

# VERIFICATION OF ROLE OF DATA SCANNING DIRECTION IN IMAGE COMPRESSION USING FUZZY COMPOSITION OPERATIONS

---

**Prashant Paikrao**

SGGSIET, Nanded, Maharashtra, (India).

E-mail: [plpaikrao@gmail.com](mailto:plpaikrao@gmail.com)

**Dharmapal Doye**

SGGSIET, Nanded, Maharashtra, (India).

**Milind Bhalerao**

SGGSIET, Nanded, Maharashtra, (India).

**Madhav Vaidya**

SGGSIET, Nanded, Maharashtra, (India).

**Reception:** 07/09/2022 **Acceptance:** 22/09/2022 **Publication:** 29/12/2022

## Suggested citation:

Paikrao, P., Doye, D., Bhalerao, M., and Vaidya, M. (2022). Verification of role of data scanning direction in image compression using fuzzy composition operations. *3C Tecnología. Glosas de innovación aplicadas a la pyme*, 11(2), 38-49. <https://doi.org/10.17993/3ctecno.2022.v11n2e42.38-49>



<https://doi.org/10.17993/3ctecno.2022.v11n2e42.38-49>

## ABSTRACT

*A digital image is a numerical representation of visual perception that can be manipulated according to specifications. In order to reduce the cost of storage and transmission, digital images are compressed. Depending upon the quality of reconstruction, compression methods are categorized as Lossy and Lossless compression. The lossless image compression techniques, where the exact recovery of data is possible, is the most challenging task considering the tradeoff between the compression ratio achieved and the quality of reconstruction. The inherent data redundancies like interpixel redundancy and coding redundancy in the image are exploited for this purpose. The interpixel redundancy is treated by decorrelation using Run-length Encoding, Predictive Coding, and other Transformation Coding techniques. While entropy-based coding can be achieved using Huffman codes, arithmetic codes, and the LZW algorithm, which eliminates the coding redundancy. During the implementation of these sequential coding algorithms, the direction used for data scanning plays an important role. A study of various image compression techniques using sequential coding schemes is presented in this paper. The experimentation on 100 gray-level images comprising 10 different classes is carried out to understand the effect of the direction of scanning of data on its compressibility. Depending upon this study the interrelation between the maximum length of the Run and compression achieved similarly the resultant number of Tuples and compression achieved is reported. Considering the fuzzy nature of these relations, fuzzy composition operations like max-min, min-max, and max-mean compositions are used for decision-making. In this way, a rational comment on which data scanning direction is suitable for a particular class of images is made in the conclusion.*

## KEYWORDS

*Image Compression, Data Scanning Direction, Sequential Coding, Fuzzy Composition.*

## 1. INTRODUCTION

The digital image is a function of brightness that corresponds to the intensity of pixels. This representation involves large data associated so that the requirements of storage space, computing power and the related communication bandwidth are very high. The technique involved is called image compression to minimize these requirements, so that information can be depicted in a reduced form (Gonzalez, 2004). The capacity of the compression technique to decrease the data size is called the compression ratio. To attain lossless and lossy compression respectively, the redundant and irrelevant data is removed (Holtz, 1993). The techniques of lossy compression have relatively higher compression ratios than that of lossless compression. Compression ratio and reconstructed image quality is always a tradeoff.

Nowadays, with the rise in mobile phone popularity, images are becoming an important record form. Image compression is required for storing and processing large numbers of such images. Depending upon the requirement of data preservation and accuracy reconstructed data quality, DC techniques can be divided into lossless and lossy compression. Compressing the data without sacrificing its originality is the main objective of lossless image compression, the reconstructed data is identical to original data in lossless compression, and it is suitable primarily for applications in compression of Text, medical imaging, law forensics, military imagery, satellite imaging, etc. In lossy compression the reconstructed data is an acceptable approximation of original data, here higher compression ratio can be achieved its applicable in compression of natural images, audio, video, etc. (Hosseini, 2012).

