4-23-2013 12:00 AM

# A quantitative method for measuring and visualizing species' relatedness in a two-dimensional Euclidean space.

Abu Sadat Md. Sayem, *The University of Western Ontario*

Supervisor: Dr. Lila Kari, *The University of Western Ontario*

A thesis submitted in partial fulfillment of the requirements for the Master of Science degree in Computer Science

# A QUANTITATIVE METHOD FOR MEASURING AND VISUALIZING SPECIES' RELATEDNESS IN A TWO-DIMENSIONAL EUCLIDEAN SPACE

(Thesis format: Monograph)

by

Abu Sadat Md. Sayem

Graduate program in Computer Science

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

The School of Graduate and Postdoctoral Studies
The University of Western Ontario
London, Ontario, Canada

# Abstract

Representing DNA sequences graphically and evaluating, as well as displaying, species' relationships have been considered to be an important aspect of molecular biology research. A novel approach is proposed in this thesis that combines three methods: a) Chaos Game Representation (CGR), to portray quantitative characteristics of a DNA sequence as a black-and-white image, b) Structural Similarity (SSIM) index, an image comparison method, to compute pair-wise distances between these images, and c) Multidimensional Scaling (MDS), to visually display each sequence as a point in a two-dimensional Euclidean space. The proposed method produces a visual representation called *Genome Distance Map* (GDM) when applied to a collection of genomic DNA sequences. In a resulting *Genome Distance Map*, the sequences can be visualized as points in a common two-dimensional Euclidean space, wherein the geometric distance between any two points is approximate to the differences between their respective DNA sequence compositions. In addition, the proposed *Genome Distance Map* provides a compelling visualization of species' relatedness in comparison to the phylogenetic trees. Moreover, the proposed method is sensitive and robust in detecting insertions, deletions, substitutions of nucleotides in a genome.

# Acknowledgements

I would like to express my respect and gratitude to my supervisor Dr. Lila Kari for her invaluable guidance, support and supervision. Her continuous motivation kept me inspired throughout this research work. She was always there with her advice and helpful direction when it was required most.

Heartfelt thanks to Dr. Kathleen A. Hill, Nikesh A. Dattani, Katelyn Davis, and Nathaniel Bryans for our constructive and helpful collaboration.

I would like to thank Dr. Mahmoud El-Sakka for his moral support and guidance throughout my stay in the department.

I would also like to thank all of my friends, especially the Shams family, for their inspiration and support. In addition, I would like to express my appreciation to all my colleagues and friends at the BioComputing Lab, Department of Computer Science, University of Western Ontario, for sharing their knowledge with me.

Special thanks are due to all my family members, especially my parents, for being a continuous source of support and inspiration.

# Dedication

*To my parents*

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

After the first successful sequencing of a genome by Fred Sanger [SNC77] in 1977, many methods have been used to explore the large amount of biological data contained in a genome. To extract information from the primary DNA sequences, some visualization methods were introduced in literature. The very first among these methods, introduced by Gates [Gat86] used a 2D coordinate system and the four different directions for the four different nucleotides of a DNA sequence. Similar representations were proposed by Nandy [Nan94] and Leong *et al.* [LM95]. All of these methods [Gat86, Nan94, LM95] have the problem of degeneracy and lack of applications. As a remedy, the vector walk method was proposed by Liao *et al.* [Lia05] that suffers from higher computation complexity, requires much memory, and has limited applications. Recently, another 2D approach was proposed by Yu *et al.* [YLY$^+$10]. The coordinates of the four nucleotides in the approach of Yu *et al.* [YLY$^+$10] are dependent on the $y$- coordinates, and highly vary with the amount of $G+C$ content of a particular DNA sequence. In [YLY$^+$10], a 2D space named "genome space" is also proposed to display relatedness among several species. This space is also completely $y$- coordinate dependent and the distance between points does not satisfy the triangular inequality, therefore it is not a 2D metric space. The method proposed in [YW04] can efficiently analyze smaller region of a DNA sequence (e.g., a specific gene), but as the length of a chromosomal DNA sequence is very long in general, the efficiency of this method remains in question. The Huffman coding was used by Qi *et al*. [QLQ11] to represent DNA sequences on a 2D Cartesian plane using the frequency information of four nucleotides. This method overcomes the problem of degeneracy, but did not show potential application to

the analysis of large genomes. To remove deficiencies of the 2D representation methods, 3D representations were introduced by many [RVNB00, YLW03, YSW09], some of which can distinguish genomes of different species. Additionally, some 4D [CD05] and 5D [LLZX07] representation methods are available in the literature. One of the most promising 2D representations was proposed by Jeffrey [Jef90] in 1990: Chaos Game Representation (CGR) is a unique 2D representation of a DNA sequence. CGR generates interesting fractals and geometric shapes for different genomes of diverse species. These interesting images of DNA sequences resulted in further research in CGR [KMC97, DD92, HSS92]. In 1993, Goldman [Gol93] analyzed the pattern of CGRs in terms of nucleotide, dinucleotide, and trinucleotide frequencies of a DNA sequence. Goldman stated that *"it is unlikely that CGRs can be more useful than simple evaluation of nucleotide, dinucleotide, and trinucleotide frequencies"*. In other words, CGR images contain no insight beyond the frequencies of different nucleotide combinations. In 1995, Karlin and Burge [KB95] introduced the concept of genomic signature and proposed Dinucleotide Relative Abundance Profile (DRAP) as a genomic signature. Afterwards, genomic signatures were widely studied for different genome datasets by researchers in [CMK99, GK01, DGV$^+$99, DGV$^+$00, HLZ00]. As CGR images exemplify the characteristics of a genomic signature, in 1999, Deschavanne *et al*. [DGV$^+$99] showed some interesting properties for CGRs and provided a link between CGR and genomic signatures. In [DGV$^+$99] a new variant of CGR was introduced, called FCGR by Almeida *et al*. [ACM$^+$01]. In 2005, Wang *et al*. [WHSK05] proposed the spectrum of genomic signatures and described some of their properties. Both DRAP and FCGR were proposed as genomic signatures in [WHSK05], and a relation between DRAP and FCGR was also proposed. Interestingly, Wang *et al*. [WHSK05] provided counterexamples to Goldman's conclusion about CGR. Moreover, some properties for FCGR were discussed along with different distance methods to compare two CGR images. In addition, CGR comparisons among 26 mitochondrial DNA sequences were analyzed by generating phylogenetic trees. Another CGR method called Temporal CGR (TCGR) was proposed by Dunham *et al*. [DQW$^+$06] to analyze shorter sequences by using a sliding window. Furthermore, in 2007, Tavassoly *et al*. [TTR$^+$07] proposed a 3D CGR that can be used to analyze the complex structure of a genome. A novel 3D CGR model was proposed by Tu [Tu09] in 2009 on a regular tetrahedron that was used to make comparisons between genomes, as well

as between melodic signatures.

To summarize, efficient genome comparison can be a powerful tool for genome analysis. Every year new genomes of different species are being sequenced. For instance, in 2012 alone, biologists classified between 16,000 and 20,000 new species [Mil12]. Moreover, it was found [MTA$^+$11] that as many as 86% of existing species on Earth and 91% of species in the ocean are still await classification. As a consequence, it is necessary to find a comprehensive, quantitative, general-purpose method to reliably identify the relationships among the already classified 1.2 million species as well as among those that have not yet been classified.

In this thesis, a novel approach is proposed that combines a) a 2D visualization method for DNA sequences called Chaos Game Representation (CGR), b) an image distance (SSIM) to compute distances between the DNA sequences' visual representations, and c) a statistical method called multidimensional scaling (MDS) for representing each genome as a point on the 2D Euclidean space, to determine the degree of relatedness among species. The proposed method produces a visual representation *Genome Distance Map*, from a collection of genomic DNA sequences. In a resulting map, sequences can be visualized as points in a common 2D Euclidean space, wherein the geometric distance between any two points approximates the differences between their respective DNA sequence compositions.

Concretely, if we want to compute and visually display the relationships between DNA sequences in a given set $S = \{s_1, s_2, ..., s_n\}$ of $n$ DNA sequences, a combination of three techniques is proposed:

- *Chaos Game Representation* (CGR), to graphically represent each DNA sequence $s_i$, $1 \leq i \leq n$, as a two-dimensional grayscale image $c_i$.

- *Structural Similarity (SSIM)* index, an image-distance measure, to compute the distances $\delta(c_i, c_j)$, $1 \leq i, j \leq n$, between all pairs of CGR images, and produce a distance matrix $\delta$, where $\delta$ is an $n \times n$ symmetric dissimilarity matrix.

- *Multidimensional Scaling* (MDS) applied to the distance matrix $\delta$ to generate a map in the Euclidean space, where each point $p_i$ with coordinates $(x_i, y_i)$ represents the DNA sequence $s_i$ whose associated CGR image is $c_i$. The distance between two points $p_i$ and $p_j$ reflects the relative distance between the DNA sequences $s_i$ and $s_j$.

The organization of the thesis can be summarized as follows:

Chapter 2 gives some fundamental knowledge of molecular biology that includes the concept and structure of DNA, the concept of mitochondrial DNA, sequence alignment, phylogenetic trees and biological classifications.

Chapter 3 provides a literature survey of various methods to represent DNA sequences graphically, including the CGR research, and the applications of these methods to the comparison of genomes. At the end of this chapter an image distance comparison method called *Structural Similarity* (SSIM) index and the Multidimensional Scaling (MDS) are discussed.

Chapter 4 presents the proposed method to compare DNA sequences in a 2D Euclidean space. Subsequently, the application of this method to compare genomes at different scales of biological taxonomies is discussed, along with introducing the concept of Genome Distance Maps (GDM). The results in this chapter show the advantages of the proposed Genome Distance Maps over DNA barcodes [HCBD03] and Klee diagrams [SSZ10]. Key biological observations include the Genome Distance Map for all primates, where the only misplaced species are two Haplorrhines that are placed with Strepsirrhines, namely *Tarsius bancanus* and *Tarsius syrichta*. These are both tarsiers, whose position within the primates has been a controversial subject for over a century [JHS+11].

Chapter 5 describes the experimental results implemented to answer several biological and mathematical questions. This includes the robustness of the proposed CGR/SSIM method under insertion, deletion and substitution of different number of nucleotides at different positions. The length experiment has as its goal finding the least number of nucleotides required to get recognizable patterns in a CGR, and have reasonable results for GDMs. Furthermore, the experiment with artificial sequences for the same genome keeping single, di- and trinucleotide frequencies similar to consider the validity of the Goldman's conclusion [Gol93] is shown. In addition, the metamorphosis experiment graph shows how the sequential substitution of a genome with a completely different genome behaves. Moreover, the SSIM distance graph of the SSIM distances between one particular genome and all other genomes and analyze the behaviour of the SSIM as a measurement metric in this setting is discussed.

Chapter 6 gives a summary of the major points of this research and presents the conclusions along with possible future work.

This thesis is part of a collaborative project with Dr. Lila Kari (Professor, Department of Computer Science, University of Western Ontario), Dr. Kathleen A. Hill (Professor, Department of Biology, University of Western Ontario), Nikesh A. Dattani (PhD candidate at the University of Oxford), and Katelyn Davis (4th year student, Department of Biology, University of Western Ontario) [KSDH]. My contribution in this project can be summarized as follows:

- Introducing the idea that the CGR images can be compared with various image comparison methods.
- Finding and implementing several image comparison methods to compare CGRs and empirically deciding that SSIM gives the optimal comparison.
- Writing the Matlab code to compare genomes using sequence files such as FASTA, and accession numbers with the CGR/SSIM method. Subsequently designing the complete tool.
- Studying different existing methods and generating phylogenetic trees using those methods.
- Introducing and implementing Multidimensional Scaling (MDS) to display relatedness in a 2D space by representing genomes as points.
- Implementation of assigning unique number and appropriate colors to each mitochondrial genome in the proposed Genome Distance Map.
- Calculating *Stress* (the error). Scaling for each of the Genome Distance Maps.
- Designing and implementing all the experimental simulations.

# Chapter 2

# Molecular biology background

This chapter provides the necessary molecular biology background for this thesis: the concept and structure of nucleic acids, DNA, mitochondrial DNA, phylogenetic trees, sequence alignment and biological classifications.

## 2.1 DNA

Deoxyribonucleic Acid or DNA is a nucleic acid that contains the genetic information used in the development and functioning of all known living organisms. DNA is often compared to a set of blueprints, like a recipe or a code, since it contains the instructions needed to construct other components of cells, such as proteins and RNA molecules. The DNA segments that carry this genetic information are called genes. DNA is built with long polymers called nucleotides and connected by phosphodiester bonds through backbones made of sugars and phosphate. The four different nucleotide bases of DNA are adenine (A), cytosine (C), guanine (G) and thymine (T). Cytosine (C) and thymine (T) are called pyrimidines. Adenine (A) and guanine (G) are called purines. DNA sequences can be single stranded or double stranded. In a double stranded DNA, the nucleotides are pairwise complementary: A is complementary to T, C is complementary to G, and two complementary nucleotides on opposite strands can bind to each other by hydrogen bonds. A single strand of DNA has a specific orientation given by the chemical properties of its sugar-phosphate backbones. The ends of a DNA single strand are denoted by 5' and 3' respectively, based on the chemical convention of naming carbon

atoms in the sugar ring. Two DNA single strands with opposite orientation and complementary nucleotides at each position will bind to each other by hydrogen bonds in a process called base-pairing. Figure 2.1 shows the double helix and chemical structure of a DNA.

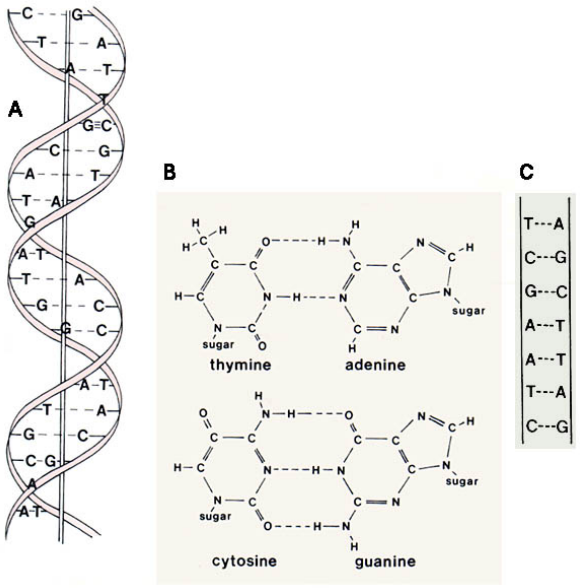



Figure 2.1: Structure of DNA [Wik13b].

## 2.2 Mitochondrial DNA

In this thesis, all the experiments and simulations were performed using a special kind of DNA named *mitochondrial* DNA. Mitochondrial DNA (mtDNA or mDNA) is the DNA located in organelles called mitochondria, which are complex organelles that generate and supply energy to the cellular organism. These organelles are widely found in eukaryotic cells. Human mitochondrial DNA was the first significant part of the human genome to be sequenced. In all species, including human, mtDNA is inherited solely from the mother [Opp12]. Mitochondria have their own genome and are circular, double-stranded DNA strands with few exceptions [Cla91]. The lengths of mtDNA are different for plants, fungi, animals and protists in terms of total number of base pairs. However, the gene content and order of all mitochondrial genes are very similar. The average plant mtDNA genomes are typically longer (40.5-710 knt) than the mtDNA genomes of animals (13.8-17.5 knt). “knt” stands for killo nucleotides. In this thesis, the term “nt” will be used to refer nucleotides. Studies of mtDNA genomes are showing great

promise in the field of evolutionary biology and diagnosis of various diseases. Most importantly, mutations of mitochondrial DNA can lead to a number of illnesses including exercise intolerance and Kearns-Sayre syndrome (KSS), which causes a person to lose full function of heart, eye, and muscle movements [HB92]. Some evidence suggests that mutations of mitochondrial DNA might be major contributors to the aging process and age-associated pathologies [ALW04]. In addition, taxonomic classification can be achieved by comparing mtDNA genomes of different species. This thesis emphasises the comparative analysis of mtDNA genomes of different species.

## 2.3 Sequence alignment

In the post genomic era, one of the main focuses of research has been to compare various biological sequences. For this purpose, sequence alignment was introduced, a method of arranging various biological sequences to find similar regions between them. This is a pairwise alignment that uses three basic operations: insertion, deletion, and substitution of base pairs.

The central concept of all alignments is to assign a score to each possible alignment and minimize the score of over all alignments. The residues of the aligned sequences are typically represented as rows in a matrix. To align the identical or similar nucleotides in successive columns, gaps are inserted between the residues. There are two major kinds of alignments: local alignment and global alignment. In local alignment, similar segments of two different sequences are searched. It does not try to align the whole sequence but attempts to find the different parts that match well between two sequences [NW70]. Global alignment attempts to align every residue in every sequence and forces the entire sequence into a single alignment. Figure 2.2 shows examples of (a) local alignment, (b) global alignment.

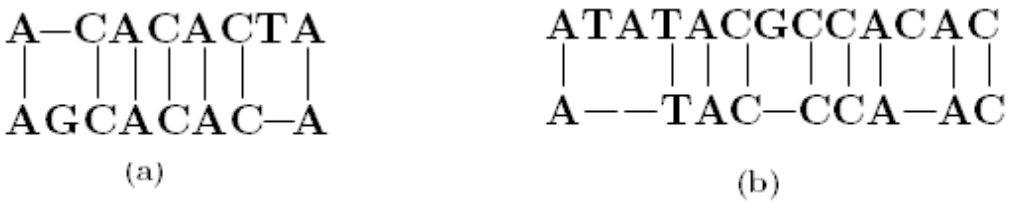



Figure 2.2: Example of (a) local alignment, (b) global alignment.

Local alignment is perhaps widely used but has a limitation; it cannot determine the overall alignment, in that case global alignment is used. Depending on the total number of sequences required to be aligned, sequence alignment can be either pairwise alignment or multiple sequence alignment. Pairwise alignment performs alignments for two sequences at a time where multiple sequence alignment can align more than two sequences at the same time. Multiple sequence alignment is used to find homologous sites of all sequences in an entire set.

## 2.4 Phylogenetic trees

A phylogenetic tree is a tree that represents the evolutionary interrelationships among various species or other entities supposed to have a common ancestor. The tree consists of branches and nodes like the trees used in the computer science data structures. The ancestral node is represented as the root. The different children nodes or leaf nodes represent taxonomic data such as genes or populations of species. In a rooted phylogenetic tree, each node contains descendants, and represents the inferred most recent common ancestor for the descendants. Branches have different lengths that can be proportional to the changes of difference between the species or the sequences. A tree displaying evolutionary relationships among different species is shown in Figure 2.3.

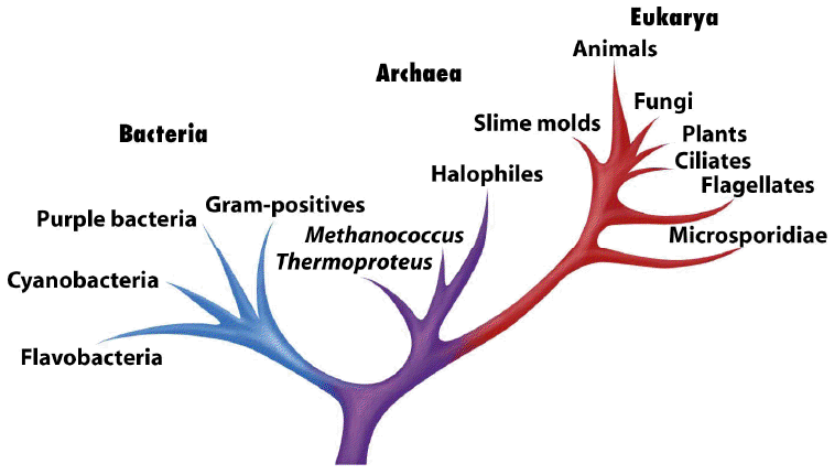



Figure 2.3: Tree of organismal evolution [VVP06].

## 2.5 Biological classifications

Biological classification, or scientific classification in biology, is a scientific taxonomy method that is used to group and categorize organisms into groups such as genus or species. These groups are defined as taxa (singular: taxon). Analysis of different biological taxonomies with the aid of DNA sequence comparison has been considered a very important research issue. The hierarchy of the common biological taxonomy is shown in Figure 2.4. A classification as defined above is hierarchical. In a biological classification, rank is the level (the relative position) in a hierarchy. There are seven main ranks defined by the international nomenclature codes: kingdom, phylum/division, class, order, family, genus, species [Wik13a]. Ranks between the seven main ones can be produced by adding prefixes such as “super-”, “sub-”. Thus a subclass has a rank between class and order, a super-family between order and family. There are slightly different ranks for zoology and for botany, including subdivisions such as tribe.

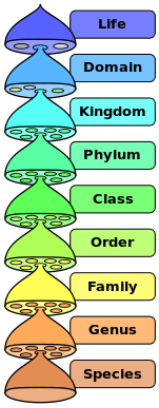



Figure 2.4: The hierarchy of biological classification’s eight major taxonomic ranks. Intermediate minor rankings are not shown [Wik13a].

## Summary

We have discussed the primary molecular biology topics related to this thesis. In the next chapter, we will see how a genome can be graphically represented and the inter genome relatedness can be calculated by applying mathematical operations on the graphical representations of genomes.

# Chapter 3

# DNA sequence visualization methods

This chapter describes some of the earlier methods to represent DNA sequences in a common coordinate system. The basic idea of all these methods is to assign a coordinate for a nucleotide of a DNA sequence and plot the whole sequence according to the coordinate assignment.

## 3.1 The 2D rectangular walk method

One of the earliest methods to represent DNA sequences graphically is the 2D rectangular method that plots a DNA sequence as a random walk on a 2D grid, and the four different nucleotides are represented in the four different quadrants. Thereafter, the whole sequence is scanned and plotted for every base according to the directions of the four nucleotides.

The 2D rectangular method was first introduced by Gates [Gat86] in 1986. In this method, if the base is $C$, then walk one step in the positive $x$ direction, if $G$, then one step in the negative $x$ direction, if $A$, then one step in the negative $y$ direction, and one step in the positive $y$ direction if the base is $T$. Figure 3.1 shows the 2D representation of the sequence $ACTCTGT$ obtained using Gate's method. Every edge represents one step in the corresponding direction of the scanned nucleotide from the DNA sequence. In Figure 3.1, the sequence starts with $A$, so the first edge is towards the negative $y$ direction. The second base is $C$, so the second edge moves towards the positive $x$ direction. Similarly, for the whole sequence the resulting picture can be achieved.

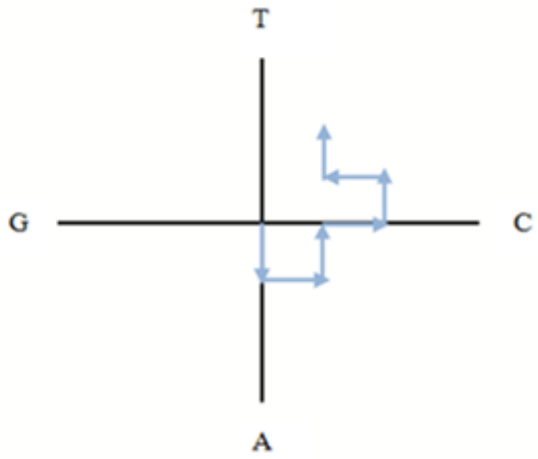



Figure 3.1: 2D rectangular walk plot obtained using Gate's method for the sequence *ACTCTGT* [Gat86].

Afterwards, Nandy [Nan94] used different directions for the four bases. According to his proposed method, if the base is *A*, then walk one step in the negative *x* direction and towards the opposite if the base is *G*. For *C*, walk one step in the positive *y* direction and in opposite direction for the base *T*. Figure 3.2 shows the 2D representation for the sequence *ACGCGTG* by Nandy's method. On the other hand, Morgenthaler [LM95] assigned *C* for a walk of one step in the negative *x* direction, *T* in the positive *y* direction, *A* in the positive *x* direction, and *G* in the negative *y* direction.

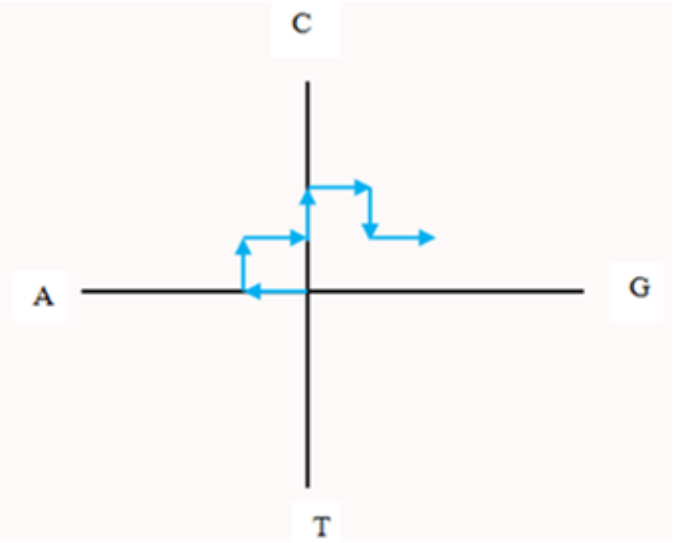



Figure 3.2: 2D rectangular walk method by Nandy's method for the sequence *ACGCGTG* [Nan94].

Though these methods are the earlier approaches to represent a DNA sequence, they have some severe limitations such as degeneracy. For example, in Figure 3.2, if there is a sequence like $CGCGCGCGCGATATATATATAT$ there will be just two edges and the image becomes uninformative. Another problem is that the original sequence cannot be retrieved, as well as, the representation does not have a one-to-one correspondence with the sequence. Most importantly, the 2D rectangular methods did not offer any significant application. These problems were taken into consideration in the later representation methods.

## 3.2 The vector walk method

The vector walk method was introduced by Liao *et al.* [Lia05] to eliminate degeneracy of the 2D rectangular walk methods. In this method, four nucleotide bases were represented by four special vectors that maintain small angles among them. In the vector walk method, the four bases are plotted on two different quadrants: $T$ and $C$ are assigned to the first quadrant, and $A$ and $G$ to the fourth quadrant. Figure 3.3 (a) shows the general representation of the vector walk method. The vectors representing the four nucleotides $A, G, C$, and $T$ can be written as

$$(m, -\sqrt{n}) \rightarrow A,\ (\sqrt{n}, -m) \rightarrow G,\ (\sqrt{n}, m) \rightarrow C,\ (m, \sqrt{n}) \rightarrow T,$$

where $m$ is a real number, $n$ is a positive real number but not a perfect square number. Thus, a DNA sequence can be reduced to a series of nodes $S_0, S_1, S_2 .... S_n$, whose coordinates $(x_i, y_i)$ ($i = 0, 1, 2, .....n$, $n$= length of the DNA sequence) satisfy

$$x_i = a_i m + g_i \sqrt{n} + c_i \sqrt{n} + t_i m,$$

$$y_i = -a_i \sqrt{n} - g_i m + c_i m + t_i \sqrt{n}.$$

Here, $a_i$, $c_i$, $g_i$ and $t_i$ are the cumulative numbers of occurrences of $A$, $C$, $G$ and $T$, respectively. Figure 3.3(b) shows the vector walk graph for the sequence $ATGGTGCACC$.

The vector walk can completely remove degeneracy. However, it possesses higher computational complexity, and requires more memory than the 2D rectangular methods.

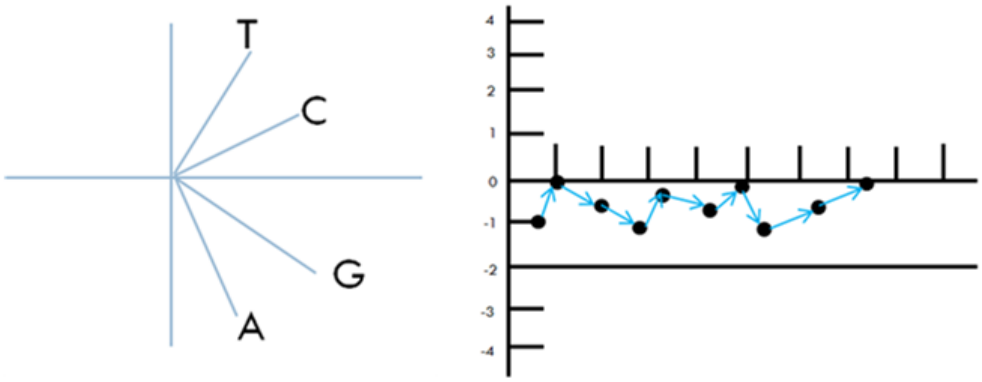



Figure 3.3: (a) General representation of vector walk method, (b): vector walk graph for the sequence *ATGGTGCACC* [Lia05].

## 3.3 Yu's method

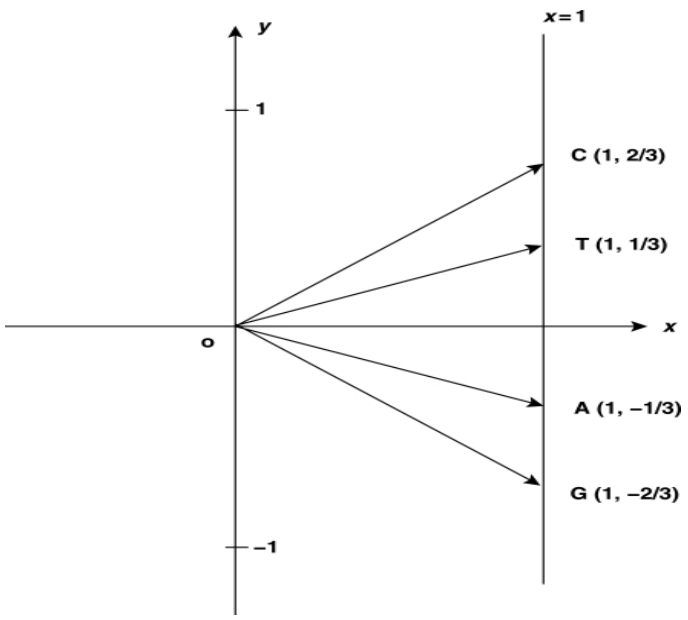
Another method was proposed by Yu *et al.* [YLY+10] recently, where a DNA sequence graph is constructed in two quadrants of the Cartesian coordinate system. Pyrimidines ($C$ and $T$) were placed in the first quadrant and purines ($A$ and $G$) in the fourth quadrant. The vectors corresponding to the four nucleotides $G$, $A$, $T$ and $C$ were defined as follows:

$$(1,-2/3) \rightarrow G, (1,-1/3) \rightarrow A,$$

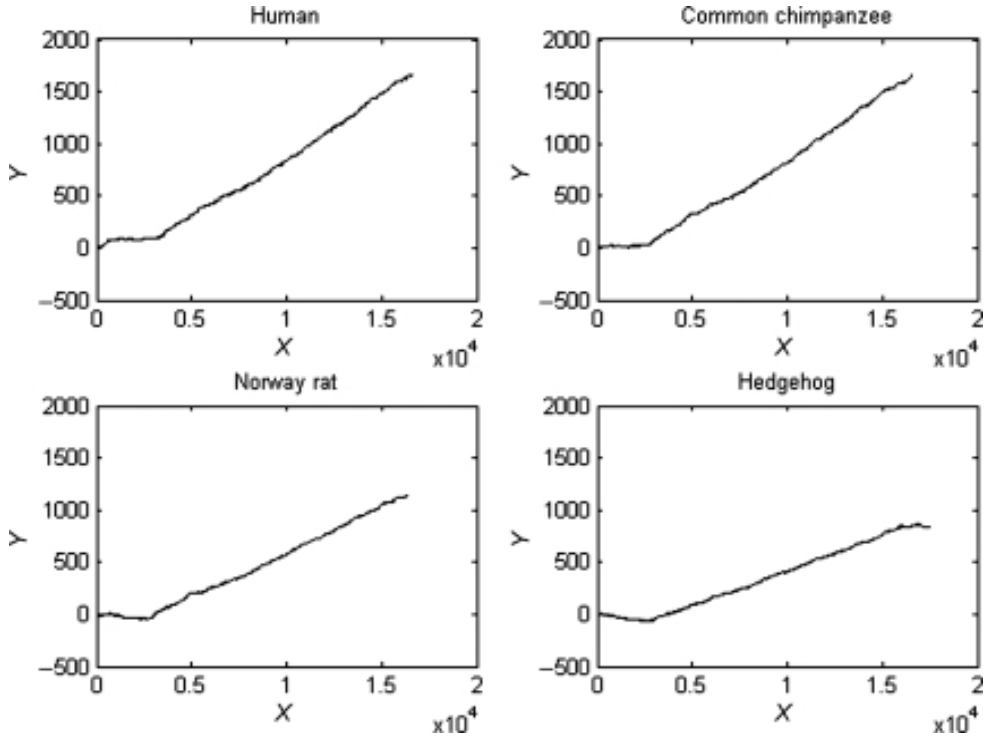
$$(1,1/3) \rightarrow T, (1,2/3) \rightarrow C.$$

With this definition, points for any of the four bases can be calculated. For instance, if the first base is $A$, then the point is $(1,-1/3)$, if the second base is $T$, then the point is (2,0), and if the third base is $G$, then the point is (3,-2/3). Figure 3.4(a) shows the distribution of the four bases and Figure 3.4(b) shows the four different plots obtained with this method for human, common chimpanzee, Norway rat and hedgehog from their mitochondrial genomes. In [YLY+10], the authors outlined some applications using their proposed method. A DNA graphical curve was characterized by the use of moment vectors. In this method, points of a sequence are $(1,y_1),(2,y_2),...,(n,y_n)$. As a consequence, a sequence of numbers $(1-y_1),(2-y_2),.....(n-y_n)$ can be computed in the reverse way and using this series, the original points can be recovered. The moment vector was defined as follows:

$$M_j = \sum_{i=1}^{n} (x_i - y_j)^j / n^j,\ j = 1, 2, .....n.$$




(a)




(b)

Figure 3.4: (a) Coordinate distribution for four bases; (b) Plots of mitochondrial genomes of four different species using the method of Yu *et al.*[YLY+10].

