

THE COMPRESSIBILITY AND THE RANDOMNESS OF
COMPRESSED DATA BASED ON FIBONACCI CODE:
A NOVEL APPROACH

BY

KAMAL AHMED MULHI AL-KHAYYAT

A thesis submitted in fulfilment of the requirement for the
degree of Doctor of Philosophy in Computer Science

Kulliyyah of Information and Communication Technology
International Islamic University Malaysia

DECEMBER 2021

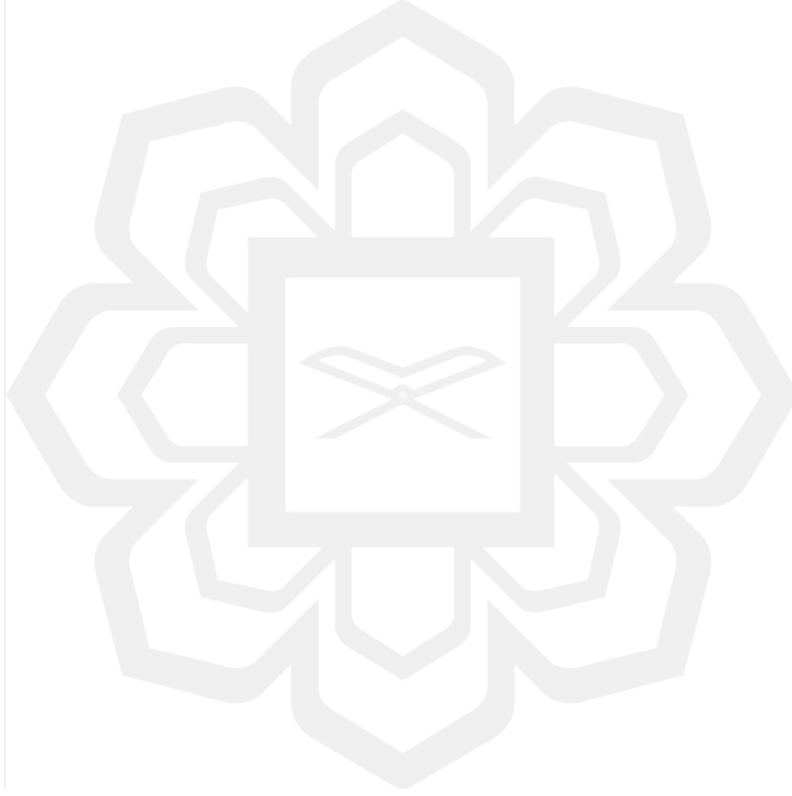
ABSTRACT

The tremendous growth of data generated daily has made the science of data compression an important and renewable field. It has become the first way to reduce the volume of data to optimize the use of storage units and accelerate the process of transferring data across various types of networks, chiefly the World Wide Web, thus reducing the cost of transport and storage. Compressed data grows with the same frequency as the data itself, which, in turn, created an urgent necessity to understand and analyze the compressed files themselves, and since efforts are focused only on inventing and developing new compression algorithms, few efforts remain trying to understand and analyze compressed files. This research invests in compressed files introducing a new way to analyze and understand compressed data from new angles. This analysis contributes to solutions to practical problems, including the problem of servers in classifying files before actually compressing them with what is known as compressibility. The issue of studying compressibility in systems servers is a sensitive and important issue, given that they provide for the optimum utilization of the physical and programmatic server resources. This research presented a new method by which server systems can distinguish between compressed files from uncompressed files on the one hand, and on the other hand, distinguish between compressed files that need more compression and those that do not need all of this in one frame. Moreover, as the randomness study programs cannot distinguish compressed data from uncompressed data in most cases, this study provided an integrated package of methods for studying the randomness of compressed files called (RTCD). This package can analyze the randomness of compressed files from new practical angles and open the way for the ability to compare compressed files with each other and distinguish between them successfully. This package includes quantitative and graphical measures all set to be standard in practice. The analysis in this study relies on the use of the Fibonacci code as a strong analytical basis capable of knowing the common characteristics of compressed files and can thus distinguish them from uncompressed files successfully. Moreover, the difference in these characteristics within the compressed files circle enables one to know the files that still need more compression. Compared to the well-known techniques that study compressibility and those that study randomness of data, this analysis shows its distinction and its ability to overcome the deficiencies of these methods.

ملخص البحث

جعل ان النمو الهائل للبيانات المتولدة على اساس يومي علم ضغط البيانات مجالا هاما ومتجددا حيث اصبح الوسيلة الأولى في تقليل حجم البيانات من أجل تحسين استخدام وحدات التخزين وتسريع عملية نقل البيانات عبر مختلف انواع الشبكات وعلى راسها الشبكة العنكبوتية مما خفض كلفة النقل والتخزين. وهكذا نمت الملفات المضغوطة بنفس وتيرة البيانات نفسها , وهذا بدوره ولد ضرورة ملحة لفهم وتحليل الملفات المضغوطة نفسها . وحيث ان الجهود متركزة فقط على اختراع وتطوير خوارزميات ضغط جديدة , تبقى قلة من الجهود من تحاول فهم وتحليل الملفات المضغوطة والسبب في ذلك أن البيانات المضغوطة عبارة عن بيانات معقدة التركيب ويصعب التنبأ بسلوكها ولذلك تعتبر عشوائية بشكل عام . ان هذا البحث يستثمر في الملفات المضغوطة ويقدم طريقة جديدة لتحليل وفهم البيانات المضغوطة من زوايا جديدة . يساهم هذا التحليل في حل مشاكل عملية من ذلك مشكلة السرفرات في تصنيف الملفات قبل ضغطها فعليا بما يعرف بقابلية الضغط (compressibility) . ان مسالة دراسة قابلية الضغط في انظمة السرفرات هي مسالة حساسة وهامة لما توفرة من استغلال امثل لمصادر السرفرات المادية والبرمجية . قدم هذا البحث طريقة جديدة تستطيع بها أنظمة السرفرات على سبيل المثال من التمييز بين الملفات المضغوطة من الملفات غير المضغوطة من جهة ومن جهة اخرى ميز بين الملفات المضغوطة المحتاجة لمزيد من الضغط وتلك التي لا تحتاج كل ذلك في اطار واحد . علاوة على ذلك , وحيث أن البيانات المضغوطة لا تستطيع برامج دراسة العشوائية من التمييز بينها وبين البيانات الغير مضغوطة في معظم الاحوال , فان هذه الدراسة قدمت حزمة متكاملة من طرق دراسة العشوائية للملفات المضغوطة سميت (RTCD) . هذه الحزمة تستطيع أن تحلل عشوائية الملفات المضغوطة من زوايا عملية جديدة وتفتح المجال للقدرة على مقارنة الملفات المضغوطة مع بعضها البعض والتمييز بينها وبين الملفات الغير مضغوطة بنجاح . ان هذه الحزمة تشتمل