There is always a limit to the compression ratio that can be achieved in Lossless Compression (Rehman, 1952). According to Shannon, on the other hand, in lossless compression techniques, the measure of the amount of information content (Entropy) in the data that can be used to find the theoretical maximum compression ratio for lossless compression techniques, data can be compressed into as small as 10 percent of its actual size, and as the compression techniques require less-complex encoders and decoders as compared to lossless techniques.

The Shannon Entropy concept is explored in the paper to point out different possibilities to increase the compression ratio to its maximum extent. The paper discusses the different concepts related to compression techniques. One alternative to deal with the tradeoff between image quality and compression ratio is to opt for Near-Lossless compression, where difference between the original and reconstructed data is within user-specified amount called as maximum absolute distortion (MAD). This may be suitable for compression of medical images, hyper spectral images, videos, etc.

In addition to the storage space requirements and the overhead of processing time, all users on a specific network are suggested to minimize the size of the data and use the network resources optimally (Kavitha, 2016). Since compression is both time-effective and cost-effective, it helps to share network resources to enhance network performance.

## 2. MACHINE LEARNING METHODS AND FEATURE IMPORTANCE

In 1999, Holtz gave a review of lossless image compression techniques, saying, "Theories are usually the starting point of any new technology." There are some lossless compression methods explained, namely Shannon's theory, Huffman code, Lempel-Ziv (LZ) code and data trees for Self-Learning Autopsy. Hosseini submitted another review in 2012, which discussed many algorithms with their performances and applications. It includes Huffman algorithm, Run Length Encoding (RLE) algorithm, LZ algorithm, Arithmetic coding algorithm, JPEG and MPEG with their applications.

Image Compression deals with the ways in which the data and space needed to represent and store the digital image are reduced. The elimination of data redundancy may be one of the strategies for achieving compression. Human Visual System Redundancy is categorized into three categories: Spatial Redundancy, Spectral Redundancy, and Temporal Redundancy. Spatial redundancy, which is the correlation of neighborhood image pixels. Spectral redundancy is a correlation measure of an image between different color planes (Spectral Bands). The Temporal Redundancy deals with the correlation between a video's consecutive image frames.

In lossless data compression, the removal of data redundancy is the key process, and the data redundancy (Rd) is given by equation 2.1.

$$Rd = 1 - \left(\frac{1}{CR}\right) \quad (2.1)$$

Where,

CR (Compression Ratio) =  $n1/n2$ ,

$n1$ : Size of compressed data,  $n2$ : Size of uncompressed data

Types of data redundancy are as follows,

1. Coding Redundancy
2. Inter-pixel Redundancy
3. Psycho-visual Redundancy

## 2.1 CODING REDUNDANCY

A code is a system of symbols used for information representation. The pieces of information are represented by a combination of symbols called a codeword, and their length is called the number of symbols in a codeword. The available grey levels in an image are assigned to various codewords. If the grey levels are coded by using longer codewords than needed, an image is said to have coding redundancy. An image's gray-level histogram is created to construct codes with reduced coding redundancy. It is suggested that the most frequent grey levels should be represented by shorter codewords and vice versa to achieve the shortest representation, i.e., to avoid coding redundancy in the data. This variable length coding process may result in an image being shorter overall than the representation of the fixed length code. And when probability-based method(s) are used to design the code, it guarantees the shortest representation. The present coding redundancy is less than optimal, meaning codewords are used for representation, not as short as possible. Below is the average number of bits required to represent each pixel, given by equation 2.2 and 2.3.

$$L_{avg} = \sum_{n=1}^N l(g_n) \cdot p(g_n) \quad (2.2)$$

Where,

$n$  – Total number of pixels

$l(g_n)$  – Number of bits used to represent gray level of individual pixel

$p(g_n)$  – Probability of occurrence of gray level  $g_n$ .