Here, $n$ is the number of nucleotides contained in a DNA sequence and $(x_i, y_i)$ represents the position of the $i^{th}$ nucleotide. By definition, each DNA sequence has an $n$-dimensional moment vector $(M_1, M_2, ...M_n)$ associated with it. A 2D space for different genomes was plotted taking the first two moment vectors. Figure 3.5 shows the plot from the first two moment vectors for some group of species by the method of [YLY+10].

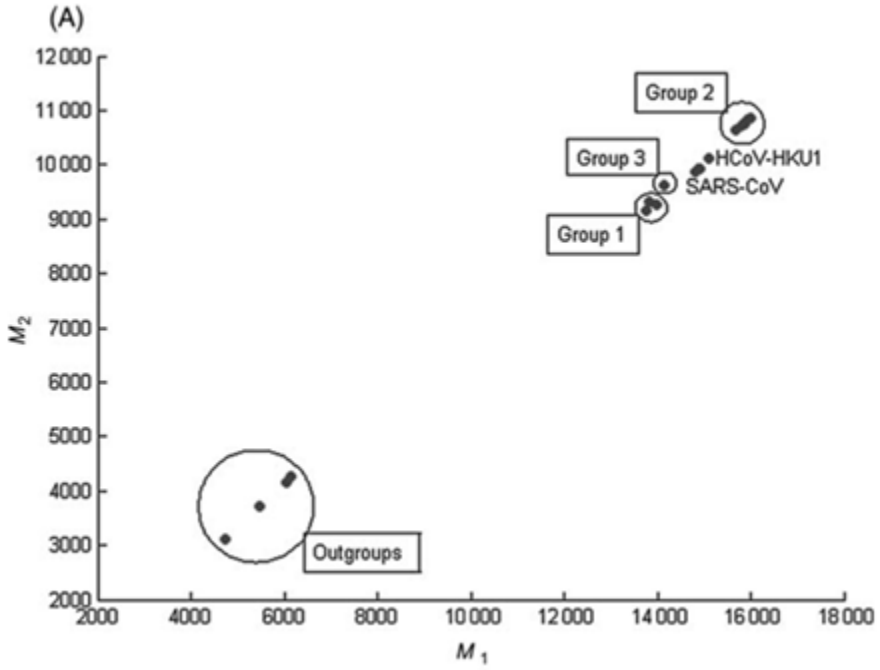



Figure 3.5: Proposed genome space of Yu *et al*. [YLY+10].

Limitations of the proposed plot of Yu *et al*. [YLY+10] includes the $y$-coordinate dependency of the points. In addition, the curve depends on the $C+G$ content of a DNA sequence and on the vectors assigned to each nucleotide. The vector assignment for the nucleotides can be changed from one quadrant to another. As a result, if the vector assignments of $C$ and $G$ are changed, the whole curve changes and it will be highly dependent on the $C+G$ content of that particular DNA sequence. Moreover, the space is also $y$ dependent, and not a real metric space: the distance between points does not satisfy the triangular inequality property. For example, if we have three points $M_1$, $M_2$, $M_3$, and distance between $M_1$ and $M_2$ is $D_{12}$, distance between $M_1$ and $M_3$ is $D_{13}$, and between $M_2$ and $M_3$ is $D_{23}$ then it must satisfy

$$D_{12} + D_{13} > D_{23}$$

$$D_{12} + D_{23} > D_{13}$$

$$D_{23} + D_{13} > D_{12}$$

However, the space proposed in [YLY+10] fails to satisfy the above condition that makes the space not a metric space.

## 3.4 The cell method

This 2D representation was introduced by Yao *et al.* [YW04], where four bases were represented by four different cells. The cell is a $2 \times 2$ matrix of four different dots, where each dot represents a nucleotide base. To plot a DNA sequence by the cell method, the sequence is scanned for every nucleotide, and moved forward according the position of the dot associated with the scanned nucleotide. The correspondence between the nucleotide bases and coordinates are calculated as follows:

$$\varphi(g_i) = \begin{cases} (2(i-1), 0), & g_i = G \\ (2(i-1), 1), & g_i = A \\ (2(i-1)+1, 0), & g_i = C \\ (2(i-1)+1, 1), & g_i = T \end{cases}$$

Figure 3.6 shows the representation of a cell. The representation of the sequence $ATGGTA$ by this method is shown in Figure 3.7.

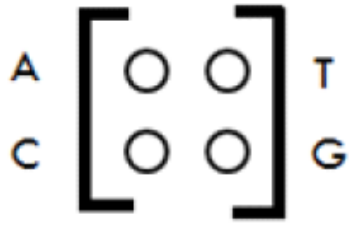



Figure 3.6: A cell [YW04].

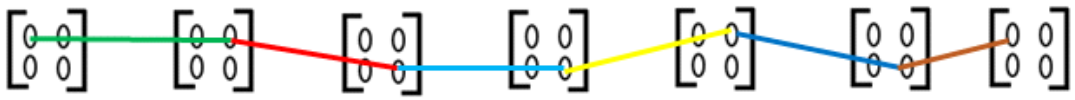


Figure 3.7: Representation of the sequence $ATGGTA$ by the cell method [YW04].

To exemplify some applications of this method, the authors [YW04] compared 11 different species by using the graphical representation of the first exon of the $\beta$-globin gene. Upon transforming the curve into a matrix, a 12-component vector consisting of the normalized eigenvalues $\lambda/N$, is calculated, where $\lambda$ is the leading eigenvalue of some characteristic matrix which is also generated from the transformed matrix and $N$ is the total number of bases of the DNA sequence. The distance between two component vectors was calculated by the Euclidean distance, and it is claimed that similar species have smaller distances between them than distant ones.

The cell method is applicable for the analysis of a small region of a DNA sequence. For the whole sequence, the cell method suffers from high computational complexity as representing each base with a cell consumes large memory. Generally, one single chromosomal DNA sequence contains more than a billion of base pairs. So, representing one DNA sequence with the cell method would be very costly in terms of memory.

## 3.5 The Huffman Coding Method

The Huffman coding method was introduced to reduce degeneracy for the repetitive nucleotides in a sequence [QLQ11]. In this method, first the frequencies of the four nucleotides in a sequence are calculated. Afterwards, a binary tree is generated based on the frequencies and the Huffman coding method. For instance, if we have a sequence with frequencies $\{f_A, f_G, f_T, f_C\}$, a binary tree is generated from left to right taking the two least probable symbols and putting them together to form another equivalent symbol having a probability that is equal to the sum of the two symbols. The process is repeated until there is just one symbol. Following this, the Huffman tree for the DNA sequence is created. The tree can be read backward, from right to left, assigning different bits (bit ‘‘0,’’ the left child with a less probability; bit ‘‘1,’’ the right

child with a larger probability) to different branches. For example, if there is a DNA sequence with its nucleotides frequencies {0.05; 0.3; 0.2; 0.45}, then the Huffman tree for the sequence can be drawn as shown in Figure 3.8. The final Huffman codes can be used from the tree, for the tree in Figure 3.8 they are: C→0, G→11, A→100, and T→101. Using this information, a 2D graphical representation can be made afterwards by a 0-1 graph, where bit 1 is plotted in the first quadrant and bit 0 in the fourth quadrant of a Cartesian coordinate system as shown in Figure 3.9. Figure 3.10 shows the Huffman tree and 2D representation for the first exon of the $\beta$-globin gene of chimpanzee. The major advantage of this method is that even if the Huffman tree is the same for two sequences where the frequencies of four nucleotides are similar, the resulting 2D representation will be different for two different sequences based on their internal organization.

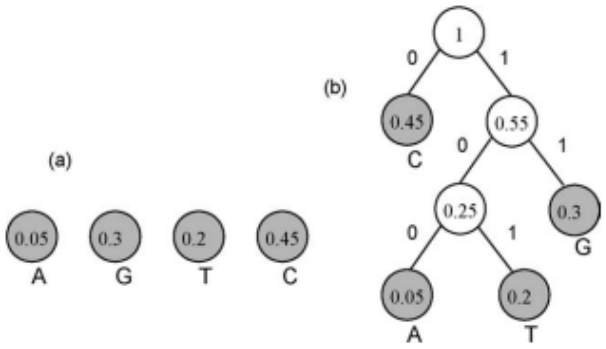



Figure 3.8: A Huffman tree for a DNA sequence with nucleotide frequencies {0.05; 0.3; 0.2; 0.45}; a) The first nodes; b) The final Huffman tree.

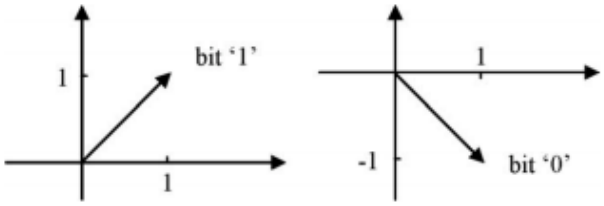



Figure 3.9: The graphical representations of bit ''1'' and bit ''0.''

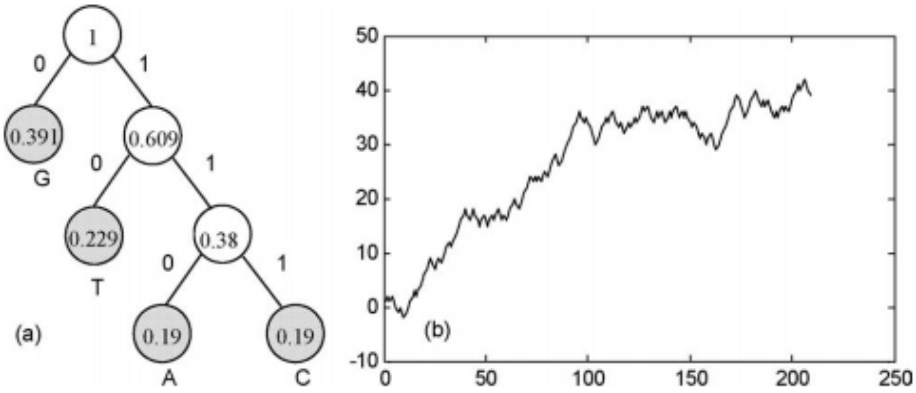



Figure 3.10: a) The Huffman tree for the first exon of the $\beta$-globin gene of chimpanzee; b) The 2D graphical representation of the first exon of the $\beta$-globin gene of chimpanzee.

## 3.6 The ColorSquare method

This method represents a DNA sequence in a big square that contains small squares of different colors for different nucleotides [ZSZ+12]. ColorSquare has several advantages: (1) no degeneracy, (2) no loss of information, (3) highly compact, (4) colorful, and (5) square. To build the ColorSquare, first the size of the big square needs to be calculated. This is done by calculating the square root of the length of the DNA sequence. If the length of the sequence is $n$ then the size of the big square is $\lceil \sqrt{n} \rceil \times \lceil \sqrt{n} \rceil$. After getting the big square, the small squares are marked. The marking is done clockwise around in the big square according to the given DNA sequence. Because $\lceil \sqrt{n} \rceil \times \lceil \sqrt{n} \rceil$ is generally greater than $n$, the big square contains more that $n$ squares. It means that there are some small squares left which do not represent DNA bases. These remaining squares are marked with '$N$'. After marking the big square, the squares are filled according to color assignments as follows: $A \rightarrow$ Red, $G \rightarrow$ Blue, $C \rightarrow$ Yellow, $T \rightarrow$ Green, $N \rightarrow$ White. Figure 3.11 shows the construction and the final ColorSquare for the first exon of the human $\beta$-globin gene.

This big square can be converted to a matrix with values assigned to nucleotides. For the conversion of the ColorSquare to a numerical matrix, values are assigned to the four different color squares and the white squares as follows:

$N \rightarrow White \rightarrow 0$, $A \rightarrow Red \rightarrow 1$, $C \rightarrow Yellow \rightarrow 2$, $G \rightarrow Blue \rightarrow 3$, $T \rightarrow Green \rightarrow 4$.

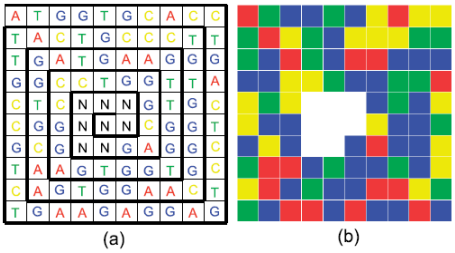



Figure 3.11: (a) The whirlpool construction of ColorSquare of the first exon of the human $\beta$-globin gene; (b) The visualization result of ColorSquare of the first exon of human $\beta$-globin gene [ZSZ+12].

$$\begin{pmatrix}
1 & 4 & 3 & 3 & 4 & 3 & 2 & 1 & 2 & 2 \\
4 & 1 & 2 & 4 & 3 & 2 & 2 & 2 & 4 & 4 \\
4 & 3 & 1 & 4 & 4 & 1 & 1 & 3 & 3 & 3 \\
3 & 3 & 2 & 2 & 4 & 3 & 3 & 4 & 4 & 1 \\
2 & 4 & 2 & 0 & 0 & 0 & 3 & 4 & 3 & 2 \\
2 & 3 & 3 & 0 & 0 & 0 & 2 & 3 & 3 & 4 \\
3 & 2 & 3 & 0 & 0 & 3 & 1 & 3 & 3 & 2 \\
4 & 1 & 1 & 3 & 4 & 3 & 3 & 4 & 3 & 2 \\
2 & 1 & 3 & 4 & 3 & 3 & 1 & 1 & 2 & 4 \\
4 & 3 & 1 & 1 & 3 & 1 & 3 & 3 & 1 & 3
\end{pmatrix}$$

Figure 3.12: The matrix representation of ColorSquare (Fig.3.11 (b)) of the sequence of the first exon of human $\beta$-globin gene [ZSZ+12].

The equivalent matrix for Figure 3.11 is shown in Figure 3.12.

To compare two ColorSquare matrices, a numeric characterization is obtained for each matrix using the leading eigenvalues and adopting 24 component vectors [ZSZ+12]. Suppose there are two sequences $i$ and $j$, and the obtained numerical characterization vectors are $D_i$ and $D_j$ from their corresponding ColorSquare matrices, then the similarity between two sequences can obtained by calculating the Euclidean distance between the two vectors, $D_i$ and $D_j$. The main problem of this method is that the construction of the ColorSquare is completely unnecessary as we are dealing with the final matrix at the end, which can be constructed independently without using the ColorSquare.

## Summary

2D representations can effectively provide visual inspection of data, which can be used to recognize major differences among several sequences. Earlier methods were limited by degeneracy and the original sequences were unrecoverable. The concerns with the methods like vector

walk and cell were memory and computational complexity. In the vector walk method, every base position has to be counted and stored in the memory to compute cumulative numbers of bases. The cell method requires each cell to be stored for every base and comparing dissimilarities among species requires additional computations. ColorSquare method involves some unnecessary computation steps. The Huffman coding method is better than the other methods as it removes the factor of degeneracy and can produce different representations for different sequences which contain the same frequencies for the four nucleotides. In the following subsections, we will look at some of the 3D representation methods for a DNA sequence.

## 3.7 Randic's method

Randic *et al*. [RVNB00] introduced a 3D representation for a DNA sequence by assigning four different nucleotides to four different coordinates in a 3D space. The coordinates for the four nucleotides are as follows:

$$(+1,-1,-1) \rightarrow A, (-1,+1,-1) \rightarrow G, (-1,-1,+1) \rightarrow C, (+1,+1,+1) \rightarrow T.$$

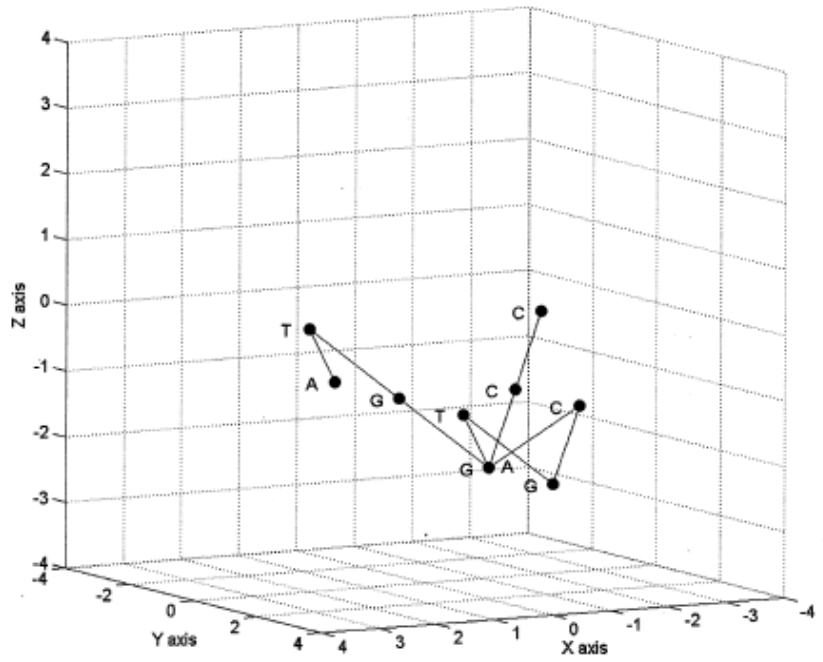



Figure 3.13: The 3D representation of the sequence *ATGGTGCACC* by the method of Randic *al.* [RVNB00].

If the first base is $A$, then the coordinate is $(+1,-1,-1)$. If the second base is $T$, then the point is calculated by adding the previous point to its assigned coordinates. Thus, the resulting point will be $(+2,0,0)$. The application described for the cell methods was first introduced by Randic in [RVNB00]. So, with this plot, we can also find the dissimilarities among several regions of a particular DNA sequence and also among several species by using specific genes. Figure 3.13 shows the plot for the sequence *ATGGTGCACC* with Randic's proposed 3D approach to represent a DNA sequence.

## 3.8 Yuan's method

Yuan *et al*. [YLW03] introduced another simple 3D method to represent a DNA sequence, where the three different coordinates were chosen as follows: $A$= negative $x$ axis, $G$= positive $y$ axis, $T$ = negative $y$ axis and $C$ = positive $y$ axis. The $z$ value of any point is $i$, where $i$ is the position of the current base. The coordinate functions are as follows:

$$\varphi(i) = \begin{cases} (-1,0,i), & g_i = A, \\ (1,0,i), & g_i = G, \\ (0,-1,i), & g_i = T, \\ (0,1,i), & g_i = C. \end{cases}$$

For example, the corresponding points for the sequence *ATGGTGCACC* are {(-1, 0, 1), (0, -1, 2), (1, 0, 3), (1, 0, 4), (0, -1, 5), (1, 0, 6), (0, 1, 7), (-1, 0, 8), (0, 1, 9), (0, 1, 10)}. Figure 3.14 shows the 3D curve generated by this method for the sequence *ATGGTGCACC*.

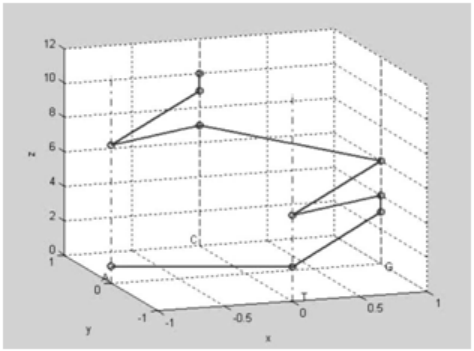



Figure 3.14: Characteristic curve of the sequence *ATGGTGCACC*; The dots denote the bases making up the sequence [YLW03].

## 3.9 The TN curve

The TN curve is one of the most sophisticated 3D representation methods to visualize a DNA sequence, offering better sensitivity in terms of calculating dissimilarity among different species. This curve was proposed by Yu *et al.* [YSW09] based on the trinucleotide organization of a DNA sequence. With four different bases, there can be 64 different combinations of trinucleotides. In this method, at first values are assigned for the first and third base of any trinucleotide as $A \rightarrow 1$, $G \rightarrow 2$, $C \rightarrow 3$, and $T \rightarrow 4$ to determine the values of the $x$ and $y$ coordinates. The sign of the coordinates are obtained using the sign assignment of the second base, e.g., $+,+ \leftrightarrow A; -,+ \leftrightarrow G; -,- \leftrightarrow C; +,- \leftrightarrow T$. The value of the $z$ coordinate increases with the successive trinucleotide numbers. The positions of each nucleotide based upon the second nucleotide are shown in Figure 3.15. To use this curve more effectively, the authors introduced two other parameters, that are the $x_i$ and $y_i$ namely the mean of $x$ and the mean of $y$. The equations for $x_i$ and $y_i$ are as follows

$$x_i^{'} = \sum_{n=1}^{i} x_n,$$

$$y_i^{'} = \sum_{n=1}^{i} y_n.$$

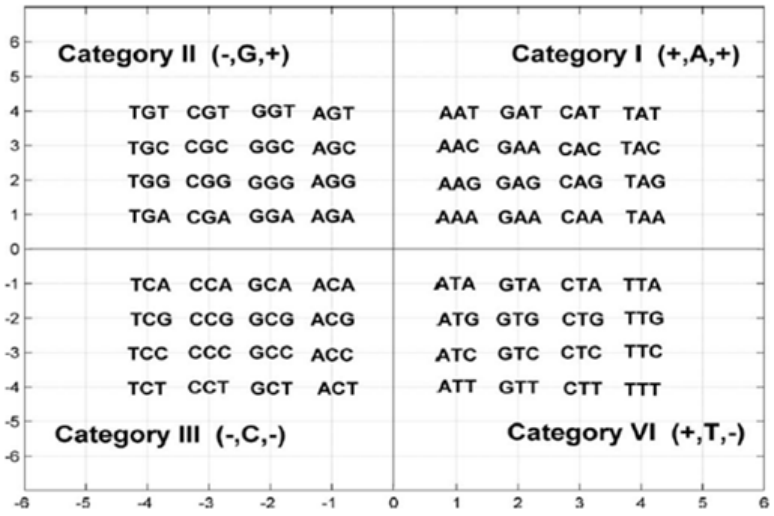



Figure 3.15: Distribution of the 64 different trinucleotides in Cartesian 2D coordinates [YSW09].

| Triplets | $x$ | $y$ | $z$ | $x'$ | $y'$ |
|---|---|---|---|---|---|
| ATG | 1 | -2 | 1 | 1 | -2 |
| TGG | -4 | 2 | 2 | -3 | 0 |
| GGT | -2 | 4 | 3 | -5 | 4 |
| GTG | 2 | -2 | 4 | -3 | 2 |
| TGC | -4 | 3 | 5 | -7 | 5 |
| GCA | -2 | -1 | 6 | -9 | 4 |
| CAC | 3 | 3 | 7 | -6 | 7 |
| ACC | -1 | -3 | 8 | -7 | 4 |

Table 3.1: 3D coordinates for the sequence *ATGGTGCACC* [YSW09].

For example, if we have the sequence *ATGGTGCACC*, the coordinate set $(x,y,z)$ and $(x',y',z)$ can be described by Table 3.1. Figure 3.16 shows the TN curve for the above sequence. The configuration of the curve can be changed by changing the value of the four bases and the sign assignment for the second base. The curve is unique for any particular coordinate distribution but can vary with the change of coordinate assignment.

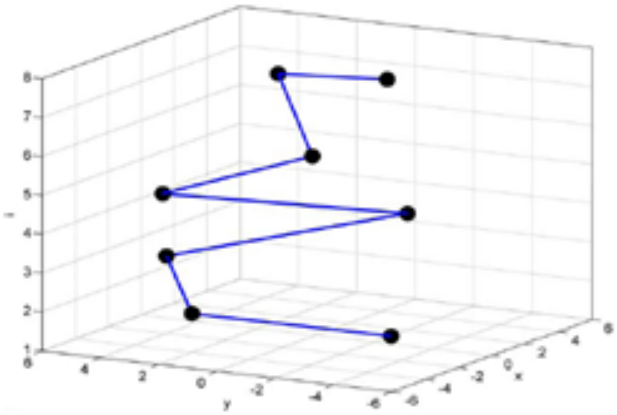


Figure 3.16: TN curve for the sequence ATGGTGCACC [YSW09].

The characteristic graph (projection on the $(x',z)$ and $(y',z)$ plane respectively) for the $x'$ and $y'$ are different for genomic sequences originating from different species. Figure 3.17 shows the $x'$ and $y'$ curve for four different species chicken, gorilla, human, and opossum. First three curves have similar kind of displays but the curve for opossum is different. With these plotted

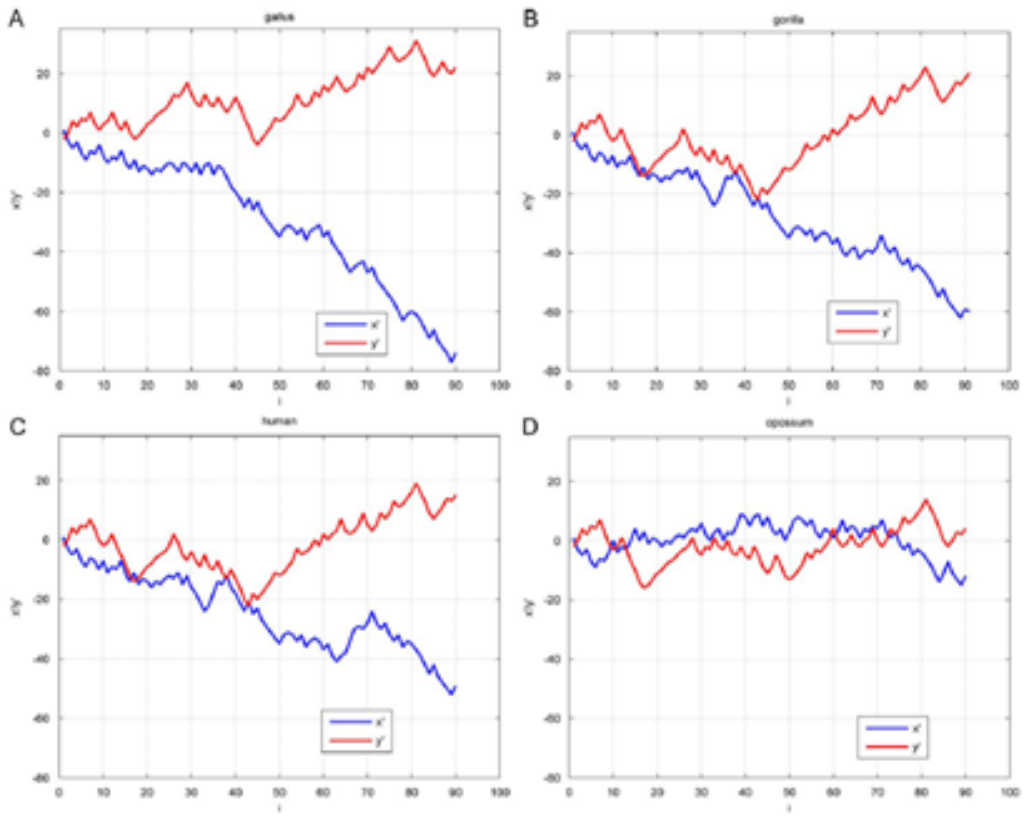



Figure 3.17: 2D plots of $x'$ and $y'$ of the coding sequences of the first exon of $\beta$- globin gene of human, gorilla, opossum and chicken [YSW09].

curves, authors calculated the dissimilarity matrix for the different curves with the Euclidean distance, which was used to analyze relatedness of different species. Though the dataset chosen was very small, the curve was better in terms of differentiability than the previous approaches. The relation between every given DNA sequence with its TN curve is one to one. The TN curve can be used as a nucleotide descriptor. The second base carries the descriptor information. If $x > 0$ then the second base must be $A$ or $T$, and $G$ or $C$ otherwise. If $y > 0$ then the second base must be $A$ or $G$, and $C$ or $T$ otherwise. So, the TN curve has important mathematical properties to represent DNA sequences and the curve is unique for each sequence. Moreover, with the points of the curve, the original sequence can be recovered. The mean value of $x$ and $y$ also carries important information to differentiate the organization of several segments of the DNA sequence or between two or more DNA sequences.

## Summary

3D representations are capable of removing the degeneracy of the 2D plots, as well as the comparison for different species retrieved from the 3D plots was more sensitive. However, 3D representations require more computation than 2D representations. Moreover, for visualization purposes, the 2D plots are more convenient than the 3D plots. TN curve is the recent proposed 3D representation and proved to be superior to the 2D and other graphical representations of DNA sequences. In the next section, we will study another sophisticated and efficient representation of DNA sequences named Chaos Game Representation for a DNA sequence.

## 3.10 Chaos Game Representation and genomic signatures

Chaos Game Representation (CGR ) was introduced by Jeffrey [Jef90] in 1990 to visualize the organization of a DNA sequence. By this approach, different DNA sequences produce different fractal structures, leading to the interesting area of research of analysing DNA sequence graphically. The CGR image of any DNA sequence is plotted in a 2D coordinate system in a unit square. The center is the center of the square. The four corners of the square are the four different nucleotides of the DNA sequence. The coordinates of the nucleotides are A=(0,0), C=(0,1), G=(1,1) and T=(1,0). The lower and upper vertices (A+T), (C+G) represent the base composition, and the diagonals indicate the purines (A, G) or the pyrimidines (C, T). The whole sequence is read base by base from left to right. The plotting steps are as follows: the first point is the center of the square, the next point is the midpoint between the current point and the corner point of the next nucleotide. The basic algorithm of CGR is shown in Algorithm 1.

**Algorithm 1** CGR_Plot.

Input : Input: A DNA sequence S

Output: CGR of the sequence S

Point = (.5, .5);

Repeat

Scan next character from S;

Point = (Point + Point of the corner of this nucleotide input character)/2;

Plot a point at this position in the square

Until there are no more characters

CGR images of different DNA sequences show interesting patterns including various geometric patterns, such as squares, parallel lines, rectangles and triangles. Some of the CGR images even show complex fractal patterns. Figure 3.18 shows the CGR images of four different species, generated using their mitochondrial DNA sequences.

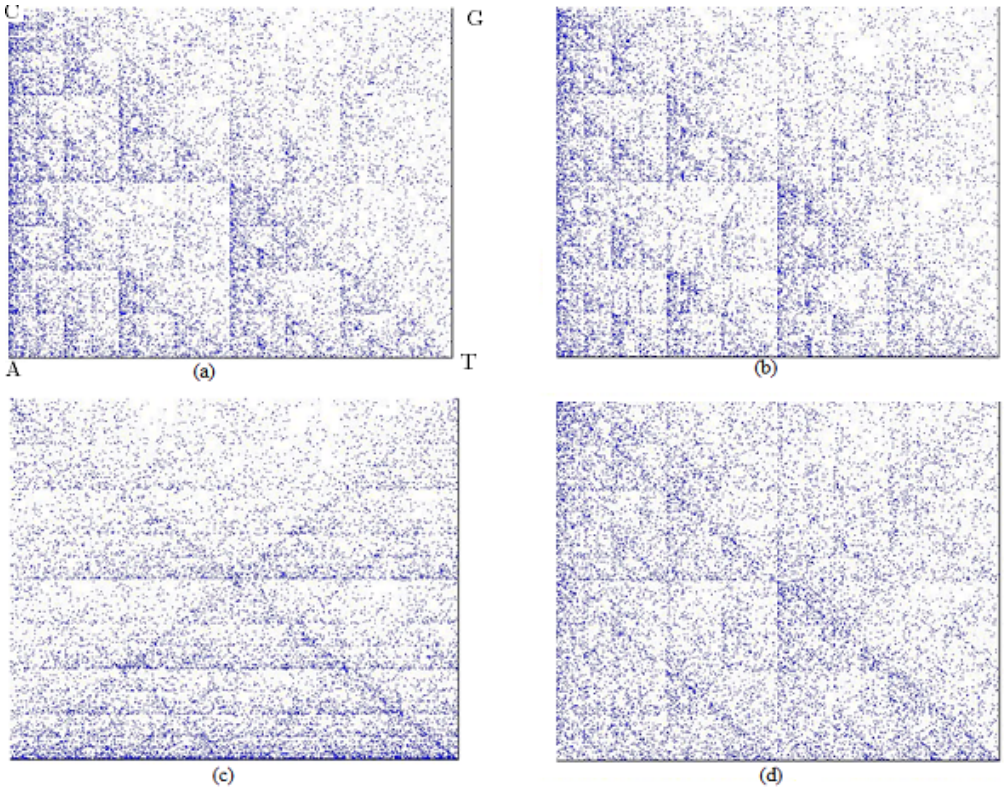



Figure 3.18: CGR image of the mitochondrial genomes of (a) baboon, (b) human, (c) shrimp, and (d) trout.

In 1993, Goldman [Gol93] analyzed the pattern of CGRs in terms of nucleotide and dinuclotide frequencies using a Markov Chain Model [LM95]. In the first order Markov Chain model [LM95], the successive bases in a simulated sequence depend only on the preceding bases. In this model, a 4×4 matrix $P$ defines a set of probabilities, and subsequent bases follow the current base in a DNA sequence using these probabilities. If the base labels $A, C, G$, and $T$ are equated with the numbers 1, 2, 3 and 4, then $P_{ij}$, the $j^{th}$ element of the $i^{th}$ row of $P$, defines the probability that base $j$ follows base $i$. The row sum of the matrix $P$ must be equal to 1. With the use of the matrix $P$, a simulated DNA sequence is obtained by selecting the first base randomly, according to the frequencies of the bases in the DNA string under study; if this is base $i$, then the probabilities $P_{i1}$, $P_{i2}$, $P_{i3}$, and $P_{i4}$ are used to select the next base, and so on until the simulated sequence is of the same length as the original DNA sequence. In the second order Markov Chain model, each base depends on the previous two bases. The probabilities are in the form of $P_{XYZ}$, which implies that this is the probability that base $Z$ follows the dinucleotide $XY$ to simulate the original sequence.

