Recent Patents on Signal Processing, **2010**, *2*, **47-62** **47**

Recent Patents on Image Compression – A Survey

V. Singh*

VNR Vignana Jyothi Institute of Technology, Hyderabad, India

Abstract: Image compression is the key technology in the development of various multimedia and communication applications. JPEG, JPEG2000, MPEG, H.26x are the different existing standards in still and moving Image Compression. Apart from these existing standards, to deal with the compression requirements of different applications such as camera, printer, scanner etc., a number of patents have been filed related to image compression. These patents explain application specific image compression technology. This paper presents an extensive survey on recent patents in the field of image compression.

Keywords: Image compression, image processing, video processing, JPEG, lossless compression, lossy compression, discrete cosine transform, quantization, entropy encoding, color image compression, image segmentation.

1. INTRODUCTION

Image compression forms the backbone for several applications such as storage of images in a data base, picture archiving, TV, facsimile transmission and video conferencing. Compression of images involves taking advantage of the redundancy in the data present with in an image. At present, image compression technology consists of three important processing stages: image transform, quantization and entropy encoding. In addition to theses technologies, new ideas emerge across different discipline and from new research. In the recent past, a vast number of patents have been filed in the area of image compression. These patents present novel ideas to solve various problems related to compression in different equipments, such as still and video cameras, scanners, printers etc.

This paper comprises four sections. Section 2 discusses existing image compression technologies and standards. Section 3 contributes the review coverage where various patents have been explained extensively. Finally, section 4 gives the concluding remarks and other possibilities of future research directions.

2. IMAGE COMPRESSION

Image compression refers to the reduction of the size of the data that images contain. Generally, image compression schemes exploit certain data redundancies to convert the image to a smaller form. A typical image compression system is shown in Fig. (**1**). The data reduction, or compression, is performed by a device known as the encoder. The encoder reduces the data size of the original image X. The compressed image Y is the output which passes through a channel (usually an actual transmission channel or a storage system) to the decoder. The decoder reconstructs, or decompresses, the image Z from the compressed data. The ratio of the size (amount of data or bandwidth) of the original image to the size of the compressed image is known as the compression ratio or compression rate. The compression ratio can also be expressed in bpp (bits per pixel). The term bit rate is a general term for bpp. The higher the compression rate, the greater is the reduction of data [1].

Depending on the application, the channel may be affected by noise which results in distortion of the compressed image during transmission. If so, the channel is known as an error-prone channel; otherwise, it is errorless. In Fig. (**1**), the channel is assumed to be error-free, hence Y is the input to the decoder. Data compression schemes can be divided into two broad classes:

a) Lossless compression schemes, in which Z is identical to X

b) Lossy compression schemes, which generally provide much higher compression than lossless compression but allow Z to be different from X.

A compression algorithm can be evaluated in a number of different ways. We can measure the relative complexity of the algorithm, the memory required to implement the algorithm, how fast the algorithm performs on a given machine, the amount of compression, and how closely the reconstructed image resembles the original image. The main goal of data compression algorithm is to represent any given data at low bit rates [2].

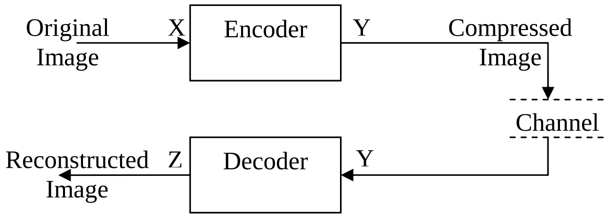


Fig. (1). Block diagram of image compression system.

2.1. Lossless Compression

In lossless image compression, the reconstructed image Z at the output of the decoder is exactly the same as the original image X (Fig. **1**) at the input of the encoder, provided the channel is errorless. One form of lossless

---

*Address correspondence to this author at the VNR Vignana Jyothi Institute of Technology, Hyderabad, India; E-mail: vipulasingh@yahoo.com

1877-6124/10 2010 Bentham Open

compression is Huffman coding [3]. In this technique, it is assumed that each pixel intensity is associated with a certain probability of occurrence and this probability is spatially invariant. Huffman coding assigns a binary code to each intensity value, with shorter codes going to intensities with higher probability. If the probabilities can be estimated a priori, then the table of Huffman codes can be fixed at both the encoder and the decoder. Otherwise, the coding table must be sent to the decoder along with the compressed image data. Other lossless compression techniques include run-length coding (RLC) [4], arithmetic coding [5], and bit plane coding. Like Huffman coding, they also have limited compression ratios and so are used only in very sensitive applications (such as medical imagery) where data loss is unacceptable, or used in conjunction with other techniques.

2.2. Lossy Compression

Lossy compression is mainly used to compress multimedia data (audio, video and still images). JPEG [6] is a still image compression standard that was developed by the “Joint Photographic Experts Group”. JPEG was formally accepted as an international standard in 1992. It is a lossy image compression method. It employs a transform coding method using the DCT (*Discrete Cosine Transform*) [7, 8]. The effectiveness of the DCT transform coding method in JPEG relies on the major observation that the useful image contents change relatively slowly across the image, i.e., it is unusual for intensity values to vary widely several times in a small area, for example, within an 8×8 image block. Much of the information in an image is repeated, hence “spatial redundancy". Psychophysical experiments suggest that humans are much less likely to notice the loss of very high spatial frequency components than the loss of lower frequency components. The spatial redundancy can be reduced by largely reducing the high spatial frequency contents.

Main Steps in JPEG image compression as shown in Fig. (**2**) are:

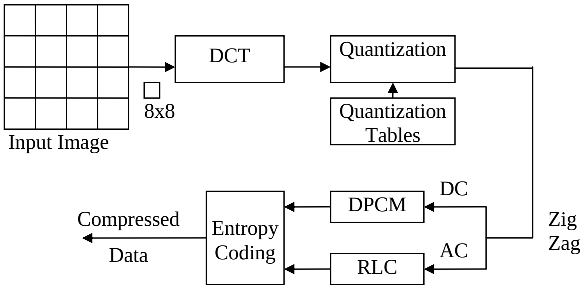


Fig. (2). Block diagram for JPEG encoder.

- Transform RGB to YIQ consisting of one luma component (Y), representing brightness, and two chroma components, (I and Q), representing color. Then each image is divided into 8×8 blocks.
- The 2D DCT is applied to each block image. Using blocks, however, has the effect of isolating each block from its neighboring context. This is the reason why JPEG images look blocky when a high *compression ratio* is specified by the user.
- The quantization step is the main source for loss in JPEG compression. DCT coefficients are quantized using predefined quantization tables. The entries of the quantization tables tend to have larger values towards the lower right corner. This aims to introduce more loss at the higher spatial frequencies maximizing the compression ratio while minimizing perceptual losses in JPEG images.
- A large value at the top left corner is the DC coefficient which is the average of all 64 input values. The remaining 63 coefficients are called AC coefficients. Zig-zag ordering (Fig. **3**) and run-length encoding (RLC) is done on quantized AC coefficients. While DC coefficients are encoded using Differential Pulse Code Modulation (DPCM).
- The DC and AC coefficients finally undergo an entropy coding step to gain a possible further compression.

Entropy Coding (EC) achieves additional compression losslessly by encoding the quantized DCT coefficients more compactly based on their statistical characteristics [8]. The JPEG proposal specifies both Huffman coding and arithmetic coding. The baseline sequential codec uses Huffman coding, but codec with both methods are specified for all modes of operation. Arithmetic coding, though more complex, normally achieves 5-10% better compression than Huffman coding.

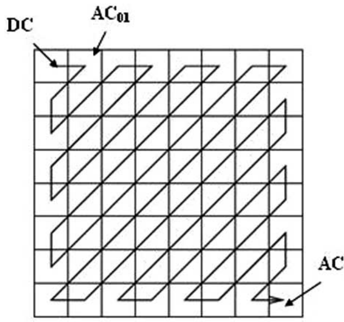


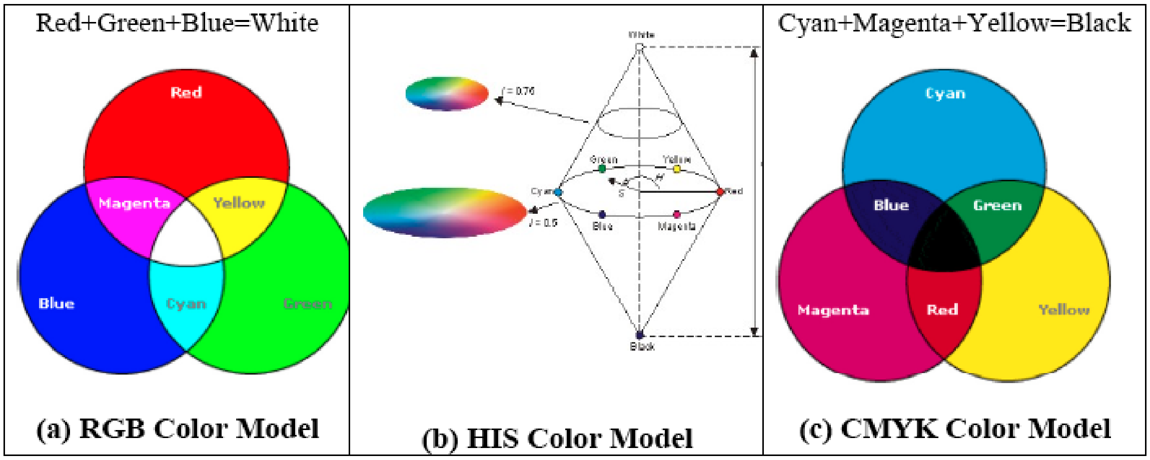
Fig. (3). Zig zag scanning.