$$p(g_n) = \frac{N}{n}, n = 1, 2, \dots, L \quad (2.3)$$

Where,

$N$  – Number of times the  $n$ th gray level appears in image

$L$  – Number of gray levels

Total number of bits: The total number of bits required to code an R X C image is given by equation 2.4,

$$L_{RXC} = R.C.L_{avg} \quad (2.4)$$

## 2.2 INTER-PIXEL REDUNDANCY

There is some connection between the pixel image data represented by the respective grey levels. It is possible to identify this inter-pixel relation of a particular pixel in terms of its neighborhood pixel. These inter-pixel relationships are responsible for the structural and geometric features of objects present in the image. In other words, by means of an Interpixel Redundancy, the intensity value of a particular pixel can be predicted by knowing the intensity value of its neighborhood pixels. An image's spatial resolution is directly proportional to the redundancy of the neighboring pixels. As this increases the likelihood of two adjacent pixels having the same intensity value, the spatial resolution of an image increases. By transforming the 2-D image matrix into a more efficient representation, this redundancy may be taken care of. Mapping is called the operation involved in this process. If it is possible to reconstruct the original image from the mapped data, the operation is called reversible mapping.

Autocorrelation Coefficient:

The autocorrelation coefficients can be computed using equation 2.5,

$$\gamma(\Delta n) = \frac{A(\Delta n)}{A(0)} \quad (2.5)$$

Where,

$A(\Delta n)$  – Scaling factor given by equation 2.6,

$$A(\Delta n) = \frac{1}{n-\Delta n} \sum_{y=0}^{n-1-\Delta n} f(x, y).f(x, y + \Delta n) \quad (2.6)$$

Where,

n - number of pixels on a line.

N- number of sum terms.

x - coordinate of the line used in the computation

## 2.3 PSYCHO-VISUAL REDUNDANCY

Some information is given more importance than others in normal visual processing and this is how the human visual system does not respond with equal sensitivity to all exposed information. This ignored data is viewed as redundant psycho-visual data. By means of a quantization method, this redundancy is minimized, but as quantization itself is a lossy process, we cannot precisely reconstruct the original image. The method of exploring psycho-visual redundancy is therefore classified under lossy data compression techniques.

## 3. VARIOUS COMPRESSION TECHNIQUES

Several ways can be employed to compress the images which are further divided into two major categories or compression techniques.

### 3.1 LOSSY COMPRESSION

Here, some of the finer details in an image are compromised and an approximate representation of an input image is the resulting image. The psycho-visual capacities of human eyes should be restricted by this degradation. JPEG (Joint Photographic Experts Group), MPEG (Moving Photographic Experts Group) and MP3 (MPEG Audio Layer 3) are some of the algorithms for lossy image compression.

### 3.2 LOSSLESS COMPRESSION

The objective of this method is to reduce the compressed representation bit rate without any distortion in the input image signal. The entire set of values is the same as the input. PNG (Portable Network Graphics), TIFF (Tagged Image File Format) and JBIG (Joint Bi-level Image Experts Group) are some lossless picture compression algorithms.

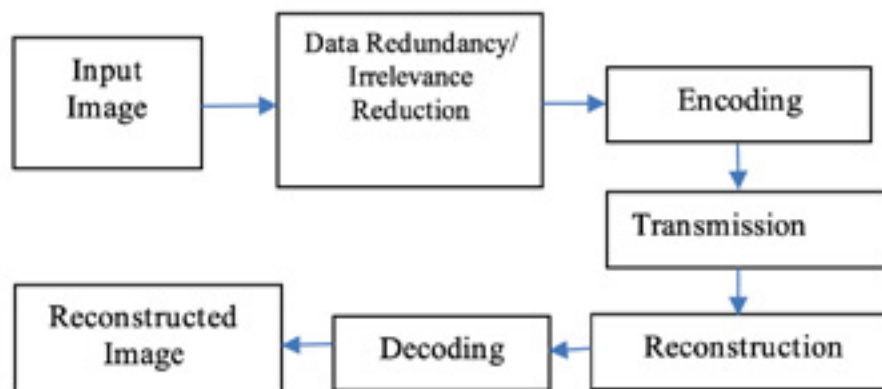


Fig. 1. Image compression model.

