3D Graphical Representation of DNA Sequences and its Application for Long Sequence Searching over Whole Genomes

[redacted] [redacted] [redacted] [redacted]

[redacted]

ABSTRACT

With the development of Next Generation Sequencing techniques, the analysis of megabyte-sized whole genome sequence has been common. In general genome sequence comparison is conducted by alignment algorithm model. It is accurate, but assuming that the length of the target sequence is short(less than a few kilobytes) since it requires the quadratic time and space complexity, $O(n^2)$ where n is the length of target and query sequences.

To overcome these drawbacks in whole genome scale comparison, we suggest a new method for finding local similar subsequences among whole genomes based on random walk visualization. So that the sequence searching problem in DNA strings can be reduced to find some parts of geometric object within a relatively small-scale geometric space. When comparing similarity by modifying sequences of similar length, we can confirm that the comparison model is appropriate by accurately reflecting the degree of similarity. When searching the query sequence comparison model based on 200MB sized whole genome sequence, using the compressed coordinate information, it was able to search the 10MB sequences in 22s, which is a very reduced time compared to alignment.

CCS CONCEPTS

• **Human-centered computing** → **Scientific visualization**; **Visual analytics**; • **Applied computing** → *Bioinformatics*; • **Information systems** → Structured text search;

KEYWORDS

Whole Genome, Genome Similarity, Genome Visualization, Genome Search

ACM Reference Format:
Da-Young Lee, Hae-Sung Tak, Han-Ho Kim, Hwan-Gue Cho*. 2017. 3D Graphical Representation of DNA Sequences and its Application for Long Sequence Searching over Whole Genomes. In *SoICT '17: Eighth International Symposium on Information and Communication Technology, December 7–8, 2017, Nha Trang City, Viet Nam.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3155133.3155208

1 MOTIVATION

Basic Local Alignment Search Tool(BLAST)[1] is the most common method to search for sequences in a database. It divides the query sequence into three characters, finds the matching region, and gradually widens the region to select candidates for alignment. Although it is very useful when searching for a short query in the whole database, since it is based on alignment, it is difficult to obtain an immediate processing result in the case of a large sequence such as a megabyte-scale chromosome owing to a large increase in computational cost. When utilizing the actual BLAST service, it is recommended to reduce the database search scope when the query size is of the order of megabytes, and it is often time consuming to search and provide results by mail, rather than providing it immediately.

In this paper, we propose a geometric-based heuristic technique that enables the rapid comparison and search of sequences in personal computers. In this regard, AMSS[10] is a model that provides shape-based similarity comparison, assuming that the time series data is a vector sequence. Instead of focusing on individual points of time series data, the model focuses on vectors and compares similarities between data using cosine similarity. This method is advantageous in that it is effective for amplitude and time shifting. In this study, we also aimed to reduce the time and space complexity by converting the genetic sequence into a geometric object such as a random walk plot and performing comparison and search, taking into account that the genetic sequence data is ordered sequence data. Instead of considering a single separate base, as in the alignment algorithm, the method compares the vector generated based on the sequence of the predetermined unit only once, and it is possible to significantly reduce the time required for comparison operation by visualizing a sequence search result and presenting the information more intuitively. In addition, the high-speed heuristic search technique can be applied to large amounts of data, and it is possible to specify the necessary precise alignment analysis.

2 RELATED WORK

2.1 Genome Sequence Visualization

Most genetic data have a huge volume, and it is difficult to find meaningful patterns in such data owing to the irregular configuration of the four bases. The visualization of sequence information and sequence analysis information can help in forming an intuitive understanding of the genomic data and enable the efficient representation of the results. Genome visualization research focuses on two aspects. The first is the visualize of a large amount of genetic

SoICT '17, December 7–8, 2017, Nha Trang City, Viet Nam
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5328-1/17/12…$15.00
[redacted]

information in a short time and a limited space, and the second is the representation of complex information as intuitively as possible.

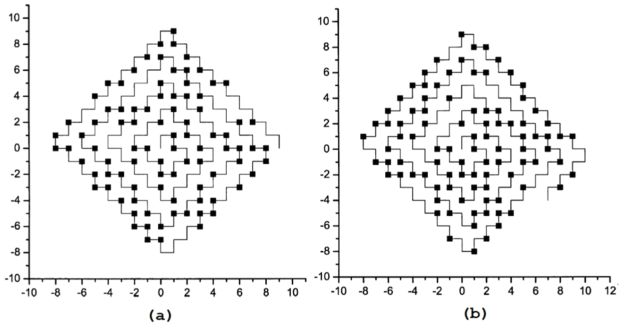

Figure 1: The compact graphical representation[14] of the first exon of human β-globin gene(a) and gorilla β-globin gene. The visualization of search result for query sequence of 10M size in human chromosome 1.