على مقاييس كمية واخرى بيانية كلها موضوعة لتكون قياسية من الناحية العملية . يعتمد التحليل في هذه الدراسة على استخدام كود الفيوناتشي كاساس تحليلي قوي قادر على معرفة الخصائص المشتركة للملفات المضغوطة ويستطيع بذلك تمييزها من الملفات الغير مضغوطة بنجاح وعلاوة على ذلك فان الاختلاف في هذه الخصائص داخل دائرة الملفات المضغوطة يمكن من معرفة الملفات التي لا تزال بحاجة الى مزيد من الضغط . بالمقارنة مع التقنيات المعروفة التي تدرس قابلية الضغط وتلك التي تدرس عشوائية البيانات يتبن مقدار تميز هذا التحليل وقدرته على تجاوز النقص في هذه الطرق.



APPROVAL PAGE

The thesis of Kamal Ahmed Mulhi Al-Khayyat has been approved by the following:

Imad Fakhri Al-Shaikhli
Supervisor

Raini Binti Hassan
Co-supervisor

Abdul Wahab Abdul Rahman
Internal Examiner

Shamala K Subramaniam
External Examiner

Hassanuddeen Abdul Aziz
Chairman

DECLARATION

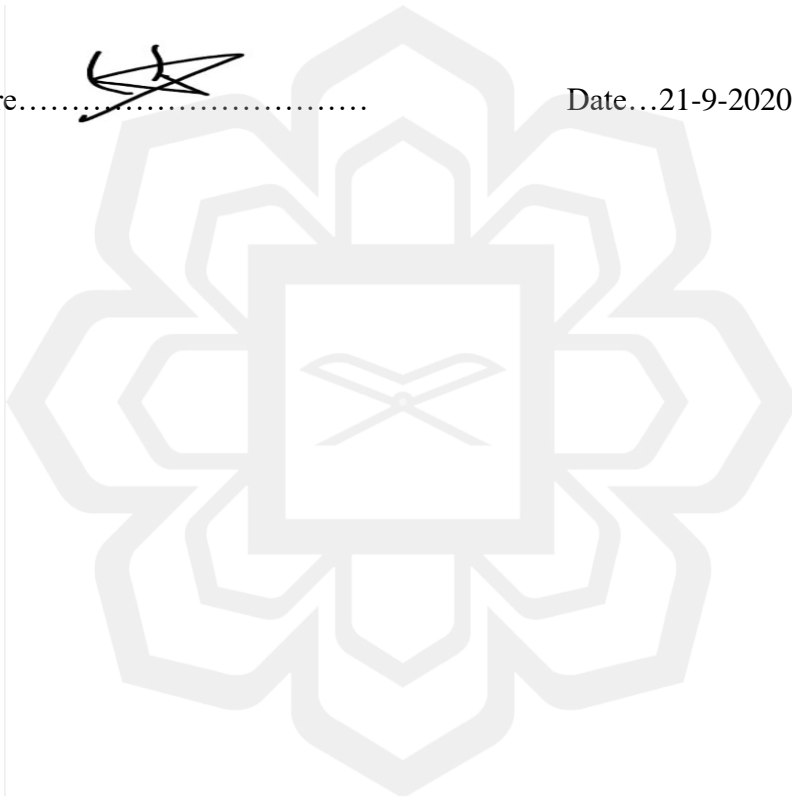
I hereby declare that this thesis is the result of my investigations, except where otherwise stated. I also declare that it has not been previously or concurrently submitted as a whole for any other degrees at IIUM or other institutions.

Kamal Ahmed Mulhi Al-Khayyat

Signature.....



Date...21-9-2020.....



INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

**DECLARATION OF COPYRIGHT AND AFFIRMATION OF
FAIR USE OF UNPUBLISHED RESEARCH**

**THE COMPRESSIBILITY AND THE RANDOMNESS OF
COMPRESSED DATA BASED ON FIBONACCI CODE:
A NOVEL APPROACH**

I declare that the copyright holder of this thesis/dissertation are jointly owned by the student and IIUM.

Copyright © 2021 Kamal Ahmed Mulhi Al-Khayyat and International Islamic University Malaysia. All rights reserved.