The modified representation of data compression is a kind of coding and therefore consists of two separate functional components; an encoder and a decoder, as shown in Fig. 1. The encoder encodes the data in a compressed form, while the decoder decodes it in a decompression form called the original form.

The image signal  $f(x, y)$  is used as input of the encoder. The encoder compresses the  $f(x, y)$  data to a  $g(r, c)$  compressed representation. This information may be stored locally or sent for a specific purpose to some remote location. This compressed representation is used as an input to the decoder whenever necessary, which decompresses this data to  $f'(x, y)$ . If  $f(x, y)$  and  $f'(x, y)$  are identical, the compression process used is otherwise referred to as a lossless compression.

## 4. SEQUENTIAL CODING SCHEMES

The lossy compression technique is used in certain image processing applications involving large image sizes, high resolutions such as satellite images and video processing to achieve a good compression ratio (Vaidya, 2017). Lossless image compression is used in other applications where every data component is important, such as medical imaging and document images. We're going to discuss a few of the techniques for lossless image processing here.

## 4.1 RUN LENGTH CODING (RLE)

Run length coding is based on sequential pixel redundancy. The number of adjacent pixels with the same grey levels are counted here, which is called Run of that grey level and the grey level snippet / tuple  $\{g_i, r_i\}$  and its run count are formed to represent the whole image. Since the run value can vary from a minimum of 1 pixel to a maximum of  $m \times n$  (total number of pixels in the image;  $m$ : number of rows,  $n$ : number of columns), this number will be very large and will require many bits to be represented. The highest size of a run is therefore considered to be multiple components in a row, i.e.,  $n$ . It is obvious that in the images where the size of runs will be maximum, such kind of coding will be applicable. The kind of images suitable for RLE are graphs, plots, line works, facsimile data, icons, and basic animation images.

Some researchers also perform the RLE on bit planes of an image and perform two-dimensional (2D) RLE.

The image is processed row wise in 1-D RLE as pixel arrays, its grey levels are regarded as source symbols, and its runs are considered to be computed. But in 2-D, some characteristic pixel blocks are treated as source symbols, and for encoding, runs of those symbols are considered.

## 4.2 ENTROPY CODING

The good coding with entropy encoding results in the information in the message being represented in just enough bits. If an unlikely symbol appears rather than the appearance of a probable symbol, more information will be obtained. The entropy of information, or Entropy of Shannon, is given by equation 4.1.

$$I = p_i \cdot \log_2 \left( \frac{1}{p_i} \right) \quad (4.1)$$

Where,

$P_i$ : Probability of occurrence of  $i^{\text{th}}$  symbol.

N: Number of symbols in a particular message/codebook.

The base2 of the logarithmic function gives an idea here that the symbol's occurrence is represented by means of 0 and 1.

Before encoding the image data using variable length codes that obtain no-prefix condition, the probabilities need to be known. So, these algorithms are generally two-pass and consume more time. Compression of data results from representing more likely symbols with shorter codewords and less likely symbols with longer codewords.

## 4.3 DICTIONARY-BASED ALGORITHMS

Symbol sub-strings are repeated in a file; these instances are recorded in a string table and referred to for encoding and decoding processes instead of repeating each time their position in the record. To prepare the dictionary, these string table techniques work on various approaches and LZW (Lempel-Ziv-Welch), LZ77 and LZ78 variants are created. In the data sequence, LZ77 uses a sliding window that generates a (position, length) tuple to point back to pre-existing substrings or symbols. LZ78 creates a string table dynamically and only replaces the substring in the message with the position index in the string table. Lossless coding systems are these coding schemes. Since some entries in the dictionary may not be referred to as frequently as others, the system is not optimal and the carrying of the dictionary in files is an overhead.