Despite all the advantages of JPEG compression schemes based on DCT namely - simplicity, satisfactory performance, and availability of special purpose hardware for implementation, these are not without their shortcomings. Since the input image needs to be “blocked”, correlation across the block boundaries is not eliminated. This causes noticeable and annoying “blocking artifacts” particularly at high compression ratio.

2.3. Different Color Models

Colors are important in our perception of the world, therefore the color models are designed to create different colors from a small set of primary colors. Though there are several color models, but generally three color models are used: RGB, CMYK HSI and YIQ [9].

RGB stands for Red, Blue and Green. It is an additive color model as shown in Fig. (**4a**), used to display the image on the computer screen. A computer screen mixes shades of red, green, and blue to create color pictures. White is the



(a) RGB Color Model **(b) HIS Color Model** **(c) CMYK Color Model**

Fig. (4). Different color models.

"additive" combination of all primary colored lights, while black is the absence of light.

CMYK stands for Cyan, Magenta, Yellow and Black. It is a subtractive color model as shown in Fig. (**4c**), generally used for printing materials. Here, white is the natural color of the paper, while black results from a full combination of colored inks.

HSI stands for Hue, Saturation and Intensity and uses these concepts to describe the image as shown in Fig. (**4b**). It is generally used for enhancing or editing images in computer graphics applications. Hue refers to pure color in the form of an angle between [0, 360] degrees, saturation refers to the color contrast and the range of S component is [0, 1]. Intensity refers to color brightness and ranges between [0, 1] where 0 means black and 1 means white. Modeled on how human beings perceive color, this color space is considered more intuitive than RGB.

As all the color spaces are either device dependent or application dependent, a conversion between them is done [10]. The conversions cannot be exact, since these spaces have very different gamuts.

3. REVIEW COVERAGE

This section is dedicated to the review of recent patents related to image compression. Sections 3.1, 3.2 and 3.3 deal with the compression of still and moving images in digital cameras. In section 3.1, the authors describe how the captured image has to be compressed according to the space available in the memory of the digital camera without deleting other photos. In section 3.2, the authors have suggested image processing a method to efficiently process images in less time and the method is suitable for high speed and low power requirements of mobile phones and digital still and video cameras. A novel way compressing still images in the digital camera is suggested in section 3.3 so that large number of images can be stored in limited memory space.

In DCT based compression algorithm, blocking effects are visible if the compression ratio is high. In section 3.4, the author suggests a novel technique to prevent blocking artifacts in the image compressed by DCT based encoder. Section 3.5 and 3.6 deal with the printing of images from digital camera. A technique to print the compressed captured image in high quality is explained in section 3.5. While in section 3.6, the author suggests a method to connect digital camera to PC and printer so that high resolution images can be viewed and edited before printing.

Section 3.7 deals with the compression of multi channel color images. In order to achieve high compression ratio, the authors suggest a method to separate the three planes of color image data and compress each plane independently. In section 3.8, the authors explain handling of compressed video data with different image compression formats to be used in broadcasting TV stations. In section 3.9, the author suggests a method of monitoring the quality of medical images by a specially trained person, so that the repetition of the acquisition process can be avoided.

3.1. Image Compression Method for Still and Moving Images

Okada, Miyuki *et al.* [11] in 2007 proposed image compression processing method, device and program that are used in processing of still image data and moving image data in digital still cameras, digital video cameras, and cell phone cameras.

In the existing image compression device, the data is compressed several times in a step wise manner, so that compressed data occupies a predetermined size. The input image data is compressed by a lossy encoder similar to the encoder in Fig. (**2**). A byte calculator calculates the amount of memory required to store the compressed data. If this value is more than the memory space available, then the quantization table is adjusted to recompress the data till it fits in the required memory space. Because of this reason, it takes a very long time to complete the compression processing. The device using this compression technique can not capture image in short imaging interval. For repeating compression process several times, large memory is required to store original data.

A solution to these problems was suggested by Miyuki Okada *et al.* [12], where the image data is compressed rapidly one time. First a thumbnail image (preparatory image) usually smaller than the size of original image is generated. Horizontal and vertical high-frequency components are extracted from the thumbnail image. Based on that, the compression rate is predicted which is generally erroneous as lesser amount of information is used to predict the compression rate. To check if the target image is captured properly or not, high-frequency component integration processing and code amount prediction processing are executed in the image monitoring mode. But, the captured image is loaded after transition to a data recording mode. Therefore, the image subjected to the high-frequency component integration processing and code amount prediction processing in the image monitoring mode does not correspond with the image loaded to be recorded. In imaging with use of a flash in particular, this point would cause deterioration of accuracy of the code amount prediction.

The previous devices have several problems:

- Long time is taken to compress the image as multiple feedbacks are required to compress the image to an appropriate amount.
- Large memory is required to store the entire image in order to implement multiple time processing.
- Errors occur in the prediction of the amount of compression as only high frequency components are used for prediction on a thumbnail image which is generally smaller than the original image.
- Image quality deteriorates because of errors in the prediction of the amount of compression.

The solution to this problem is suggested by the authors [11] by compressing the data one time to an appropriate rate. An appropriate compression rate is decided by generating a thumbnail image and finding the high and low frequency components of this image as shown in Fig. (**5**). The compression rate determined by this method is not unnecessarily high, thus avoiding image data deterioration.

An image compression processing method, device and program includes

1. **Record** the image signal in a storage medium.
2. **Detect the characteristics** of horizontal high frequency component and a vertical high-frequency component of the input signal.
3. **Calculate the compression rate** based on the results of step 2.
4. **Compress the image** signal at a compression rate calculated at step 3 by one time compression coding process.
5. **Detect the characteristics** of horizontal low frequency component and a vertical low frequency component of the input signal.
6. **Calculate the compression rate** based on the results of step 5.
7. Based on the results of step 3 and step 6, it is decided whether or not to **adjust the information amount** of the image signal to be stored in the storage medium. If the image includes more low frequency components than high frequency components then the information amount of the image signal is adjusted in step 8.
8. If the data has to be adjusted as per the result of step 7, the image signal is adjusted towards a reduced amount of data before **recording of the resultant image signal** in the storage medium.

The characteristics of the horizontal high/low frequency components can be detected from the image in the following steps

1. A luminance signal is produced from the signal. This luminance signal contains image dependent low and high frequency components.
2. Extract horizontal high/low frequency signal from the luminance signal.
3. Convert this high/low frequency horizontal component into absolute value.

Similarly characteristics of the vertical high/low frequency components can be detected from the image in the following steps.

1. Band limit the luminance signal in order to adjust the information amount.
2. Horizontal thinning or interpolation of a pixel is executed on band limited luminance signal.
3. The above signal is stored in the line memory in order

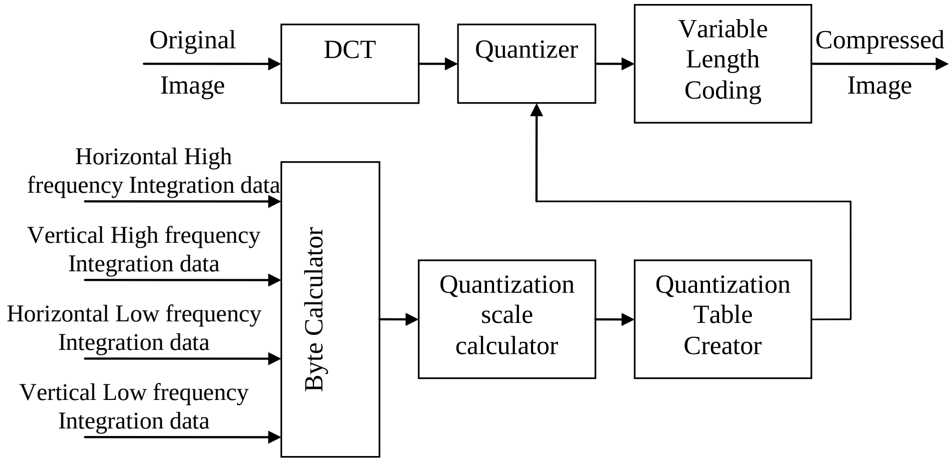


Fig. (5). Block diagram of code amount predictor [11].

to delay it by one horizontal line.