No part of this unpublished research may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without prior written permission of the copyright holder except as provided below

1. Any material contained in or derived from this unpublished research may only be used by others in their writing with due acknowledgement.
2. IIUM or its library will have the right to make and transmit copies (print or electronic) for institutional and academic purpose.
3. The IIUM library will have the right to make, store in a retrieval system and supply copies of this unpublished research if requested by other universities and research libraries.

By signing this form, I acknowledged that I have read and understand the IIUM Intellectual Property Right and Commercialization policy.

Affirmed by Kamal Ahmed Mulhi Al-Khayyat



.....

Signature

.....21-9-2020.....

Date

DEDICATION

I dedicate this thesis to my dear parents, without whom I would not have reached this stage, to my wife, who was an example of patience and sacrifice, and to my children Mustafa, Fatima, Abdel-Kareem, and Abdel-Hameed, who were indeed a treasured gift from Allah.

ACKNOWLEDGEMENTS

All glory is due to Allah, the Almighty, whose Grace and Mercies have been with me throughout my program. Although it has been tasking, His Mercies and Blessings on me ease the herculean task of completing this thesis.

I am most indebted to my supervisor, Prof. Dr. Imad Fakhri Al-Shaikhli, whose enduring disposition, kindness, promptitude, thoroughness, and friendship have facilitated the successful completion of my work. I put on record and appreciate his detailed comments, useful suggestions, and inspiring queries which have considerably improved this thesis. His brilliant grasp of the aim and content of this work led to his insightful comments, suggestions, and queries which helped me a great deal. Despite his commitments, he took the time to listen and attend to me whenever requested. The moral support he extended to me is in no doubt a boost that helped in building and writing the draft of this research work. I am also grateful to my co-supervisor, Dr. Raini Binti Hassan whose support and cooperation contributed to the outcome of this work.

Lastly, my gratitude goes to my beloved wife and lovely children; for their prayers, understanding, and endurance while away.

Once again, we glorify Allah for His endless mercy on us one of which is enabling us to successfully round off the efforts of writing this thesis. Alhamdulillah

TABLE OF CONTENTS

Abstract.....	ii
Abstract in Arabic.....	iii
Approval Page.....	v
Declaration.....	vi
Dedication.....	viii
Acknowledgements.....	xi
Table of Contents.....	xii
List of Tables.....	xiii
List of Figures.....	xiv
CHAPTER ONE: INTRODUCTION	1
1.1 Background of the Study.....	1
1.2 Problem Statement.....	4
1.3 Research Questions.....	5
1.4 Research Objectives.....	6
1.5 The Importance of the Study.....	6
1.6 The Scope of the Study.....	8
1.7 The Organization of the Thesis.....	9
CHAPTER TWO: LITERATURE REVIEW.....	11
2.1 Introduction.....	11
2.2 Compressibility Tests.....	11
2.3 Fibonacci Code.....	17
2.4 JPEG-LS Image Compressor.....	21
2.4.1 Enhancing Jpeg-ls by Pre-processing Technique.....	25
2.4.2 Enhancing jpeg-ls by Modifying Thea algorithm.....	26
2.5 The Randomness of Compressed Data.....	27
2.6 Attempts to Re-Compress Compressed Data.....	29
2.7 Some Works on Increasing the Compression Ratio of the Compression Algorithms.....	30
2.8 Non-Parametric Randomness Tests.....	32
3.8.1 The Difference-sign Test.....	33
3.8.2 The Turning Point Test.....	33
3.8.3 Cox-Stuart Test.....	34
2.9 Fibonacci Code.....	34
3.9.1 Fraenkel and Klein Codes.....	36
2.10 JPEG-LS Algorithm.....	38
2.10.1 LOCO-I Algorithm.....	39
2.10.2 Context Modelling.....	40
2.10.3 Golomb-Rice Encoding.....	41
2.11 Entropy and Compression Ratio Measurements.....	43
2.12 Summary	44

CHAPTER THREE: INTRODUCTION.....	45
3.2 Stage 1: The Randomness of Compressed Data.....	46
3.2.1 Study the Randomness of Compressed Data.....	46
3.2.2 Alternative Method of Analyzing Compressed Data.....	51
3.2.3 Analysis Compressed Data Based on Predefined patterns.....	52
3.3 Stage 2: Fixed Reference Patterns To Analyze Compressed Data.....	53
3.3.1 Predefined Variable patterns: Universal Coding.....	53
3.3.1.1 The problem of creating a smaller version.....	55
3.3.1.2 Flagging Problem.....	56
3.3.1.3 The problem of increasing the size by flags number.	58
3.3.1.4 Summary of the criteria behind choosing universal coding.....	59
3.3.2 Nominating Fibonacci Coding.....	60
3.4 Stage 3: Using Fibonacci Code In Randomness And Transformation...	63
3.4.1 Analysis Compressed Data Based on Fibonacci Code.....	64
3.4.2 Transforming Compressed Data Based on Fibonacci Coding..	65
3.4.2.1 Extraction Algorithm.....	66
3.4.2.2 Maintaining the Original Size.....	69
3.4.2.3 Example of using The Suggested Transformation....	77
3.4.3 The Statistical Randomness of Compressed Data Based on Fibonacci Coding.....	78
3.4.3.1 Checking the Optimality of Fibonacci Codewords (First Condition).....	79
3.4.3.2 Steps of Computing the Violations in Fibonacci Codewords.....	80
3.4.3.3 Overhead of Swapping.....	82
3.4.3.4 Missing Fibonacci Words (second condition).....	85
3.4.3.5 Getting the missing words.....	89
3.4.3.6 Utilizing Missing and Combined Words.....	90
3.4.3.7 Summary of the possible exploiting optimality concept for reducing the size.....	91
3.4.3.8 Randomness Based on The Graph of The Fibonacci Sequence.....	91
3.5 Stage 4: Using Fibonacci Code for The Compressibility.....	94
3.5.1 The Compressibility of Compressed Data Based on Fibonacci Coding.....	94
3.5.1.1 Detecting the Long Fibonacci Words in Compressible Files.....	95
3.5.1.2 The Effect of Long Fibonacci Words on Compressibility.....	96
3.5.1.3 Comparing the Fibonacci Method with The Existing Methods.....	98
3.6 Stage 5: Using JPEG-LS as a Case Study.....	98
3.6.1 JPEG-LS.....	99
3.6.1.1 Compressibility of JPEG-LS Based on Fibonacci Code.....	101
3.6.1.2 JPEG-LS with Long Fibonacci Words.....	102
3.6.1.3 Inside JPEG-LS Algorithm.....	106
3.6.1.4 Randomness of JPEG-LS Based on Fibonacci Code	108