## 4.4 ARITHMETIC CODING

Arithmetic coding is a variable length coding scheme that performs better than Huffman coding with skewed probabilities for data with small alphabets (Langdon, 1984). Its implementation is a bit



complex, but it is possible to segregate the modelling and coding features of the compression technique. It has two forms, Adaptive Arithmetic Coding, which is a method of lossy coding, and Binary Arithmetic Coding, which is a technique of lossless coding. The real number line is split into smaller intervals corresponding to the probabilities of source symbols in this encoding. The first symbol that appears in the message then selects its corresponding interval, which is further divided into the same number of smaller proportional intervals. The successive symbol selects the corresponding interval from the message and this process of further splitting the smaller intervals continues until the last symbol in the message. This way the entire message is coded most efficiently according to the source symbol probability.

#### 4.5 HUFFMAN CODING

Huffman code is a well-known technique that is efficiently suited to almost all file formats (Huffman, 1952). This probability-based, variable length, optimally coded non-prefix code is used in lossless compression. For more likely codes, this minimum redundancy code uses shorter codes and vice-versa. Using a code tree, the symbols are encoded; the codeword starts from the root node and traverses until the leaf node, where the encoded symbol is placed. The codes of leaf nodes that are closer to the root node are shorter in length and more likely to occur in the given message. If the number of symbols and their probabilities are very close, then the achieved compression ratio is less and far from the entropy of the source. In this case, it is possible to create symbol pairs / group of larger size and to codify these meta-symbols using their joint probabilities. This is a two-pass algorithm that works faster if probabilities are previously known, providing excellent average codeword length otherwise the two-pass procedure is followed. Faller and Gallagher have proposed one-pass adaptive Huffman codes and Knuth and Vitter have further developed them. If we want to code the  $(i+1)$ <sup>th</sup> symbol using earlier  $i$ th symbol statistics, the two binary tree parameter procedure is proposed and later developed as adaptive Huffman Code.

### 5. SCANNING OF DATA

An image is a function of space  $f(x, y)$  that is two-dimensional (2D). For sequential data, the coding schemes discussed in the following section are appropriate. To convert the 2D images to one-dimensional (1D) sequential data, a scanning process should be used. Local and global redundancies are represented in an image. In order to deal with the coherence and correlation of image pixels (Vemuri, 2014), local redundancies need to be more explored. The pixels in any direction may be related to each other. The schemes are sensitive to the direction of scanning of the image which is nothing but the direction of scanning the pixels of an image. Horizontal, vertical, and diagonal directions of scanning can be achieved by padding row-after row, column-after-column and pixel intensity scanning in a zig-zag manner.

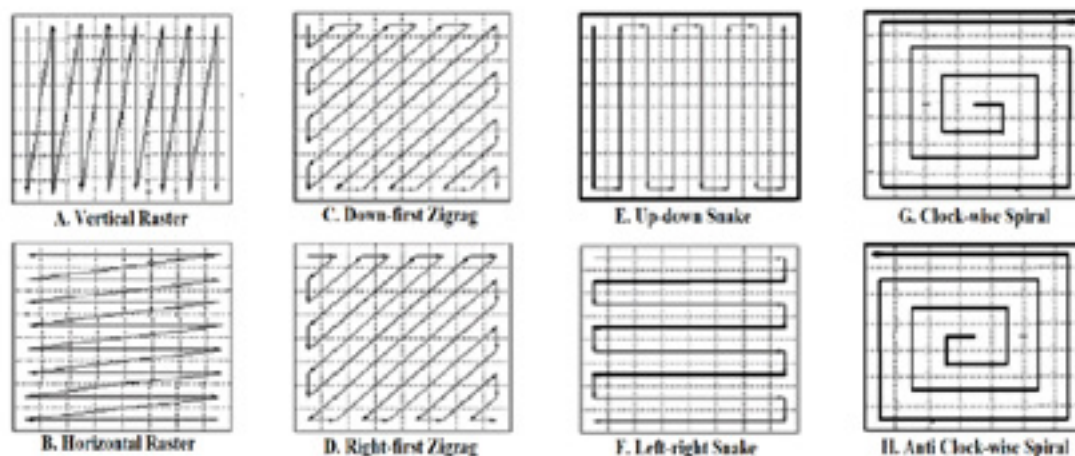


Fig. 2. Directions of Data Scanning.