4. Signal from step 2 and step 3 are arithmetically operated to extract vertical high/low frequency component.
5. Convert this high/low frequency vertical component into absolute value.

The **advantages** of the above methods are:

- Compression can be done rapidly
- By calculating horizontal and vertical high frequency components, small as well as large changes in the data can be identified and compression rate can be calculated without error.
- Image distortion is avoided at the time of decompression.
- Storage medium is used efficiently as compression is completed in one step thus allowing large volume of image data to be stored.

3.2. An Image Processing Method for High Resolution Digital Cameras

Saito Masatake *et al.* [13, 14] suggested an image processing method, apparatus and camera system to efficiently process the captured images in very less time without deteriorating them.

In recent days, high resolution images are captured by the digital cameras which lead to high volume of data to be handled and processed by large amount of expensive integrated systems. These include high width data bus, high operating frequency, large capacity recording and storage devices. If the device is portable such as *mobile phone, digital camera or a video camera* which requires high speed performance and low power consumption, compression has to be performed on the image data while recording it to the memory storage devices. This memory device is later accessed by data transmission unit, data reception unit and data processing unit. Data transfer between these blocks cause several problems such as:

1. It is difficult to maintain the quality of the compressed data and the compression performance. To increase the compression rate, lossy compression has to be used which degrades the quality of the image.
2. For high compression efficiency algorithms, processing speed is very low and processing time is very large. e.g a video camera captures 30 images in one second where as it is difficult to process one image in 1/30 seconds. Decompression also takes equally longer time which makes it unsuitable for broadcast purpose.
3. Before actual compression occurs, it is virtually impossible to know data capacity after compression. Even for the worst compression efficiency, targeted data reduction should be achieved.
4. Random access property is generally lost during compression. While compressing the image, the pixels on the screen are “raster” scanned from top left to bottom right. Decompression is done at the same scanning order from the top of the compressed data. Current variable length encoding data is not capable of expanding at the middle of the encoding data to obtain a freely selected part, so that the random access property is totally lost.

In order to tackle these issues, the authors [13] have proposed an image processing apparatus, method and camera to efficiently process the pictures in very less time without deteriorating them and ensuring the random access property also. The description is as follows.

1. As shown in the Fig. (**6**), a data transmission block, a data reception block and a data processing block are attached to a storage unit which can be accessed by all of them. A compression unit acts as an interface between data transmission block and the memory block to compress the input image before writing it to the storage unit. Similarly a data expansion unit expands the image before it is read from memory by the data receiving unit. Another data compression expansion block reads the data from memory to perform the data processing. Due to this arrangement, amount of data passing through the bus reduces substantially and a capacity of a memory device in the integrated circuit is also reduced.
2. The data compression unit is designed to keep **four goals** in mind.
   a) To maintain high picture quality by choosing the

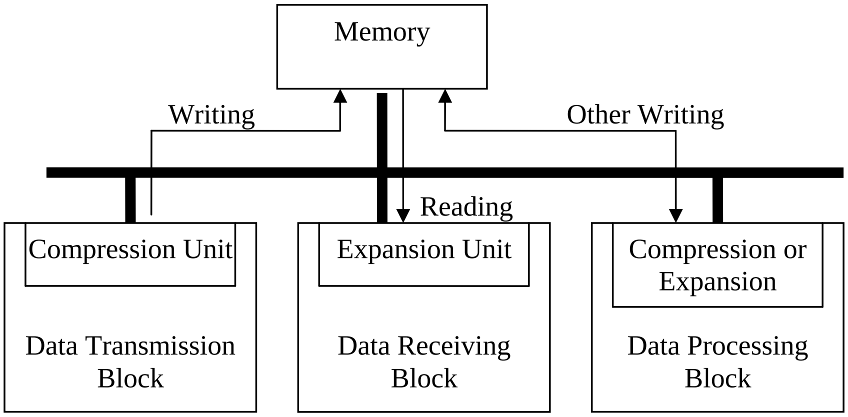


Fig. (6). Conceptual block diagram of Image processing Apparatus [13].

compression algorithm suitable to the Human Visual characteristics.

b) To reduce the data flowing through the bus by adapting 1-path encoding method for the entire image.

c) Certain compression rate is always ensured for all input images.

d) Random access property should be retained.

In order to achieve these goals, the following **actions are taken**

- Compression is performed block wise containing only limited number of pixels.
- By realizing 1-path encoding processing for an entire image, an increase of the band is prevented. A target compression rate for the entire image is attained. Based on the target compression rate, a lossless (preferably) or a lossy (if compression rate is not satisfied by the lossless method) compression method is chosen.
- A high picture quality is maintained by multi-path encoding processing for each block in one image. A target compression rate for each block is also attained. Based on the compression rate, a lossless (preferably) or a lossy (if compression rate is not satisfied by the lossless method) method is chosen for that block.
- Based on the complexity of the block in the image, bit resolution of the pixels is varied. Adaptively, bit resolution of the pixel is reduced for the block for complicated patterns. The compression unit makes it harder to reduce the resolution of the pixel if its value is low as compared to the pixels with high values. This prevents visual picture quality deterioration as human eye is more sensitive to delicate changes in the dark part. In a step wise algorithm the bit resolution is unchanged as much as possible for low pixel values, lower order bits if high pixel values are reduced in small steps until the target compression rate is achieved. Thus the reduction degree of the bit resolution is controlled to an optimal point for each block.
- The address determined by each block is recorded to maintain random access quality. Each block is compressed by its target compression rate and the blocks are recorded in the storage region secured for that block, so that an address of top data of each block is fixed in the memory regardless of an actual compression rate of the block.
- Data is divided into different bit resolution parts such as the higher-order bit side and lower-order bit side, etc. and suitable compression methods are separately used. Lossless compression is performed on the higher-order bit side to assure a minimum picture quality. For example, 12-bit resolution image data is divided to a higher-order side 8-bit part and lower-order side 4-bit part, lossless compression in combination of a difference PCM and Huffman encoding based on neighbor correlation is adopted to the higher-order 8-bit part and uncompressed data (PCM) is adopted as it is to the lower-order side 4-bit part.
- Trail calculation of the encoding amount is done with different schemes and finally the scheme that satisfies the aim is selected and actual encoding is done according to that scheme. If none of the schemes is found to be good, the higher-order data is stored as it assures a minimum capacity after compression.
- For example, the data is divided into higher order 8 bit part and lower order 4 bit part. The lower order 4 bits are respectively coded with 1) lossless compression 2) lossy compression 3) second lossy compression for assuring a band in the worst case until target rate is not achieved. The higher order 8 bit part is coded separately by 1) lossless compression 2) lossy compression 3) second lossy method for assuring a band in the worst case till the target rate is achieved. If the target rate is not achieved, the higher order 8 bits are saved uncompressed.

3.3. Special Image Compression Method for Digital Cameras

Jang Sung-kyu [15] in 2009 proposed digital image processing apparatus and image compression method to significantly reduce the image file size so that large number of image can be stored in limited memory.

In digital camera, image files have to be stored in the limited amount of storage space. The images are captured and compressed to reduce the file size by conventional image compression methods before storing it. But in the existing system there is a limit of reducing the file size. If the user can not further store an image file in the storage medium, he has no option but to delete some of the previously stored images or replace a new storage medium.

In order to remove this drawback, the authors [15] has suggested a digital image processing apparatus, as shown in the Fig. (**7**), having the following blocks.

- **A sub-image generator** to generate multiple sub images from a still image. It splits the still image into a number of sub images having same size or size acceptable to moving image compressor. Each of the sub-images is considered as a single frame image of a moving image, thus still image can be considered as a single moving image having multiple frames.
- **A moving image compressor** that treats the sub images of the still image as frames of the moving image and compress moving image having multiple frames. It uses MPEG or H.264 for compressing the moving image, where the first frame is compressed using JPEG and the difference between the subsequent two frames is extracted and coded. This procedure significantly reduces the amount of data stored in the memory. In addition to compression, the moving image compressor also stores the additional information regarding the order of sub-images, which represents the position of extracted data, in the storage medium of the digital image processing apparatus.