Goldman simulated CGRs of the *Bacteriophage lambda* genome with the original sequence and with second order Markov chain simulated sequence and found similar images [Gol93]. Figure 3.19 shows the resulting CGR images of the simulation in [Gol93]. Thus, he concluded by saying "it is unlikely that CGRs can be more useful than simple evaluation of nucleotide, dinucleotide and trinucleotide frequencies", which implies CGR is only a pictorial representation of nucleotide, dinucleotide and trinucleotide frequencies of a DNA sequence. After this conclusion, the research on CGR continued with less frequency.

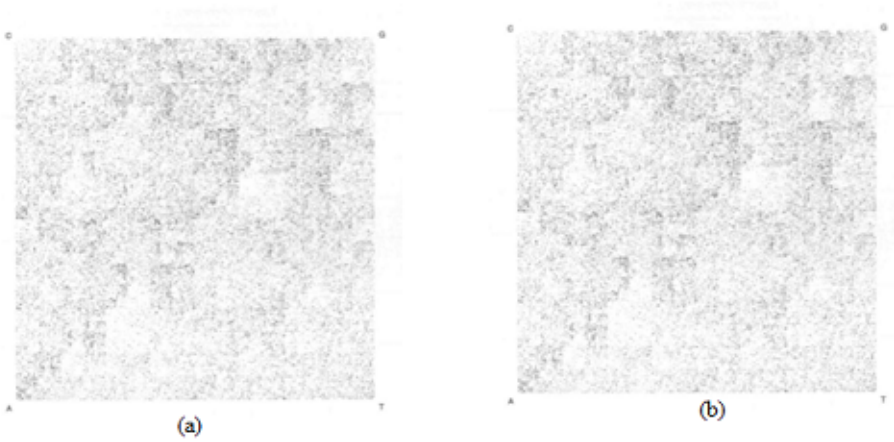



Figure 3.19: CGR images simulated in [Gol93].

After two years, the concept of the genomic signature was introduced by Karlin *et al.* [KB95]. A genomic signature is a specific arithmetic characteristic dissimilar for different organisms but pervasive in the genome of same organism [KB95]. A genomic signature has two major properties: pervasiveness and differentiability. This means the signature will be pervasive in the genome of same the organism and must be different for unlike organisms. The signature proposed by Karlin and Burge [KB95] was the ratio of dinucleotide frequencies to the total number of mononucleotide frequencies. DRAP can be defined as follows:

**Definition 1** *For a sequence s, the dinucleotide relative abundance profile DRAP(s) is an array* $\rho_{XY} = f_{XY}/f_X f_Y$*, where XY stands for all combinations of dinucleotides,* $f_X$ *denotes the frequency of the mononucleotide X in s, and* $f_{XY}$ *denotes the frequency of the dinucleotide XY in s.*

Afterwards, Deshavanne *et al.* [DGV+99] showed that CGR images inside a particular genome vary less than among different genomes. Furthermore, in this paper [DGV+99], another type of CGR which is called Frequency Chaos Game Representation (FCGR) was introduced. The name FCGR was given by Almeida *et al.* [ACM+01]. FCGR can quantitatively express the structure and complexity of a DNA sequence. Though CGR images provide a visualization for human eyes, it has a limitation of resolution if they are computer generated, which can be solved by FCGR. A $k^{th}$-order FCGR can be obtained in two ways: a) counting the number of points inside the grid that can be obtained by dividing the CGR plot into $2^k \times 2^k$

grid, and putting the number into a corresponding matrix element, b) directly counting the number of occurrences of each length $k$ oligonucleotide in the sequence, and putting it into the corresponding place in the matrix. A $k^{th}$-order FCGR is a $2^k \times 2^k$ matrix, and can be defined as follows.

**Definition 2** *A $k^{th}$-order FCGR of a sequences s, denoted by $FCGR_k(S)$, is a $2^k \times 2^k$ matrix. A first order FCGR and second order FCGR are shown below where $N_w$ is the number of occurrences of the oligonucleotide w in the sequence s.*

$$FCGR_1(s) = \begin{pmatrix} N_C & N_G \\ N_A & N_T \end{pmatrix}$$

$$FGCR_2(s) = \begin{pmatrix} N_{CC} & N_{GC} & N_{CG} & N_{GG} \\ N_{AC} & N_{TC} & N_{AG} & N_{TG} \\ N_{CA} & N_{GA} & N_{CT} & N_{GT} \\ N_{AA} & N_{TA} & N_{AT} & N_{TT} \end{pmatrix}$$

*Then, the $(k+1)^{th}$ - order $FCGR_{k+1}(s)$ can be obtained by replacing each element in $N_x$ in $FCGR_k(s)$ with four elements*

$$\begin{pmatrix} N_{CX} & N_{GX} \\ N_{AX} & N_{TX} \end{pmatrix}$$

The $k^{th}$-order FCGR of a DNA sequence $s$ concatenated with its reverse complement $s'$ is denoted by $FCGR_k(ss')$.

In 2005, Wang *et al*. [WHSK05] proposed the concept of a *spectrum of genomic signatures*. Some common features for genomic signatures were also discussed such as a) each genomic signature is a numerical matrix and can be visualized in a CGR, b) a positive integer number called *order* determines its granularity, and c) if the order is $k$, the numerical matrix has $2^k \times 2^k$ elements. Wang *et al*. [WHSK05] claimed that a $k^{th}$-order FCGR is equivalent to a CGR of resolution $1/2^k$, and if the resolution of a CGR is $1/2^k$ and the length of a DNA sequence is much longer than $k$, then the numbers of length $k$ oligonucleotide occurrences are the complete determinants of the CGR pattern. Both DRAP and FCGR were proposed as genomic signatures

by Wang *et al*. [WHSK05], and there is a direct relation between the two. Both the second order FCGR and DRAP have 16 elements that correspond to one dinucleotide. An element of a second order FCGR represents only the frequency of a specific dinucleotide. On the other hand, an element of a DRAP is the ratio of the dinucleotide frequency to the frequencies of the two single nucleotides composing this dinucleotide. Thus, an element of DRAP can be called a relative frequency, and DRAP can be defined as a second order relative FCGR and can be denoted as $rFCGR_2(s)$.

$$rFCGR_2(s) = \begin{pmatrix} \rho_{CC} & \rho_{GC} & \rho_{CG} & \rho_{GG} \\ \rho_{AC} & \rho_{TC} & \rho_{AG} & \rho_{TG} \\ \rho_{CA} & \rho_{GA} & \rho_{CT} & \rho_{GT} \\ \rho_{AA} & \rho_{TA} & \rho_{AT} & \rho_{TT} \end{pmatrix}$$

In a DRAP, all the elements are organized in an array; whereas in a second-order relative FCGR the same elements are organized in a matrix. So by organizing the elements of a DRAP as a second-order FCGR the similarity of a DRAP and a FCGR is revealed.

In [WHSK05], the authors proved that Goldman's conclusion does not always hold, using simulations. Figure 3.20 shows the counterexamples to the conclusions of [Gol93]. The three CGRs on the left column of the figure are plotted from human DNA sequence, human mtDNA sequence, and a *Neurospora crassa* mtDNA sequence respectively. The CGRs on the right column are plotted from the sequences constructed by simulating the length and single-nucleotide, dinucleotide, and trinucleotide frequencies of the corresponding sequences in the left column. Interestingly, the CGR images of each pair of sequences are not similar at all, despite the fact that the length, single-nucleotide, dinucleotide, and trinucleotide frequencies are the same. This result shows that Goldman's conclusion does not always hold.

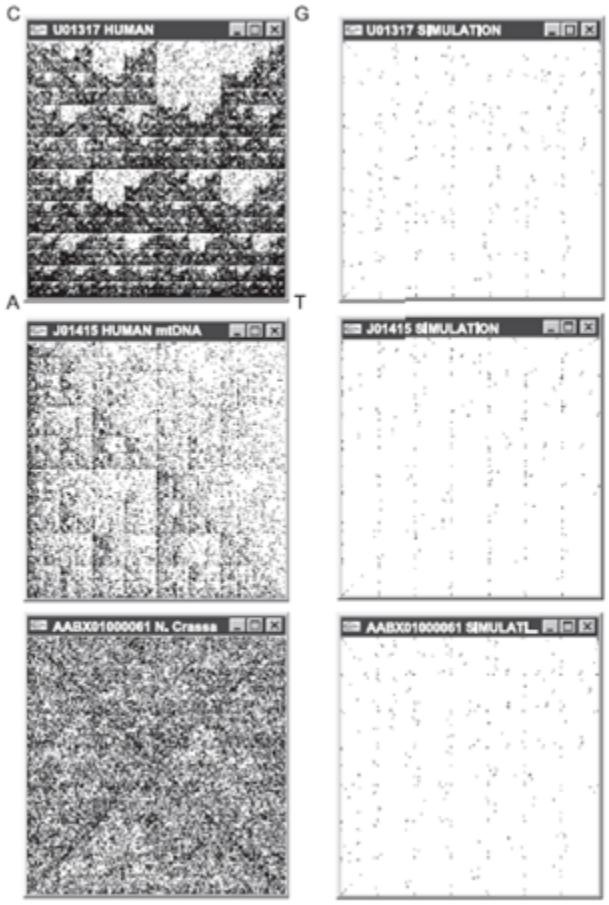



Figure 3.20: Counter example to Goldman's conclusion showed in [WHSK05].

Another version of CGR was proposed by Dunham *et al.* [DQW$^+$06] called Temporal Chaos Game Representation (TCGR). The major objective of TCGR was to deal with shorter sequences and analyzing the variability in distribution based on positions within the long sequence by using a sliding window. The term "temporal" refers to the starting point of a sliding window [DQW$^+$06]. Traditional CGR methods have no ability to deal with a set of short sequences, and are not suitable for evaluating any kind of changes that occur inside a genomic sequence. In order to overcome these imperfections of traditional CGR, TCGR was introduced. With experimental results on different groups of miRNAs, TCGRs showed the ability to visualize similarities and differences among the miRNA groups of viruses, nematodes, rodents and primates [DQW$^+$06]. As TCGR uses a sliding window to capture the variation of the distribution of nucleotides, it can be used to differentiate between the coding and non coding regions in the genome. The sliding window can also be useful in examining the internal structure of a

DNA sequence.

In 2007, Tavassoly *et al*. [TTR+07] proposed a 3D CGR on a cube to analyze coding and non coding regions in genomic sequences. The eight corners of the cube were used to map the four nucleotides of the coding regions and four nucleotides of non-coding regions of a DNA sequence. Afterwards, the points for each nucleotide were calculated with a similar approach of Algorithm 1 with an additional $z$ value for the $z$ axis. The resulting CGR image contains two parallel 2D CGRs with very few dots between them.

Another 3D-CGR to visualize a DNA sequence was proposed by Tu [Tu09] on a regular tetrahedron. A tetrahedron is a polyhedron with four vertices, six edges, and four triangle faces. A regular tetrahedron is one of the platonic solids with the faces all being equilateral triangles. The four nucleotide bases were placed at the four corners of the regular tetrahedron. The center of the tetrahedron is the starting point. The next steps are similar to the traditional CGR plotting. Figure 3.21 shows the 3D-CGR of human and mouse mtDNA genome plotted in [Tu09]. 3D-CGR is useful in exploring the complex structure of the whole genome, but in terms of complexity and application, it could not outperform 2D-CGRs.

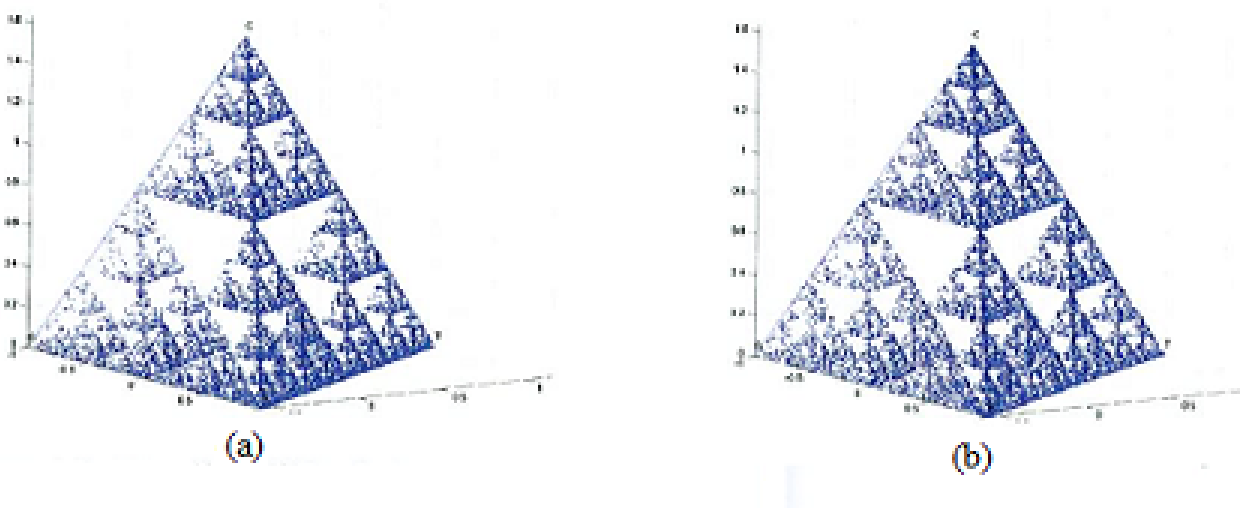



Figure 3.21: 3D-CGR images of mouse and human mtDNA sequence [Tu09].

### 3.10.1 Applications of CGR: Computing species' relatedness

CGR images show distinct patterns for unlike species and similar patterns for closely related species. Thus, by comparing CGR images, species' relatedness can be efficiently measured. Different methods have been proposed to compare CGR images of diverse species in literature [WHSK05, ACM+01, DGV+99], but the most efficient result was achieved by Wang *et al*. [WHSK05]. In [WHSK05], three different distance methods for calculating FCGR distance were implemented for 26 mitochondrial genomes from a diverse range of species. They compared their resulting phylogenetic trees with trees obtained using ClustalW and showed how FCGR can effectively outperform the traditional alignment based measurement of species' relatedness. The three distance methods compared are: a) Euclidean distance, b) image distance, and c) Pearson distance. Before calculating the distance between two FCGR matrices, they must first be standardized. The sum of all elements in a FCGR is proportional to the total number of base pairs in the DNA sequence. So, to make comparisons fair in terms of length and to be able to compare FCGRs obtained from DNA sequences of different lengths, we need to eliminate the length factor of FCGRs. This method of eliminating the sequence length parameter from FCGR is called standardization. A quantitative method of standardization was described in [WHSK05].

Suppose, $A$ is a $k^{th}$-order FCGR. From the definition of a FCGR, we know that $A$ is a $2^k \times 2^k$ matrix. Let $a_{i,j} (1 \leq i \leq 2^k, 1 \leq j \leq 2^k)$ be the elements of this matrix. If the standardized matrix of $A$ is $\bar{A}$, then

$$\bar{A} = \frac{4^k}{\sum_i \sum_j a_{i,j}} A.$$

Assuming that the elements in $\bar{A}$ are denoted by $b_{i,j}$, the following property holds:

$$\sum_{i=1}^{2^k} \sum_{j=1}^{2^k} b_{i,j} = 4^k.$$

This property implies that in a standardized $k^{th}$-order FCGR, the sum of all elements is equal to the number of elements, and therefore the average value of the elements of the FCGR matrix is 1. If the FCGRs are standardized, we can compare DNA sequences of different lengths. We will now look at some of the distance methods for comparing FCGR matrices.

**Definition 3** *(Euclidean distance). If* $\bar{A} = (a)_{2^k \times 2^k}$ *and* $\bar{B} = (b)_{2^k \times 2^k}$ *are two standardized* $k^{th}$*-order FCGRs, then we define Euclidean distance between* $\bar{A}$ *and* $\bar{B}$ *as:*

$$dE(\bar{A}, \bar{B}) = \frac{\sqrt{2^k}}{4^k} \sqrt{\sum_{i=1}^{2^k} \sum_{j=1}^{2^k} (a_{i,j} - b_{i,j})^2}.$$

The constant $\frac{\sqrt{2^k}}{4^k}$ is related to the definition of a standardized FCGR.

**Definition 4** *(Hamming distance). If* $\bar{A} = (a)_{2^k \times 2^k}$ *and* $\bar{B} = (b)_{2^k \times 2^k}$ *are two standardized* $k^{th}$*-order FCGRs, then we define Hamming distance between* $\bar{A}$ *and* $\bar{B}$ *as:*

$$dH(\bar{A}, \bar{B}) = \frac{1}{4^k} \sum_{i=1}^{2^k} \sum_{j=1}^{2^k} |a_{i,j} - b_{i,j}|.$$

There is an another method which compares two CGR images according to the image similarity between the two CGRs, called Image distance. This is an expansion of Hamming distance. Before defining the Image distance, we need the following definitions.

**Definition 5** *(Neighbourhood). Let A be a matrix* $A = (a)_{n \times n}$*, R be a positive integer, and* $(i, j)$ *be a pair where* $1 \leq i, j \leq n$*. A neighbourhood of radius R, centered at (i,j), denoted as* $\theta_R(i, j)$*, consists of all integer pairs (s,t), where* $1 \leq s \leq n, 1 \leq t \leq n$*,* $s \in [i - R, i + R]$*, and* $t \in [j - R, j + R]$*.*

**Definition 6** *(Density). For a matrix $A = (a)_{n\times n}$, the density matrix $(density_R(A))_{n\times n}$, where for any $(i, j)$, $1 \le i \le n, 1 \le j \le n$, is defined as:*

$$density_R(A)_{i,j} = \frac{\sum_{(s,t)\in\theta_R(i,j)} a_{s,t}}{\sum_{(s,t)\in\theta_R(i,j)} 1}.$$

Now the Image distance of two FCGRs can be defined as follows:

**Definition 7** *(Image distance). Suppose we have two $k^{th}$-order standardized FCGR matrices $\bar{A} = (a)_{2^k\times 2^k}$ and $\bar{B} = (b)_{2^k\times 2^k}$, then the Image distance between $\bar{A}$ and $\bar{A}$ is:*

$$dI_R(\bar{A}, \bar{B}) = \frac{1}{4^k} \sum_{i=1}^{2^k} \sum_{j=1}^{2^k} |density_R(\bar{A})_{i,j} - density_R(\bar{B})_{i,j}|.$$

Precisely, while we are calculating image distance, we are comparing two neighbourhoods of certain radius of the two different CGR images.

**Definition 8** *(Pearson distance). If we express FCGR A as an array $(x_i)_{(1\le i\le n)}$, and FCGR B as $(y_i)_{(1\le i\le n)}$, then the Pearson distance $dP(A, B)$ can be defined as:*

$$dP(A, B) = 1 - rw_{x,y},$$

where

$$nw = \sum_{i=1}^{n} x_i \cdot y_i,\ \bar{x}w = \frac{\sum_{i=1}^{n} x_i^2 \cdot y_i}{nw},\ \bar{y}w = \frac{\sum_{i=1}^{n} x_i \cdot y_i^2}{nw}$$
$$sx = \frac{\sum_{i=1}^{n} (s_i - \bar{x}w)^2 \cdot x_i \cdot y_i}{nw},\ sy = \frac{\sum_{i=1}^{n} (y_i - \bar{y}w)^2 \cdot x_i \cdot y_i}{nw}.$$

$$rw_{x,y} = \frac{\sum_{i=1}^{n} \frac{s_i - \bar{s}w}{\sqrt{sx}} \cdot \frac{y_i - \bar{y}w}{\sqrt{sy}} \cdot x_i \cdot y_i}{nw}.$$

In the case of Pearson distance, one advantage is that we do not need to standardize the FCGR matrices before calculating the distance. These distance methods can be effectively

used to compare different FCGRs, which can be then used to analyze phylogenetic evolutionary relations among diverse species. ClustalW is a tool used to align multiple sequences in order to compute their phylogenetic relations. In [WHSK05], FCGRs of 26 mitochondrial sequences were compared with the four distance method discussed here. The authors compared the phylogenetic trees generated by comparing the FCGRs of those 26 species computed by the various distance methods. They also compared the result with the conventional ClustalW tool. Table 3.2 shows the dataset chosen in [WHSK05]. The tree generated using Euclidean distance was the best among all in terms of species' relatedness. This tree can distinguish all the vertebrates from other organisms. The invertebrates are also differentiated from all other organism with the exception of yeast B. The other trees generated using the image distance, Pearson distance, and ClustalW did not produce results as accurate as the Euclidean distance.

## Summary

We have seen how FCGR can be constructed and how the calculation of the distances among different FCGRs can be used to generate phylogenetic trees for diverse species. ClustalW was one of the widely used alignment tools used to analyze phylogenetic relationship, but FCGR shows strong evidence in outperforming it. Thus, CGR can be an efficient and effective tool for measuring species' relationships.

One important point to consider while constructing FCGR is the value of the order $k$. Experimental results show [WHSK05] that the higher order FCGR are preferable to the lower ones, but no definitive conclusion was made. Also, when increasing the order of FCGR, after some point, higher order FCGR does not give better result than the lower ones. The increment of order increases the time and space complexity exponentially. As there is no theoretical or mathematical conclusion on what should be the optimal order for FCGR, in [WHSK05], it was empirically proposed that the value for $k = 10$ is an upper bound for the order of FCGR. The authors recommended choosing a value of $k$ between 1 and 10. A $k^{th}$-order FCGR becomes sparse with the increase of $k$, and thus, higher values of $k$ should be avoided. The FCGR matrices are not sparse as long as $k \leq 10$, when the DNA sequence is long enough (10,000 nt).

| Genome Number | Accession number | Name | Short name | mtDNA length |
|---|---|---|---|---|
| 1321 | NC_012920 | *Homo sapiens* | human | 16569 |
| 2514 | NC_006853 | *Bos taurus* | cow | 16338 |
| 2757 | NC_005089 | *Mus musculus* | mouse | 16299 |
| 3012 | NC_001329 | *Podospora anserina* | fungus | 100314 |
| 3047 | NC_001324 | *Paramecium aurelia* | protozoan | 40469 |
| 3070 | NC_001224 | *Saccharomyces cerevisiae* S288c | yeast B | 85779 |
| 3079 | NC_001327 | *Ascaris suum* | roundworm | 14284 |
| 3081 | NC_001453 | *Strongylocentrotus purpuratus* | urchin B | 15650 |
| 3082 | NC_001620 | *Artemia franciscana* | shrimp | 15822 |
| 3102 | NC_001321 | *Balaenoptera physalus* | whale A | 16398 |
| 3103 | NC_001322 | *Drosophila yakuba* | fruitfly | 16019 |
| 3104 | NC_001323 | *Gallus gallus* | chicken | 16775 |
| 3105 | NC_001325 | *Phoca vitulina* | seal B | 16826 |
| 3106 | NC_001566 | *Apis mellifera ligustica* | honeybee | 16343 |
| 3107 | NC_001572 | *Paracentrotus lividus* | urchin A | 15696 |
| 3109 | NC_001601 | *Balaenoptera musculus* | whale B | 16402 |
| 3110 | NC_001602 | *Halichoerus grypus* | seal A | 16797 |
| 3111 | NC_001606 | *Cyprinus carpio* | carp | 16575 |
| 3112 | NC_001610 | *Didelphis virginiana* | opposum | 17084 |
| 3116 | NC_001665 | *Rattus norvegicus* | rat | 16313 |
| 3122 | NC_001717 | *Oncorhynchus mykiss* | trout | 16642 |
| 3123 | NC_001727 | *Formosania lacustris* | loach | 16558 |
| 3155 | NC_002084 | *Anopheles gambiae* | mosquito | 15363 |
| 3172 | NC_001613 | *Prototheca wickerhamii* | alga | 55328 |
| 3174 | NC_001660 | *Marchantia polymorpha* | plant | 186609 |
| 3175 | NC_001326 | *Schizosaccharomyces pombe* | yeast A | 19431 |

Table 3.2: The dataset of mitochondrial genomes used by Wang *el al.* [WHSK05].

## 3.11 SSIM: Measuring species' relatedness with CGR

In previous sections, we have seen how the CGR of a genome can give detailed information and insight about the genome and how making efficient comparison among CGRs of different genomes is capable of providing relatedness analyses of those genomes. This section gives a brief discussion of an image comparison method that can be efficiently used to compare CGRs of various genomes. The method is called *Structural Similarity* (SSIM) index. This method was proposed in [WBSS04] to asses the similarity of two different images. Figure 3.22 shows the system diagram of SSIM.

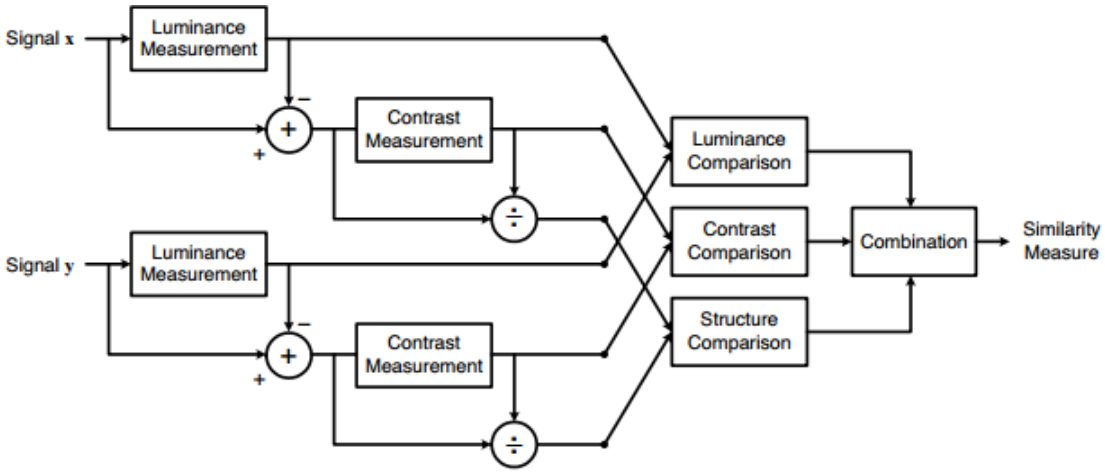



Figure 3.22: Diagram of the structural similarity (SSIM) measurement system.

Precisely, suppose we have two non-negative image signals $x$ and $y$, which are aligned to each other. If we consider that one of the images has perfect quality, then the similarity measure of these two images can serve as a quantitative measurement of the quality of the second signal. The task of the similarity measurement is achieved by three comparison steps: luminance, contrast and structure. First, the luminance of each signal is compared. The function of the luminance comparison, denoted by $l(x,y)$ is a function of $\mu_x$ and $\mu_y$, where $\mu$ is the mean intensity and can be defined as

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i.$$

Consequently $l(x,y)$ is defined as

$$l(x,y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}.$$

Here, the constant $C_1$ is included to avoid instability when $\mu_x^2 + \mu_y^2$ is very close to zero. In [WBSS04], the value of the constant $C_1$ was chosen as follows

$$C_1 = (K_1 L)^2,$$

where $L$ is the dynamic range of the pixel values (255 for 8-bit grayscale images), and $K_1$ is a small constant. Similar considerations also apply to contrast comparison and structure comparison described later.

In the second step, the mean intensity is removed from the signal. The resulting signal $(x - \mu_x)$ corresponds to the projection of vector $x$ onto the hyperplane defined by

$$\sum_{i=1}^{N} x_i = 0.$$

The standard deviation is used as an estimate of the signal contrast. An unbiased estimate in discrete form is given by

$$\sigma_x = \left(\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)^2\right)^{1/2}.$$

The contrast between two signals $x$ and $y$ is then compared by the following definition

$$c(x,y) = \frac{2\sigma_x \sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2},$$

where $C_2$ is another constant and defined by $C_2 = (K_2 L)^2$, used to avoid instability when $\sigma_x^2 + \sigma_y^2$ is very close to zero.

In the third step, the signal is normalized by its own deviation, so the two signals being compared have unit standard deviation. Afterwards, the structure comparison $s(x,y)$ is conducted on these normalized signals $(x - \mu_x)/\sigma_x$ and $(y - \mu_y)/\sigma_y$. The definition of $s(x,y)$ is as follows:

$$s(x,y) = \frac{2\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3},$$

where the constant $C_3$ is used to avoid instability when $\sigma_x \sigma_y$ is very close to zero, and $\sigma_{xy}$ is defined as

$$\sigma_{xy} = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)(y_i - \mu_y).$$

Finally, the three components are combined to output the overall similarity measure that can be defined as

$$\text{SSIM}(x,y) = \frac{(2\mu_x\mu_y+C_1)(2\sigma_{xy}+C_2)}{(\mu_x^2+\mu_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}.$$

Here, $C_2$=$C_3$.

To achieve a single overall similarity measure of the entire image, a mean SSIM (MSSIM) index is used, and defined as

$$\text{MSSIM}(X,Y) = \frac{1}{M}\sum_{j=1}^{M}\text{SSIM}(x_j,y_j).$$

where $X$ and $Y$ are the reference and the distorted images, respectively; $x_j$ and $y_j$ are the image contents at the $j^{th}$ local window; and $M$ is the number of local windows of the image. The three components (luminance, contrast, and structure) are relatively independent. The detailed analysis of all the terms of SSIM can be found in [WBSS04].

To summarize, the *Structural Similarity* (SSIM) index is an image distance measure that is widely used in the context of image processing and computer vision to compare two gray images from the point of view of their structural similarities [WBSS04]. SSIM was designed to perform most similarly to the human visual system, which is highly adapted to extract structural information from viewing.

The theoretical range of the MSSIM distance is $[-1,1]$, with the distance being 1 between two identical images, 0 between a black image and a white image, and -1 if the two images are negatively correlated, that is, $\text{MSSIM}(X,Y) = -1$ if and only if every pixel of image $X$ has the inverted value of the corresponding pixel in $Y$. To compute dissimilarity between two CGR images, we first compute $\text{MSSIM}(X,Y)$. In this thesis, the dissimilarity for the CGR images, DMSSIM is calculated performing $1 - \text{MSSIM}(X,Y)$. As a result, for our case the range of the DMSSIM is $[0,2]$. More precisely, if we have two identical CGRs then the DMSSIM will return a distance of 0, if we have one black and a white CGR image, the DSSIM will give a distance of 1, and if the two CGR images are negatively correlated then the DSSIM distance will be 2. For the dataset of genomic CGR images used in this thesis, all distances range between 0 and 1.

Using SSIM as a distance calculation method to compare two CGR images has some advantages. The value of the order $k$ in the calculation of FCGR [WHSK05] is no longer required. The increment of $k$ increases the computation complexity of FCGR exponentially. All of these problems are bypassed if we use SSIM to compare two CGR images. Moreover, SSIM normalizes the image signals with standard deviation, which implies we no longer have to normalize the CGR images separately.

SSIM is a sensitive comparison method, as will be seen in subsequent sections. However, one problem with SSIM is that the distance calculated by SSIM for a set of images may not satisfy the triangular inequality property.

## Summary

We have seen how to produce CGR images for genomes and discussed an image comparison method to compare two CGR images. As a result, we can generate a dissimilarity or similarity matrix for any number of given genomes using their CGRs and comparing them with SSIM. Next, we want to display the relatedness. A method called Multidimensional Scaling (MDS) will be employed to display species relatedness in a 2D Euclidean space. The following section briefly describes MDS.

## 3.12 Multi Dimensional Scaling (MDS)

MDS has been used for the visualization of data relatedness in various fields such as cognitive science, information science, psychophysics, psychometrics, marketing, and ecology [BG10]. MDS takes as input a distance matrix containing the pairwise distances between $n$ given items and outputs a plot wherein each item is represented by a point, and the geometric distances between points are a linear function of the distances between the corresponding items in the distance matrix. In a classic example [CC01], if we have distances among 10 different north American cities, MDS generates the map of these cities taking the distances as input. To illustrate more precisely, if we have the following distance matrix for the ten North American cities showed in Table 3.3, then MDS will plot the map of these cities as shown in Figure 3.23.

| | London | Toronto | Vancou-ver | Chicago | New York | Seattle | Montreal | Washing-ton Dc | Detroit | Texas |
|---|---|---|---|---|---|---|---|---|---|---|
| London | 0 | 192 | 4181 | 647 | 862 | 3965 | 721 | 847 | 199 | 2379 |
| Toronto | 192 | 0 | 4371 | 837 | 790 | 4155 | 543 | 775 | 372 | 2557 |
| Vancou-ver | 4181 | 4371 | 0 | 3537 | 4086 | 227 | 4900 | 4660 | 3989 | 3324 |
| Chicago | 647 | 837 | 3537 | 0 | 1271 | 3321 | 850 | 1126 | 455 | 1813 |
| New York | 862 | 790 | 4086 | 1271 | 0 | 4590 | 595 | 366 | 988 | 2846 |
| Seattle | 3965 | 4155 | 227 | 3321 | 4590 | 0 | 4685 | 4444 | 3773 | 3108 |
| Montreal | 721 | 543 | 4900 | 850 | 595 | 4685 | 0 | 944 | 901 | 3086 |
| Washing-ton DC | 847 | 775 | 4660 | 1126 | 366 | 4444 | 944 | 0 | 845 | 2493 |
| Detroit | 199 | 372 | 3989 | 455 | 988 | 3773 | 901 | 845 | 0 | 2189 |
| Texas | 2379 | 2557 | 3324 | 1813 | 2846 | 3108 | 3086 | 2493 | 2189 | 0 |

Table 3.3: Distance in kilometers among ten North American cities obtained from Google map (not real straight line distances).