3.7 Summary.....	110
CHAPTER FOUR: TESTING THE RANDOMNESS OF COMPRESSED DATA.....	112
4.1 Introduction.....	112
4.2 Experimental Setup.....	112
4.2.1 Compression Algorithms Used.....	113
4.2.2 Image Dataset Used.....	113
4.2.3 Randomness Tests Used.....	114
4.3 Experimental Results.....	115
4.3.1 Randomness and Compression Ratio.....	116
4.3.2 Use of Statistical Information Provided by Randomness Tests.....	117
4.3.2.1 Correlations Between Ratios and Randomness.....	119
4.3.2.2 Correlations Between Ratios and Compression Ratio.....	121
4.3.2.3 Statistical Information and Compression Ratio.....	122
4.3.2.4 Using Statistical Ratio as A Replacement for Randomness Tests.....	126
4.4 Summary.....	129
CHAPTER FIVE: FIBONACCI ANALYSIS.....	131
5.1 Introduction.....	131
5.2 Experimental Setup.....	131
5.2.1 Data Samples and Compressor Algorithms Used.....	132
5.2.2 Steps of the Experiments.....	132
5.3 Experimental Result.....	134
5.3.1 The Curves of JPEG-LS.....	135
5.3.2 The Curves of BZIP2.....	138
5.3.3 The Curves of JPEG-2000.....	138
5.3.4 Summary of The Common Features of The Compressed Data Curves.....	142
5.3.5 The Curves of Non-Compressed Data.....	142
5.4 Summary.....	145
CHAPTER SIX: CASE STUDY: COMPRESSIBILITY OF JPEG-LS.....	146
6.1 Introduction.....	146
6.2 Description of the Experiments.....	146
6.3 The Compressibility of Group1.....	148
6.3.1 The Fibonacci Histogram of Group1 Images.....	150
6.3.2 The Second Compression of Group1 Images.....	153
6.4 The Compressibility of Group2 Images.....	154
6.4.1 The Fibonacci Histogram of Group2 Images.....	155
6.4.2 The Second Compression of Group2 Images.....	156
6.5 The Compressibility of Group3 Images.....	159
6.5.1 6.4.1 The Fibonacci Histogram of Group3 Images.....	159
6.5.2 The Second Compression of Group3 Images.....	163
6.6 General Notes on Pixelated Images.....	165
6.7 Second Compression Using General-Purpose Compressors.....	165
6.8 Comparing the Second Compression for Jpeg-Ls, Png And Jpeg-2000..	167

6.9 Comparing Between the Double Compression and the Lossy Compression.....	167
6.10 The Frequency of Bits.....	171
6.11 Compare the Compressibility Based on Fibonacci with other Approaches.....	173
7.11.1 Prefix Estimation.....	174
7.11.2 Entropy and Unique Alphabets.....	175
6.12 Inside Jpeg-Ls Algorithm.....	178
6.13 Summary.....	180

CHAPTER SEVEN: PROPOSED RANDOMNESS TESTS FOR COMPRESSED DATA (RTCD)..... 182

7.1 Introduction.....	182
7.2 Statistical Randomness Tests Based on Fibonacci Code.....	182
7.2.1 Violation in Frequency.....	182
7.2.1.1 Violation of Group1.....	184
7.2.1.2 The Violation of Group 2.....	185
7.2.1.3 The Violation of Group3.....	187
7.2.1.4 Comparing the Three Groups.....	188
7.2.2 Missing Fibonacci Words.....	189
8.2.3 Randomness Based on The Number of Fibonacci Words.....	193
7.3 Randomness Based on Graph.....	195
7.3.1 The Curves of Group1.....	195
7.3.2 The Curves of Group2.....	196
7.3.3 The Curves of Group3.....	202
7.4 All the Suggested Randomness Tests (RTCD).....	205
7.5 Comparing (RTCD) with NIST.....	205
7.6 Summary.....	208

CHAPTER EIGHT: CONCLUSION AND FUTURE WORKS..... 209

8.1 Introduction.....	209
8.2 Research Objectives.....	209
8.3 Summarising the Study.....	209
8.3.1 Randomness Tests and Compressed Data.....	210
8.3.2 Alternative Method to Analyse the Compressed Data.....	211
8.3.3 Fibonacci Code as Reference Patterns.....	211
8.3.4 Compressibility based on the Fibonacci code.....	212
8.3.5 Randomness Based on Fibonacci Code.....	213
8.4 Promising Fibonacci Analysis Possibilities.....	213
8.4.1 Compressibility Test.....	214
8.4.2 Selecting the Best Compression Algorithm.....	215
8.4.3 Compression Ratio Estimation.....	218
8.4.4 Randomness of Compressed Data.....	220
8.5 Limitations.....	221
8.6 Future Work.....	221