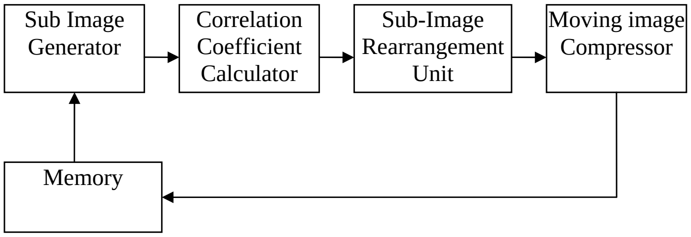


Fig. (7). Block Diagram of Digital Image Processing Apparatus [15].

- **A display unit** where one of the sub-images is displayed.
- **Correlation coefficient calculator** for calculating a correlation coefficient.
- **A sub image rearrangement unit** to rearrange the sub image with high correlation adjacent to each other. This is done to achieve high compression from the moving image compressor. When the correlation between the sub images is high, the cost is small; hence these sub images can be arranged adjacent to each other.
- **Memory**, a recording medium for storing the program for executing the method for compression.

The working description is as follows:

The overall operation of the digital photographing apparatus is controlled by a CPU (central processing unit). In a photographing mode, the image is captured by image sensing element where the position of the lens, degree of the opening of the iris and sensitivity of the image sensing element is controlled by CPU. The captured image is converted to digital image data by an analog-to-digital converter and stored in the memory and given to digital signal processing unit. Some preprocessing operations such as gamma correction, white balancing etc. are applied on the captured image. Multiple sub images of the still image are generated by the sub image generator. These sub images are considered as a single frame of a moving image. These frames of the moving image are compressed by MPEG or H264 to reduce the size of the image substantially. The compressed data is stored in the storage medium.

Fig. (**8**) illustrates the configuration of the image file stored in the memory of the image processing apparatus. The image file includes a header, a main image data part and a screen nail image data part. The screen nail image is the single image that is displayed on the screen. It is stored in the screen nail image data part. The image file includes a header, a main image data part and a screen nail image data part.

This method is **advantageous** as it is capable of significantly reducing the size of the image so that more images can be stored in the memory.

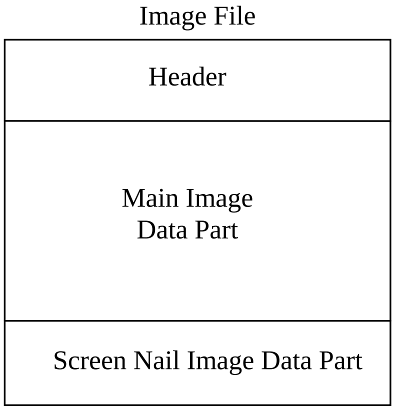


Fig. (8). Configuration of image file stored in memory [15].

3.4. Method to Deblock Compressed Image

Koo, Jae-hoon [16] suggested a method and apparatus for generating the assistant images to deblock the compressed image.

A general image compression encoder, as shown in Fig. (**9**), has an image-dividing unit, a Discrete Cosine Transform (DCT) unit, a quantization unit, and a Huffman Coding unit. To compress an image, it is divided in 8 x 8 blocks and then these blocks are compressed individually. When this image is decompressed, it has several discontinuities at the boundaries of the blocks. The restored image looks as if several tiles are joined together. These blocking artifacts are severe if the compression ratio is high. The acceptable quality of the reconstructed image reduces the amount of compression done on the image.

Earlier a number of deblocking approaches have been proposed in literature

- In the first approach, a deblocking algorithm block is introduced between quantization block and Huffman coding block. To implement this, the encoder has to be redesigned and the existing compression encoder can not be used.
- To avoid the above mentioned problem, another solution is suggested. The compressed image is Huffman decoded, Deblocking algorithm is applied to it and again the result is Huffman encoded. Though this approach allows the reuse of existing encoder, it overloads the processor thus lowering the processing

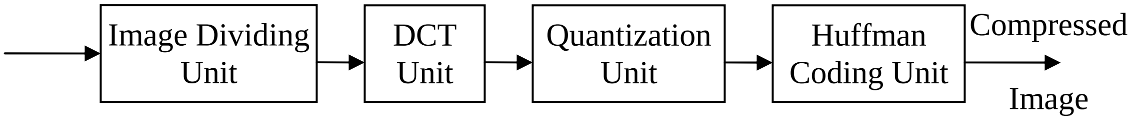


Fig. (9). Image compression encoder.

rate and increasing the power consumption of the processor.

- Another approach is to decompress the image and then apply deblocking algorithm. But if the user's decompression algorithm does have deblocking algorithm, the blocking artifacts remain in the decompressed image.

Therefore a method and an apparatus are suggested by the authors [16] that can prevent the blocking artifacts and reduce the processing requirements and the power consumption of the processor. An assistant image is generated from the original image and using the assistant image, the general image compression encoder performs deblocking processing as shown in Fig. (**10**). Various steps of the deblocking process are

1. An **assistant image is generated** from the original image by the assistant image generation unit. To generate the assistant image, original image is divided into a number of blocks. The DC coefficient (defined in section 2.2) of each block is calculated. Number of DC coefficients is the same as the number of blocks (one DC coefficient per block is calculated). From these DC coefficients AC coefficients of the divided blocks are estimated. One block has 63 AC coefficients, to estimate all the AC coefficients can cause overload and processing delays. Thus only low frequency AC coefficients are estimated which contain maximum information.
2. **The original image and assistant images are compressed** by the encoder.
3. **The compressed image is the deblocked** using the compressed assistant image. Initially the compressed original image is stored in the buffer. Then AC Huffman code of the compressed original image is replaced with the AC Huffman code of an assistant image corresponding to the compressed original image. As shown in Fig. (**11**) only five Ac coefficients are estimated and the rest of the AC coefficients per block are considered to be zero. Thus, the Huffman code for the five AC coefficients is the AC Huffman code of the compressed assistant image used for the replacement. The replacement process is carried out block by block. After the AC Huffman code replacement is over, the new compressed image is the final output which can be stored in the storage unit.

The Assistant Image Generation Unit divides the original image into blocks of 8 pixels by 8 pixels. Each block has one DC coefficient and 63 AC coefficients. DC coefficient (the average value) of the block is calculated and AC coefficients are estimated. Estimation of all 63 coefficients requires lot of computational efforts which leads to the processing delay and overloads. Therefore the assistant image generation unit estimates only five low frequency AC coefficients containing maximum information as shown in the Fig. (**11**). For example, to estimate the AC coefficients of block 5, DC coefficients of adjoining 1 to 9 blocks are used. The shaded five AC coefficients AC_{01}, AC_{02}, AC_{10}, AC_{20} and AC_{11} are from DC coefficients of blocks 1 to 9. Estimation equations are given in following equations

$$AC_{01}=1.13885(DC_4-DC_6) \qquad 1$$

$$AC_{10}=1.13885(DC_2-DC_8) \qquad 2$$

$$AC_{20}=0.27881(DC_2+DC_8-2DC_5) \qquad 3$$

$$AC_{11}=0.16213((DC_1-DC_3)-(DC_7-DC_9)) \qquad 4$$

$$AC_{02}=0.27881(DC_1+DC_6-2DC_5) \qquad 5$$

Once the AC coefficients estimation for all the blocks are over, inverse DCT (IDCT) is applied to the blocks. An assistant image is generated by combining all the blocks for which IDCT has been applied. The output of the assistant image generation unit goes to the image compression encoder.

3.5. Modified Image Compression Method for Printers

Ogawa Kazuma [17] in 2005 suggested image compression and decompression method, device and image processing program that produces less amount of high quality data in less time from the captured image and also suggested a *copier and printer* that prints high quality image at low cost. The proposed method can be used in *machine, scanner, digital camera and video player.*

In conventional printers, original image is stored in the memory after lossy compression as compressed image takes less space to store it. To print the compressed image from the memory, it is first decompressed and converted into CMYK format. The decompressed data usually has distortions which lead to low reliability for image reproduction because image segmentation is performed on decompressed data.

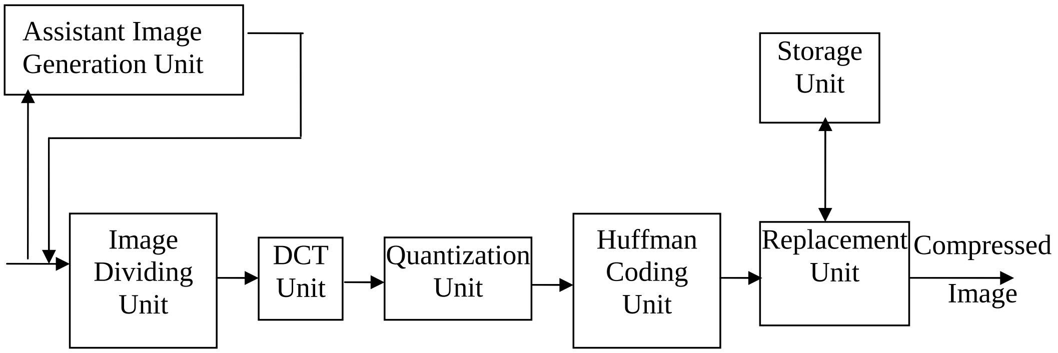


Fig. (10). Image compression encoder with deblocking [16].