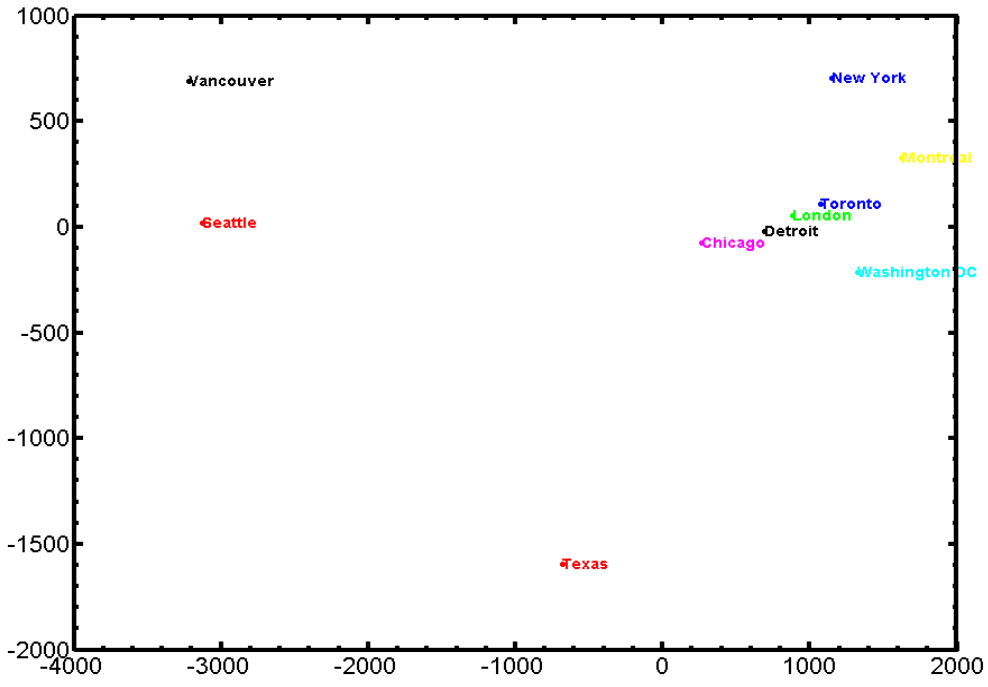



Figure 3.23: MDS plot of the ten cities from Table 3.3.

The application of MDS is not limited to reconstructing maps. MDS can be used with a wide range of dissimilarities and similarities arising from various situations. It covers any method which produces a graphical representation of objects from multivariate data. For example, dissimilarities obtained from analyzing DNA sequences could be used for an interrelationship study for various taxonomic groups. First let us look into several data types that can be analyzed by MDS.

### 3.12.1 Types of data

Variables in the analysis of similarity and dissimilarity of objects are classified according to their "measurement scale". The different scales are the nominal scale, the ordinal scale, the interval scale, and the ratio scale. The following subsections will briefly discuss these scales.

***Nominal scale***

Nominal data are classificatory. Nominal scale refers to quality more than quantity. A nominal level of measurement is simply a matter of distinguishing by name, e.g., 1 = male, 2 = female. Even though we are using the numbers 1 and 2, they do not denote quantity. The binary category of 0 and 1 used for computers is a nominal level of measurement. They are categories or classifications. We can think about some more examples such as meal preference: vegetarian, non-vegetarian; religious affiliation: 1 = Buddhist, 2 = Muslim, 3 = Christian, 4 = Jewish, 5 = Other; political orientation: Republican, Democratic, Libertarian, Green etc..

***Ordinal scale***

Data on the ordinal scale can be ordered, but are not quantitative. Ordinal scale refers to order in measurement. For instance, rank first can be judged to be better than rank seven. More examples, level of speed: slow, medium, fast; alignment: left, center, right etc..

***Interval scale***

Data on the interval scale are quantitative data, where the numerical difference between two values is meaningful. An example of an interval scale is temperature, either measured on a Fahrenheit or Celsius scale. A degree represents the same underlying amount of heat, regardless of where it occurs on the scale. Measured in Fahrenheit units, the difference between a temperature of 46 and 42 is the same as the difference between 72 and 68. Another example of interval scale is time of day on a 12-hour clock.

***Ratio scale***

Data measured on the ratio scale is similar to that on the interval scale, with the exception of having a meaningful zero point, for instance, weight, height, temperature recorded in degrees Kelvin. We can define two more examples, number of children: 3; height : 173 cm.

So, we have seen the different data scales that can be modelled by MDS. Depending on variations of data types there are different models of MDS. The different MDSs are Classical MDS, Metric MDS, Non-metric MDS, and Generalized MDS.

Furthermore, some more types can be found in [CC01] such as Metric Least Square Scaling, Procrustes analysis, Unidimensional Scaling, Biplots, Unfolding etc. Different models of MDS are used based on the data types. For instance, if the distances in the configuration space are to be Euclidean and the dissimilarities are precisely Euclidean distances, then Classical MDS model is used. Both the Classical MDS and Metric Least Square Scaling are example of metric scaling, where the term metric means that the dissimilarity of data is a metric. In contrast, non-metric MDS is used if we have a non-metric dissimilarity matrix. For example, for ordinal data, non-metric MDS is used, where the rank order of the dissimilarities has to be preserved. In this thesis, the distance we have, using the CGR and comparing with SSIM method, is a metric, that is it satisfies a) non-negativity, b) identity of indiscernibles, c) symmetry, and d) triangular inequality . The data falls in the category of *"ratio scale"*. As a result, for our proposed method, the classical MDS has been used. Let us have a brief look at the classical MDS.

**Classical MDS**

Classical MDS receives as input a parameter $n$, corresponding to a set of $n$ items, and an $n \times n$ distance matrix that contains the pairwise distances between any two items in the set:

$$\triangle = \begin{pmatrix} \delta_{1,1} & \delta_{1,2} & . & . & \delta_{1,n} \\ \delta_{2,1} & \delta_{2,2} & . & . & \delta_{2,n} \\ . & . & . & . & . \\ . & . & . & . & . \\ \delta_{n,1} & \delta_{i,2} & . & . & \delta_{i,n} \end{pmatrix}$$

The output of classical MDS consists of $n$ points in a $k$-dimensional space, whose pairwise Euclidean distances are similar to the distances between the corresponding items in the input distance matrix. More precisely, MDS will return $n$ points $x_1, x_2 \ldots, x \in R^k$ such that

$$\|x_i - x_j\| \approx \delta_{(i,j)}$$

for all $i, j \in n$.

Here, $k$ can be at most $n - 1$, and the points are recovered from the eigenvalues and eigenvectors of the input $n \times n$ distance matrix [CC01]. In this thesis, we have used the first two sets of points returned by MDS (*i.e* value of $k = 2$).

In general, the purpose of MDS is to provide a visual representation of the pattern of proximities among a set of objects. MDS has not been widely used so far in the field of molecular biology. It was used in [Les90] for the analysis of the geographic genetic distribution of some natural populations and, in [HCBD03], to provide a graphical summary of the distances among the CO1 genes from various species. There are some advantages of the MDS to represent species' relatedness over phylogenetic trees. The representation and construction of the conventional phylogenetic trees comes with some limitations. In a phylogenetic tree, branches can be rotated. As a consequence, the adjacency of two species-representing leaves is not always informative. We can visually see and determine if two nodes are closer but it is difficult to see how much difference is there when the two nodes are far away from each other.

Though MDS can be efficiently used for similarity or dissimilarity analysis, it comes with some limitations. Firstly, the points $x_i$ are not unique. Indeed, one can translate or rotate the

main map without affecting the pairwise Euclidean distances $\|x_i - x_j\|$. In addition, when more data items are added to the input set, the obtained points in an MDS map may change coordinates as the output of the MDS aims to preserve only the pairwise Euclidean distance between points, and this can be achieved even when some of the points change their coordinates. Secondly, the method does not work properly for a dataset that contains less than eight data points. Moreover, for too many data points, the plot can be cumbersome to visualize. Lastly, each MDS map has some error, which we will define by the term *"Stress"*. The *Stress* is in general lower for an MDS map in a higher-dimensional space when using the same dataset. In this thesis, stress defined in [Kru64] has been used to study the errors of different MDS plots. The *Stress-1* (Kruskal stress, [Kru64]) is defined as follows:

$$\text{Stress-1} = \sigma_1 = \sqrt{\frac{\Sigma_{i<j}[f(\delta_{i,j}) - d_{i,j}]^2}{\Sigma_{i<j} d_{i,j}^2}}.$$

Here, $d_{i,j}$ is the Euclidean distance between $x_i$ and $x_j$.

The linear function $f$ is used to determine the relation between $\delta_{i,j}$ and $d_{i,j}$. To calculate the function $f$ of $f(\delta_{i,j})$, we have calculated two coefficients $a$ and $b$ with the use of linear regression and define $f(\delta_{i,j}) = a \times \delta_{i,j} + b$. Using a built-in Matlab function *ployfit*, the best line of fit is calculated for the pairwise distances of the points over the associated original distance in the input distance matrix and the value of $a$ and $b$ are retrieved . The function is defined as follows, $p = polyfit(x, y, n)$ finds the coefficients of a polynomial $p(x)$ of degree $n$ that fits the data, $p(x(k))$ to $y(k)$, in a least squares sense. For our case, $x_k = \delta_k$, $y_k = d_k$, and $n = 1$.

## Summary

In this chapter, some of the 2D and 3D representation methods for a DNA sequence were discussed. In addition, CGR and different applications of CGR images were described. At the end, an image comparison method SSIM, and a method, called MDS, to represent species relatedness in an Euclidean space were introduced. In the next chapter, Genome Distance Maps will be introduced that use CGR, SSIM and MDS to give a visual representation of species' interrelationships.

# Chapter 4

# Proposed method and results

## 4.1 Quantitatively measuring DNA sequence distances and displaying the inferred genome relatedness

A new method to visualize species' relatedness in a common Cartesian coordinate system is proposed in this chapter. Herein, the novel combination of CGR, SSIM and MDS is used to implement the method. More precisely, if we want to compare genomes from a diverse set and display their relatedness, we can use this method very efficiently. The step by step operation can be summarized according to Algorithm 2.

**Algorithm 2** Steps to produce Genome Distance Map (GDM)

Input : Input: $n$ DNA sequences

Output: $n$ points in a 2D Euclidean space

1. Compute the CGRs of the sequences using Algorithm 1.
2. Compare the CGRs generated in Step 1 with SSIM and produce an $n \times n$ distance matrix.
3. Apply MDS to the distance matrix produced in Step 2 and generate $n$ vectors of dimension $(n-1)$.
4. Take the first two components of each vector in Step 3 and plot the $n$ points in a 2D space that have them as coordinates.

This method can take any set of DNA sequences as input and output the interrelation-

ship in a 2D map between the sequences. For instance, if we give 10 accession numbers of 10 mitochondrial genomes from NCBI, the final output will be a 2D plot that contains 10 points representing the corresponding genomes. We will call the plot a *Genome Distance Map* (GDM).

In this thesis, every sequence has been downloaded from NCBI (url:http://www.ncbi.nlm.nih.gov/) and only mtDNA sequences are analyzed. The last updated dataset as of July 12, 2012 for all mtDNA genomes is used for all the experiments and simulations.

A total of 3,176 mtDNA sequences were contained in this dataset. An excel file has been created describing the entire dataset along with specific information for each mitochondrial genome. The information includes the accession number, sequence length, biological name and taxonomic description from kingdom to genus. For an efficient and consistent analysis, a unique number was assigned to each mtDNA sequences. For instance, if the number 1321 represents the mitochondrial genome of the *Homo sapiens* (human), then in every GDM, 1321 will be representing the mtDNA sequence of the *Homo sapiens*. We will use the abbreviation GN (Genome Number) to refer to the associated number for a genome. Furthermore, in every plot, genomes are given different colors according to their biological classifications.

Each of the map comes with legends containing useful biological and mathematical information about the map, such as total number of sequences from each taxonomy. For readability purposes, all maps are scaled so that the $x$- and the $y$- coordinates always span the interval $[-1, 1]$. To make the interval consistent for all the maps, the following scaling formula is used $x_{sca} = 2 \cdot (\frac{x - x_{min}}{x_{max} - x_{min}}) - 1$, $y_{sca} = 2 \cdot (\frac{y - y_{min}}{y_{max} - y_{min}}) - 1$, where $x_{min}$ and $x_{max}$ are the minimum and maximum of the $x$-coordinates of all the points in the map, and similarly for $y_{min}$ and $y_{max}$.

Furthermore, the *Stress* or error for each of the plots as discussed in Section 3.12 is calculated. After taking the first two sets of points returned by MDS, the pairwise distances between every pair is calculated and consequently $d_{i,j}$ is computed. Afterwards, applying the the *polyfit* function to the original DSSIM distances, we have generated $a$ and $b$ for each entries of $d_{i,j}$ over $\delta_{i,j}$. Subsequently, the the $f(\delta_{i,j})$ was calculated by the following formula $f(\delta_{i,j}) = \delta_{i,j} \times a + b$. At the end the Stress-1 for each map was computed.

## 4.2 Genome Distance Maps

From Algorithm 2, we can see that the combination of CGR, SSIM, and MDS applied to a particular set of genomic sequences yields a so-called *Genome Distance Map*, which visually illustrates the quantitative relationships and patterns of proximities among the given genomic sequences and the species they represent.

In a Genome Distance Map, we can display relatedness between any two sequences as well as among a set of sequences. We have applied this method to experiment species' relationship at different taxonomic levels. For example, Figure 4.1 shows the Genome Distance Map for all vertebrate mitochondrial sequences in our database. In this map and all other Genome Distance Maps in this thesis, the numbers refer to the associated Genome Numbers for the mitochondrial sequences of our database.

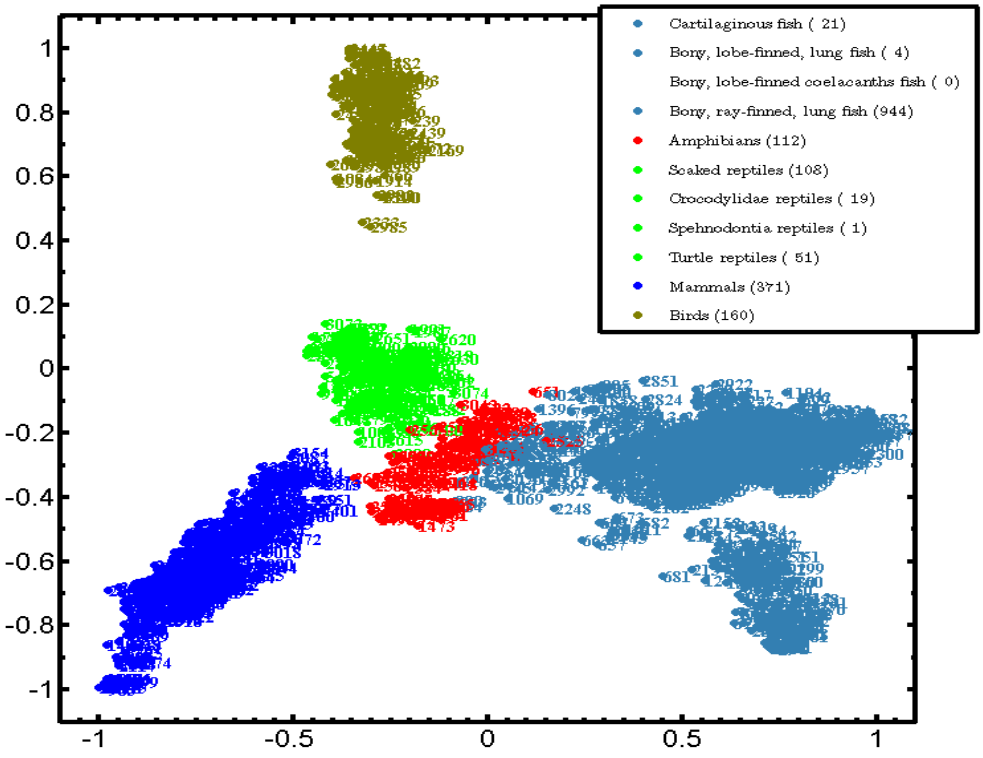



Figure 4.1: Genome Distance Map of the phylum Vertebrata, with its five subphyla: mammals, amphibians, reptiles, birds and fishes.

Figure 4.1 displays the five different vertebrates clusters Birds, Fishes, Amphibians, Mammals, and Reptiles, all clearly separated. By the term clear separation, it is meant that the different colored clusters are not mixed up with any other cluster. In this figure, a total of 371 mammals, 112 amphibians, 179 reptiles, 969 fishes and 160 birds (a total of 1791 organisms) are represented. The *Stress* for this particular figure is 0.12. To make the map consistent with all other maps, we did some scaling. The original $x_{min}$, $x_{max}$, $y_{min}$, $y_{max}$ are -0.16172, 0.19668, -0.1396, and 0.2312 respectively.

## 4.3 Genome Distance Map of all eukaryotes

To begin with, we first have the GDM of all eukaryotes (the entire database) shown in Figure 4.2.

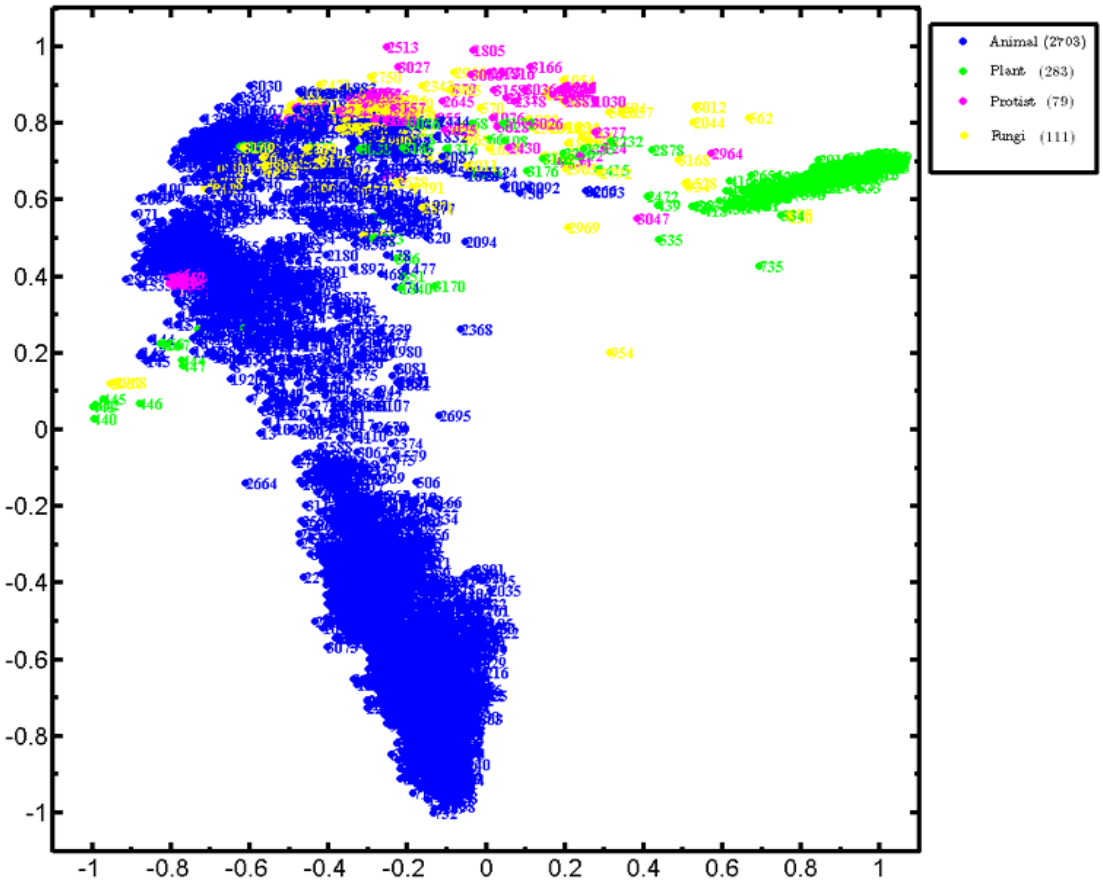



Figure 4.2: Genome Distance Map of all eukaryotes.

This is the map at the kingdom level. We have four different kingdoms with a certain number of representatives in the database. The database contains mitochondrial DNA sequence of 2703 animals, 283 plants, 79 protists, and 111 fungi. This map is dominated by the total number of animals. We can see the plants are not all grouped together. This may be due to the high total number of species in this map and the over representation of the kingdom Animalia. We have a total 3,176 organisms in this plot. The $x_{min}$, $x_{max}$, $y_{min}$, $y_{max}$, and *Stress* for the map are -0.2692, 0.44666, -0.18498, 0.2175, and 0.142 respectively.

## 4.4 Across classifications

We next applied our method to investigate large scale differences, plotting together mitochondrial genomes across classifications.

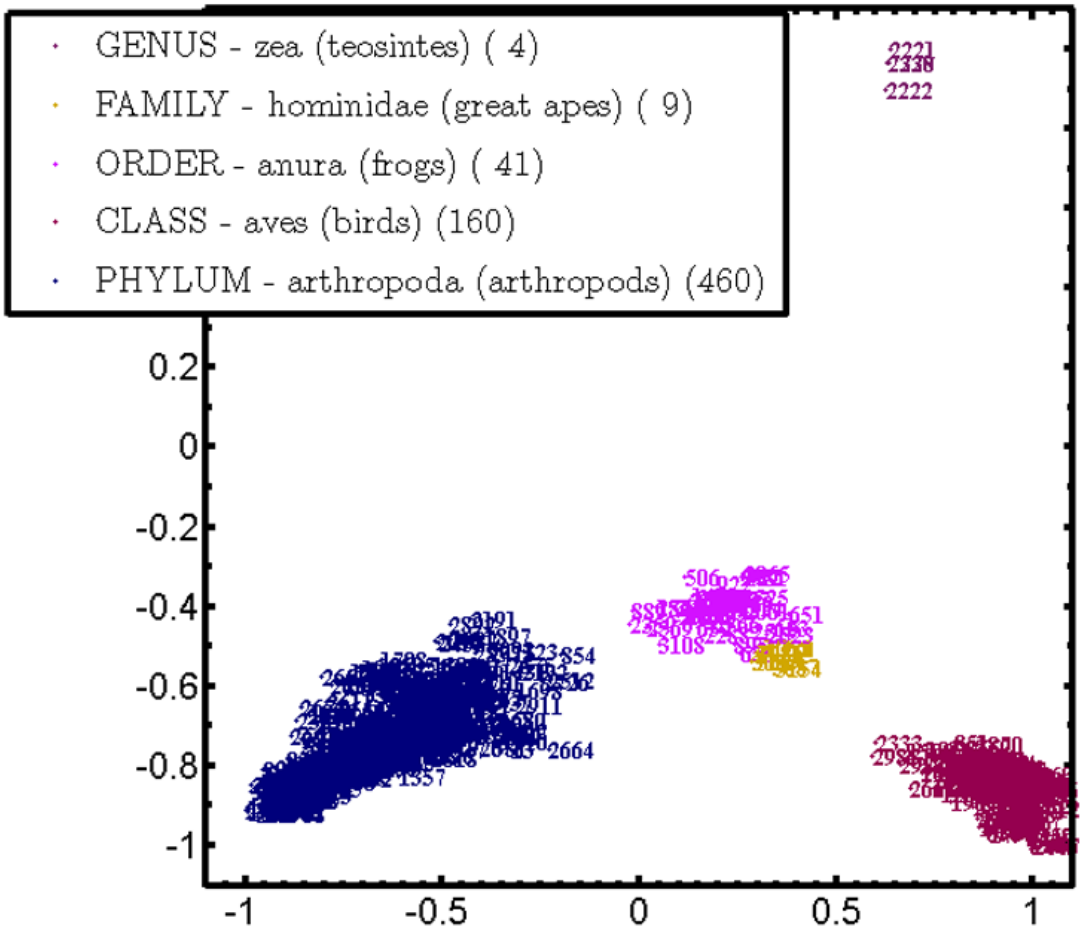



Figure 4.3: Genome Distance Map across classifications.

The map in Figure 4.3 comprises all represented species from a genus (Zea, plants, 4 species), a family (Hominidae, great apes, 9 species), an order (Anura, frogs, 41 species), a class (Aves, birds, 160 species) and a phylum (Arthropoda, 460 species).

Figure 4.3 shows that our method can distinguish species across different classifications. All species from the genus, family, order and class are grouped together in a separate cluster. The main motivation of plotting this dataset was to observe effectiveness of our method to analyze large scale phylogeny. We took representatives from the genus to the phylum level and succeeded to show applicability of the method over the hierarchy of the biological classification.

The information for this map is as follows, total number of organisms 674 (4 plants, 41 frogs, 160 birds and 460 arthropods), $Stress = 0.15504$, $x_{min} = -0.16392$, $x_{max} = 0.28038$, $y_{max} = 0.65699$, $y_{min} = -0.099185$.

## 4.5 Three classes : Amphibia, Insecta and Mammalia

Subsequently, we compared different classes and plotted together all available mitochondrial genomes from three classes, Amphibia, Insecta and Mammalia (Figure 4.4).

The three classes are grouped together without any kind of mix ups with another class. As a consequence, we can claim our proposed method can be successfully applied where relatedness analyses at class level is required.

The total number of organisms in Figure 4.4 is 790 (307 insects, 371 mammals, and 112 amphibians), $Stress = 0.16291$, $x_{max} = 0.187$, $x_{min} = -0.19862$, $y_{max} = 0.18092$, $y_{min} = -0.19862$.

On a finer scale, in the next subsections we applied this method to observe relationships within a Class: Class Amphibia and its three orders in Figure 4.5, Class Insecta and its major orders in Figure 4.6, Class Mammalia with primates highlighted in Figure 4.7.

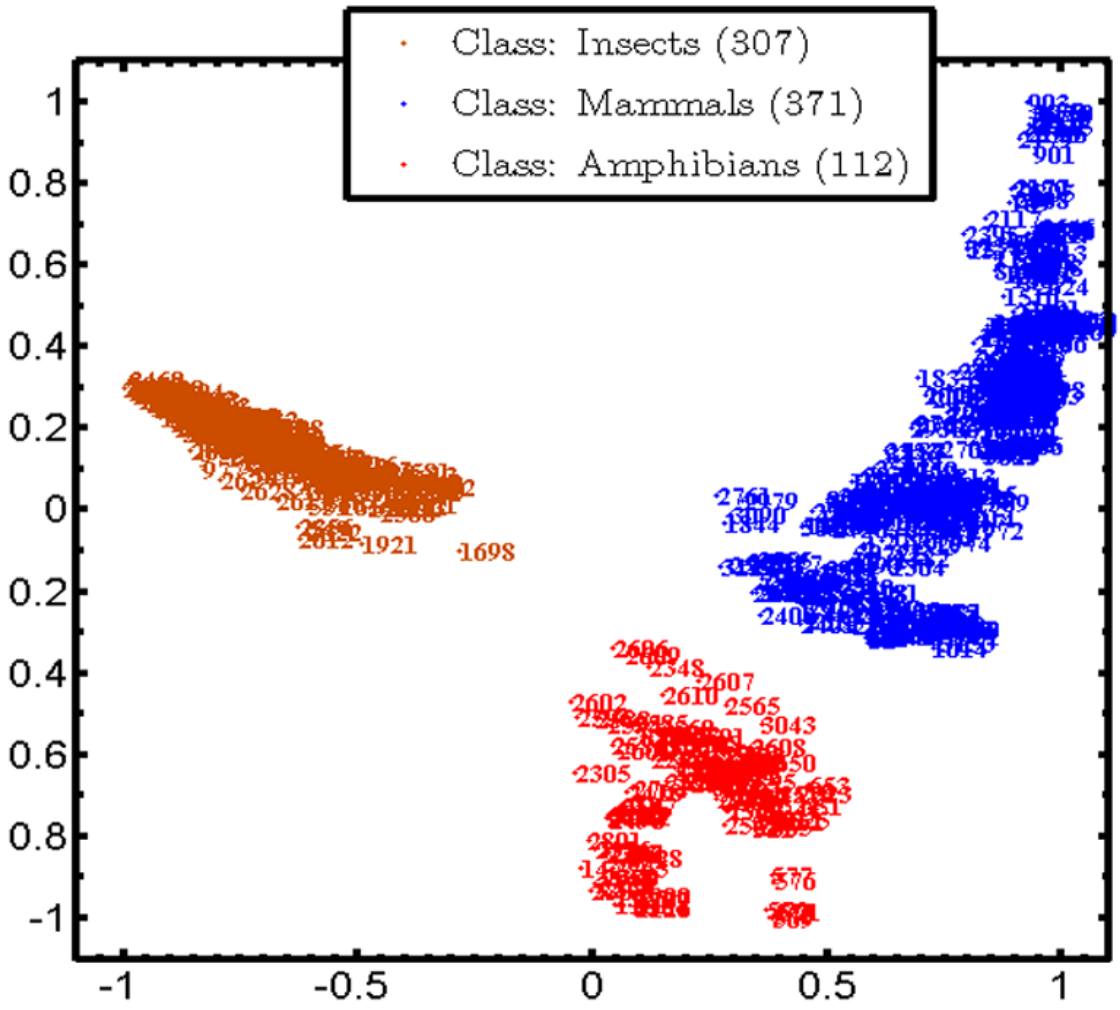



Figure 4.4: Genome Distance Map of three classes: amphibians, insects, and mammals.

## 4.6 Class Amphibia and its three orders

The dataset for Figure 4.5 consists of all the 112 amphibian mtDNA represented in our dataset, classified into three groups: 8 Caecilians (order Gymnophiona), 41 Frogs (Order Anura), and 63 Newts/Salamanders (order Caudata).

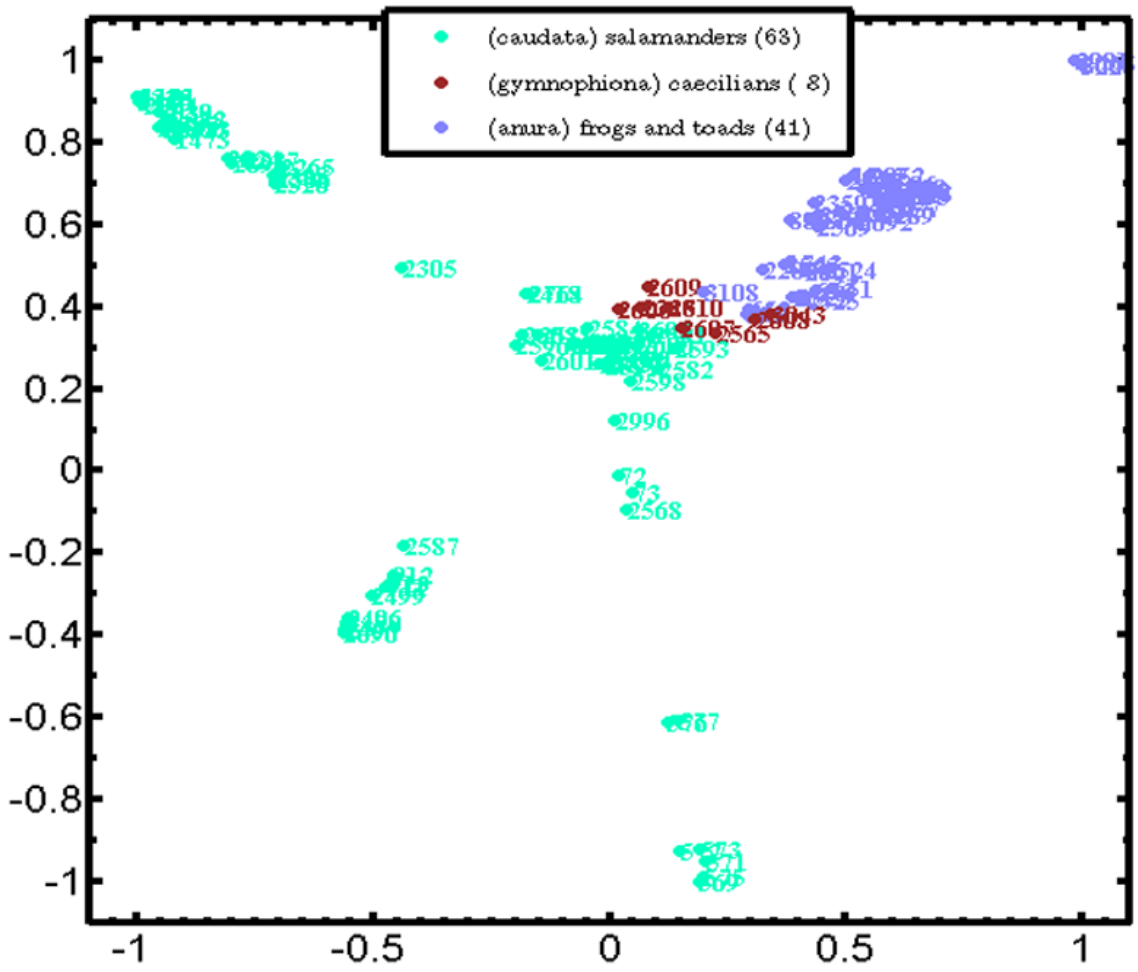



Figure 4.5: Genome Distance Map of the Class Amphibia and its three orders: Gymnophiona, Anura, and Caudata.

The map contains three separate clusters with little overlaps of the order Gymnophiona with the other two orders. These mix ups may not be anomalies. As the total number species of the order Gymnophiona is lower than the other two orders and the SSIM distances among these orders are very low, these 8 Gymnophionas took position in the middle of the other two orders. The numerical characteristics for this figure are $Stress = 0.16973$, $x_{max} = 0.23309$, $x_{min} = -0.23772$, $y_{max} = 0.1445$, $y_{min} = -0.32096$.