The 'Worm Curve'[12, 14] represents genome information in a limited space, and it assigns a binary code to each base. It is plotted on a Cartesian coordinate system, and its most significant biggest advantage is that the curve can represent all the information in a relatively small space, despite how little the point intersects with each other. Studies have been actively conducted using a variety of curves to intuitively represent complex information. For example, the 'Dual-Base Curve'(DB-Curve)[16] has been designed to visualize the features of a genome sequence at a glance. In this curve, the two different bases are configured as a combination, and a two-dimensional vector is assigned, where the y component is assigned as a constant (+1) and the x components are assigned separately. In this visualized, since the curve is continuous in the positive direction of the y axis, there is no point at which it crosses with itself. Obtaining a ratio of the x-coordinates of the end points can confirm the relative existing ratio of the two bases to obtain the statistical information of the sequence in an intuitive manner.

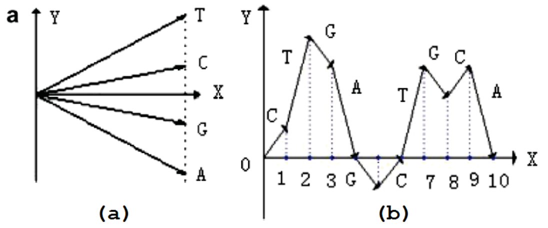

Figure 2: The vector design of 'H-L curve'[8](a) and graphical representation for the DNA sequence s ='ATGGCATGCA'(b).

In contrast, the 'H-L curve'[8] is a method of assigning a two-dimensional vector for the four bases with a constant x component, and this curve avoids intersection with itself because different y-components are assigned. Since the progress of a DNA sequence matches one-to-one with the 'H-L Curve,' it has the advantage that the main difference of each sequence with other sequences can be checked quickly.

In addition to visualizing curves, there is a 'Four-Color Map'[13], which assigns colors to each base and fills areas proportional to the frequency of occurrence with the corresponding color, and 'Circos'[2, 6], which visualizes the whole genome in a circular track form. 'Circos' represents a chromosome as a piece of a circular track, and connects the interactive chromosome tracks with a curve, thereby effectively expressing the internal relation of the whole genome. Although most relational connection visualization methods express only one-to-one associations, 'Circos' can express many-to-many associations as well by using circular tracks.

2.2 Genome Sequence Analysis with Visualization Tool

To compensate for the drawbacks of the sequence alignment method in terms of processing speed, a heuristic method based on visualization is utilized. By converting a large amount of text information composed of only four kinds of bases, the meaning of which is difficult to intuitively grasp, to geometry information, heuristic methods are able to identify the type of data through visual examination to easily find patterns that cannot be revealed using computational methods[11]. Furthermore, geometric rules found in the visible results often have a meaningful relationship with genomic analysis in the field. Heuristic methods are especially useful when utilized for quickly calculating similarity or dissimilarity.

For example, large-scale genomic sequence information is converted into information on a polygon domain, and the problem of finding similarity is solved by replacing the comparison of similarity of sequences with the comparison of image similarity[5]. By setting a direction for each base, the sequence is converted to a walk plot in which the polygon area is simplified with the k-convex hull, and the homology of two walk plots is compared. Studies [3, 7] have considered the extended space up to three dimensions in the vector assignment for each base. Consequently, a random walk plot can be visualized on three dimensions, and the similarity can be compared by simplifying it to be close to the actual walk plot.

Since direct comparison is difficult for a walk-plot object in three dimensions, a walk plot is populated in a certain space around the polygon area, and the orthogonal projection of this space on each plane (X-Y, Y-Z, and X-Z) is used to compare the degree of similarity using the overlap area ratio. However, the comparison method based on the overlapping area has a drawback in that it does not take into account the walk plot present in the local area. To overcome this drawback without simplifying the walk plot, the shape of the line is maintained while the shortest distance between any points of two walk plots is calculated for comparing the degree of similarity between two sequences[4].

Previously, an alignment method called 'Four Line' involving graphical-domain sequence alignment, rather than string alignment, was proposed[15]. By assigning the four bases to different points on the Y-axis and connecting the matched points in the sequence to be subjected to alignment in the X-axis to make a visualization of the zigzag curve, the visualization result of the two sequences are compared to conduct alignment.

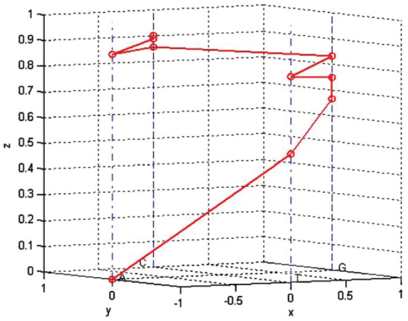

Figure 3: 3D graphical representation of DNA sequence using Z-axis as time axis[9]. The graphical representation for the sequence 'ATGGTGCACC'.