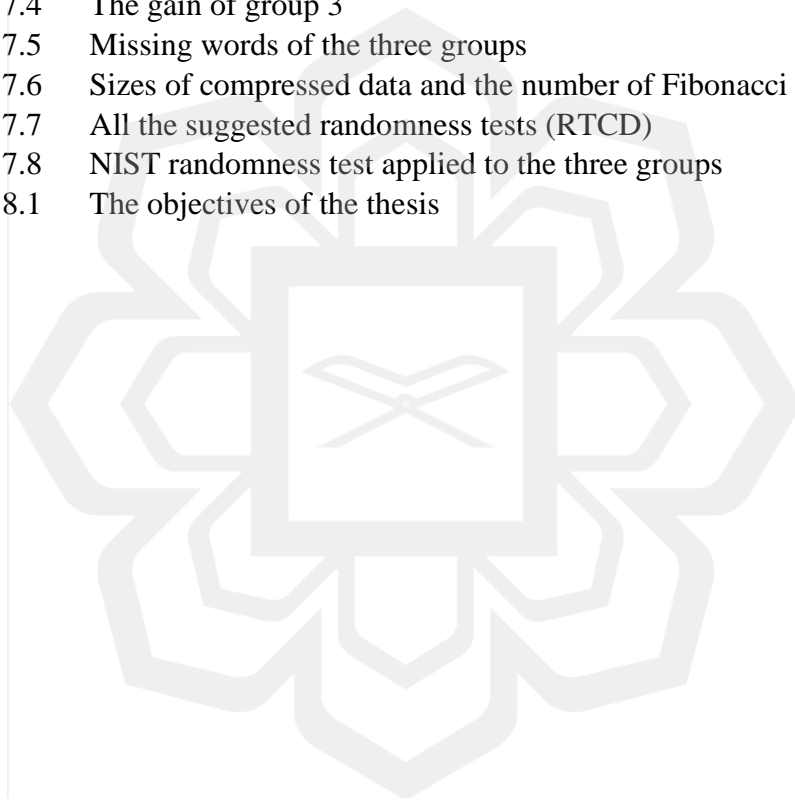
REFERENCES..... 223

LIST OF PUBLICATIONS..... 234

LIST OF TABLES

Table 2.1	The methods used in the literature for compressibility test	18
Table 2.2	Pros and cons of current compressibility tests	19
Table 2.3	Some literature reviews on prediction coding	23
Table 3.1	Famous universal coding	54
Table 3.2	The first words of five of the universal coding systems	55
Table 3.3	Number of variations for the first six Fibonacci words' length	62
Table 3.4	Number of variations for the first codewords of C2	63
Table 3.5	The difference between the curves of compressed and non-compressed data	65
Table 3.6	Fibonacci codes, C1, and C2, and the flipped version of C2	70
Table 3.7	The conversion from C ² into C ¹	72
Table 3.9	The original size of JPEG-LS files and the total of part1 and compressed part2	77
Table 3.10	The number of Fibonacci words and their lengths	86
Table 3.11	The peak length and the first missing words.	87
Table 3.12	An example of how to find the exact missing words	89
Table 3.13	Different context orders used by a predictor-based compressor	97
Table 3.14	Comparison between JPEG-LS, PNG, and JPEG-2000	100
Table 3.15	Golomb-Rice coding integer values	106
Table 3.16	Pixelated images with k and error statistics of JPEG-LS compressor.	107
Table 3.17	non-pixelated images with k and error statistics of JPEG-LS	108
Table 4.1	Correlation between randomness and compression ratio	117
Table 4.2	Correlation between the ratios and the randomness	119
Table 4.3	Correlation between ratios and compressed ratios	122
Table 4.4	Correlation between statistical information and compression ratio	124
Table 4.5	The sizes of compressed data with the number of turning points	125
Table 4.6	The range of ratios that match the randomness tests	128
Table 5.1	The sizes of ten images before and after compression	133
Table 5.2	Shows the missing words before and after the peak length for “#1.jls”	136
Table 6.1	The percentage of Fibonacci words longer than 50	150
Table 6.2	The First and the second Compression of the group1 images	153
Table 6.3	The percentage of Fibonacci words longer than 50	155
Table 6.4	The first and second compression for group2.	158
Table 6.5	The percentage of Fibonacci words longer than 50	163
Table 6.6	The result of the first and the second compression of group3	164

Table 6.7	The second compression using general-purpose compressors	166
Table 6.8	Comparison between the second compression of JPEG-LS, PNG, and JPEG-2000	168
Table 6.9	Comparing the second compression with the lossy jpeg.	169
Table 6.10	Bit Frequency of group1, group2, and group3	171
Table 6.11	Prefix estimation of the three groups	176
Table 6.12	The entropy and the unique alphabets of group2	177
Table 6.13	The k parameter of group3	179
Table 6.14	The k parameter of group1	179
Table 6.15	Analysis the errors when $k = 0$ for group1	180
Table 7.1	Frequency of the first words of Fibonacci words	183
Table 7.2	The gain of group1	185
Table 7.3	The gain of group 2	186
Table 7.4	The gain of group 3	188
Table 7.5	Missing words of the three groups	192
Table 7.6	Sizes of compressed data and the number of Fibonacci words	194
Table 7.7	All the suggested randomness tests (RTCD)	206
Table 7.8	NIST randomness test applied to the three groups	207
Table 8.1	The objectives of the thesis	210