## 4.7 Class Insecta and its major orders

The dataset of Figure 4.6 consists of the mtDNA of the 307 insect species in the database, 55 from the order hemiptera (true bugs), 23 from the order isoptera (termites), 37 from the order coleoptera (beetles), 51 from the order diptera (true flies), 55 from the order lepidoptera (lepidopterans), 19 from the order hymenoptera (sawflies, wasps, bees and ants), and 52 from the order orthoptera (grasshoppers, crickets, weta and locusts). In addition, it contains 3 Mecopteras, 3 Ephemeropteras, 2 Odonatas, 3 Thysanuras and 4 Archaeognathas. *Stress* for this map is 0.13896, $x_{min} = -0.13514$, $x_{max} = 0.23813$, $y_{min} = -0.23779$, $y_{max} = 0.07835$.

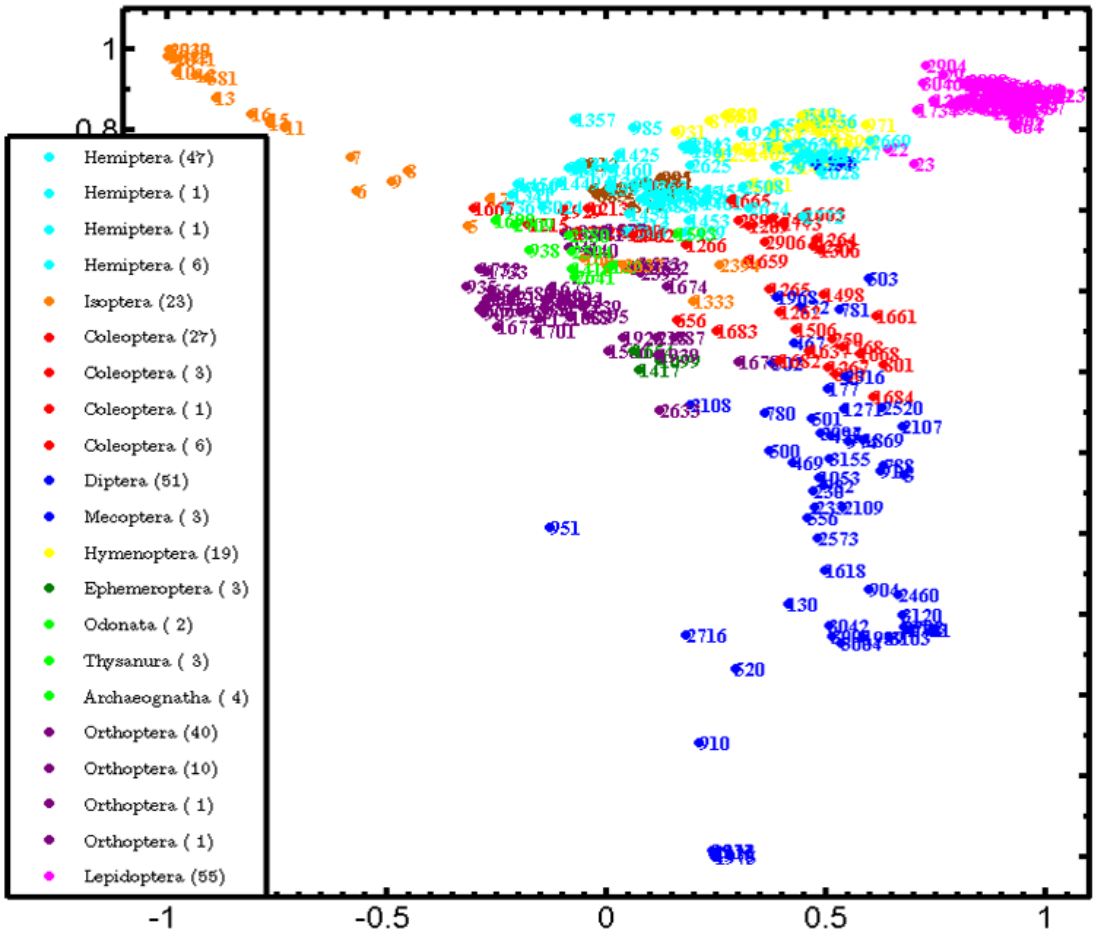



Figure 4.6: Genome Distance Map of the Class Insecta and its major orders.

## 4.8 Class Mammalia with primates highlighted

Figure 4.7 displays the genome distance map of all mammal mtDNA genomes, with the primates highlighted. The map in Figure 4.7, has a total of 371 mammals, out of which 62 are primates. The shape of the map takes on a shape of three outstretched arms, with the primates occupying a distinct arm from the other mammals. The $x_{min}, x_{max}, y_{min}, y_{max}$, and *Stress* for the map are -0.16992, 0.31409, -0.25783, 0.15513, and 0.15 respectively. To zoom in further, in the next section, we have a map of interrelationship within an order, the Primates order.

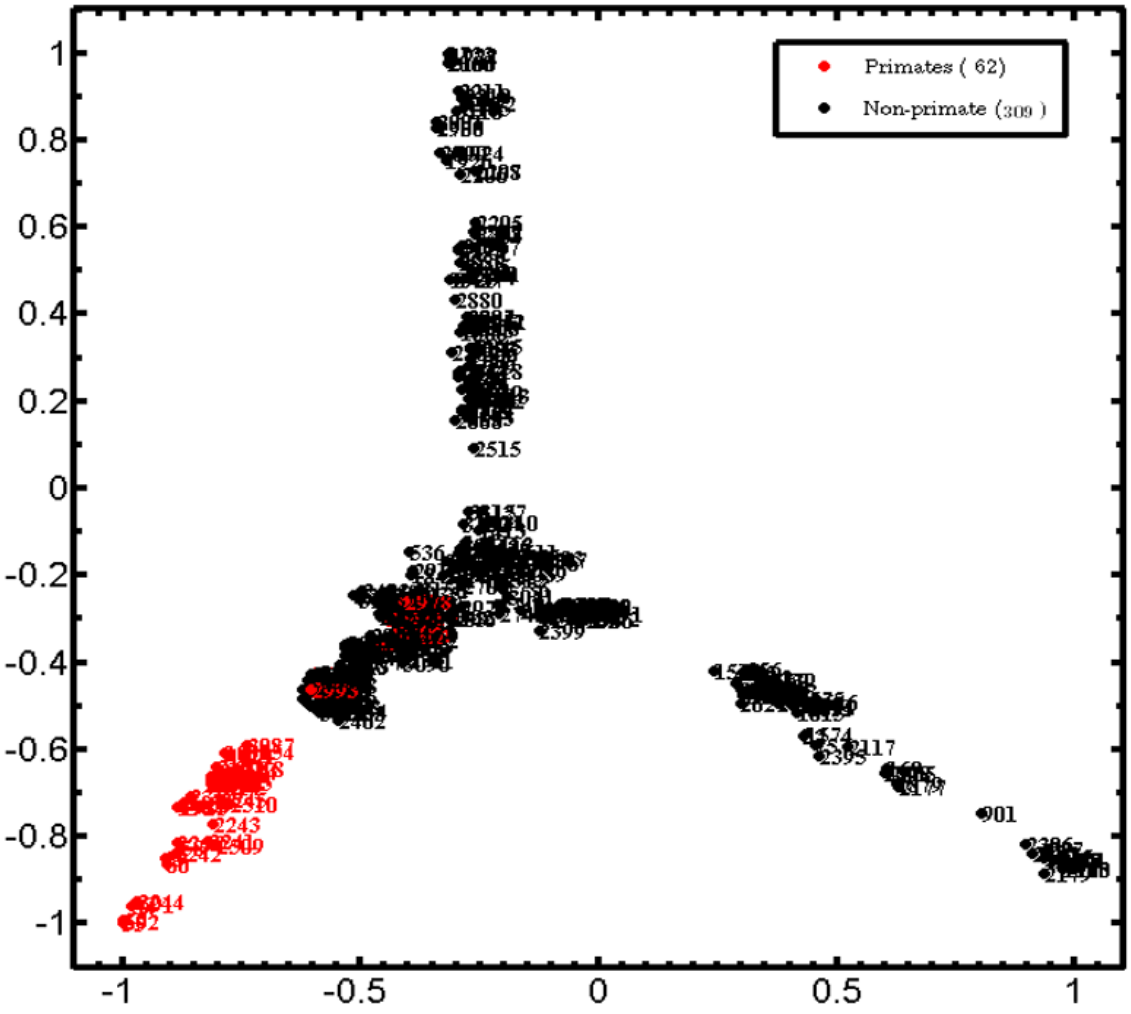



Figure 4.7: Genome Distance Map of Class Mammalia with the order Primates highlighted.

## 4.9 Primates

The two different suborders of the Order Primate are Strepsirrhini and Haplorrhini. The Genome Distance Map for the order Primates is shown in Figure 4.8. In this map, we have two misplaced Haplorrhines that are placed with the Strepsirrhines, namely Tarsius bancanus (GN: 2978) and Tarsius syrichta (GN: 1381). These are both tarsiers, whose position within the primates has been a controversial subject for over a century [JHS+11]. This map can thus support the claim of Chatterjee *et al.* [CHBG09] that these two Tarsius should actually classified in to Strepshirrini group. In contrast, the reverse is claimed by Jameson *et al.* [JHS+11]. Total number of organisms is 62 (14 Strepsirrhini, 48 Haplorrhini), $Stress = 0.19524$, $x_{max} = 0.15117$, $x_{min} = -0.33695$, $y_{max} = 0.24929$, $y_{min} = -0.28896$.

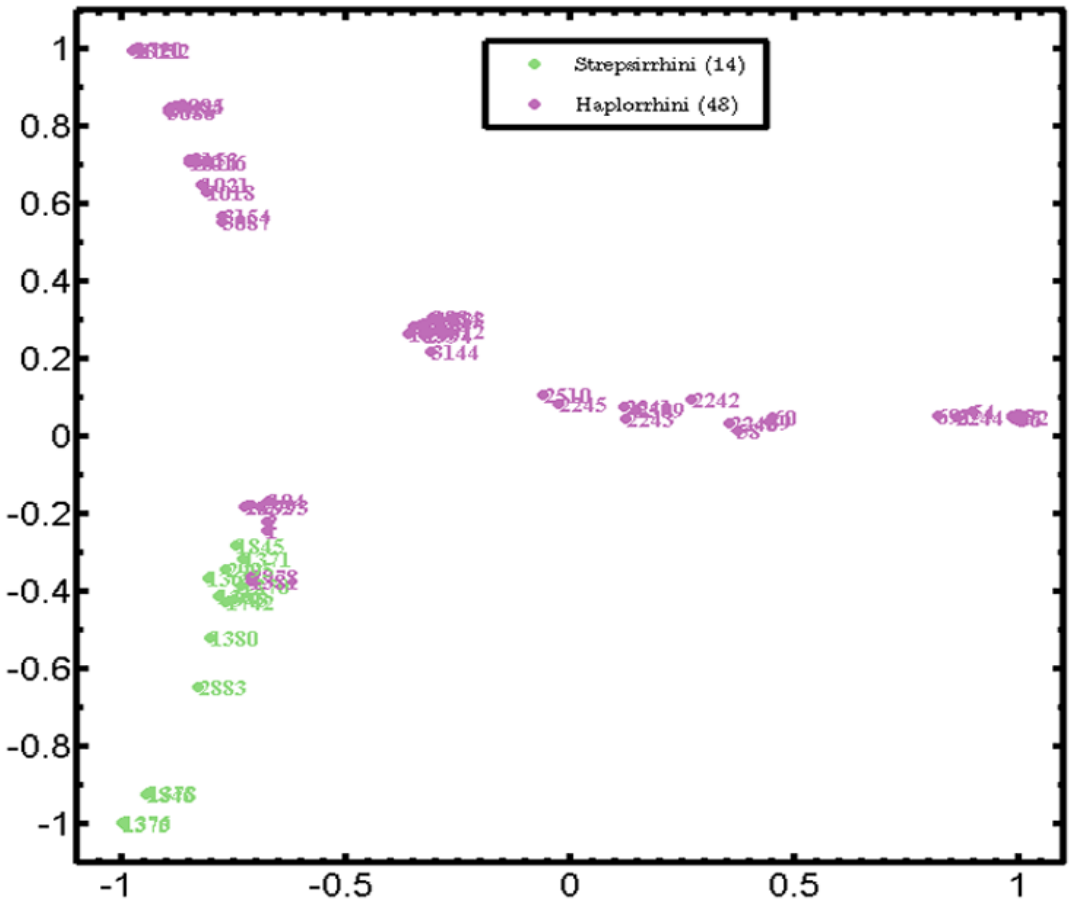



Figure 4.8: Genome Distance Map of the Order Primate and its two suborders Strepsirrhini and Haplorrhini.

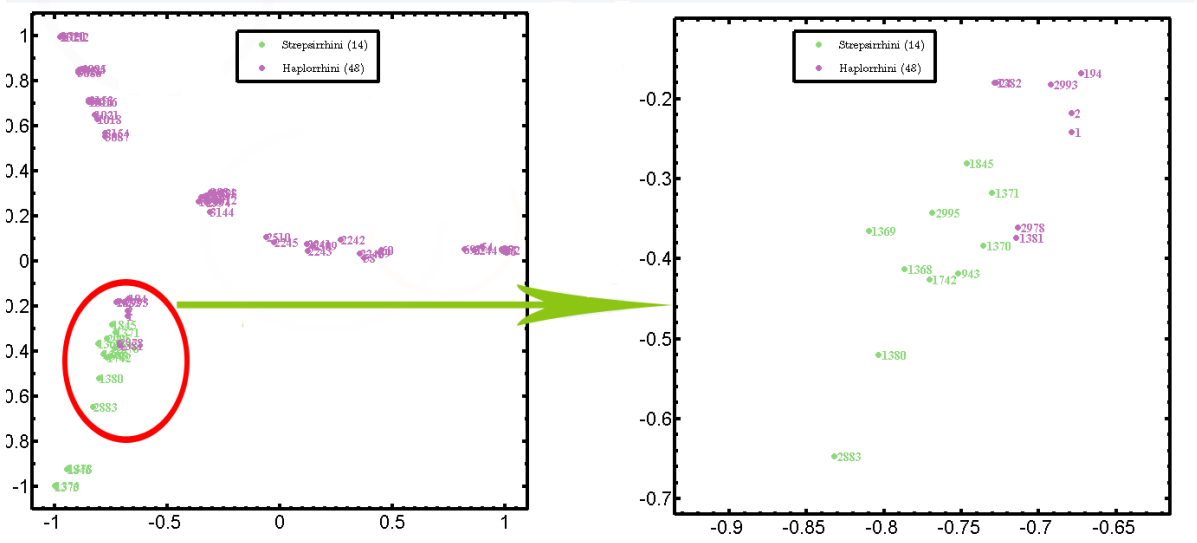



Figure 4.9: Zoomed in part of a particular region of the GDM of Figure 4.8

.

Whenever we need to look into a particular region of a GDM to observe mix ups or close relatedness, we can do it by zooming in on that particular region of a GDM. For instance, Figure 4.9 shows the zoomed in part of the misplaced Haplorrhines of Figure 4.8.

## 4.10 All protists

None of the datasets we explored so far showed the applicability of our method to the study of protists. Protists are unicellular organisms and did not fit into other kingdoms, and historically they were treated as a biological kingdom called Protista. With the use of molecular information, this group was redefined in modern taxonomy as diverse and often distantly related phyla. Consequently, the group of protists is now considered to mean diverse phyla, which are not closely related through evolution and have different life cycles, trophic levels, modes of locomotion, and cellular structures [Sim05]. Furthermore, besides their relatively simple levels of organization, the protists do not have much in common. They can be unicellular or multicellular without containing any kind of specialized tissues that makes them different from other eukaryotes such as animals, fungi and plants. Figure 4.10 is the Genome Distance Map of all protists in the dataset. This map contains a total of 79 protists, 31 from the phylum Alveolata, 7 from the phylum Amoebozoa, 1 from the Class Choanoflagellida, 1 from the phylum Het-

erolobosea, 1 from the Class Jakobida, 1 from the Class Malawimonadidae, 2 from the Class Cryptophyta, 1 from the Class Haptophyceae, and 34 from the phylum Stramenopiles.

*Stress* for this map is 0.27, $x_{min} = -0.47439$, $x_{max} = 0.29071$, $y_{min} = -0.25916$, $y_{max} = 0.31939$.

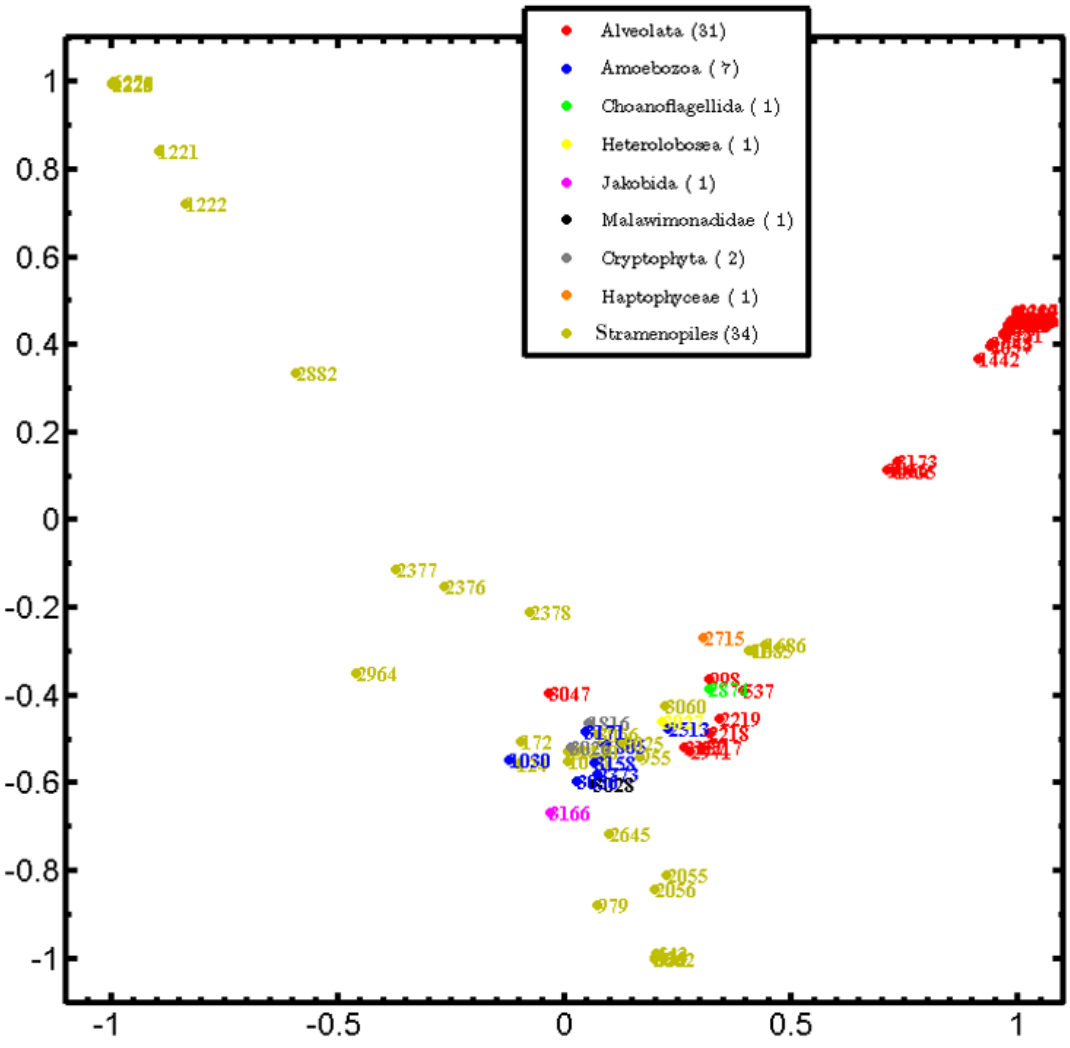



Figure 4.10: Genome Distance Map of all protists.

# Chapter 5

# Empirical analysis

While using the CGR/SSIM/MDS method to analyze species relatedness using their mitochondrial genomes, there are some issues that should be considered. At the beginning of this chapter, we will try to find the minimum number of nucleotides required to get detectable patterns in a CGR. Afterwards, we will experiment with the robustness and sensitivity of the CGR/SSIM method by observing the effect of the insertion, deletion and substitution of different number of nucleotides. In addition, we will perform length experiment on two datasets to see the differences on the final GDMs and decide the minimum sequence length requirement needed to get results for GDMs with clear separation of different clusters.

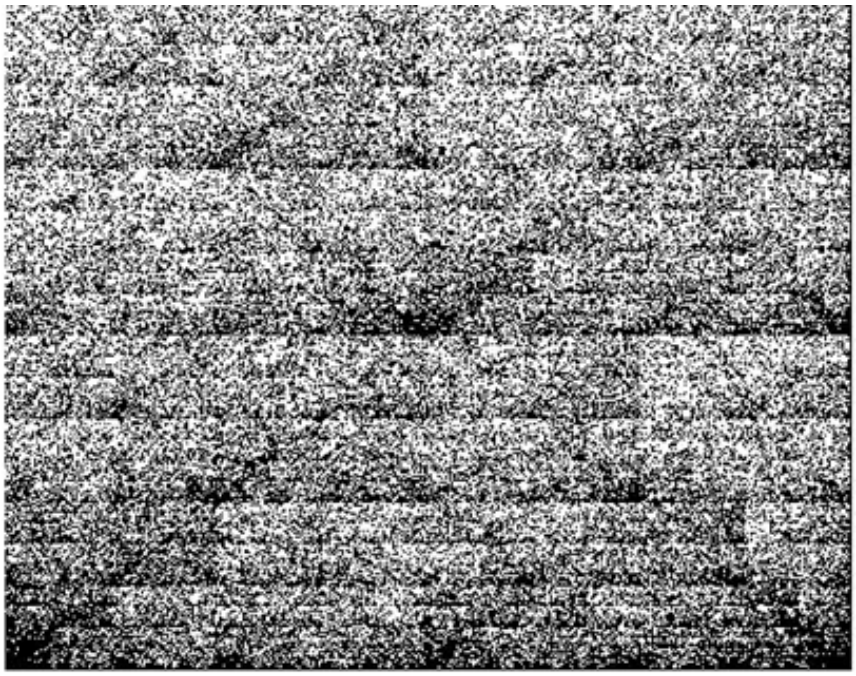
One additional experiment is implemented to see if the CGR of a genome contains more information than the mono, di-, and trinucleotides frequencies.

Furthermore, to asses the overall behaviour of the SSIM as a distance measurement method, we will see the graph of the SSIM distances between *Homo sapiens* (GN: 1321) and *Malawimonas jakobiformis* (GN: 3028) and all other organisms of our database.

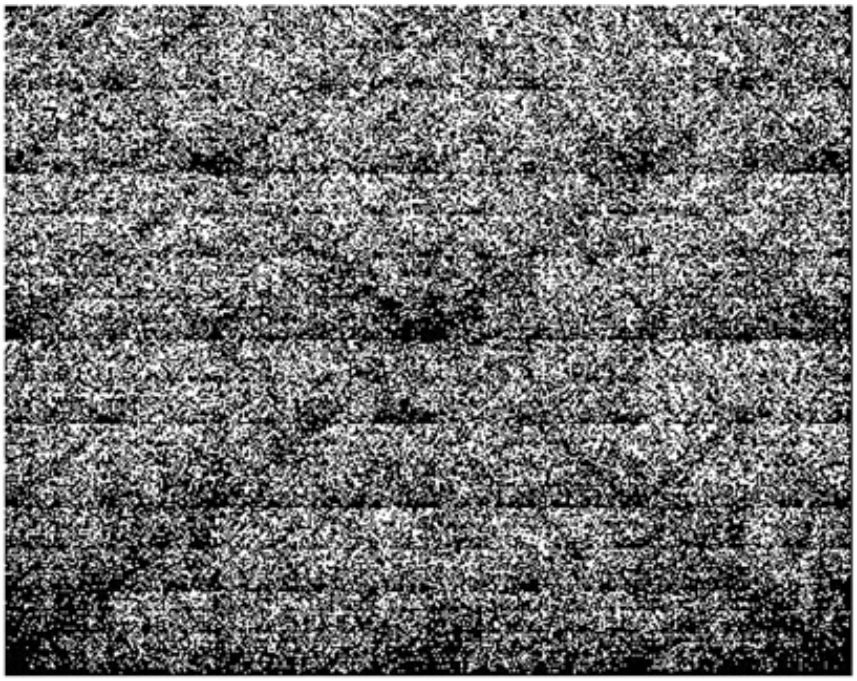
## 5.1 CGRs of plants with long mtDNA

The contrast (dark or white) of a CGR image depends on the total number nucleotides in a genome. Generally, the mtDNA sequences of plants are longer than other eukaryote genomes. In this section we will see CGRs of some plant mitochondrial genomes that contain a large number of nucleotides and try to observe how dark the CGRs can be.

In Figure 5.1, the CGRs of four different plants are shown. Figure 5.1a shows the CGR of the mtDNA genome of *Anomodon regelii*, Figure 5.1b shows the CGR of the mtDNA genome of *Treubia lacunosa*, Figure 5.1c CGR of the mtDNA genome of *Huperzia squarrosa*, and Figure 5.1d shows the CGR of the mtDNA genome of *Phoenix dactylifera*. We can see that the CGRs get darker as the total number of nucleotides increases. Consequently, we can conclude saying that the contrast or darkness of a CGR is proportional to the total number of nucleotides in a genome. As DSSIM returns 1, when comparing one black and one white image, comparison of long mitochondrial plant genomes with a genome of shorter length is always expected to be higher with our proposed CGR/SSIM method.



(a) CGR of the mitochondrial genome of *Anomodon rugelii*, sequence length 104,239 nt, GN:508.



(b) CGR of the mitochondrial genome of *Treubia lacunosa*, sequence length 151,983 nt, GN:509.

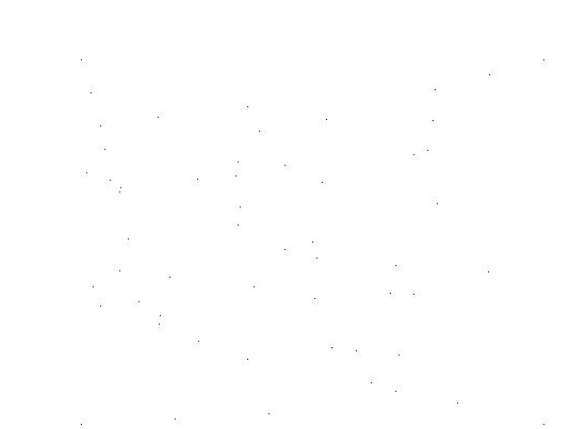
(c) CGR of the mitochondrial genome of *Huperzia squarrosa*, sequence length 413,530 nt, GN:118.

(d) CGR of the mitochondrial genome of *Phoenix dactylifera*, sequence length 715,001 nt, GN:243.

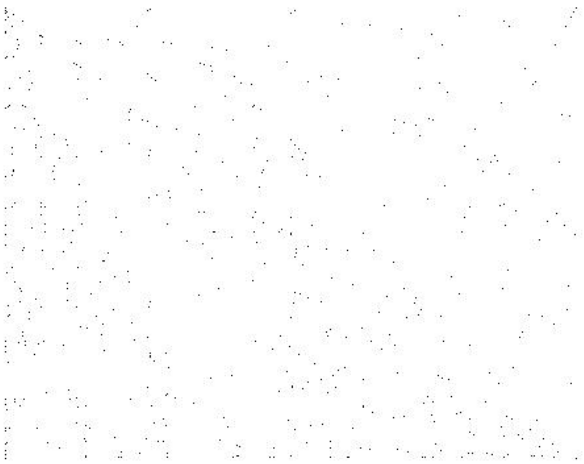
Figure 5.1: CGRs of plants with long mtDNA.

## 5.2 CGR of human mtDNA genome and *Phoenix dactylifera* mtDNA genome truncated at different length positions

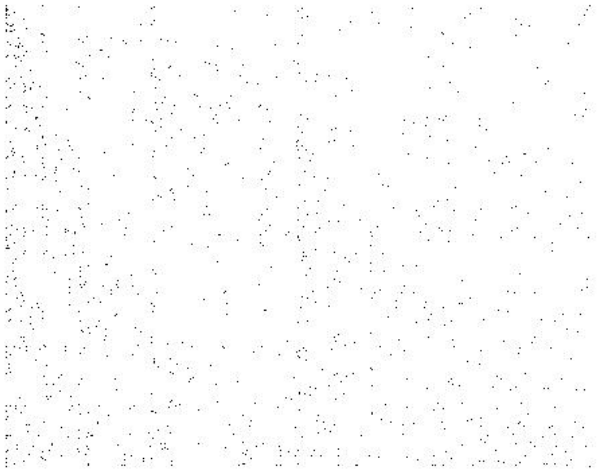
This experiment was performed to see what should be the minimum number of nucleotides required for a DNA sequence to produce a recognizable pattern in a CGR. At first, CGRs of the human mtDNA genome at different length truncations were imaged. The consecutive CGRs shown in Table 5.1 are generated taking the first 50, 500, 1000, 2000, 4000, 5000, 10000, 12000, 15000 nucleotides of the human mtDNA sequence and the last image shows the CGR of the whole human mtDNA sequence. For these cases, we can say that for a mtDNA sequence with shorter length, the first 12,000 nt sequence is sufficient to get a recognizable pattern in its corresponding CGR.