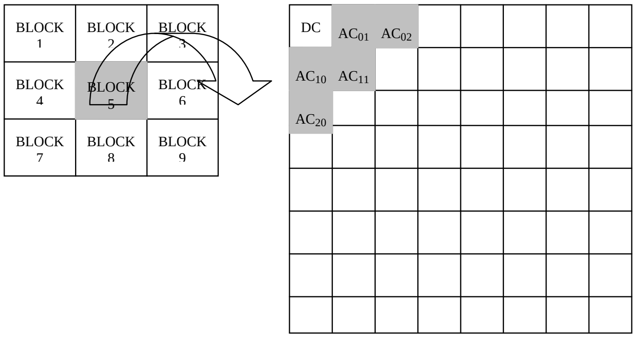


Fig. (11). AC coefficient estimation [16].

A remedy of this is suggested by Ogawa Kazuma [18], where a segmentation signal is generated from input image and stored in the memory along with the compressed image. This segmentation signal is used to perform correction on the decompressed signal. The problem in this technique is that it requires large space to store this segmentation signal and it takes lot of time for data transmission.

The author [17] suggested an image compression device to perform lossy image compression on the input signal and generate the complete segmentation signal. The input processing device reads the input image in the RGB format and applies reading corrections (shading correction, gamma correction, filtering, and color correction) to each piece of RGB data. After that this input image data is compressed as shown in Fig. (**12**) using the following steps:

1. A **first segmentation section** performs segmentation on the input image and generates the segmentation signal.
2. An **image compression section** performs lossy compression on the input image.
3. First image decompression section decompresses the compressed image stored in the memory.
4. A **second segmentation signal** performs the segmentation on the decompressed image and generates a decompression segmentation signal.
5. **Differential extracting section** extracts the difference between the segmentation signal and decompression segmentation signal and generates the differential signal. Decompression segmentation signal is usually similar to segmentation signal except at the high frequency where the signal is lost. Because of this, the differential signal has lot of “0” which are easily removed by the lossless compression technique and the size of differential signal becomes extremely small. The storage requirement and transmission time requirement of the differential signal is considerably reduced because of its small size.
6. **Differential signal compression section** losslessly compresses the differential signal and generates the compression differential signal. As the differential signal is compressed losslessly, it is possible to completely recover the differential signal after decompression. Sometimes compressed differential signal is larger in size as compared to compressed segmentation signal. In order to avoid this situation, size comparing section is included.

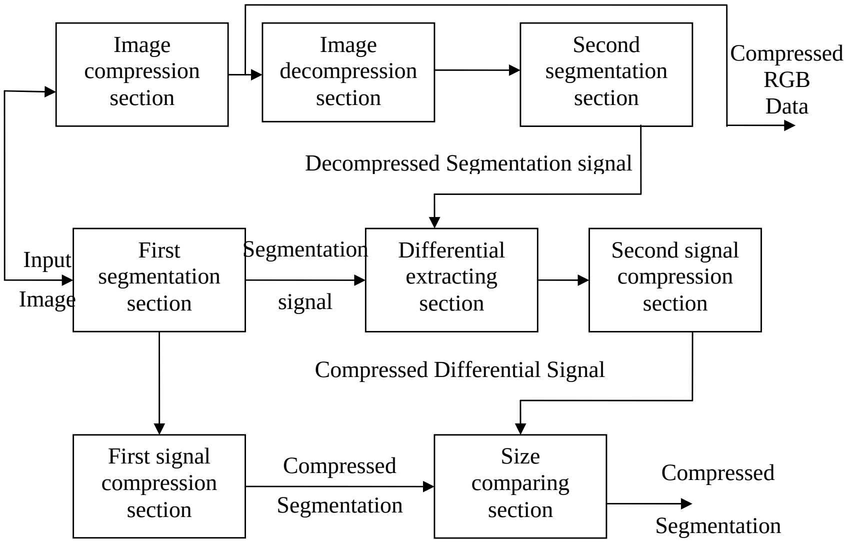


Fig. (12). Data compression unit [17].

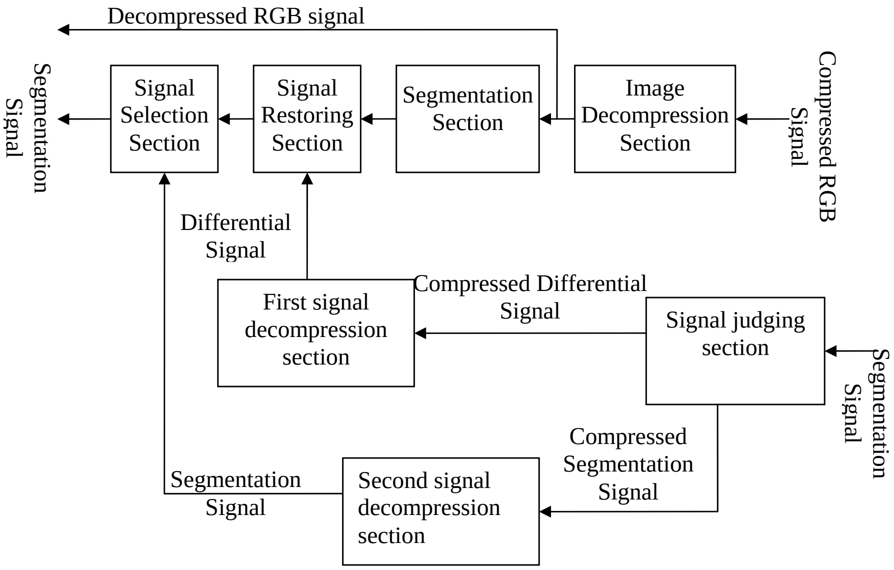


Fig. (13). Data decompression unit [17].

7. **Signal compression section** compresses losslessly the segmentation signal to generate compressed segmentation signal.
8. **Size comparing section** compares the size compares the size of compressed differential signal and compressed segmentation signal and stores the signal which ever is of smaller size.

The author [17] also suggested image decompression device as shown in Fig. (**13**), where following steps are followed:

1. Image decompression section performs decompression of the compressed image data.
2. Segmentation section performs the segmentation of decompressed image data to generate decompression segmentation signal.
3. Signal judging section uses the stored segmentation signal as an input and decides whether the segmentation information is a compression segmentation signal or a compression differential signal.
4. First signal decompression section performs decompression of the compressed differential signal to generate a differential signal.
5. Signal restoration section combines the decompression segmentation signal and differential signal and generates the segmentation signal.
6. Second decompression section decompresses the compressed segmentation signal to generate the segmentation signal.
7. Signal selection section outputs segmentation signal from signal restoring section or from second signal decompression section based on the results of the decision made by the signal judging section.

The data decompression section produces decompressed RGB signal and segmentation signal. Output corrections (filtering, gamma correction, RGB-to-CMYK conversion, and halftoning such as dithering and error diffusion) are applied to these signals and are given to the printer section for printing.

The **advantages** of the above mentioned method are that an electronic video device such as *facsimile machine, scanner, digital camera, or video player* can produce less amount of output data. This reduces the processing time and the memory required to store the images. This makes it possible to realize a high-quality printing at low cost, with the use of a printer which can process the compression RGB. Such a printer prints high-quality images at low cost.

3.6. Image Capturing and Processing Method with External Printer

Sekine Masayoshi [19] in 2009 proposed image capturing and processing method, control method and storage medium for printing the captured image with external printer. This is specifically applicable to digital cameras connected with USB to personal computer and printer.

High resolution images are taken with today's digital cameras which need to be stored on memory card and sent to the PC through a USB cable for viewing or editing before printing. It is nearly impossible to enlarge, print, edit and view the images without computer. Sekine, Masayoshi [20] gives the option of outputting the image to external device for editing and printing. But it has several problems:

1. It takes very long time to transfer large image data to printer through RS-232C (Serial communication) and providing a parallel port in the camera makes it very large and expensive.
2. Power consumption becomes very high as high speed CPU is required to execute printing processing in cameras.
3. Cost of the camera increases as the images need to be stored in the memory chip on the camera before being printed as it takes lot of time to print it.