In fig. 2 Directions of Data Scanning are shown, where A. is Vertical Raster direction, B. is Horizontal Raster direction, C. shows the Down-first Zigzag direction, D. shows the Right-first Zigzag direction, E. represents Up-down Snake, F. shows Left-right Snakes, G. is Clockwise Spiral, and H. is Anticlockwise Spiral. Which will be stored in a 1D vector representing image as well as suitable for below discussed coding schemes. Depending upon the nature of input image and the chosen direction of data scanning, variations may be observed in incident data redundancies. So, choosing this direction of scanning is a crucial task before feeding it to subsequent sequential compression algorithms mentioned in the next section.

In Table I, the effect of choosing different directions of data scanning on the interpixel redundancy is visible. Here, the preloaded image in Matlab ‘cameraman.tif’ is read, then its 2D intensity matrix is converted to 1D vector, by using mentioned data scanning methods one by one. Then the maximum, minimum Runs in the data, and number of resultant tuples are observed in each case. It may be seen that the Horizontal Scanning direction can be more suitable for this image over other methods as the Max. Run observed is longer and the number of tuples generated are less. Further it may be noted that there is certain variation in the compression ratios achieved by applying the Huffman’s encoding over the results of RLE algorithm, applied over different vectors resulted from the different scanning directions. This variation ranges from -37.66% to 60.40%, that means the CR achieved can approximately double of one direction scanning to the other. This underlines the importance of data scanning direction applied.

Table I. Effect of choosing different Direction of Scanning.

Name of Image	Direction of Scanning	Max. RUN	No. of resultant Tuples	CR using Huffman's code	Percentage variation in CR
cameraman.tif	Vertical Raster	13	56112	0.2920	--
	Horizontal Raster	15	55387	0.3323	13.89
	Down-first Zigzag	9	57548	0.1820	-37.66
	Right-first Zigzag	9	57550	0.2952	1.10
	Up-down Snake	15	55385	0.4684	60.40
	Left-right Snake	14	56110	0.2761	-5.44
	Clockwise Spiral	15	56104	0.3128	7.13
	Anticlockwise Spiral	15	56114	0.2921	0.02

The distribution of Runs using the mentioned data scanning directions is also shown in the Fig. 3. below.