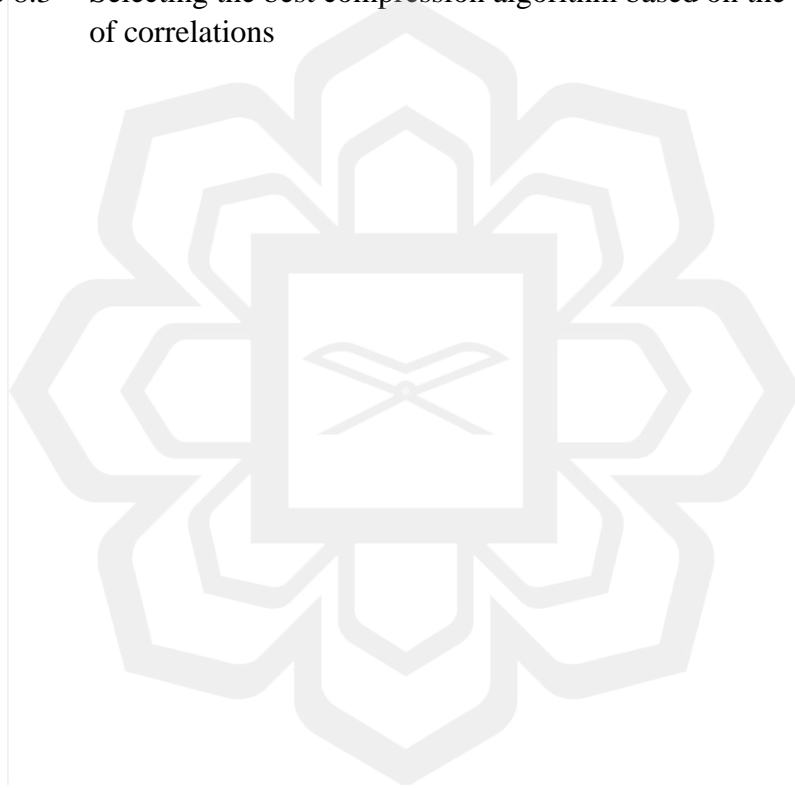


LIST OF FIGURES

Figure 1.1	Shows the position of compressibility test in server systems.	3
Figure 2.1	Lengths of Fibonacci code compared to some other codes	20
Figure 2.2	Lengths of Fibonacci code compared to some other coding	36
Figure 2.3	Frankel and Klein's codes for the first ten integers.	38
Figure 2.4	The Main components of the LOCO-I algorithm.	38
Figure 2.5	Block diagram of jpeg-ls	39
Figure 2.6	The neighboring pixels around the current pixel.	40
Figure 2.7	Golomb-Rice code examples	42
Figure 3.1	Research Methodology	47
Figure 3.2	The problem of extracting most of the compressed data	56
Figure 3.3	The process of separating universal coding	57
Figure 3.4	The remaining part (part1) only are zeros (or ones)	58
Figure 3.5	The problem of increasing the total size of both parts by flags	59
Figure 3.6	Summarization of the ideas regarding choosing the qualified reference patterns	61
Figure 3.7	Transformation of the extracted Fibonacci code to reduce the size	66
Figure 3.8	Splitting (extracting) compressed data file into two parts	67
Figure 3.9	The algorithm flowchart of splitting a random file into two parts	68
Figure 3.10	The algorithm that converts each word from the form C2 into C1	71
Figure 3.11	An example of converting C2 into C1	72
Figure 3.12	Huffman tree for the symbols, A, B, and C	73
Figure 3.13	Example of converting patterns into A, B, and C	74
Figure 3.14	The process of dropping '1' from '001' and getting 'A,' 'B,' and 'C.'	76
Figure 3.15	Example of the possible use of the suggested transformation	78
Figure 3.16	Step 2 is to store all possible violations in Fibonacci frequencies	83
Figure 3.17	Remove repetition based on shorter words (step 3).	84
Figure 3.18	An example of recording the violation in frequencies without repetition	85
Figure 3.19	The maximum length of full Fibonacci words	86
Figure 3.20	The algorithm of computing the number of missing words	88
Figure 3.21	Multiple peaks	93
Figure 3.22	Flat areas on the curve reveal the non-randomness	94
Figure 3.23	The histogram of Fibonacci words' lengths (A) existing of long Fibonacci words (B) no long Fibonacci words have existed.	95

Figure 3.24	(A) image number 49 (B) the upper left part of the image	101
Figure 3.25	The histogram of the image of Figure 3. 24B	102
Figure 3.26	An example of the image that is ideal for recompression	104
Figure 3.27	The regular mode and the run mode	105
Figure 3.28	The proposed Randomness tests package (RTCD)	109
Figure 4.1	Eight samples of the test images	114
Figure 4.2	Number of files that passed the tests	116
Figure 4.3	Relation between the randomness and the ratios.	120
Figure 4.4	The relation between the statistical ratios and the compression ratio.	121
Figure 4.5	The relation between the size of the output and the statistical information	123
Figure 4.6	The relationship between the compressed data and the statistical information.	125
Figure 4.7	Randomness test and the statistical ratio for JPEG-LS	128
Figure 4.8	Turning points test for 7z compressed data with ratios	129
Figure 4.9	Turning points test for jpeg-2000 compressed data with ratios	130
Figure 5.1	The steps of the experiments	132
Figure 5.2	The sample images used, all in PGM format	134
Figure 5.3	The Curve of “#1.jls”	136
Figure 5.4	The curves of “#2.jls” to “#10.jls”	137
Figure 5.5	The curve of “#1.bz2”	139
Figure 5.6	The curve of “#1.jp2”	139
Figure 5.7	The curves of “#2.bz2” to “#10.bz2”	140
Figure 5.8	The curves of “#2.jp2” to “#10.jp2”	141
Figure 5.9	The curve of the “#1.pgm” file	143
Figure 5.10	The curves ‘#2.pgm’ to ‘#10.pgm’	144
Figure 6.1	Compressibility of JPEG-LS	147
Figure 6.2	Compare Fibonacci analysis against some of the compressibility tests and check for the mono-bit test	148
Figure 6.3	Group1: Non-pixelated sample images	149
Figure 6.4	The histograms of Fibonacci words of the first six files.	151
Figure 6.5	Histograms of Fibonacci words for the second five files.	152
Figure 6.6	Computer generated pixelated images (group2)	154
Figure 6.7	Fibonacci histogram of group2	157
Figure 6.8	Natural pixelated images (group3)	160
Figure 6.9	The histograms of the first six files of group3	161
Figure 6.10	The histograms of the last seven files of group3.	162
Figure 6.11	(a) Original image (b) lossy jpeg (c) The removed details by JPEG	170
Figure 6.12	Scatter plot of the zeros and one of the three groups	173
Figure 6.13	The original image and the histogram of its JPEG-LS	175
Figure 7.1	Comparison between the ratio of gains for three groups	189
Figure 7.2	The size of the compressed data files with their Fibonacci words numbers	191

