Abstract

Data compression is a technique to represent data using less number of bits than original data. There are various data compression techniques available, but still there is a need to achieve more compression ratio. This paper proposes an algorithm that combines the features of both Huffman’s algorithm and LZW algorithm to achieve more compression ratio. This algorithm is named as VJ Zip. In the new algorithm VJ Zip, for compression, firstly every duplicate occurrence of data is replaced with the pointer to its previous occurrence to obtain partially compressed data. From this partially compressed data, the literals and pointers are further compressed using two separate Huffman trees. We measure the performance of this new algorithm in terms of compression ratio and also compare the performance of this new modified algorithm with the two algorithms viz., Huffman’s algorithm and LZW algorithm,
individually. Comparing the results it is inferred that new modified algorithm, VJ Zip, is more efficient than Huffman's algorithm and LZW algorithm applied individually. On an Average, it achieves 26% & 54% more compression ratio for .txt and .xml format respectively, as compared to Huffman's algorithm and 16% & 18% more compression ratio for .txt and .xml format respectively, as compared to LZW. Also this paper compares the performance of new algorithm with the existing software 7Zip. As compared to 7Zip new modified algorithm gives almost same compression ratios for text format while achieves 1% more compression ratio for images and videos.

References

- Khalid Sayood, "Introduction to Data Compression", 3rd edition.

Index Terms

Computer Science

Algorithms

Keywords

Data compression decompression compression ratio efficiency encoding decoding. (Keywords)