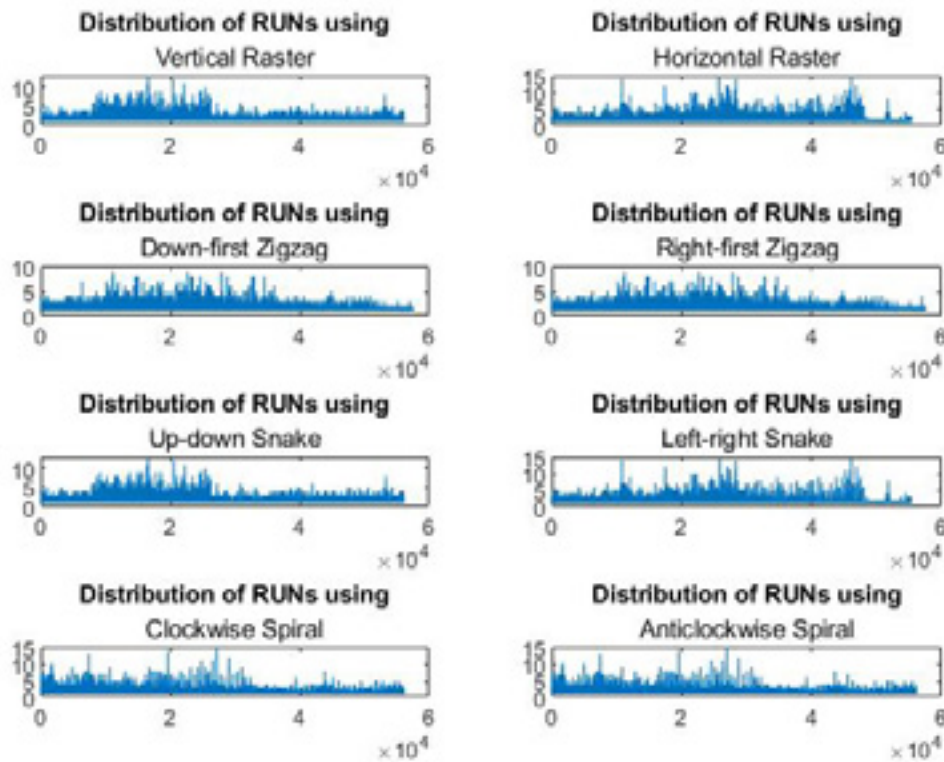


Fig 3. The distribution of Runs using various data scanning directions.

## 6. LENGTH OF THE RUN, NUMBER OF TUPLES AND CODABLE COMPRESSION ACHIEVED

Here inter-relation between Length of the Run and Number of Tuples generated in the compression procedures involving RLE is discussed. A situation where, when one thing increases, the other thing is supposed to decrease.

If the length of the Run achieved is higher, then the number of resulting Tuples will be less, and

If the length of the Run achieved is smaller, then the number of resulting Tuples will be more.

For achieving a good compression, the first situation should occur, that the maximum size Runs should be registered with smaller number of resultant Tuples. But, in this case, number of bits required to represent the Run count will grow in terms of 8-bits and the overall representation will require more data. On the other hand, if the Runs are smaller and they are representable in lesser number of bits, the number of Tuples generated will be high in number and to represent those greater number of Tuples again the data required will be more. So, to achieve good codable compression it is suggested that those parameters should be evaluated carefully. As there may not be any clear-cut range or threshold limits for length of Runs and the Tuple length. But a vague, imprecise boundary can be thought of which would separate the ‘Good’ length of Run and ‘Average’ number of Tuples from the results to achieve better codable compression. Considering this fuzzy nature of the variables it will be more significant if the methods of Fuzzy Composition are employed to form the Relation between them.

## 7. FUZZY COMPOSITION FOR FINDING THE RELATION BETWEEN VARIABLES

The experimentation is carried on 100 gray scale images of 256 X 256 size, involving 10 images each of 10 classes like Text, Person, Flower, Animal, Bird, Object, Vegetable, Pet, Graph and Diagram.

Table II. Result of choosing different direction of data scanning.

Sr. No.	Linearisation Direction	Min Run	Max Run	No. of Tuples
---------	-------------------------	---------	---------	---------------

1	Vertical Raster	1	1780	36321
2	Horizontal Raster	1	2282	36767
3	Down-first Zigzag	1	1297	39681
4	Right-first Zigzag	1	1297	39681
5	Up-down Snake	1	1806	36251
6	Left-right Snake	1	2294	36702
7	Clockwise Spiral	1	922	36043
8	Anticlockwise Spiral	1	938	36040

The length of the Run and the number of Tuples for the mentioned eight different data scanning directions are calculated. The inter-relation between the length of the maximum Run and compression achieved, similarly the resultant number of Tuples and compression achieved are formed. Composition of fuzzy relations is used to find interaction of fuzzy variables across different fuzzy relations (Ross, 2010). So, compositions of these relations are computed to find the most suitable direction for data scanning. The composition operations studied here are max-min, min-max, and max-mean. The results of applying them is shown in Table III.