To overcome these problems, the author [19] proposed a means to control the operating frequency so that power

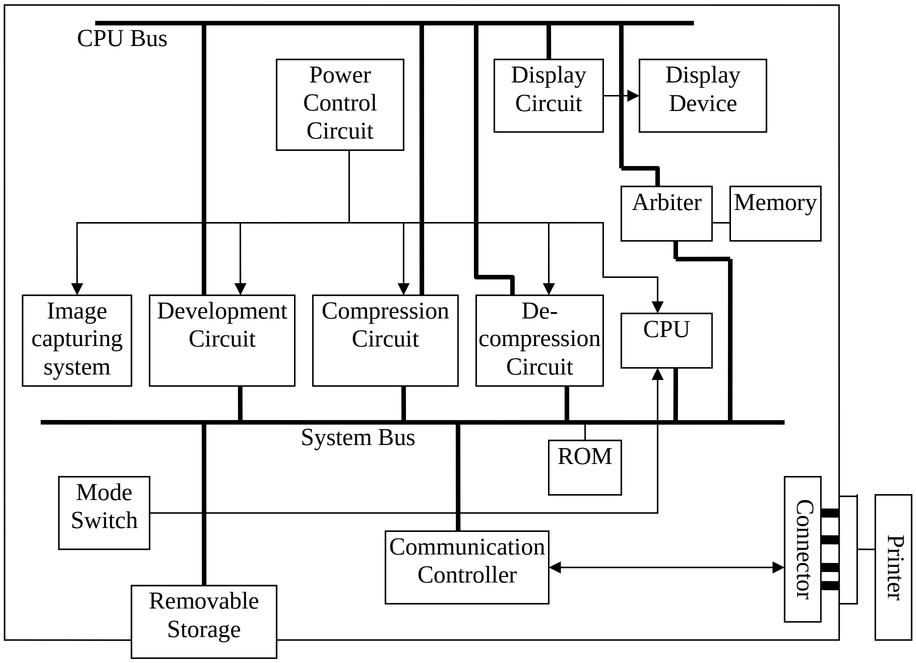


Fig. (14). Block diagram of digital camera [19].

consumption is reduced while image capturing and printing at high speed. He proposed

- An image capturing apparatus to capture the input image.
- A clock generation block that generates the block at different frequencies and outputs clock at high operating frequency in printing mode than that of the image capturing mode.
- An external printer is directly connected to the image capturing apparatus.

The operation of the digital camera as shown in the Fig. (**14**) is as follows:

The image is captured by the image capturing system. The image capturing system includes a lens, an aperture, a focus control section and a zooming control section. CPU controls the operation of the digital camera, memory control of the memory and operation control of the electric-power control circuit through CPU bus. The memory is connected to both the system bus and CPU bus for cost reduction. The external printer is connected through a connector to the communication controllers. An LCD (liquid crystal display) display circuit is provided to display the errors and the other information. The image capturing operations such as development, compression, decompression are done in hardware by special circuits as development circuit, the image compression circuit, and the image decompression circuit which operate at high speeds with low power consumption. The electric power control circuit controls the clock for various operations. The power control circuit includes a crystal oscillator, a PLL (phase lock loop) circuit, a selector to select between different frequencies, a scaling circuit that scales the frequency by half, one fourth, one eighth and so on and a setting register that controls various clocks.

The mode switch is used to select a desired operation mode from among an image-capturing mode, a reproduction mode, a communication mode, and a printing mode by the user. When the mode switch is in **image-capturing mode**, the reproduction mode, or the communication mode, all these functions are implemented by special hardware circuits and don't require processing power of the CPU. Therefore all these operations are carried out at low frequency that is less than half of the highest operating frequency of the camera.

When the mode switch is in the **printing mode**, the camera is operated at highest frequency so that printing processing can be operated at highest speed. To print a compressed image stored in the memory, first it is decompressed, and then printing processing is applied and sent to printer for printing. Once the printing is over, camera is switched to low speed mode. Thus the digital camera operates in the high-speed mode only for very short period required for the printing processing, and wasteful use of electric power is prevented.

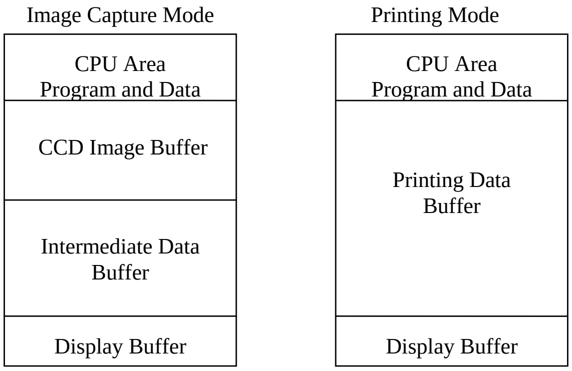


Fig. (15). Memory structure in the digital camera [19].

Fig. (**15**) shows the **memory structure** of the image capturing mode and the printing mode. In image capturing mode, the memory includes CPU area which stores program and data stored by the CPU, CCD (charge couple device) image buffer area to store the input images in large storage area, intermediate data buffer to store data during the compression processing, and display area to show various indications in the display device. The printing mode has the CPU area, printing data buffer and display buffer. CCD buffer and intermediate data buffer is not required in the printing mode. This memory structure allows printing processing efficiently without providing additional memory.

3.7. Multi-Channel Color Image Compression

Roylance *et al.* [21, 22] in 2005 proposed a method for multi-channel color image compression, transmission and decompression that included mixed resolution. The described method can be used in copy machines, facsimile machines and laser printers.

Generally, for color images three planes of data is processed. First plane has foreground colors which is compressed losslessly to retain the detail information. The second plane has background color with fewer details, hence compressed with lossy technique. The third plane is a binary plane that serves as a switch command between the foreground and background color planes. In the conventional multi-channel color compression, all the three planes have the same resolution equal to the targeted device which limits the amount of compression achieved.

The authors [21] claim that by separating the foreground colors, background colors and edge information, higher compression ratios can be achieved. The **advantage** of separating out edge information from color information is that the reconstructed image contains high details of foreground and background color at high compression ratio. The color information is defined at a lower resolution as colors generally do not change at a high frequency and at lower resolution higher compression ratios can be obtained. The edge information contains the commands switch between the foreground and background colors. The edge information occurs at high resolution and high frequency rates relative to color information. The lower resolution color planes can be pixel replicated to the resolution of the edge information after compression for proper edge selection to occur. The resulting reconstruction image of foreground and background colors retains high detail.

The authors [21] have described a method of multi-channel color image processing, multi-channel color image compression, multi-channel color image decompression, transmission and reception the color image. They also described the data storage medium containing computer instructions for multi-channel color image processing, compression, decompression. A codec for multi-channel color image processing is also described.

The entire method as shown in Figs. (**16**, **17**) is described below:

1. Separate the original color image into background colors, foreground colors and edge information.
2. Reduce resolution of the background colors and foreground colors
3. The edge information is having higher resolution than the reduced resolution background colors and foreground colors.
4. Compress the reduced resolution background colors and foreground colors and the edge information. The compression techniques selected for each of the color planes may be optimized for that color plane. Foreground and background colors are independently compressed using lossy compression method, while edge information is compressed using lossless compression method.
5. Package the compressed background colors, foreground colors and edge information to obtain an encoded color image.
6. Transmit the encoded color image.
7. Receive the encoded color image.
8. Unpack the encoded color image to obtain compressed foreground colors, background colors and edge information.
9. Decompress the compressed foreground colors, background colors and edge information.
10. Do resolution synthesis of the decompressed foreground colors and background colors to a resolution level equal to the decompressed edge information.
11. Multiplex the resolution synthesized foreground colors and background colors in accordance with the decompressed edge information to obtain a reconstructed color image. For example, if the binary plane indicates "1", that pixel can have foreground color in the reconstructed image. If the binary plane indicates "0", that pixel can have background color. Thus based on the binary plane, the multiplexer selects between the foreground and the background colors.

The multi-channel color image processing system includes an input device, a processor, an imaging device to generate a reconstructed and a memory for storing computer instructions for implementing above described multi-channel color image processing.

3.8. Compressed Video Data Processing

Higuchi *et al.* [23, 24] in 2006 proposed a method of compressed video data processing with conversion of image compression format.

In a broadcasting TV station, the news material and other records are recorded by the reporters by digital video cassette recorders (VCR). These records have to be viewed and edited for broadcasting. These records have to be transferred to the server, so that any other connected computer can access it. This video data can be transferred with or without compression. Compression format of digital video cassette recorders is generally DV (Digital Video) format while compression format of the video data at the server is Motion-JPEG. Therefore, a format converter for converting data compression format is needed between the digital video cassette recorder and the server. Generally recorders are able to reproduce the data at a faster speed as compared to the data transfer speed of the sever.