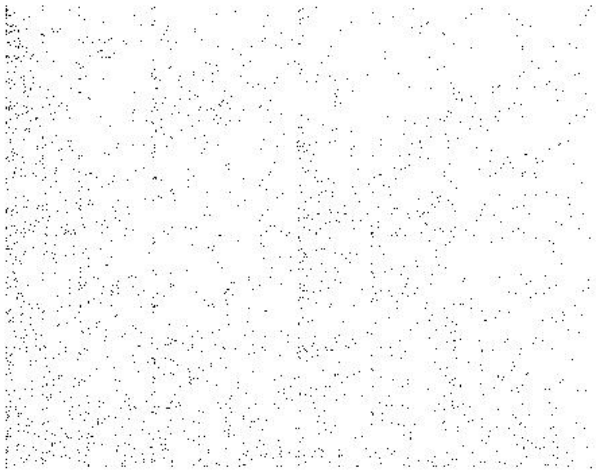
(a)



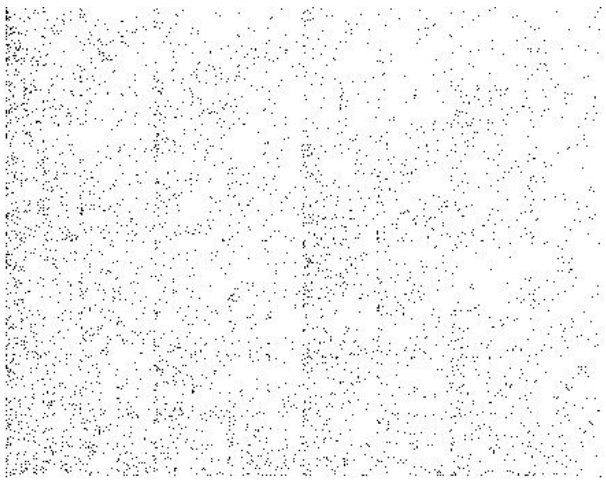
(b)



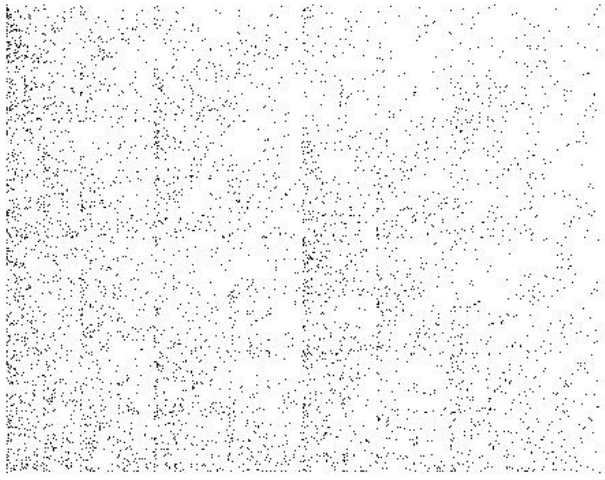
(c)



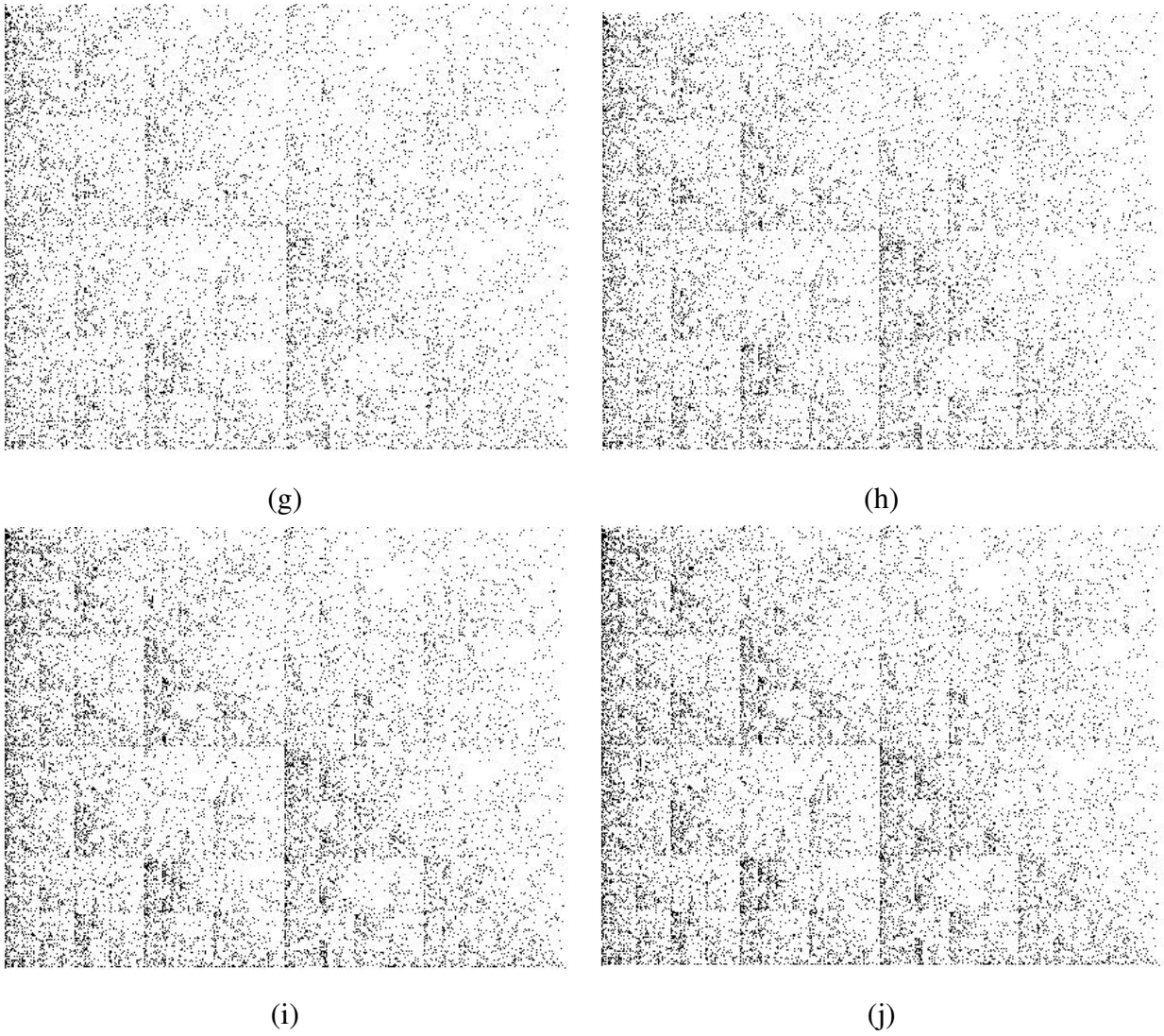
(d)



(e)



(f)



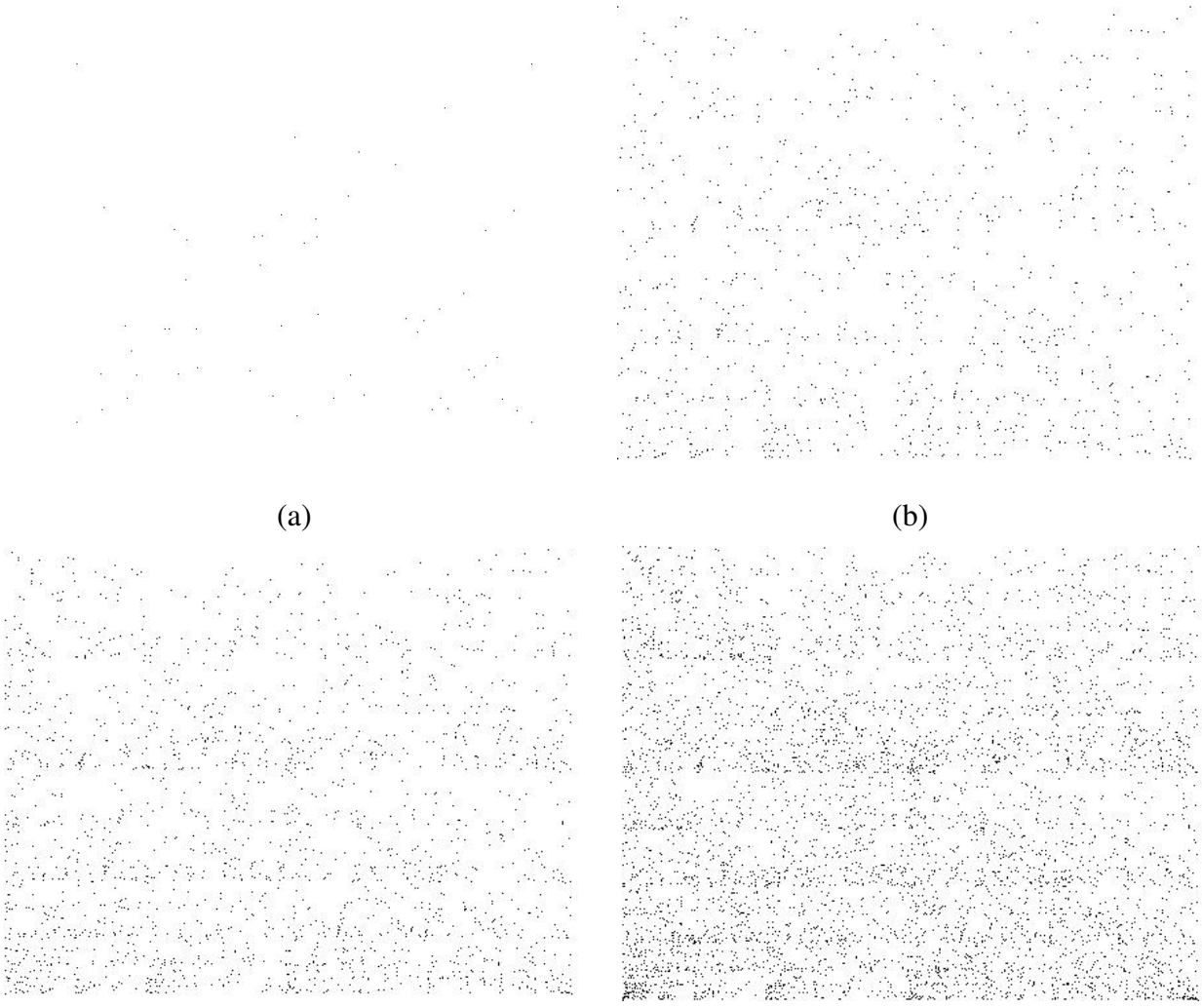
(g) (h)

(i) (j)

Table 5.1: CGRs of human mtDNA taking the first (a) 50, (b) 500, (c) 1000, (d) 2000, (e) 4000, (f) 5000, (g) 10000, (h) 12000, (i) 15000, and (j) 16569 nucleotide positions (left to right and top to bottom).

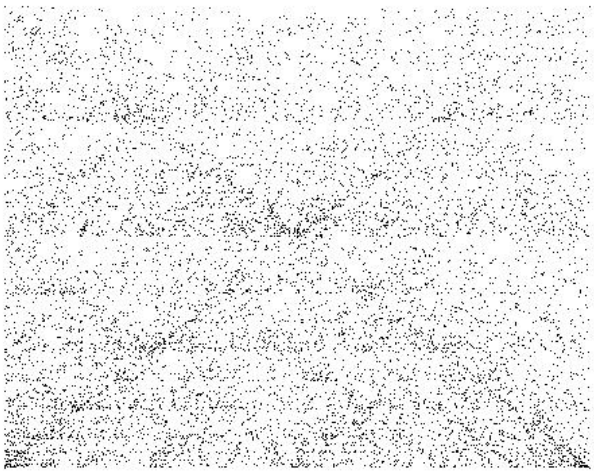
CGRs of Table 5.2 are generated taking the first 50, 1000, 2000, 5000, 10000, 20000, 50000, 100000, 200000, 300000, 700000 nucleotides of the mtDNA of date palm (GN: 243, *Phoenix dactylifera*). The last image shows the CGR of the whole mtDNA sequence. In this case, the first 20,000 nt generate detectable patterns in CGR. In general, it is difficult to draw a general conclusion about the minimum number of nucleotides required for different genomes, where the corresponding CGR image of one genome starts to generate noticeable patterns. For these two experiments, the nucleotide range of 12,000-20,000 nt seems reasonable to get no-

ticeable patterns in CGR and well suited for CGR based analyses. This does not imply that sequences of other lengths can not be imaged and compared for particular applications.

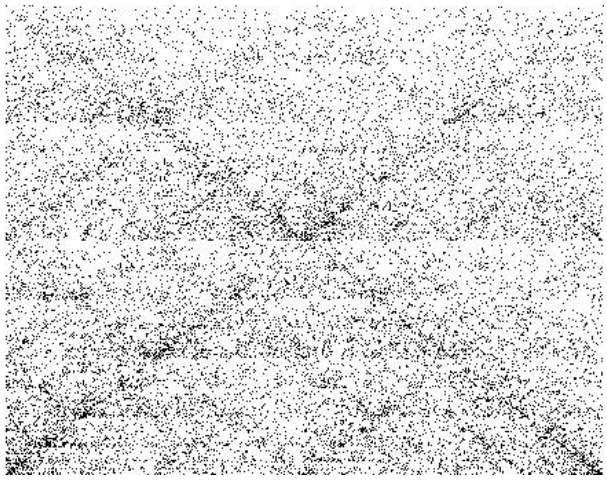


(a) (b)

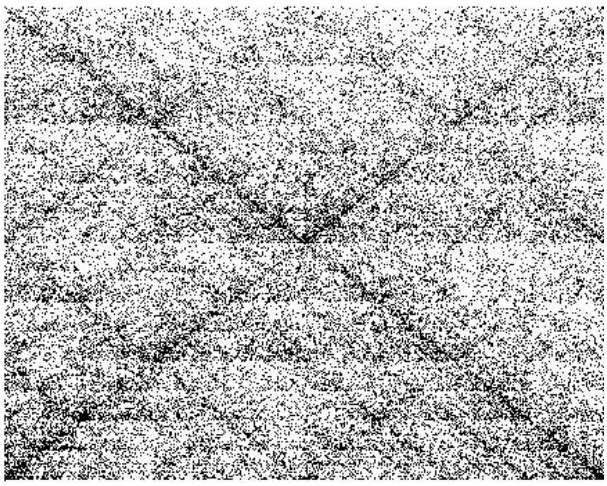
(c) (d)



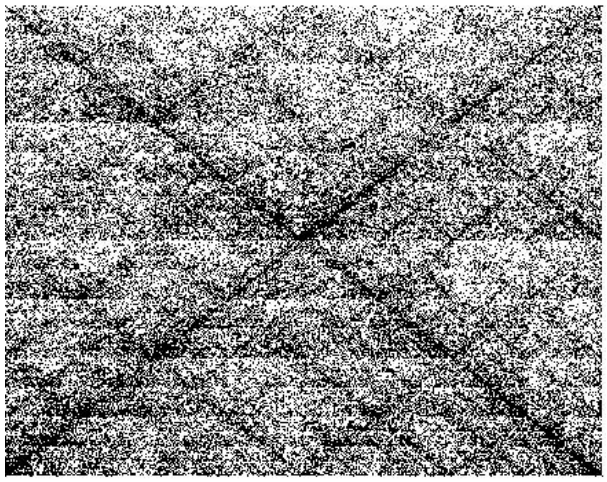
(e)



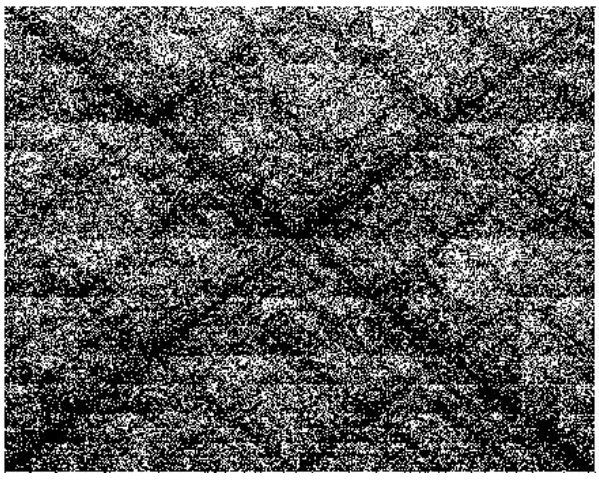
(f)



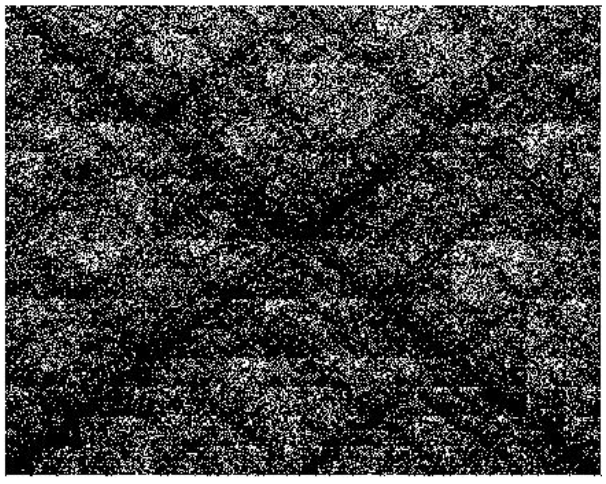
(g)



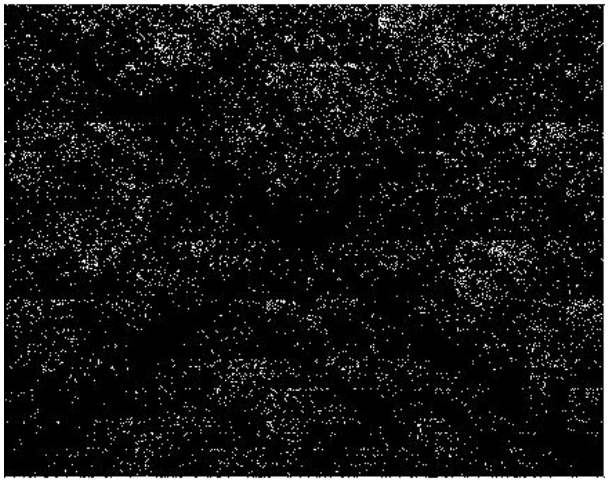
(h)



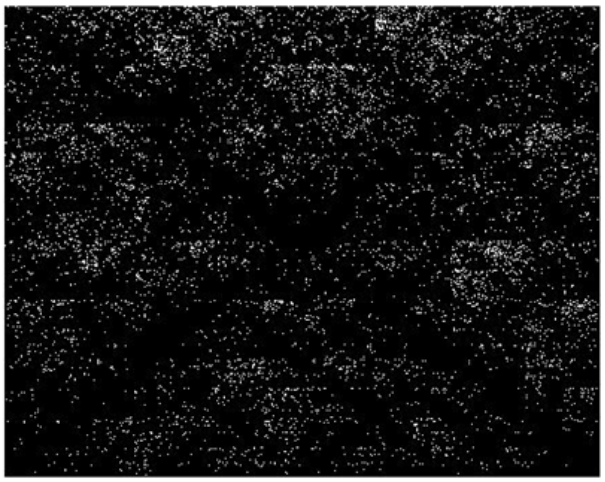
(i)



(j)



(k)



(l)

Table 5.2: CGRs of the mtDNA of date palm (GN: 243, *Phoenix dactylifera*) taking the first (a) 50, (b) 1000, (c) 2000, (d) 5000, (e) 10000, (f) 20000, (g) 50000, (h) 100000, (i) 200000, (j) 300000, (k) 700000, and (l) 715001 nucleotide positions (left to right and top to bottom).

## 5.3 Robustness and sensitivity of CGR/SSIM: Insertion deletion experiment

The purpose of the following experiments is to asses the impact of insertions/deletions and substitutions of varying number of nucleotides at different positions of a genome. We aimed to answer the following questions:

1. How sensitive is SSIM to detect single nucleotide insertion, deletion and substitution in a genome?
2. Which operation has the highest impact while doing comparison?
3. How do substitutions of subsequences of a genome with subsequences from completely different genomes impact the distance?

To answer these questions the human mitochondrial genome was used, wherein 1,000 nt were deleted from the middle, which resulted in a DSSIM distance of 0.0299 from the original

sequence. The insertion of 1,000 nt from the same human genome resulted in a DSSIM distance of 0.0007 from the original sequence. The insertion into the human mitochondrial genome of a 1,000 nt sequence from an unrelated species, *Marchantia polymorpha* (GN:3174), at the same position, resulted in a much larger distance of 0.0515. A replacement of a 1,000 nt sequence from *Marchantia* in the middle of the human mitochondrial genomes resulted in a distance of 0.084. In addition, SSIM may be sensitive enough to detect single nucleotide insertion, deletion and substitution. Deletion of one single nucleotide from the middle of human mtDNA result in a distance of 0.0001 from the original sequence, whereas insertion of one nucleotide results in a 0.0002 distance.

Furthermore, to analyze the effect of subsequence substitutions, a "metamorphosis" was simulated of one species into another, starting with the human mtDNA and substituting 100, 200, 300, ... all nucleotides with the subsequences of the same length and starting at the same position from the mtDNA of *Marchantia polymorpha* (GN:3174). Subsequently, the distances between the human mtDNA and all the intermediate chimera-organisms were plotted and the shape of the trajectory was investigated . Interestingly, the trajectory showed a very small increase in distance when the length of the substituted sequences increased from zero to 1000 nt (where the distance was 0.0941), and afterwards the distance increased linearly until the entire 16,569 nt from the human mitochondrial genome was substituted with *Marchantia*. The maximum distance of 0.974 was found when the mtDNA genome of human was completely substituted by the plant genome. The distance between the human mtDNA genome and the full-length 186,609 nt *Marchantia* mitochondrial genome is 0.9838.

Together with the genome-specificity of CGR, it can be said that the above experiments indicate that insertions of sequences from the same mtDNA would have a negligible effect on the position of a species-point in the map, while insertions or substitutions of relatively large sequence from the mtDNA of a distant species may significantly change the position of the species-point in a genome distance map. Table 5.3 shows the resulting impact of the insertion, deletion and substitution of a human mtDNA sequence and Figure 5.2 shows the metamorphosis graph.

| | Human mtDNA | Deletion of 1nt from the middle of human mtDNA | Deletion of 1000 nt from the middle of human mtDNA | Insertion of 1,000 nt human mtDNA in the middle of human mtDNA | Insertion of 1nt in the middle of human mtDNA | Substitution of 1000 nt (middle) human mtDNA with plant mtDNA | Insertion of 1000 nt of plant mtDNA in the middle of human mtDNA genome |
|---|---|---|---|---|---|---|---|
| Human genome | 0 | 0.0001 | 0.0299 | 0.0007 | 0.0002 | 0.084 | 0.0515 |
| Deletion of 1nt from the middle of human mtDNA | | 0 | 0.03 | 0.0007 | 0.0002 | 0.0841 | 0.0515 |
| Deletion of 1000 nt from the middle of human mtDNA | | | 0 | 0.0306 | 0.0301 | 0.0714 | 0.0804 |
| Insertion of 1,000 nt human mtDNA in the middle of human mtDNA | | | | 0 | 0.0007 | 0.0846 | 0.0521 |
| Insertion of 1nt in the middle of human mtDNA | | | | | 0 | 0.0842 | 0.0516 |
| Substitution of 1000 nt (middle) human mtDNA with plant mtDNA | | | | | | 0 | 0.0316 |
| Insertion of 1000 nt of plant mtDNA in the middle of human mtDNA genome | | | | | | | 0 |

Table 5.3: Effect of modifications of the human mitochondrial genome on the DSSIM distance from the unaltered sequence.

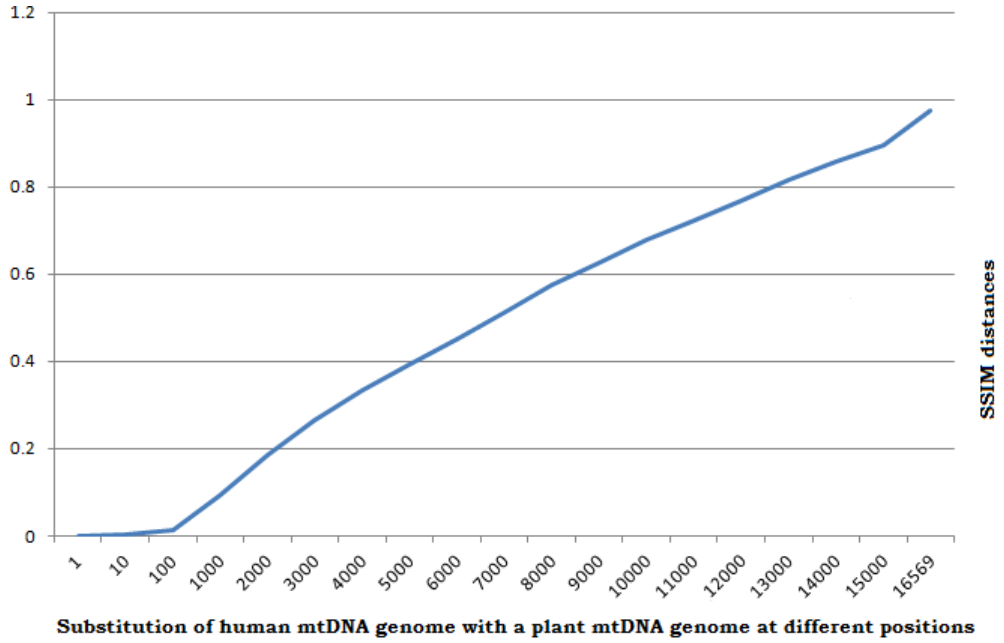


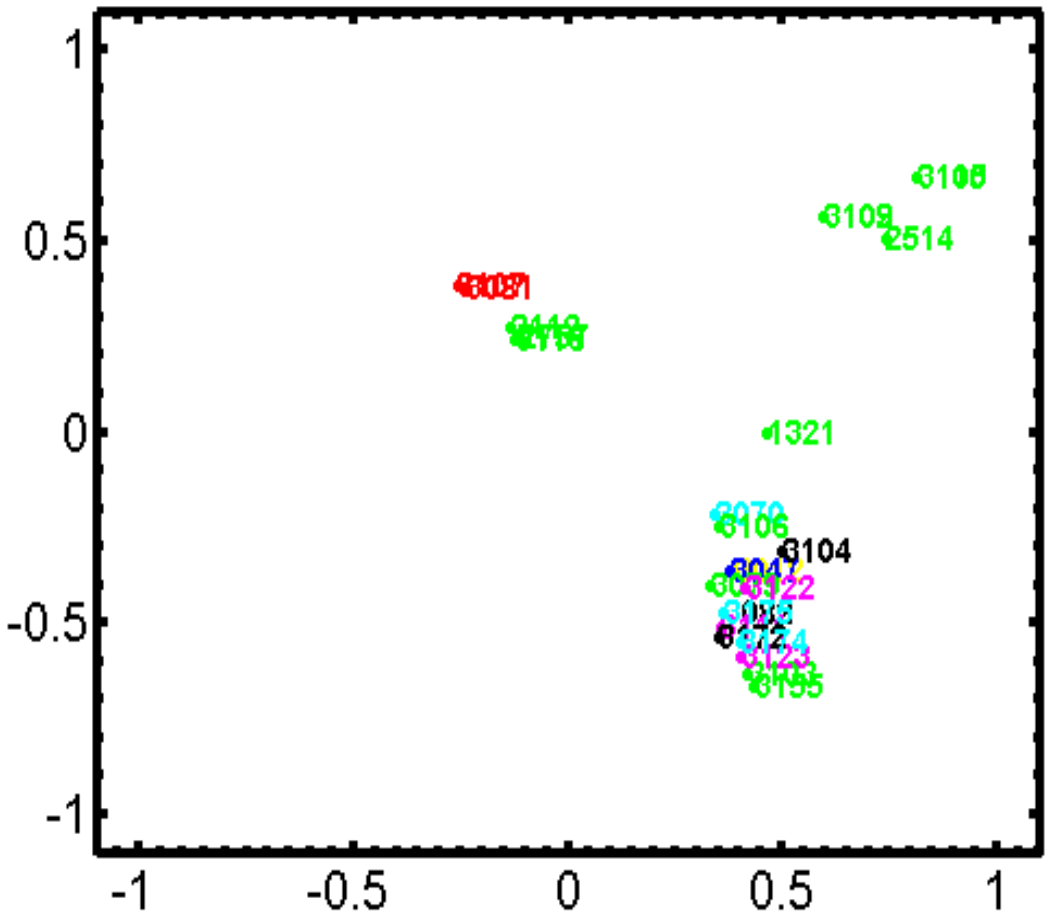

Figure 5.2: The graph plots the DSSIM distances (measured with SSIM of CGRs), between the original human mtDNA and artificial DNA sequences obtained by substituting the beginning sequence of the human mtDNA with *Marchantia Polymorpha* mtDNA. The process was repeated with subsequences of increasing length, until the entire human mtDNA was substituted with plant mtDNA.

## 5.4 Genome Distance Maps at different length truncations

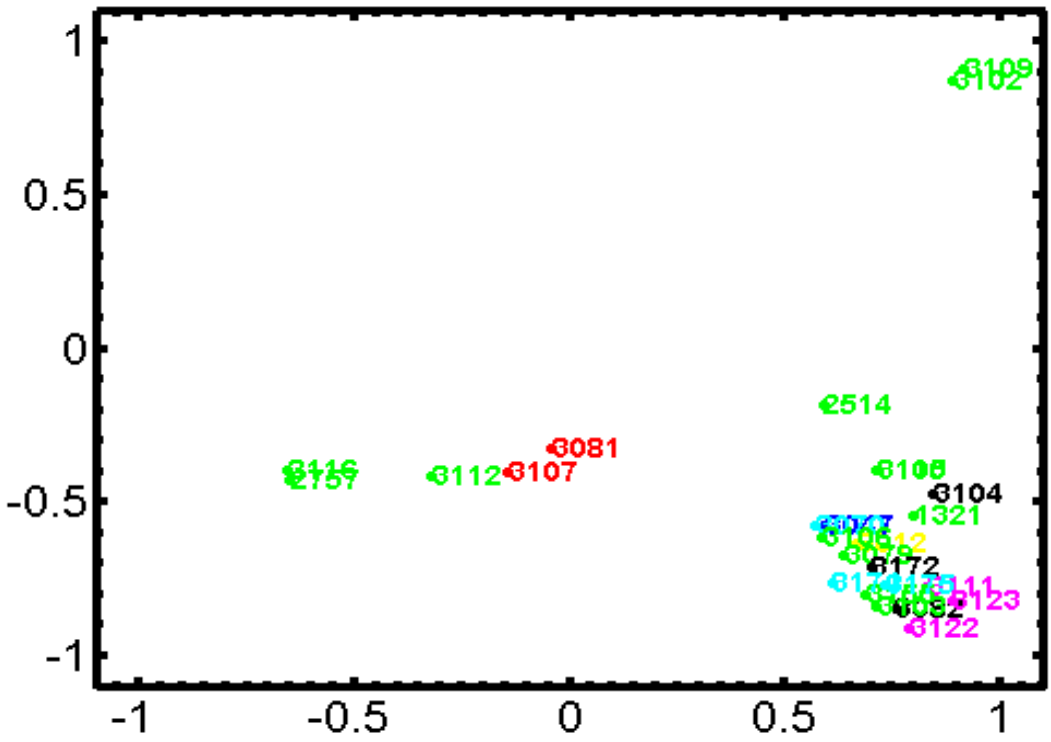
In this experiment, the effect of sequence length on the resulting Genome Distance Maps is investigated.

At first, the dataset used in Wang *et al.* [WHSK05] has been considered. The following figures are Genome Distance Maps obtained using successive truncations of the mtDNA of the organisms in the dataset: The first Genome Distance Map uses as input data the first 100 nt from each mitochondrial genome in the dataset, the second map uses the first 500 nt from each genome, and the subsequent ones use the first 1,000 nt, 2,000 nt, 4,000 nt, 6,000 nt, 8,000 nt, 10,000 nt, 12,000 nt, 14,000 nt, 15,000 nt, and full genomes respectively.

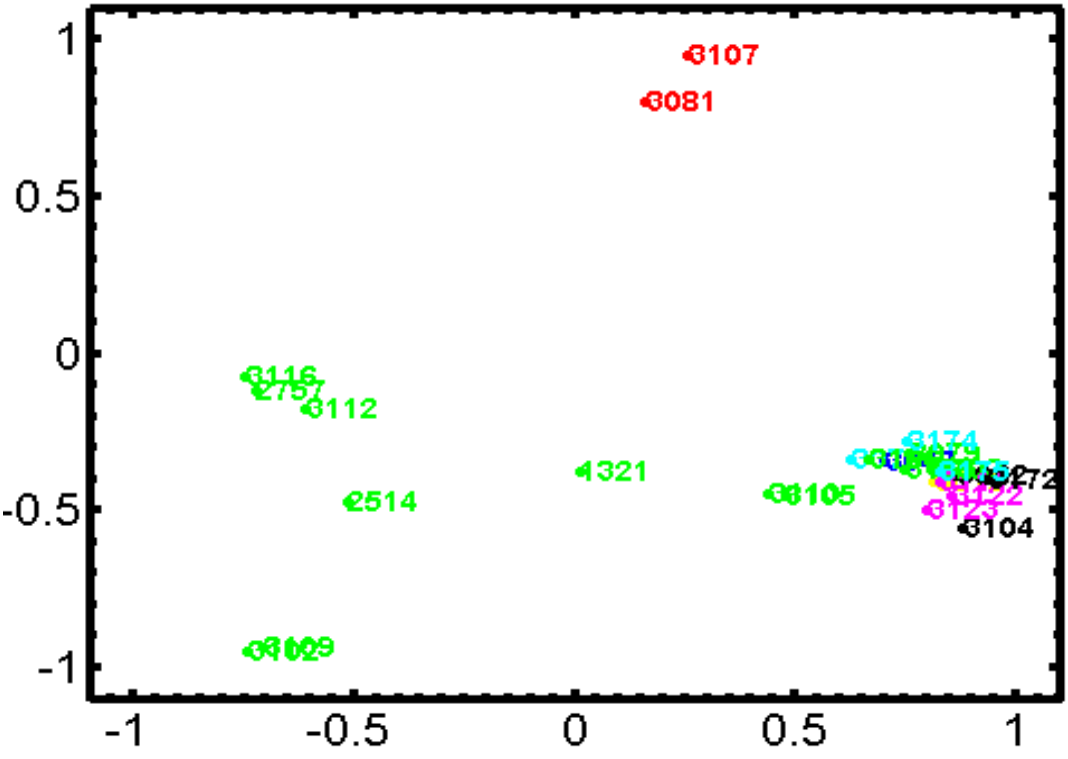
For this experiment, we start to get reasonable separation at 12,000 nucleotides and get complete visual separation of clusters at 15,000 nucleotides. The different 26 species start to group according to their biological classifications after the first 12,000 nucleotides were taken.



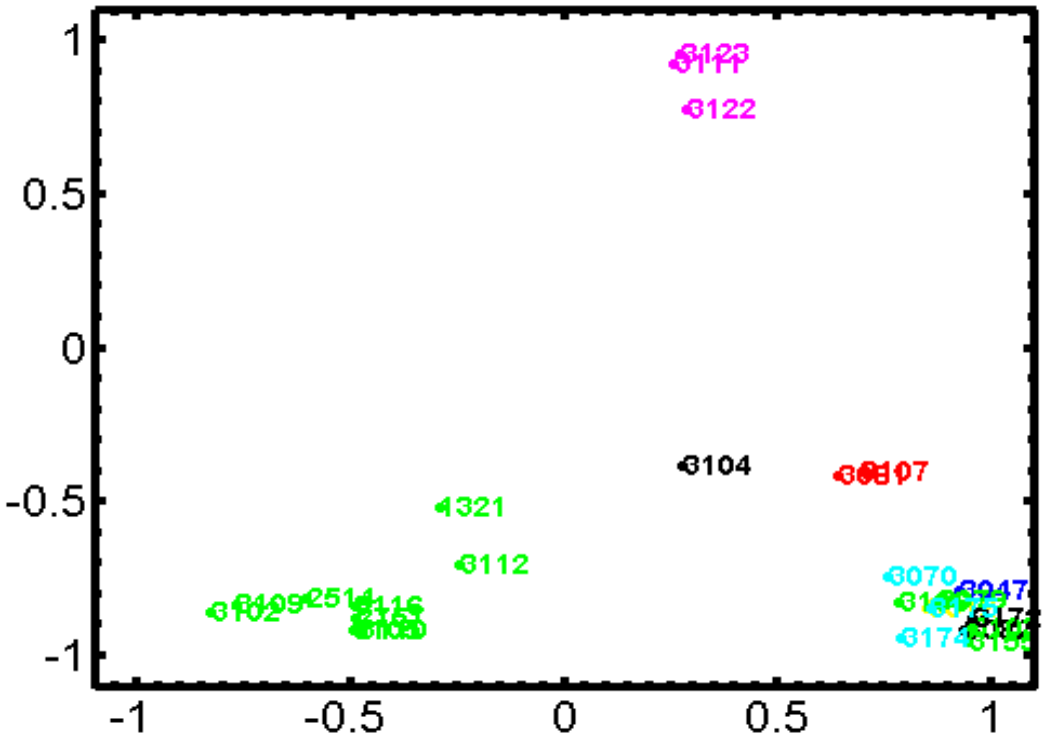

(a) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 100 nt of each mtDNA genome.