Table III. Suggested direction of data scanning, considering Fuzzy Compositions.

Sr. No.	Name of Class	Run Length			No. of Tuples			Suggested Data Scanning Direction
		Min-max	Max-min	Max-mean	Min-max	Max-min	Max-mean	
1	Text	1089	8260	11669	17994	5931	18222	L-R Snake
2	Person	42	15	51	59515	37489	59682	Both zigzag
3	Flower	1661	12	5215	57382	23755	57809	L-R Snake
4	Animal	2584	14	9937	62639	12925	61170	Both Spiral
5	Bird	212	11	326	50581	25577	53817	R F zigzag
6	Object	4704	12	10686	56746	27120	55686	U-D Snake
7	Vegetable	239	37	2780	62499	27242	57826	R F zigzag
8	Pet	70	21	83	56202	43821	57572	A Spiral
9	Graph	2764	246	5665	13003	4742	17477	U-D Snake
10	Diagram	1457	23	5272	54948	8186	52815	L-R Snake

## 7. RESULTS AND DISCUSSION

The experimentation on 100 gray level images comprising of 10 different classes is carried out. Depending upon this study the inter-relation between the maximum length of the Run and compression achieved is studied. Similarly, relation between resultant number of Tuples and compression achieved is also underlined. The fuzzy compositions have application in this type of scenarios, where the inter-relation between various experiment parameters is vague.

## 8. CONCLUSIONS

There is difference between the compressibility of the resultant data vector, i.e., 1D equivalent of 2D input image signal, if different data scanning directions are employed. Estimating the most suitable data scanning direction for a particular image in view of compression should result in higher compression ratio. The scanning directions suggested for a particular dataset in this study are very

specific about these datasets. Generally, a spiral direction should be suitable for the images having object at the center. The raster and snake scanning directions are suitable for document images. The Zigzag scanning directions are suitable for the images in which the information is scattered overspread.

## REFERENCES

- [1] D.A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes.", (1952), pp. 1098–1102.
- [2] G. G. Langdon, "An Introduction to Arithmetic Coding." IBM Journal of Research and Development", (1984), 28(2): pp. 135–149.
- [3] Holtz, Klaus. "The evolution of lossless data compression techniques" In Proceedings of WESCON'93, (1993), IEEE, pp. 140-145.
- [4] M. Hosseini, "A survey of data compression algorithms and their applications", (2012), Network Systems Laboratory, School of Computing Science, Simon Fraser University, BC, Canada.
- [5] M. Rehman, M. Sharif, M. Raza, "Image Compression: a survey", (2014), Res. I. Appl. Sci. Eng. Technol. 7, pp. 656-672.
- [6] M. V. Vaidya, Y. V. Joshi, and M. V. Bhalerao. "Marathi numeral identification system in Devanagari script using discrete cosine transform.", (2017), IJIES 10, no. 6.
- [7] M. V. Vaidya, Y. V. Joshi. "Marathi numeral recognition using statistical distribution features.", (2015), International Conference on Information Processing (ICIP), pp. 586-591.
- [8] M. V. Bhalerao, S. V. Bonde, A. V. Nandedkar, and S. D. Pilawan. "Combined classifier approach for offline handwritten Devanagari character recognition using multiple features.", (2018) In Computational Vision and Bio Inspired Computing, pp. 45-54. Springer, Cham.
- [9] P. Kavitha, "A Survey on Lossless and Lossy Data Compression Methods", (2016), 7: pp. 110-114, 2016.
- [10] Vemuri, BABA C., S. Sahni, F. Chen, C. Kapoor, C. Leonard, and J. Fitzsimmons. "Lossless image compression.", (2014), Igarss 2014 45, no. 1, pp: 78-86.
- [11] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", (2004), 2<sup>nd</sup> edition, Prentice Hall.
- [12] T. J. Ross, "Fuzzy Logic with Engineering Applications", (2010), 3<sup>rd</sup> edition, John Wiley and Sons.