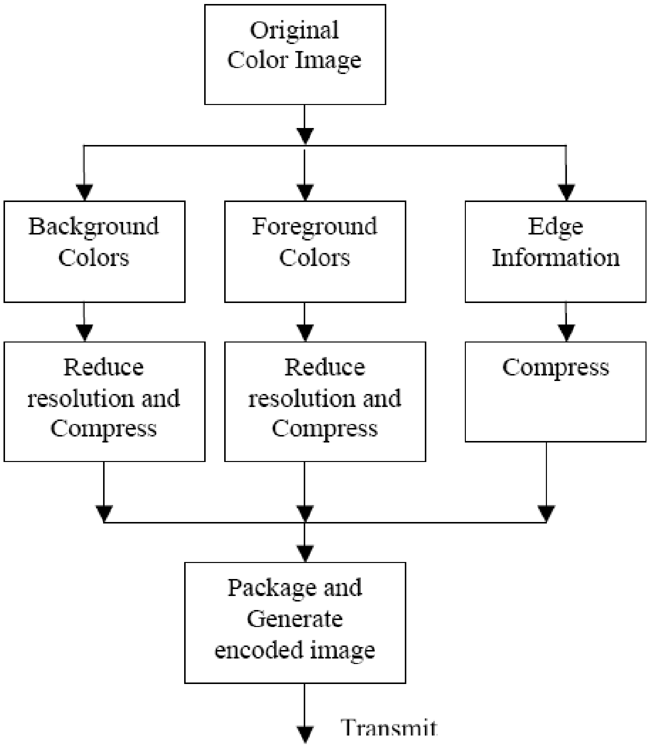


Fig. (16). Block diagram of transmitter of codec for mixed resolution color image processing [21].

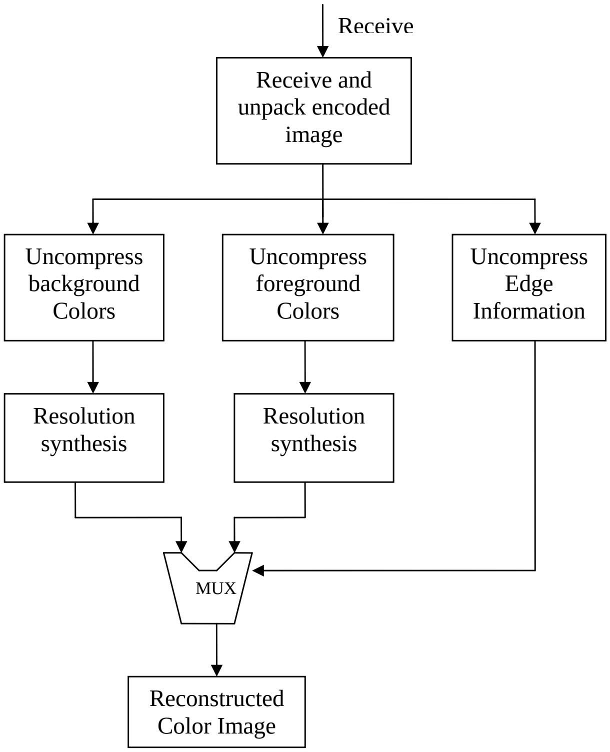


Fig. (17). Block diagram of receiver of codec for mixed resolution color image processing [21].

Thus the authors [23] have proposed a system which can deal with compressed video data of different compression formats at different speed. The authors provide a compressed video data processing system which:

- Satisfies **compatibility** of compressed digital video data of two or more different image compression formats.
- Can record or reproduce compressed video data at a **high speed** even if conversion between two or more different image compression formats is processed.

A compressed video data processing system as shown in Fig. (**18**) comprises of the following units.

1. **A digital video cassette recorder** for recording and reproducing video data according to a first image compression format (DV format).
2. A **local server** which is first recording and reproducing device.
3. **Image format converter** connected between the digital video cassette recorder and the local server. The image format converter converts compressed digital video data between the first image compression format (DV format) to the digital component serial data format (SMPTE (Society of Motion Picture and Television Engineers) 259M standard) and vice-versa.
4. **A main server** which is connected to the local server. The local server and the main server record and reproduce compressed digital video data according to a second image compression format (Motion-JPEG format) different from the first image compression format. By providing local server between the image format converter and main server, the main server can operate normally even when accessed by many computers.

The digital video cassette recorder and local server record and reproduce video data according to a first image compression format (generally DV format). The main sever records and reproduces data according to second image compression format (generally Motion-JPEG). Compressed video data are transferred between the digital video cassette recorder and local sever at a first transfer speed. The local server can record and reproduce the data at either of the first or second recording speed. Compressed data is transferred at second transfer speed between local server and main server. To transfer the data in the digital video cassette recorder at a high transfer speed from local server, the first transfer speed is much faster than the second one, usually four times faster.

The working of video data processing system is as follows:

Digital video cassette recorder records the digital data in the first image format (DV format without compression) at normal speed. This data is converted to other image format (SMPTE 259M standard) by image format converter. This data is stored and edited at local server. The received data is converted to second image compression format such as Motion-JPEG and stored. The local server also receives the data from main server in Motion-JPEG format. The image format changer converts this Motion-JPEG format data to serial data SMPTE 259M standard and sends them to digital video cassette recorder through the image format converter.

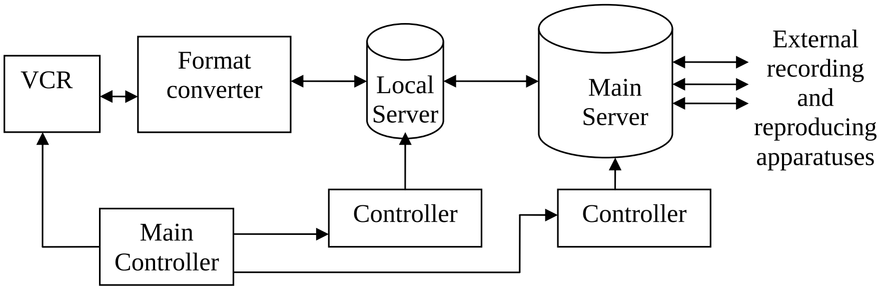


Fig. (18). Block diagram of video data processing [23].

The data in Motion-JPEG format is sent to main server from local server so that the video data can be accessed by other computers through the network.

Various advantages of the described system are

1. When the video data of the first format is recorded and reproduced after image compression format conversion, the main server can operate normally.
2. Video data is transferred at a faster rate to the local server.
3. Different image compression formats can be used without the knowledge of the user.

There are few problems in the above-mentioned video data processing system.

- Data transfer speed between digital video cassette recorder and local server can not be increased because image format has to be converted by image format converter.
- Video data is deteriorated because of the conversion of image compression format before they are edited at the local server.

A slight change in the video data processing system can remove these problems. The modified system as shown in Fig. (**19**) is as follows:

A digital video cassette recorder records and reproduces compressed digital video data at four times speed in DV format. The data received by the local server is converted to normal speed and stored and edited. The local server can also reproduce the video data in DV format at four times speed and transmit it to the digital video cassette recorder. The image format converter receives the data in DV format from local server, converts it to Motion-JPEG format and sends it to main server. It also receives the data in Motion-JPEG format from main server and sends it to the local server in DV format. The main server stores the data received from the image format changer and communicates through a network to other large capacity recording and reproducing apparatuses operating as main or local servers.

The image format converter converts compressed digital video data from first compression format (DV format) to second image compression format of Motion-JPEG standard. It has a format detector which selects one of the conversion processes automatically according to the format. Thus user can use video data without knowing the image compression format.

The procedure as shown in the Fig. (**20**) is as follows: first the compressed video data is expanded by data expansion circuit. Then the non compressed data is converted to digital composite signal by digital component signal converter unit and then to 4:2:2 component signal by 4:2:2 converter unit. Then the 4:2:2 component signals is converted to digital compression data of Motion-JPEG standard by serial data compression unit.

3.9. Progressive Image Compression for Medical Images

Martin Uwe-Erik [25] in 2007 proposed a method of monitoring the quality of the medical images using progressive image compression and transfer method and simultaneously implementing image pre-processing.

Medical image acquisition systems such as Computer Tomography (CT) system or nuclear magnetic resonance tomography (MR) system generate high quality images which need to be examined immediately in order to repeat the acquisition process if necessary. Image is acquired by an image acquisition system and brightness and contrast adjustments are done on the image by first image processing block. This entire process takes several minutes. Then the image is sent to Picture Archival and Communications System (PACS) and stored. Further processing and appraisal of the acquired image is done at image processing work

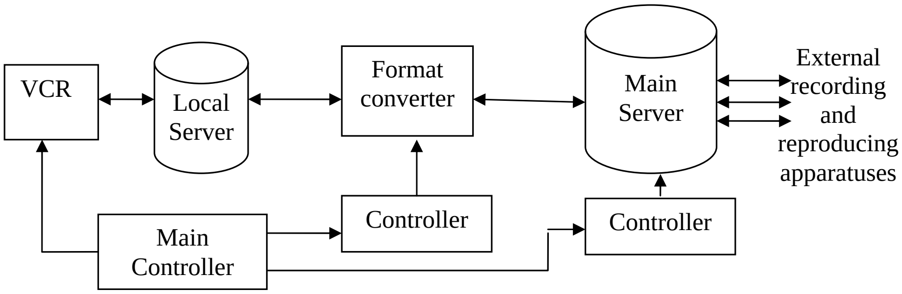


Fig. (19). Block diagram of modified video data processing [23].