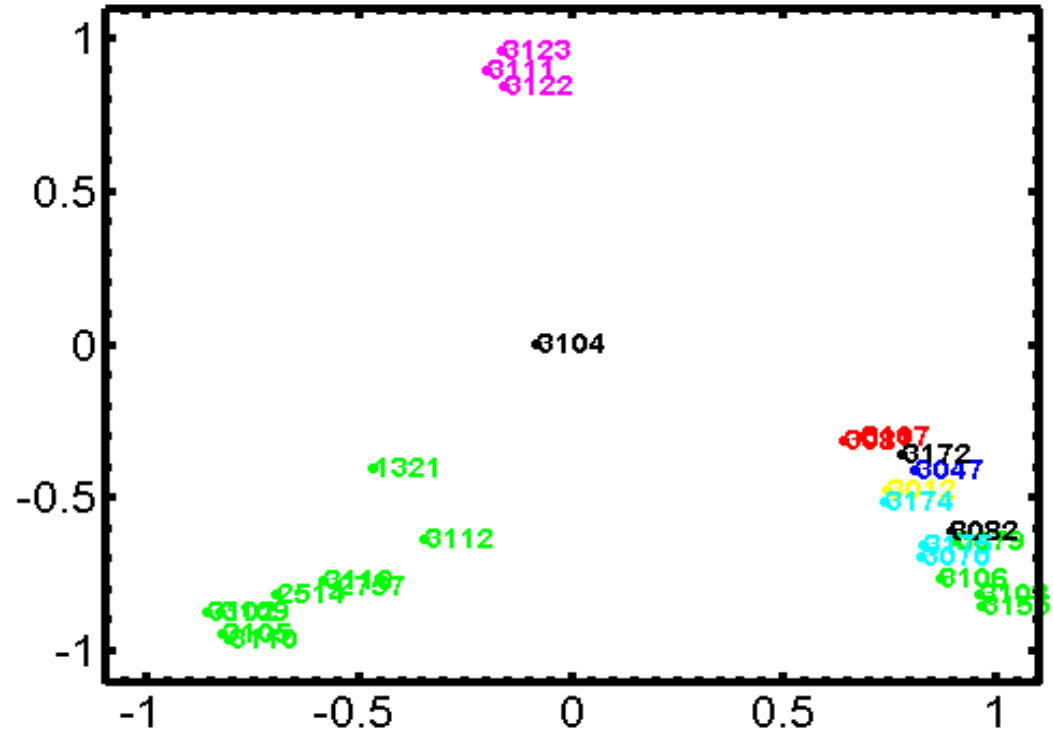

(b) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 500 nt of each mtDNA genome.



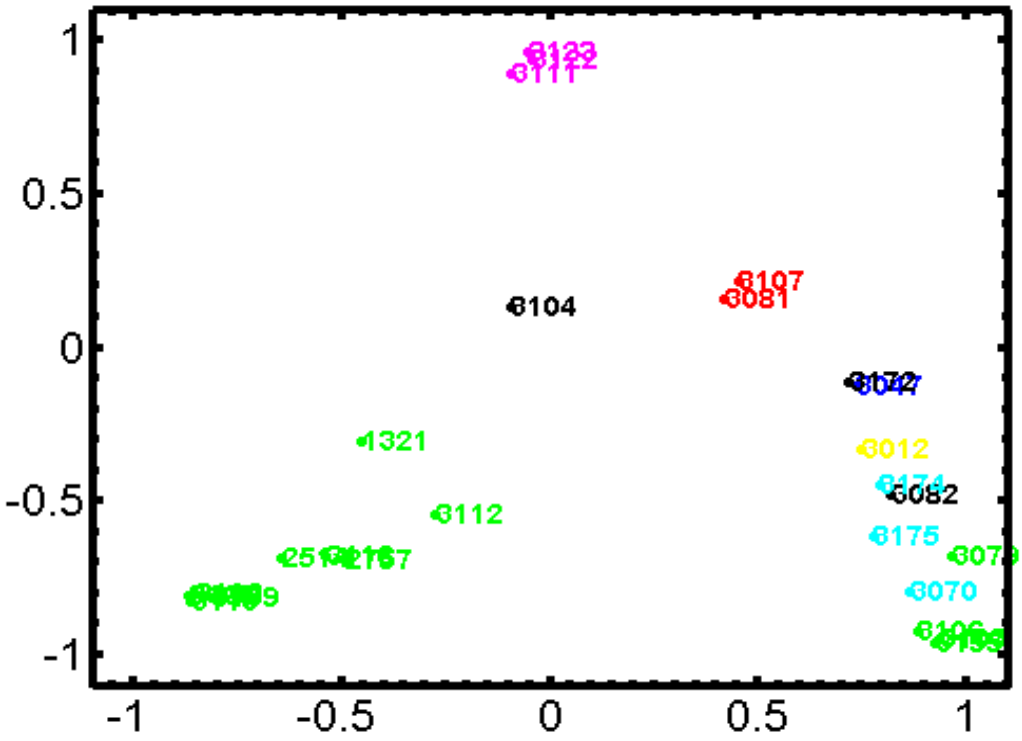

(c) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 1,000 nt of each mtDNA genome.



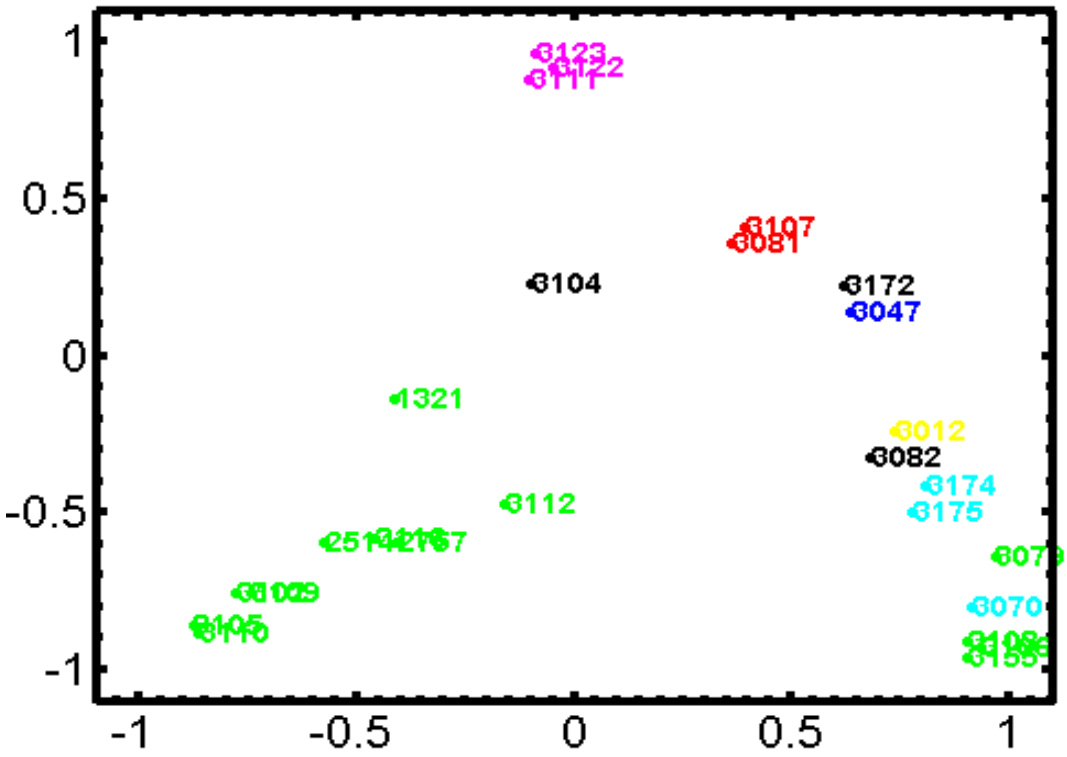

(d) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 2,000 nt of each mtDNA genome.



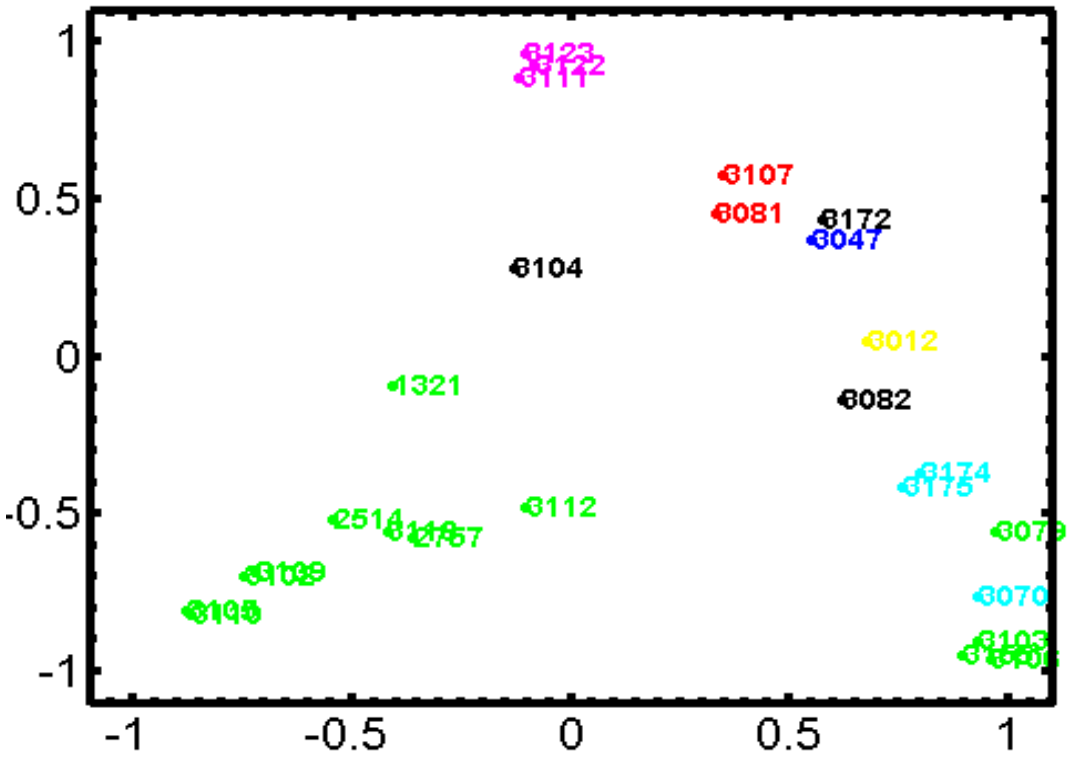

(e) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 4,000 nt of each mtDNA genome.



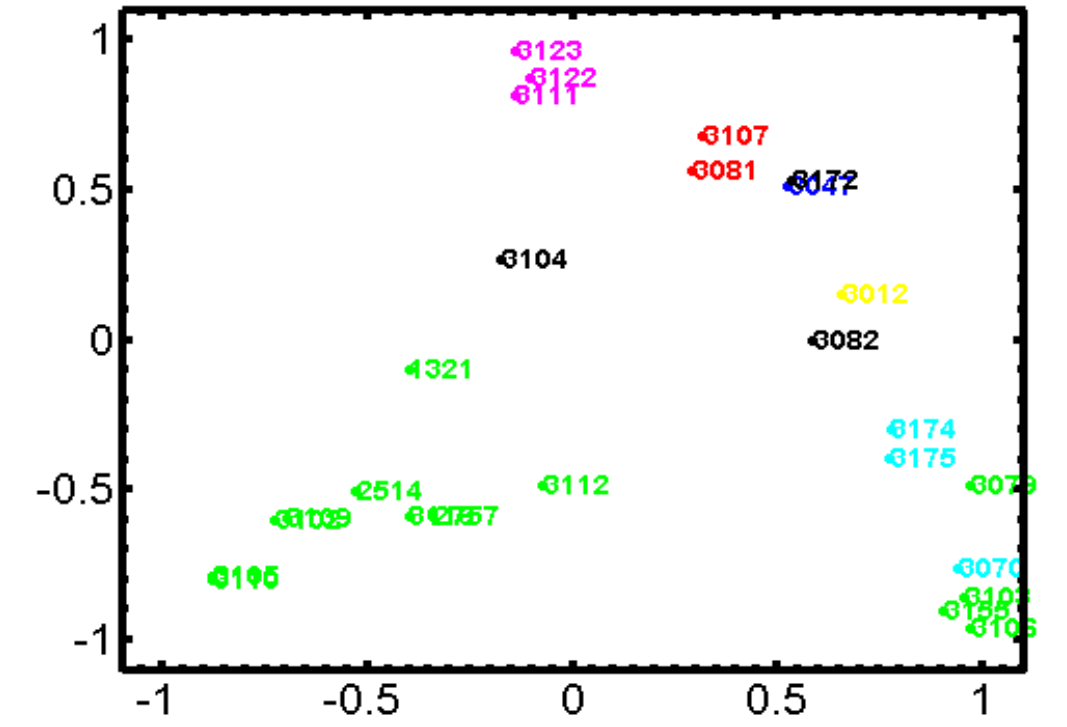

(f) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 6,000 nt of each mtDNA genome.



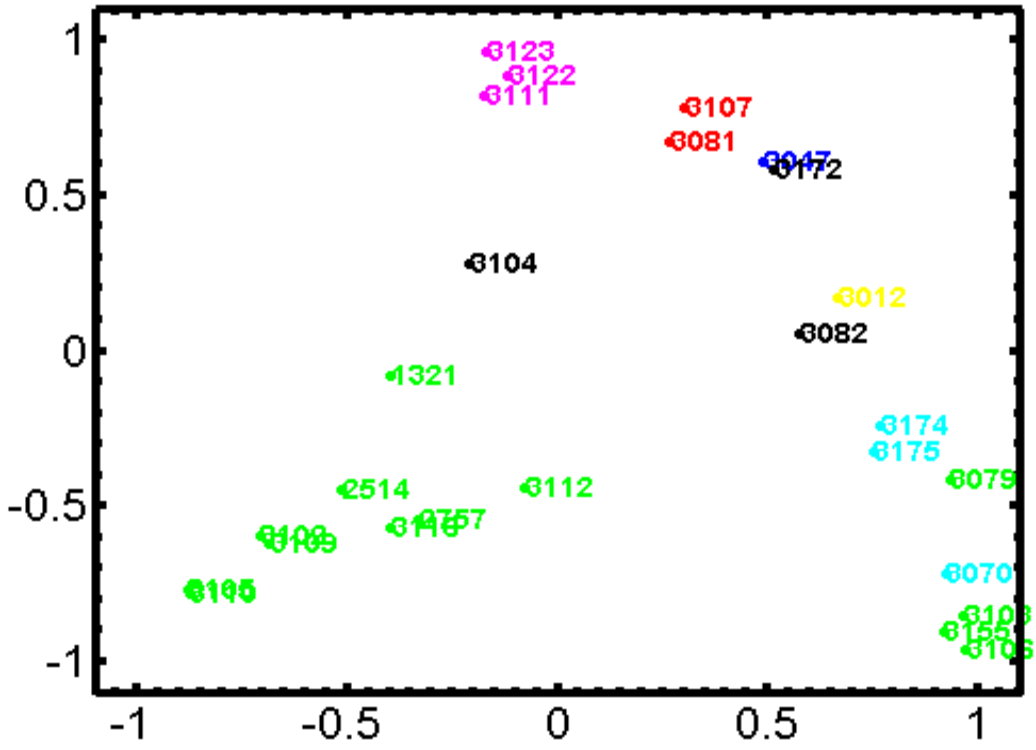

(g) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 8,000 nt of each mtDNA genome.



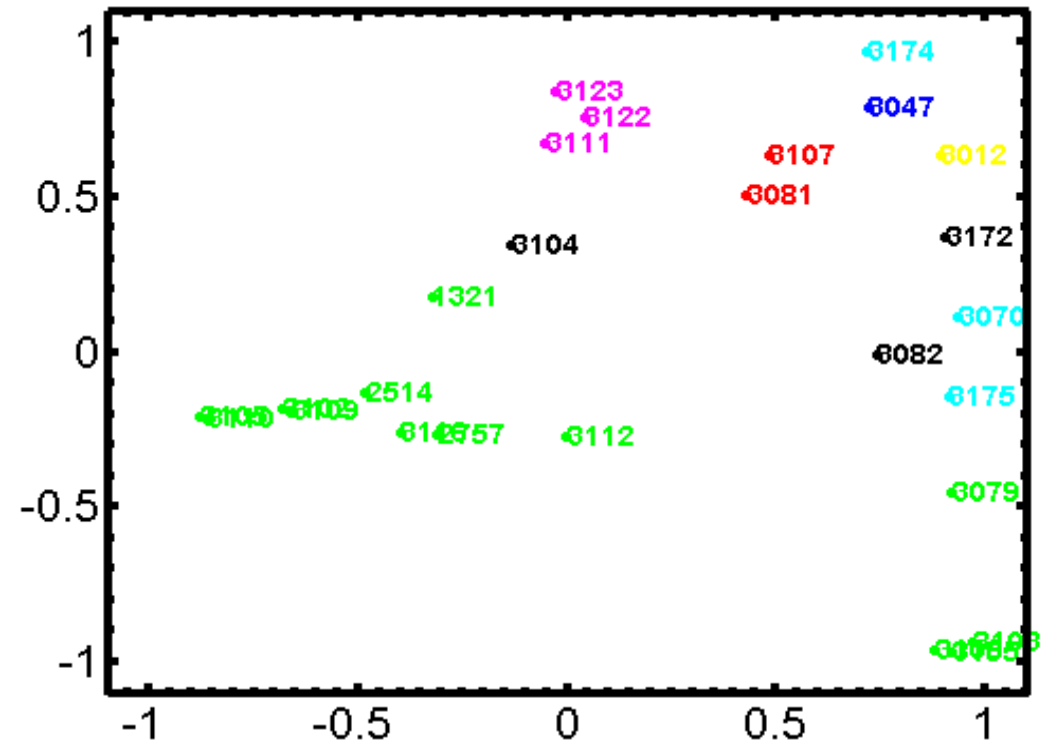

(h) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 10,000 nt of each mtDNA genome.



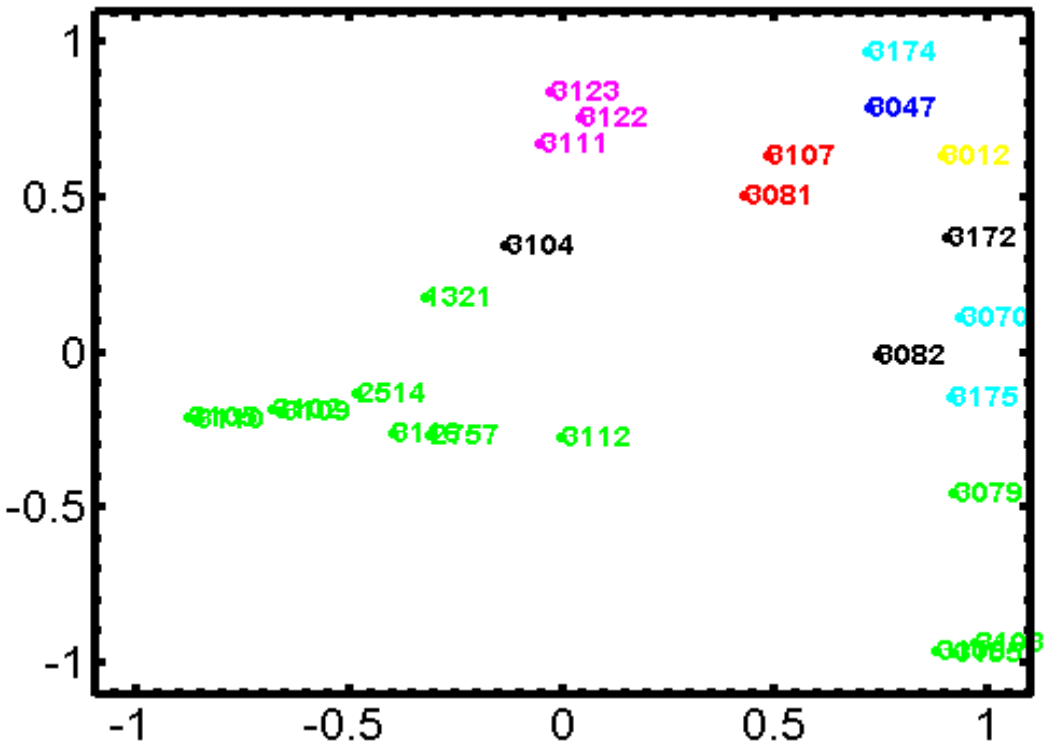

(i) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 12,000 nt of each mtDNA genome.




(j) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 14,000 nt of each mtDNA genome.




(k) GDM for the dataset of Wang *et al.* [WHSK05] taking the first 15,000 nt of each mtDNA genome.

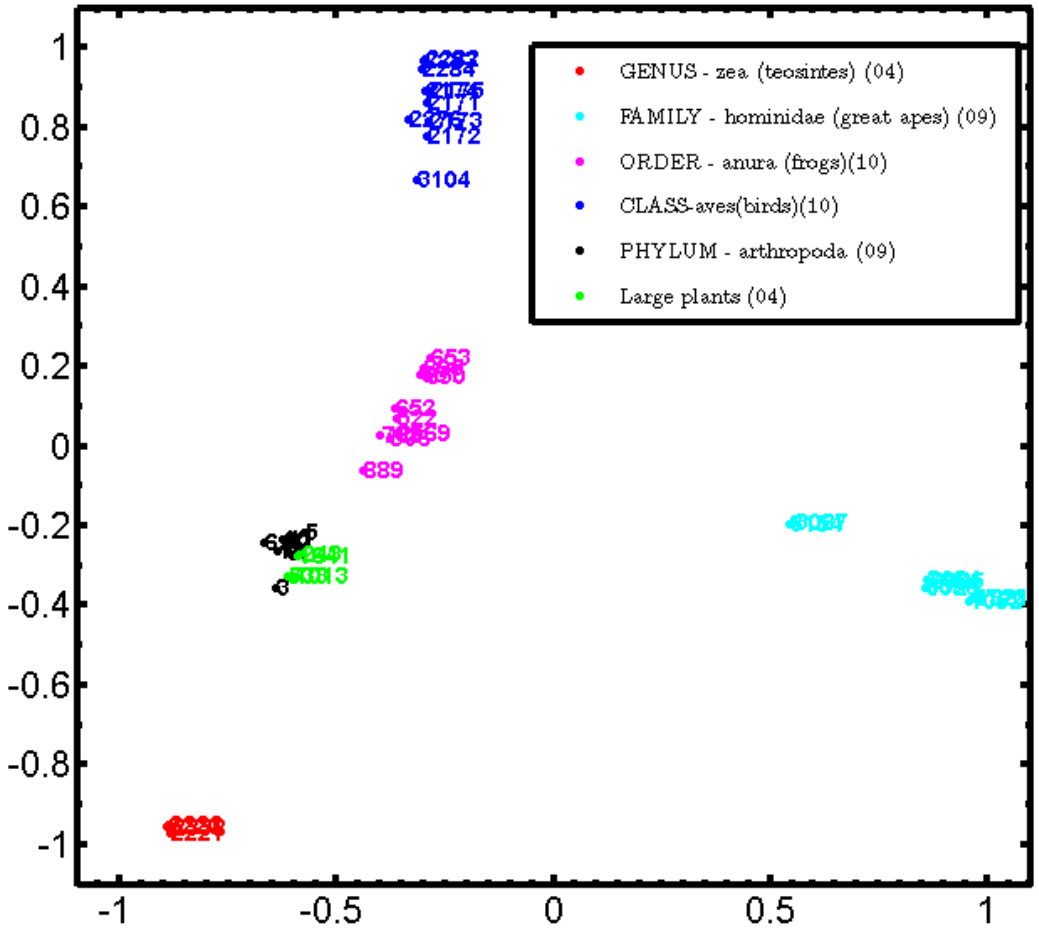



(l) GDM for the dataset of Wang *et al.* [WHSK05] taking the complete mtDNA genome.

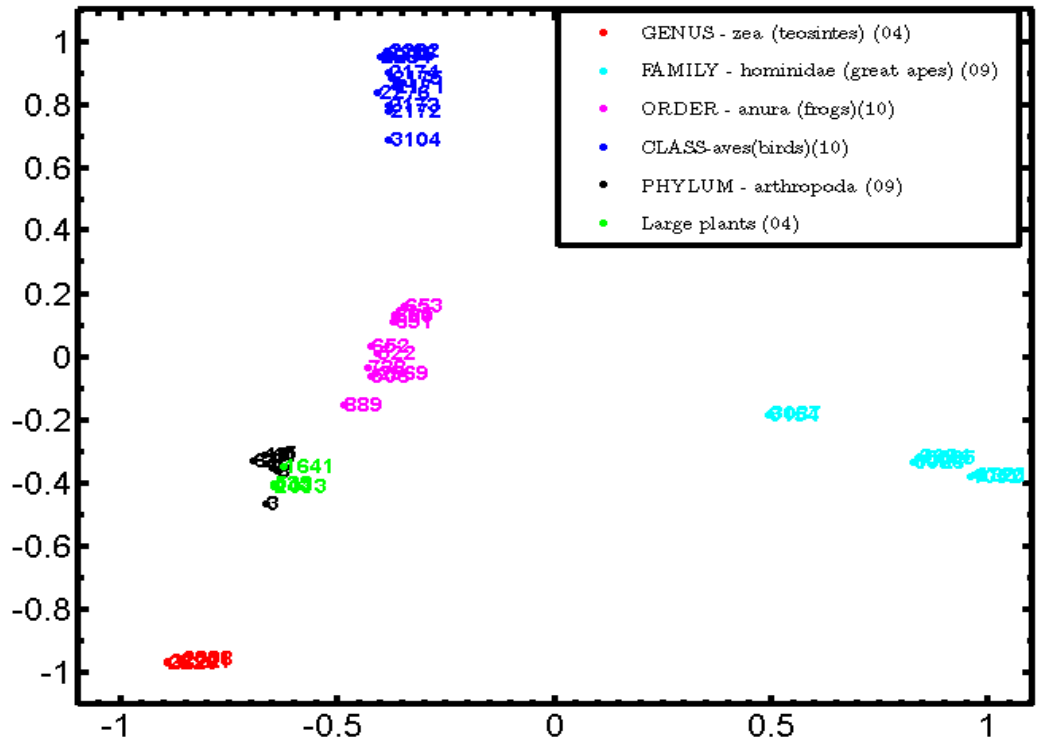
Figure 5.3: GDMs for the entire dataset of Wang *et al.* (2005) using the mtDNA genomes at different length truncations.

A similar experiment was performed using a dataset of mtDNA from species across classifications, which is a subset of the dataset for Figure 4.3, including species from a genus (Zea, plants, 4 species), a family (Hominidae, 9 species), an order (Anura, frogs, 10 species), a class (Aves, birds, 10 species) and a phylum (Arthropoda, 9 species), and additional plants with very long mitochondrial genomes (4 species). The dataset is described in Table 5.4. The following figures are Genome Distance Maps obtained using successive truncations of each mtDNA of the organisms in the dataset. The first Genome Distance Map of Figure 5.4, uses as input data the first 8,000 nt from each mitochondrial genome in the dataset, the second map uses the first 10,000 nt from each genome, and the subsequent ones use the first 12,000 nt, 15,000 nt and full genomes respectively. If we compare the subsequent figures with the last figure, where the original length is preserved, we cannot draw a definite conclusion as to which figure works best. For 8,000 nt there is one mix up of arthropods with large plants, for 10,000 nt, there are a couple of mix ups of arthropods with the same plants, but for 12,000 nt there are more

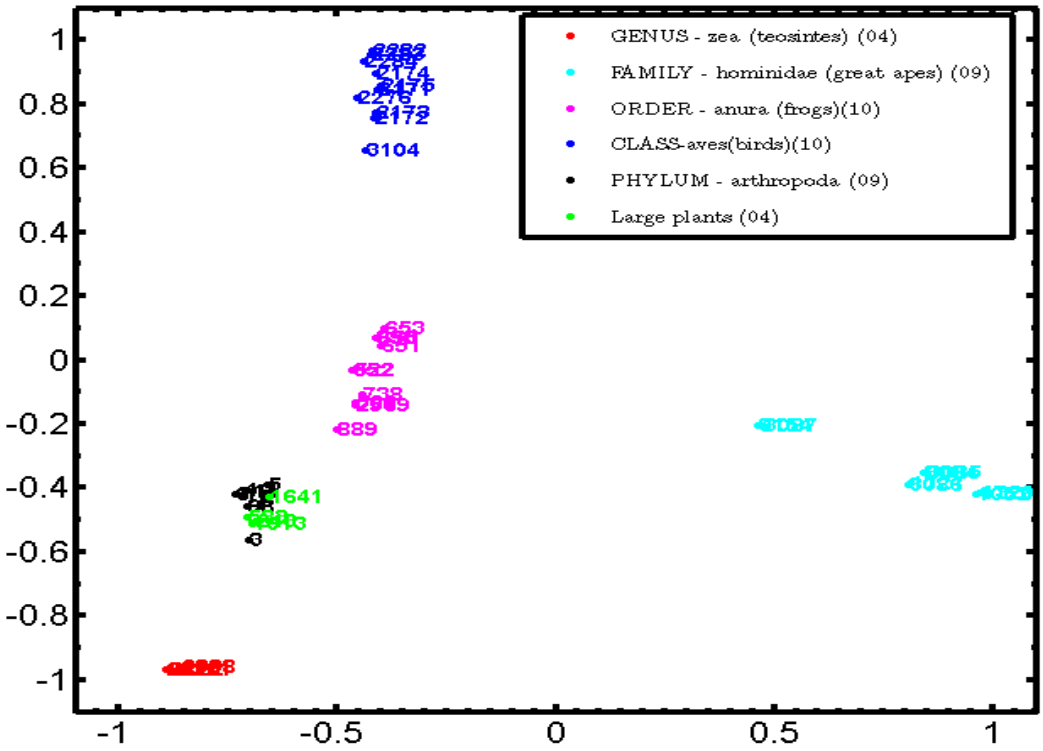
than 3 mix ups, and for 15,000 nt only one mix up. On the other hand, with the exception of these mix ups of arthropods with large plants, all the other genomes group together very well even if we take the first 8,000 nt. Consequently, for this experiment, a nucleotide range of 12,000-20,000 seems well suited for CGR/SSIM analyses. Nevertheless, this range can vary for different situations and different applications.



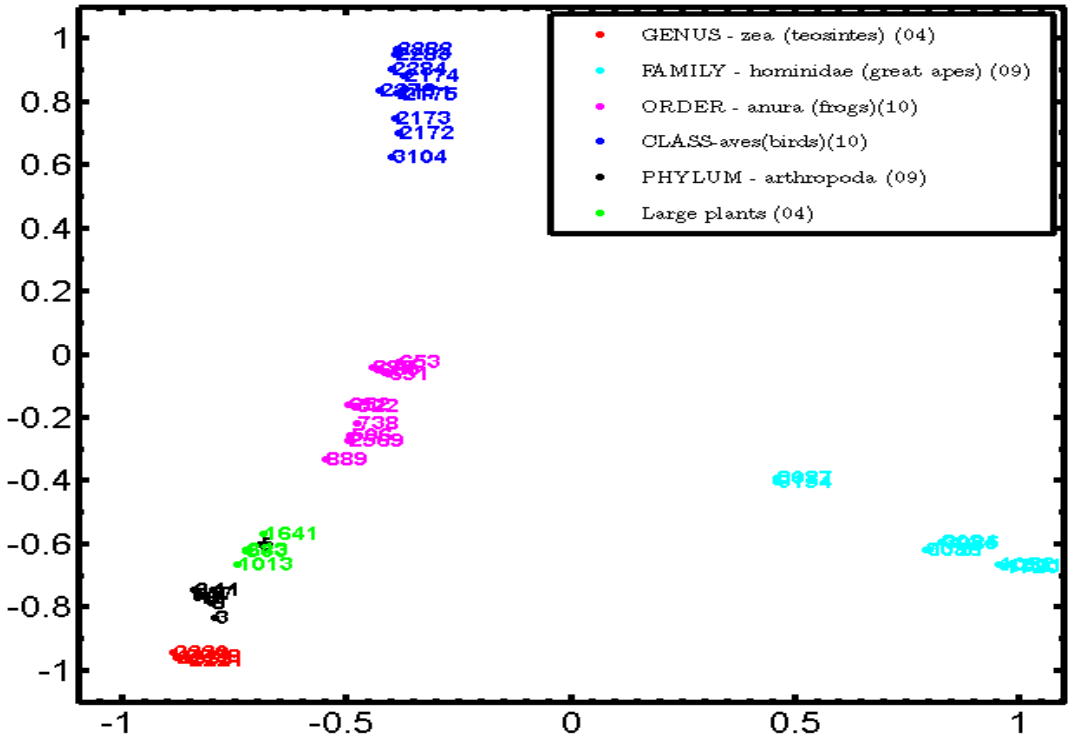

(a) GDM for a sub-dataset of the dataset in Figure 4.3 taking the first 8,000 nt of each mtDNA genome.