Figure 7.3	The relationship between the missing words and the file's size	191
Figure 7.4	Fibonacci counting graph of group1(part1)	197
Figure 7.5	Fibonacci counting graph of group1 (part2)	198
Figure 7.6	The curve of "Cat.jls"	199
Figure 7.7	The curve of "Face.jls"	199
Figure 7.8	The curves of (a)" Fish.jls" and (b) "Flower.jls"	200
Figure 7.9	The curves of "Room.jls" and "Tree.jls".	201
Figure 7.10	The curve of "Squares.jls"	202
Figure 7.11	Fibonacci counting graph of group3 (part1)	203
Figure 7.12	Fibonacci counting graph of group3(part2)	204
Figure 8.1	The location of compressibility tests in cloud system	215
Figure 8.2	Building table of correlations	216
Figure 8.3	Selecting the best compression algorithm based on the table of correlations	216



CHAPTER ONE

INTRODUCTION

1.1 BACKGROUND OF THE STUDY

Despite technological advancements, information grows exponentially, far outpacing growth in hardware. The expected growth in data in 2020 is 44 Zettabytes (ZB), and for 2025, 463 Exabytes (EB) (Desjardins, 2019). Millions of data are created and transmitted worldwide daily. For example, ~147,000 photos are uploaded to Facebook every 60 s, over ~95 million photos and videos are shared on Instagram every day (Aslam, 2020), and ~1.2 billion photos are uploaded to Google Photos per day (Porter, 2019).

The huge data produced on a daily based known as “big data” which become an active research area. (Kolajo et al., 2019), (Ghani et al., 2019)

Not only the social media creates huge of data daily, but the advancement in various sciences also resulted in enormous data, for example in the field of genomics science each personal genome produces a data file of around 100 GB, which will be very costly when thinking about recording the genome of the population for the whole country (Greenfield et al., 2019).

The tremendous growth in data necessitates the use of data compression to minimize the use of storage spaces, increasing the throughput of the network and saving other resources (Jin et al., 2019). Data compression is the science of reducing the size of data which is essential to decrease the financial costs of accommodating and transmitting data. The importance of data compression is even more evident when we are aware of the available storage capacity only being able to store less than 15% of produced data in 2020 (EMC, 2014).

As the data grows, the compressed format of that data grows as well, either stored in storage spaces or transmitted from a point to another on the web. This compressed form is considered the final form of data and it is smaller than the original.

In the end, the compressed data needs to be restored to its original form. If the restoration (called decompression), resulted in the exact original data, the compression is considered lossless compression, but if the restoration resulted in a similar version of the original, it is considered lossy compression. Both lossy and lossless are used to reduce the size of data, and each has its applications.

Data compression has become an integral part of any storage system, including cloud computing which is widely used nowadays. (Barik et al., 2020),(Azar et al., 2020),and(Lu et al., 2021). Since the server should store the data in a compact form so the cost of storage will be minimum, there is a problem if the data itself is incompressible. The server in this case will expend resources on an ultimately futile task. For larger-sized incompressible data, for example, the process of such compressing may take several hours, whereas for the same size of compressible data, it may take several minutes (Harnik et al., 2013) and (Kim et al., 2020).

This longer processing time will also cause other problems such as data delivery latency due to the share of the resources being committed to compressing incompressible data.

The amount and growth of data now and in the future begs the question of how do we identify compressible and non-compressible data?

This question leads to the proposal of a compressibility test for data. The genesis of the test is to minimize the allocation of resources for an ultimately futile undertaking. Larger data are also harder to compress, and the rate of data growth and its size would make this undertaking ever more challenging.