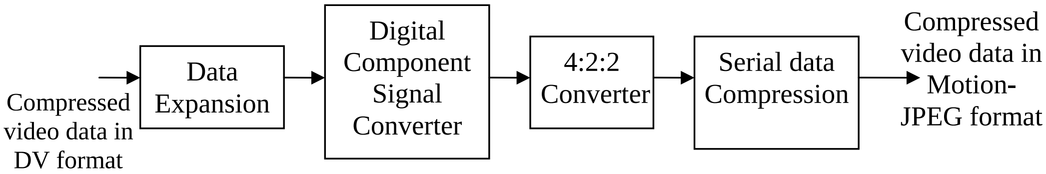


Fig. (20). Block diagram of image format converter [23].

station by specially trained person in this field. The block diagram of the existing system is shown in Fig. (**21**).

There are several problems in the present system.

1. Inspection and quality control at the image processing block is not possible.
2. Image acquisition block and image processing workstation can be at different place. Tools used in the image processing block may not be available in the image processing workstation.

These problems can be solved by ensuring the quality of the acquired image at the image acquisition system by a trained person so that a complete repetition of the acquisition procedure can be avoided. Therefore the authors propose the quality control of the acquired image with image processing tools at the image acquisition block by a radiologist. The block diagram of the proposed system is shown in Fig. (**22**) and the working description is given below.

- Image is acquired by the image acquisition system and a preview image of low quality and resolution is generated and sent to PAC over the network. The low resolution progressive image is generated by implementing progressive image compression technique such as JPEG2000. It is a multi-resolution coding that separates low-resolution data from high resolution.
- PAC forwards the progressive image collected from image acquisition system to image processing workstation, where the description of the image is generated and stored in the store unit along with the high resolution images which are available later.
- Image processing workstation provides image processing capabilities such as brightness and contrast adjustments, providing additional information such as drawings and textual description, highlighting the region of interest (ROI). The workstation has a display unit where the preview image can be viewed and image processing operations are implemented on it and the saved in the storage unit. The preview image has low resolution data and the processing and storage happens very fast while the image acquisition and image transfer are simultaneously still occurring.

Various **advantages** of this method are:

1. Image processing tools are provided only at one point that is at the workstation.
2. There is no need for a radiologist at the image acquisition system as it is possible to simultaneously implement the pre-processing on the acquired image at the workstation.

4. FUTURE DEVELOPMENTS

In this paper, we have discussed various image compression technologies applied in digital cameras, printers, copiers, facsimile machine, medical imaging, TV broadcast stations. The technologies give a solution to the problems incurred during image compression such as: long time taken while compression, how much should be the compression ratio, how much degradation of the image is allowed while compressing, how much memory the compressed data is going to take after compression, how to compress the colored image without losing the details, how to handle compressed video data with different image compression formats, how to remove blocking artifacts from the highly compressed image. At present, image compression is limited by conventional technology namely, *transform + quantization + entropy encoding*. In image transforms also mainly DCT and wavelet transform is used.

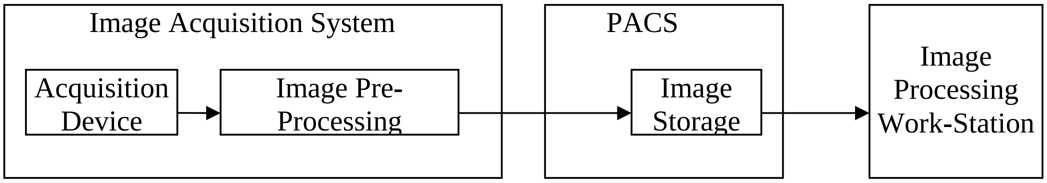


Fig. (21). Block diagram of existing image acquisition and processing system [25].

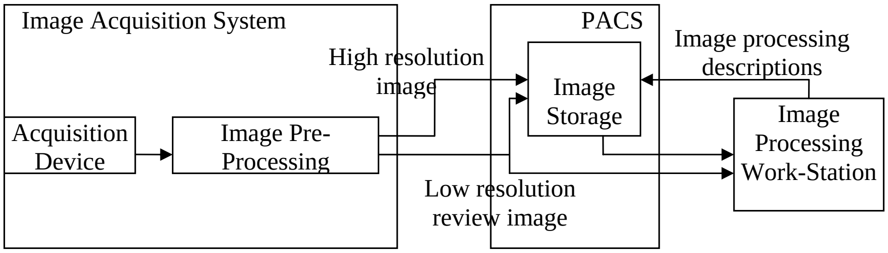


Fig. (22). Block diagram of image acquisition and processing system [25].

Further research can use projection based and adaptive reversible integer wavelet transform and multi wavelets constructed using B-spline super function, as image transforms. Combining various technologies such as neural networks especially topology-preserving property of self-organizing Kohonen feature map, fractal image coding and fuzzy logic can help achieve better compression algorithms.

REFERENCES

[1] Nelson M. The Data Compression Book. 2nd ed. New York: M&T Books 1995.

[2] Khalid S. Introduction to Data Compression. 2nd ed. New York: Elsevier 2005.

[3] Huffman D. *A method for construction of minimum redundancy codes*. Proceedings IRE 1952; 40: 1098- 1101.

[4] Goloumb G. Run length encoding. IEEE Trans Inf Theory 1966; 12: 399-401.

[5] Witten IH, Neal RM, Cleary JG. Arithmetic coding for data compression. Commun ACM 1987; 30: 520-540.

[6] Pennebaker WB, Mitchell JL. JPEG - Still Image Data Compression Standards. New York: Van Nostrand Reinhold 1993.

[7] Ahmed N, Natarajan T, Rao KR. Discrete cosine transform. IEEE Trans Comput 1974; C-23: 90-93.

[8] Rao KR, Yip P. Discrete Cosine Transforms - Algorithms, Advantages, Application. Boston: Academic Press 1990.

[9] Jain AK. Fundamental of digital image processing. India: Princeton Hall of India 1997.

[10] Gonzalez RC, Woods P. Digital Image processing. 3rd ed. New Jersey: Pearson Education 2008.

[11] Okada M, Kawanishi I, Hama M, Harano T. Image compression processing device, image compression processing method, and image compression processing program. US Patent 20070110328, 2007.

[12] Miyuki O, Isao K, Mikiko H, Takeshi H. Image compression device, image compression method, and image compression program. JP Patent 199019, 2007.

[13] Saito M, Tsumura T. Image signal processing apparatus, camera system and image signal processing method. US Patent 20060232451, 2006.

[14] Saito M, Tsumura T. Image signal processing apparatus, camera system and image signal processing method. US Patent 20060233446, 2006.

[15] Jang S. Digital image processing apparatus, a method of controlling the same, and a digital image compression method. US Patent 20090103817, 2009.

[16] Koo JH. Image compression apparatus generating and using assistant images for deblocking processing and a method thereof. US Patent 20050238241, 2005.

[17] Ogawa K. Image compression device, image output device, image decompression device, printer, image processing device, copier, image compression method, image decompression method, image processing program, and storage medium storing the image processing program. US Patent 20050259876, 2005.

[18] Ogawa K. Image compression device, image output device, image decompression device, printer, image processing device, copier, image compression method, image decompression method, image processing program, and storage medium storing the image processing program. JP Patent 162382, 2003.

[19] Sekine M. Image-capturing apparatus, image processing system, control method, and storage medium for use with external printer with determination of operating state of external printer and corresponding adjustment of clock signal. US Patent 7483052, 2009.

[20] Sekine M. Image-capturing apparatus, image processing system, control method, and storage medium. JP Patent 046331, 2002.

[21] Roylance EA, Grohs RE. Method and system for image processing including mixed resolution, multi-channel color compression, transmission and decompression. US Patent 6941011, 2005.

[22] Roylance EA, Grohs RE. Method and system for image processing including mixed resolution, multi-channel color compression, transmission and decompression. US Patent 20040001622, 2004.

[23] Higuchi M, Asari S, Asada S. Compressed video data processing with conversion of image compression format. US Patent 6996331, 2006.

[24] Higuchi M, Asari S, Asada S. Compressed video data processing with conversion of image compression format. US Patent 20030091332, 2003.

[25] Martin UE. Compressed video data processing with conversion of image compression format. US Patent 20070092119, 2007.

Received: October 8, 2009 Revised: November 11, 2009 Accepted: December 4, 2009