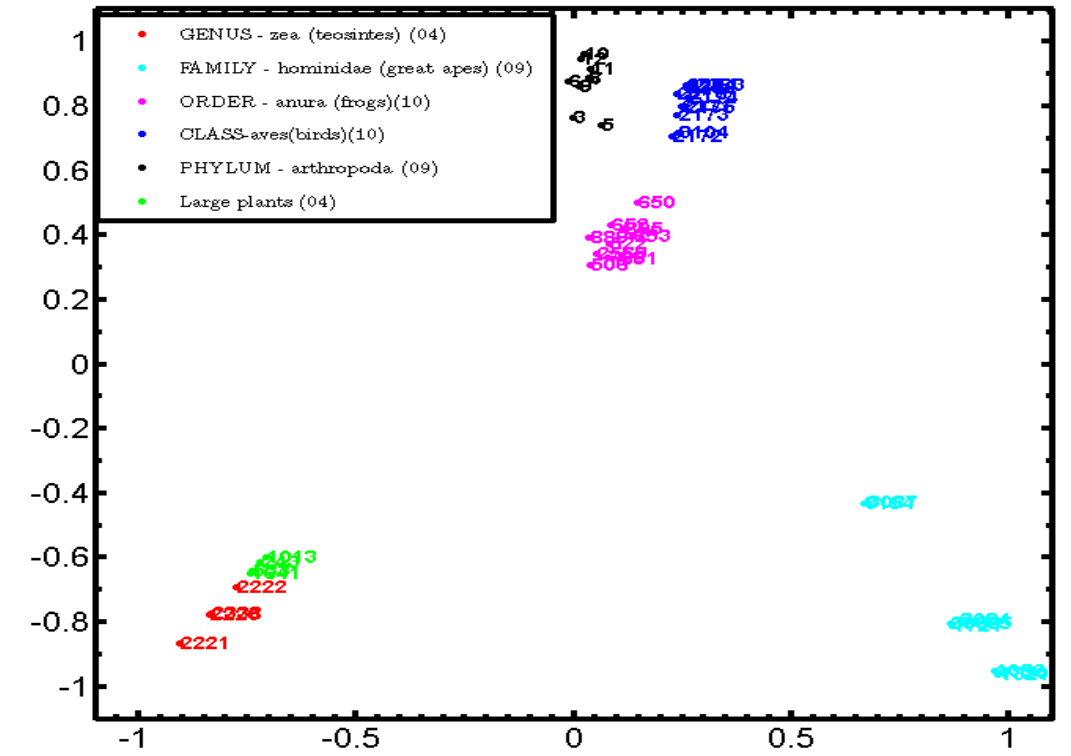

(b) GDM for a sub-dataset of the dataset in Figure 4.3 taking the first 10,000 nt of each mtDNA genome.




(c) GDM for a sub-dataset of the dataset in Figure 4.3 taking the first 12,000 nt of each mtDNA genome.




(d) GDM for a sub-dataset of the dataset in Figure 4.3 taking the first 15,000 nt of each mtDNA genome.




(e) GDM for a sub-dataset of the dataset in Figure 4.3 taking the complete mtDNA genome.

Figure 5.4: Length experiment: Across classifications.

| Number | Accesion_number | Name | Sequence length |
|---|---|---|---|
| 3 | NC_018118 | *Elodia flavipalpis* | 14932 |
| 5 | NC_018120 | *Mastotermes darwiniensis* | 15487 |
| 6 | NC_018121 | *Porotermes adamsoni* | 16039 |
| 7 | NC_018122 | *Microhodotermes viator* | 15704 |
| 8 | NC_018123 | *Zootermopsis angusticollis* | 15483 |
| 9 | NC_018124 | *Neotermes insularis* | 15799 |
| 10 | NC_018125 | *Coptotermes lacteus* | 16326 |
| 11 | NC_018126 | *Schedorhinotermes breinli* | 15864 |
| 12 | NC_018127 | *Heterotermes sp. SLC-2012* | 16370 |
| 2220 | NC_008331 | *Zea perennis* | 570354 |
| 2221 | NC_008332 | *Zea mays subsp. parviglumis* | 680603 |
| 2222 | NC_008333 | *Zea luxurians* | 539368 |
| 2338 | NC_007982 | *Zea mays subsp. mays* | 569630 |
| 2171 | NC_008548 | *Micrastur gilvicollis* | 17344 |
| 2172 | NC_008549 | *Pteroglossus azara flavirostris* | 18736 |
| 2173 | NC_008550 | *Pandion haliaetus* | 17864 |
| 2174 | NC_008551 | *Ardea novaehollandiae* | 17511 |
| 2175 | NC_008540 | *Apus apus* | 17037 |
| 2276 | NC_008132 | *Nipponia nippon* | 16732 |
| 2282 | NC_008138 | *Eudyptes chrysocome* | 16930 |
| 2283 | NC_008139 | *Gavia pacifica* | 15574 |
| 2284 | NC_008140 | *Podiceps cristatus* | 16134 |
| 1052 | NC_013993 | *Homo sp. Altai* | 16570 |
| 1321 | NC_012920 | *Homo sapiens* | 16569 |
| 1720 | NC_011137 | *Homo sapiens neanderthalensis* | 16565 |
| 1721 | NC_011120 | *Gorilla gorilla gorilla* | 16412 |
| 3084 | NC_001643 | *Pan troglodytes* | 16554 |

| | | | |
|---|---|---|---|
| 3085 | NC_001644 | *Pan paniscus* | 16563 |
| 3086 | NC_001645 | *Gorilla gorilla* | 16364 |
| 3087 | NC_001646 | *Pongo pygmaeus* | 16389 |
| 3154 | NC_002083 | *Pongo abelii* | 16499 |
| 506 | NC_016119 | *Nanorana pleskei* | 17660 |
| 522 | NC_016059 | *Rana chosenica* | 18357 |
| 650 | NC_015615 | *Hymenochirus boettgeri* | 18007 |
| 651 | NC_015617 | *Pipa carvalhoi* | 19534 |
| 652 | NC_015618 | *Pseudhymenochirus merlini* | 18029 |
| 653 | NC_015620 | *Rhinophrynus dorsalis* | 17299 |
| 738 | NC_015305 | *Rana ishikawae* | 21020 |
| 889 | NC_014685 | *Occidozyga martensii* | 18321 |
| 895 | NC_014691 | *Leiopelma archeyi* | 16593 |
| 2569 | NC_006408 | *Polypedates megacephalus* | 16473 |
| 3104 | NC_001323 | *Gallus gallus* | 16775 |
| 243 | NC_016740 | *Phoenix dactylifera* | 715001 |
| 1641 | NC_012119 | *Vitis vinifera* | 773279 |
| 1013 | NC_014050 | *Cucurbita pepo* | 982833 |
| 533 | NC_016005 | *Cucumis sativus* | 1555935 |

Table 5.4: Dataset of the species in Figure 5.4.

## 5.5 Pseudo genome experiment: Robustness of CGR

This experiment was done to verify the validity of the conclusion made by Goldman in 1993 [Gol93] that CGR does not give more insight than the mono, di- and trinucleotide frequencies of a genome. To analyze this statement, artificial genomes of *Ustilago maydis* (GN: 2214, fungus), *Arabidopsis thaliana* (GN: 3167, plant), *Apis mellifera ligustica* (GN: 3106, insect), *Danio rerio* (GN: 3048, fish), *Gallus gallus* (GN: 3104, bird), and *Homo Sapiens* (GN: 1321)

were added to the dataset from [WHSK05].

For each of these genomes, three pseudo-genomes were generated. For example, for the human mitochondrial genome, GN: 1321, the pseudo-genome 1321a, a DNA sequence with the same length and single nucleotide frequency as the original; the pseudo-genome 1321b with the same length and dinucleotide frequency as the original; the pseudo-genome 1321c with the same length and trinucleotide frequency as the original , and similar for the other five genomes were added.

Figure 5.5 represents the Genome Distance Map of the original genomes along with the above pseudo-genomes. We can note the dramatic change in the positions of the human and chicken pseudo-genomes compared with the original genomes' position. This experiment contradicts the statement made by Goldman [Gol93] for CGRs. Even having same length and same single, di- and trinucleotide frequencies as the original genome, the artificial sequences are farther away from the original genome. For instance, the human pseudo-genomes (1321a, 1321b, 1321c), moved dramatically away from the real human genome (GN: 1321). The distances, $\delta(GN:1321, GN:1321a) = 0.9540$, $\delta(GN:1321, GN:1321b) = 0.9467$, and $\delta(GN:1321, GN:1321c) = 0.935$, mean that all three human pseudo-genomes were farther away from the original human genome than *Drosophila yakuba* (GN: 3103, fruit fly) is, with $\delta(3103, 1321) = 0.934$. In addition, the human pseudo-genomes were farther away from the mammal cluster, e.g., from *Bos taurus* (GN: 2514, cow), *Mus musculus* (GN: 2757, mouse), *Balaenoptera physalus* (GN: 3102, finback whale), *Phoca vitulina* (GN: 3105, harbor seal), *Balaenoptera musculus* (GN: 3109, blue whale), *Halichoerus grypus* (GN: 3110, grey seal), *Rattus norvegicus* (GN: 3116, rat). Consequently, this indicates that single, di- and trinucleotide frequency may not contain sufficient information to classify a genomic sequence, contradicting thus Goldman's claim [Gol93] that "CGR gives no futher insight into the structure of the DNA sequence than is given by the dinucleotide and trinucleotide frequencies".

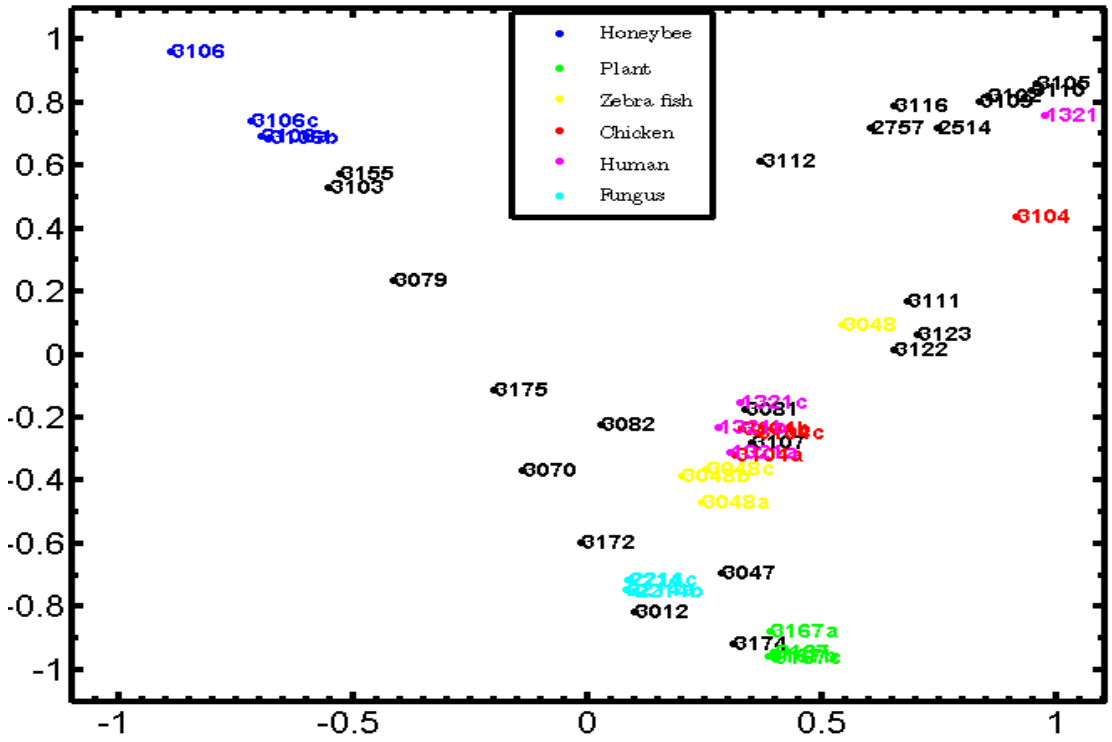



Figure 5.5: Genome Distance Map of the organisms from the data set of Want *et al.*[WHSK05] (in black) together with six other mitochondrial genomes (in colour) and their respective pseudo-genomes. The pseudo-genomes are marked by the letter *a* (same length, same single nucleotide frequency), *b* (same length, same single dinucleotide frequency) and *c* (same length, same single trinucleotide frequency) following the organism's identification number.

## 5.6 Graphs of DSSIM distances between the CGR images of human mtDNA and each of the 3,176 mitochondrial genomes of the dataset

To observe the overall behaviour of SSIM as a distance measurement method, this section describes the sorted and unsorted graphs of the DSSIM distances between the CGR images of the human mitochondrial genome and each of the 3,176 mitochondrial genomes. The minimum distance to *Homo sapiens* was found to be $\delta(1321, 1720) = 0.109$, the distance to *Homo sapiens neanderthalensis* (GN: 1720), and the second smallest distance is $\delta(1321, 1052) = 0.18$, the distance to *Homo sp. altai* (GN: 1052), with the third smallest distance being $\delta(1321, 3084) =$

0.4655 to *Pan troglodytes* (GN: 3084, chimp). In contrast, the maximum distance from the human mtDNA was found to be $\delta(1321,533) = 0.9946$, the distance from *Cucumas sadivus* (GN: 533, cucumber), the plant with the longest mitochondrial genome in the dataset, with a length of 1,555,935 nt. Overall the graph showed that the distance rose quickly to 0.65 with a second marked increase after reaching 0.80, and mostly ranging between 0.65 and 0.80.

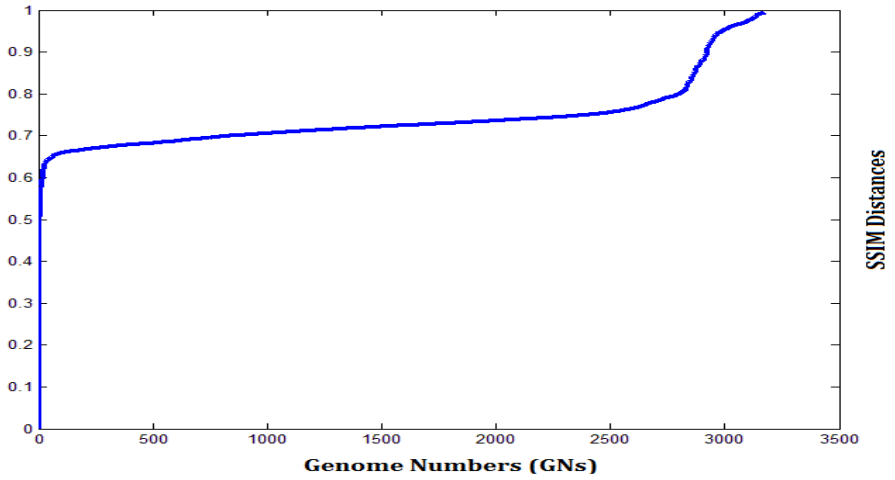



Figure 5.6: Graph of the SSIM distances between the CGR images of human mtDNA and each of the 3,176 mitochondrial genomes (sorted).

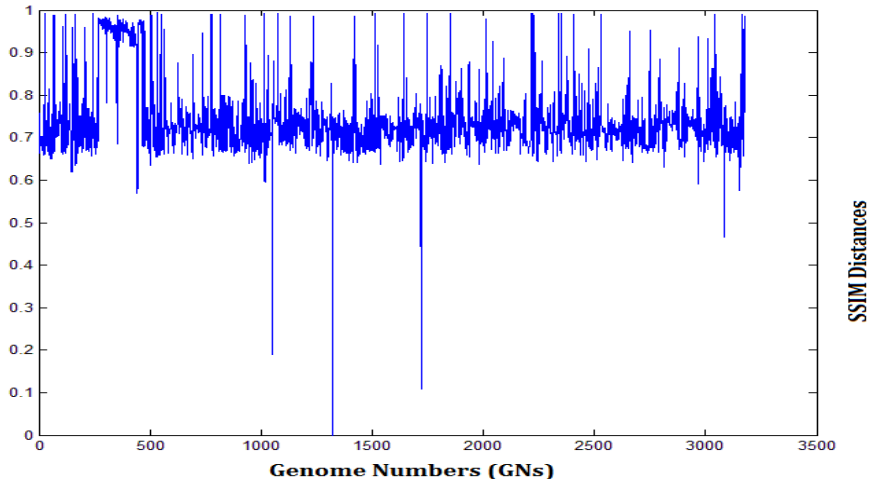



Figure 5.7: Graph of the DSSIM distances between the CGR images of human mtDNA and each of the 3,176 mitochondrial genomes (unsorted).

## 5.7 Graphs of DSSIM distances between the CGR images of the mitochondrial genome of an ancient eukaryote *Malawimonas jakobiformis*, and each of the 3,176 mitochondrial genomes

This section contains the sorted and unsorted graphs of the DSSIM distances between the CGR images of the mitochondrial genome of an ancient eukaryote *Malawimonas jakobiformis* (GN: 3028) and each of the 3,176 mitochondrial genomes.

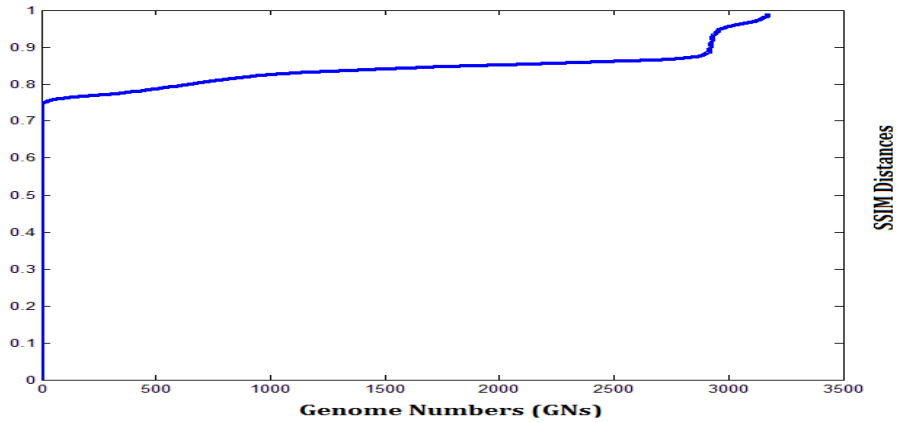



Figure 5.8: Graph of the SSIM distances between the CGR images of an ancient euraryote *Malawimonas jakobiformis* (GN: 3028) mtDNA and each of the 3,176 mitochondrial genomes (sorted).

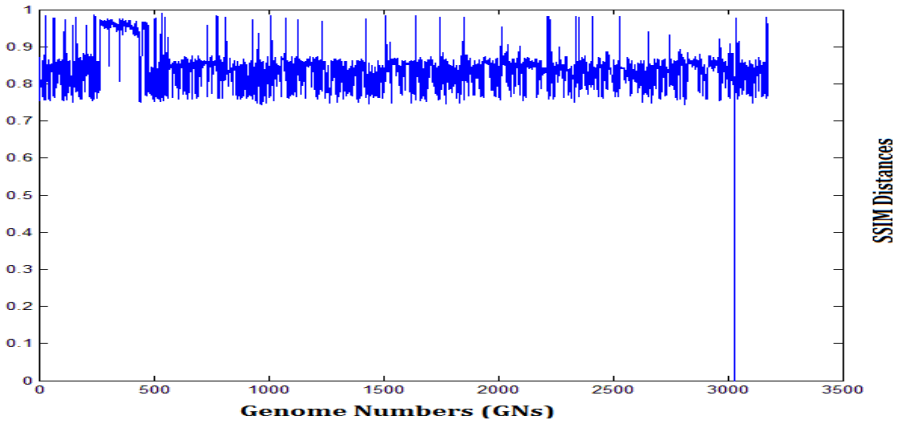



Figure 5.9: Graph of the SSIM distances between the CGR images of an ancient euraryote *Malawimonas jakobiformis* (GN: 3028) mtDNA and each of the other 3,176 mitochondrial genomes (unsorted).

The distances rise sharply to 0.75, with the majority of distances being between 0.75 and 0.85. Figure 5.8 and 5.9 shows the sorted and the unsorted graphs.

## 5.8 Software and tools used to implement this project

All the maps and implementations were implemented using MALTLAB R2010a. Pseudo genomes were generated using the software CLC Sequence Viewer, version 6, and a C++ program[1]. Phylogenetic trees were generated using the PHYLIP 3.69 package and the web interface of ITOL [LB11].

## 5.9 Discussion

The proposed method is successful in comparing a wide range of DNA sequences. Any genome can be imaged by CGR and subsequently compared with SSIM. CGR images can be successfully used to analyze species' relatedness starting from genomic sequences of any length.

The SSIM is a very sensitive image comparison method with the ability of detecting single nucleotide variation in two genomes. Substitution of a genome with a genome of a different species has the most impact (among insertion, deletion and substitutions) while using CGR/SSIM method. For our dataset, the computed DSSIM distances between all pairs of full length mitochondrial genomes varied from 0 to 0.9969. The minimum DSSIM distance 0 was found between the *Rhinomugil nasutus* (GN: 98, commonly known as shark mulet, sequence length 16,974 nt) and *Moolgarda cunnesius* (GN: 103, commonly known as longarm mullet, sequence length 16,974 nt). These two genomes are actually two identical genomes, which is exposed by a base by base sequence comparison. The maximum DSSIM distance 0.9969 was found between *Huperzia squarrosa* (GN:118, a firmoss, sequence length 413,530 nt) and *Candida subhashii* (GN:954, a yeast, sequence length 29,795nt). One interesting observation for the maximum SSIM is that the maximum DSSIM distance is not between the longest (GN: 533, cucumber, sequence length 1,555,935 nt) and the shortest (GN: 440, *Silene conica*, sequence length: 288 nt) mitochondrial DNA sequence in our dataset. This may be because,

[1] A C++ code written by S. Kopecki that generates DNA sequences with the same length and trinucleotide frequency as a given input DNA sequence

while comparing two CGRs with DSSIM, only the luminance distortion directly depends on length. In contrast, the two other parameters (contrast distortion and linear distortion) weakly depend on length of a genome.

The DSSIM distance matrix for our entire distance matrix fails to satisfy the property of a metric distance completely. The first exception was for the case of two different species $x, y$ with $\delta(x, y) = 0$ discussed earlier. The other violation was found for a triplet of three corals. The three genomes *Montastraea annularis* (GN: 2432, length 16,138 nt), *Montastraea franksi* (GN: 2433, length 16,137 nt), and *Montastraea faveolata* (GN: 2434, length 16,138 nt). The corresponding SSIM distances are as $\delta(2432, 2434) = 0.0008$, $\delta(2433, 2432) = 0.0096$, $\delta(2433, 2434) = 0.0087$. The reason for this violation could be due to the similarity among these three corals. The genomes are so close to each other that they could not maintain a property of metric distance.

The resulting Genome Distance Maps are generated using the MDS method. All the maps are scaled so that the range of $x$ and $y$ is [-1 1]. The *Stress* values for each of the maps are less than 0.2, which is within the acceptable range proposed in [Kru64], with the exception of Figure 4.10.

# Chapter 6

# Conclusions and future work

Representing DNA sequences graphically and measuring, as well as displaying, species' relationships have been considered to be major aspects of molecular biological research. This thesis discusses some of the 2D and 3D methods used to represent genomes and their potential applications. In addition, we discuss Chaos Game Representation and genomic signatures for DNA sequences.

In this thesis, a novel way to quantitatively measure species' relatedness in a Euclidean space using the mtDNA genomes of the species is proposed. The proposed method can be effectively used to compare species using their mitochondrial DNA. Moreover, the proposed Genome Distance Maps might be more informative than the phylogenetic trees, where relationships among distinct species are difficult to judge and a large number data points is challenging to visualize. Furthermore, whenever a new genome is sequenced, this method can be used to define the taxonomic classification for that genome.

This method is also applicable where we have an alphabet that can be transformed and partitioned into four subsets. For example, any binary sequence can be mapped with the CGR method by labelling the corners as 00, 01, 10, and 11.

The proposed method is usable for genome comparison. However, to make it more user friendly, one complete software project can be a possible future work, where the tool will be a web interface, the end users can give inputs, and the software would output a "google" map of genomes, with complete information displayed along with relatedness.

The SSIM algorithm can be improved so that it can work regardless of the sequence length.

In addition, higher-dimension representations for MDS can be experimented with to produce maps with lower *Stress*. Moreover, more image comparison methods can be empirically tested to gain better sensitivity when comparing two CGR images.

# Bibliography

[ACM+01] J. Almeida, J. Carrio, A. Maretzek, P. Noble, and M. Fletcher. Analysis of genomic sequences by Chaos Game Representation. *Bioinformatics*, 17(5):429–437, 2001.

[ALW04] F. Alexeyev, P. Ledoux, and L. Wilson. Mitochondrial DNA and aging. *Clin. Sci.*, 107(4):355–364, 2004.

[BG10] I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 2nd edition, August 2010.

[CC01] T. Cox and M. Cox. *Multidimensional Scaling*. Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, 2001.

[CD05] R. Chi and K. Ding. Novel 4D numerical representation of DNA sequences. *Chemical Physics Letters*, 407(1-3):63 – 67, 2005.

[CHBG09] H. Chatterjee, S. Ho, I. Barnes, and C. Groves. Estimating the phylogeny and divergence times of primates using a supermatrix approach. *BMC Evolutionary Biology*, 9(259), 2009.

[Cla91] D. Clayton. Replication and transcription of vertebrate mitochondrial DNA. *Annual Review of Cell Biology*, 7:453–478, 1991.

[CMK99] A. Campbel, J. Mrazek, and S. Karlin. Genome signature comparisons among prokaryote, plasmid, and mitochondrial DNA. *Proceedings of the National Academy Sciences of the USA*, 96(16):9184–9, 1999.

[DD92] C. Dutta and J. Das. Mathematical characterization of Chaos Game Representation: New algorithms for nucleotide sequence analysis. *Journal of Molecular Biology*, 228(3):715–719, 1992.

[DGV+99] P. Deschavanne, A. Giron, J. Vilain, G. Fagot, and B. Fertil. Genomic signature: characterization and classification of species assessed by Chaos Game Representation of sequences. *Molecular Biology and Evolution*, 16(10):1391–1399, 1999.

[DGV+00] P. Deschavanne, A. Giron, J. Vilain, C. Dufraigne, and B. Fertil. Genomic signature is preserved in short DNA fragments. In *Bio-Informatics and Biomedical Engineering, 2000. Proceedings. IEEE International Symposium*, number 115, pages 161–167, 2000.

[DQW+06] M. Dunham, D. Quick, Y. Wang, M. McGee, and J. Waddle. Visualization of DNA / RNA Structure using Temporal CGRs . *Bioengineering*, pages 171–178, 2006.

[Gat86] M. Gates. A simple way to look at DNA. *Journal of Theoretical Biology*, 119(3):319–328, 1986.

[GK01] A. Gentles and S. Karlin. Genome-scale compositional comparisons in eukaryotes. *Genome Research*, 11(4):540–546, 2001.

[Gol93] N. Goldman. Nucleotide, dinucleotide and trinucleotide frequencies explain patterns observed in Chaos Game Representations of DNA sequences. *Nucleic Acids Research*, 21(10):2487–2491, 1993.

[HB92] J. Harvey and D. Barnett. Endocrine dysfunction in Kearns-Sayre syndrome. *Clin. Endocrinol. (Oxf.)*, 37(1):97–103, 1992.

[HCBD03] P. Hebert, A. Cywinska, S. Ball, and J. Dewaard. Biological identifications through DNA barcodes. *Proc. Biol. Sci*, 270:313–321, 2003.

[HLZ00] B. Hao, H. Lee, and S. Zhang. Fractals related to long DNA sequences and complete genomes. *Chaos, Solitons and Fractals*, 11(6):825–836, 2000.

[HSS92] K. Hill, N. Schisler, and S. Singh. Chaos Game Representation of coding regions of human globin genes and alcohol dehydrogenase genes of phylogenetically divergent species. *J. Mol. Evol.*, 35(3):261–9, 1992.

[Jef90] H. Jeffrey. Chaos Game Representation of gene structure. *Nucleic Acids Research*, 18(8):2163–2170, 1990.

[JHS+11] N. Jameson, Z. Hou, K. Sterner, A. Weckle, M. Goodman, M. Steiper, and D. Wildman. Genomic data reject the hypothesis of a prosimian primate clade. *Journal of Human Evolution*, 61(3):295 – 305, 2011.

[KB95] S. Karlin and C. Burge. Dinucleotide relative abundance extremes: a genomic signature. *Trends in Genetics*, 11(7):283 – 290, 1995.

[KMC97] S. Karlin, J. Mrazek, and A. Campbell. Compositional biases of bacterial genomes and evolutionary implications. *J Bacteriol*, 179(12):3899–3913, 1997.

[Kru64] J. Kruskal. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29(1):1–27, 1964.

[KSDH] L. Kari, A. Sayem, N. Dattani, and K. Hill. Map of Life: Measuring and Visualizing Species Relatedness with Genome Distance Maps. *In press*.

[LB11] I. Letunic and P. Bork. Interactive tree of life v2: online annotation and display of phylogenetic trees made easy. *Nucleic Acids Res.*, 39:475–478, 2011.

[Les90] E. Lessa. Multidimensional analysis of geographic genetic structure. *Systematic Zoology*, 39(3):242–252, 1990.

[Lia05] B. Liao. A 2D graphical representation of DNA sequences. *Chemical Physics Letters*, 401(1-3):196–199, 2005.

[LLZX07] B. Liao, R. Li, W. Zhu, and X. Xiang. On the Similarity of DNA Primary Sequences Based on 5-D Representation. *Journal of Mathematical Chemistry*, 42(1):47–57, 2007.

[LM95] P. Leong and S. Morgenthaler. Random walk and gap plots of DNA sequences. *Computer applications in the biosciences : CABIOS*, 11(5):503–507, 1995.

[Mil12] S. Milius. New species of the year. *Science News*, 182(13):30, 2012.

[MTA+11] C. Mora, D. Tittensor, S. Adl, A. Simpson, and B. Worm. How many species are there on earth and in the ocean? *PLoS Biology*, 9(8):1–8, 2011. e1001127.

[Nan94] A. Nandy. A new graphical representation and analysis of DNA sequence structure: Methodology and application to globin genes. *Current Science*, 66(4):309 – 314, 1994.

[NW70] S. Needleman and C. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443 – 453, 1970.

[Opp12] S. Oppenheimer. Mitochondrial DNA: The Eve Gene, 2012.

[QLQ11] Z. Qi, L. Li, and X. Qi. Using Huffman coding method to visualize and analyze DNA sequences. *Journal of Computational Chemistry*, 32(15):3233–3240, 2011.

[RVNB00] M. Randic, M. Vracko, A. Nandy, and S. Basak. On 3D graphical representation of DNA primary sequences and their numerical characterization. *Journal of Chemical Information and Computer Sciences*, 40(5):1235–1244, 2000.

[Sim05] T. Simonite. Protists push animals aside in rule revamp. *Nature*, 438:8–9, 2005.

[SNC77] F. Sanger, S. Nicklen, and A. Coulson. DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America (PNAS)*, 74:5463–5467, 1977.

[SSZ10] L. Sirovich, M. Stoeckle, and Y. Zhang. Structural analysis of biodiversity. *PLoS ONE*, 5(2):e9266, 2010.

[TTR+07] I. Tavassoly, O. Tavassoly, M. Rad, N. Dastjerdi, and N. Mottaghi. Three dimensional Chaos Game Representation of genomic sequences. In *Proceedings of the 2007 Frontiers in the Convergence of Bioscience and Information Technologies*, FBIT '07, pages 219–223, 2007.

[Tu09] R. Tu. Three dimensional Chaos Game Graphical Representation of quaternary and binary data, and application to genomic and melodic signatures. *M.Sc. thesis, Department of Computer Science, University of Western Ontario*, 2009.

[VVP06] D. Voet, J. Voet, and C. Pratt. *Fundamentals of Biochemistry*. Wiley & Sons, 2 edition, 2006.

[WBSS04] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[WHSK05] Y. Wang, K. Hill, S. Singh, and L. Kari. The spectrum of genomic signatures: From dinucleotides to Chaos Game Representation. *Gene*, 346:173–185, 2005.

[Wik13a] Wikipedia. Biological classification, the free encyclopedia, 2013. [Online; accessed 22-January-2013].

[Wik13b] Wikipedia. Deoxyribonucleic acid DNA, the free encyclopedia, 2013. [Online; accessed 22-January-2013].

[YLW03] C. Yuan, B. Liao, and T. Wang. New 3D graphical representation of DNA sequences and their numerical characterization. *Chemical Physics Letters*, 379:412 – 417, 2003.

[YLY+10] C. Yu, Q. Liang, C. Yin, R. He, and S. Yau. A novel construction of genome space with biological geometry. *DNA Research*, 17(3):155–168, 2010.

[YSW09] J. Yu, X. Sun, and J. Wang. TN curve: A novel 3D graphical representation of DNA sequence based on trinucleotides and its applications. *Journal of Theoretical Biology*, 261(3):459 – 468, 2009.

[YW04] Y. Yao and T. Wang. A class of new 2D graphical representation of DNA sequences and their application. *Chemical Physics Letters*, 398(4-6):318–323, 2004.

[ZSZ+12] Z. Zhang, T. Song, X. Zeng, Y. Niu, Y. Jiang, L. Pan, and Y. Ye. Colorsquare: A colorful square visualization of DNA sequences. *Communications in Mathematical and in Computer Chemistry*, 68(2):621–637, 2012.

# Vita

| | |
|---|---|
| NAME: | Abu Sadat Md. Sayem |
| PLACE OF BIRTH: | Nilphamari, Bangladesh |
| YEAR OF BIRTH: | 1985 |
| POST-SECONDARY EDUCATION AND DEGREES: | University of Western Ontario<br>London, Ontario, Canada<br>2010-2013 M.Sc. studies |
| | University of Dhaka<br>Dhaka, Bangladesh<br>2004-2009 B.Sc. |
| RELATED WORK EXPERIENCE: | Research Intern<br>National Institute of Informatics.<br>February 2010 - June 2010 |
| | Research and Teaching Assistant<br>University of Western Ontario<br>2010 - 2013 |

# PUBLICATION

1. L. Kari, A. Sayem, N. Dattani, K. Hill. Map of Life: Measuring and Visualizing Species' Relatedness with Genome Distance Maps. *(In press)*.

2. A. Sayem, S. Mitra. Efficient approach to design low power Reversible Logic Blocks for Field Programmable Gate Arrays (FPGA). *IEEE International Conference on Computer Science and Automation Engineering*, 251-255, 2011.

3. A. Sayem, M. Ueda. Optimization of reversible sequential circuits. *Journal of Computing*. 2(6): 208-214, 2010.

4. A. Sayem, M. Polash, H. Babu. Design of a reversible logic block of field Programmable Gate Array (FPGA). *Silver Jubilee Conference on Communication Technologies and VLSI design (CommV09)*, 500-501, 2009. **Best student paper award**.