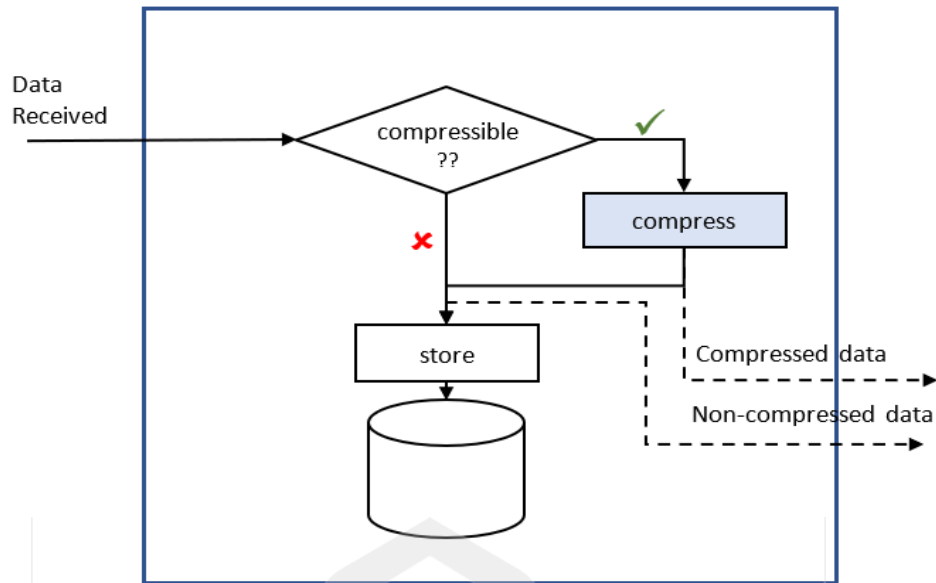


Figure 1.1 Shows the position of compressibility test in server systems.

Upon receiving the data, the compressibility is taken place, so the decision is made to compress or not, whether this process is resulted of online compression (as shown in dotted lines) or for allocating storage spaces, this process is important to save the resources.

The importance of the compressibility test in increasing the throughput as in (Kim et al., 2020) is about 34.15% with sacrificing only 0.09% of the compression ratio for incompressible data, and this percentage is very high in saving resources.

Regarding the storage spaces, the compressibility test can save from 40% to up to 60% of storage spaces, see (Nicolae, 2011). The other resources, such as CPU time, I/O, and memory, will be dramatically affected too.

1.2 PROBLEM STATEMENT

Compressed data is considered as the final minimum form of any data because compressed data, most of the time, is incompressible data (Jon Tate, Christian Burns, Bosmat Tuv-El, 2018). But when the compressed data can be compressed for a second time, do the compressibility tests have the ability to detect their compressibility?

Based on the compressibility tests commonly used, it is possible to figure out the answer.

Starting with the most used compressibility test, named "Prefix estimation". This compressibility test examines the compressibility by compressing a small chunk taken from the head of the data. The main drawback of this method is, no guarantee that the first portion of data will represent the whole data, and for compressed data, this is not an exception.

It is important to note that the search for alternatives or supplements to the test reflects the existence of shortcomings of this test as discussed in (Harnik et al., 2013) which we will refer to next.

"Entropy" is an alternative compressibility test (Balakrishnan & Touba, 2007; Harnik et al., 2013; Oltean et al., 2014b). This compressibility test will not work with compressed data because compressed data have high entropy, whether this data is compressible or not. Compressed data tends to occupy the full range of symbols. Another problem with entropy appeared in (Harnik et al., 2013) which stated that entropy does not imply the compressibility of data in all cases because entropy does not consider the repetition within the data.

Some compressibility tests are not different than entropy, such are "corset" (Harnik et al., 2013), and "byte counting" (Peterson & Reiher, 2016).

To overcome the drawback of entropy, some compressibility tests count on the repetition of patterns (W. Huang et al., 2018; Kipnis & Dror, 2016), others invent a new randomness measurement such as "Pairs distance " (Harnik et al., 2013).

The compressibility tests of this kind are also, not working with compressed data for the same reason of entropy.

Compressible compressed data is not as random as most compressed data, but at the same time, the obvious inner correlations and redundancies are removed by the compression algorithm (D Salomon & Motta, 2010).

Since the compressibility tests will not work on compressed data, we can think of compressibility tests as randomness tests of data.

Despite the word "random" is a vague term and has no exact meaning (Shen et al., 2017), the researchers will follow the Kolmogorov algorithmic definition of randomness (Allender et al., 2006), which implies that data is not random if it is compressible and vice versa. In this case, we can consider the compressibility test as a randomness test too. But can other randomness tests discover the randomness of compressed data and work as a compressibility test?

Like the problem of compressibility tests for compressed data, randomness tests also cannot deal with compressed data predictably. Unfortunately, the result of applying randomness tests on compressed data varies so much from a file to another without any certain rules (Klein & Shapira, 2020).

The previous discussion resulted in two main problems, the first is regarding the compressibility tests which cannot deal with the compressibility of the compressed data, and the second is the inability of randomness tests to recognize the randomness of compressed data (hence the compressibility) in a standard manner.

Both problems reflect the absence of a sophisticated analysis method dedicated to the compressed data, so the compressibility and the randomness of such data be identified.

1.3 RESEARCH QUESTIONS

The research questions are listed as follows:

1. What are the current compressibility tests used to determine the compressibility of data? And can these tests determine the compressibility of compressed data?

2. What are the randomness tests used to measure the randomness of data? and can they work to identify the randomness of compressed data?
3. How do we develop an alternative compressibility test and randomness test sophisticated to deal with compressed data?

1.4 RESEARCH OBJECTIVES

The objectives of this study are:

1. To explore the various methods of compressibility tests used to examine data for compressibility, and whether or not it works for already compressed data.
2. To study the randomness of compressed data and assess whether the available methods are reliable analytical tools that can provide a better understanding of compressed data and serve for detecting its randomness in terms of compressibility.
3. To identify what differentiate compressed data from its uncompressed counterpart and look for possible commonality between compressed data.
4. To develop a new approach that can determine the compressibility and the randomness of compressed data in a measurable and practical way.

1.5 THE IMPORTANCE OF THE STUDY

This study suggests a better understanding of the compressed data by introducing a new approach for analyzing compressed data, with a two-fold approach:

1. It will provide an alternative for studying the randomness of compressed data instead of test packages such as that of NIST (Rukhin et al., 2001)