In order to overcome the disadvantages such as loss of information and self-intersection of existing two-dimensional visualization methods, there is a study in which a DNA sequence is three-dimensionally utilized as a time axis[9]. Regardless of the information of the base to the z-axis will always increases, and by assigning vectors x, y axis is increased or decreased for each base. Not only it limited to visualization, to derive the geometrical center of the curve, this time the center of this curve is important information indicating the distribution of each base. In this study, a similarity comparison model was devised by assigning vectors to each other in different ways and using the Euclidean distance and angle correlation of the distance to the start and end points of the vector through eight transform. As a result, they could construct the similarity matrix, it shown that the similar species such as human and gorilla have high similarity.

In this manner, visualization results can be used not only for the intuitive delivery of sequence information but also as an analysis target to improve the processing speed and to obtain meaningful results. In this study, by focusing on this point, we convert a whole genome sequence to a walk-plot object in three-dimensional space, extract a vector, and compare and search for the sequence with improved speed. Furthermore, by visualizing a search query sequence together with the walk plot of the whole genome sequence, the position and distribution of the obtained similar sequence can be transferred in an intuitive form.

3 NEW METHOD USING 3D RANDOM PLOT

3.1 3D Random Plot Representation for DNA Sequence

An overview of our algorithm framework is shown in Figure 4. Generally, all types of biological sequence comparison exploit the sequence alignment based on a dynamic programming approach. One popular alignment algorithm is the Needlemann–Wunsch algorithm, which is widely used in molecular biology. There are many variations in sequence alignment, such as global alignment, local alignment, and semi-global alignment. Though the alignment approach has many advantages, it has a critical drawback in that it involves high complexity in terms of execution-time complexity and space complexity. The complexity of the basic alignment algorithm is $O(m \cdot n)$ if the lengths of two input sequences are n and m. If $\Theta(n) = \Theta(m)$, the complexity is quadratic: $O(n^2)$. When the size of the input sequence is greater than 100 megabytes, this alignment is impractical, because it requires a main memory greater than the order of gigabytes. To overcome these problems, researchers developed heuristic alignment techniques such as BLAST-like tools. Another problem in the alignment algorithm is that it is not easy to define the score/penalty matrix to meet the many different constraints in biological sequence comparison.

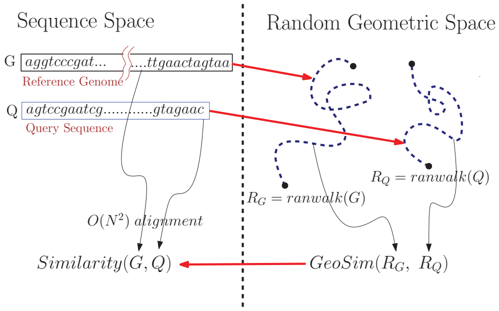

Figure 4: Space transform diagram.

The basic idea of our approach is that we compute the similarity of two sequences in "geometric random walk" space, rather than "string(sequence)" space. As shown in Figure 4, we first transform the input sequences into random walks in 3D space. Then, we compare or search for a target sequence in 3D geometric object. Since the size of the transformed random walk can be represented in a $500 \times 500 \times 500$ grid, the sequence, tens of megabytes in length, can be represented as a list with less than 100,000 linked pixels. Thus, we can say that our geometric transformation is a type of approximation with visualization. The advantage of our transformation is that the global structure can be shown by hiding the biological noise embedded in the sequence. The main merit of our approach is that it is useful and efficient in comparing very long sequences. Assume that we are asked to find the location of a sequence that is a few megabytes in length in a whole genome longer than 100 megabytes.

3.2 Vector Allocation for Walk Plot

Sequence data are string information composed of {a,g,t,c}; therefore, they must be converted into graphical information for visualization. Previous 2-D visualization methods have visualized genome sequences by assigning a separate base in the positive and negative directions of each axis (x and y). This method has a disadvantage in that a large amount of information is lost when a base having a vector in opposite directions is continuously repeated. Furthermore, if the same pattern is continuously repeated, it is impossible to visualize a large volume of data in a limited space. To overcome this disadvantage, [3] used a 3D vector. A vector is assigned to each base, but a combination of two bases constitutes a walk plot. When the two bases are coupled together with the vector in the opposite direction, the representation is made three-dimensional with a z-axis to minimize the lost information. In this study, by using a 3D

Table 1: Vector allocation method for each 2-mer base in a genome sequence in three-dimensional geometric space

2mer	Vector	2mer	Vector
AA	(2, 0, 0)	AG(GA)	(1, 1, 0)
AC(CA)	(1, -1, 0)	AT(TA)	(0, 0, -2)
CC	(0, -2, 0)	CG(GC)	(0, 0, +2)
CT(TC)	(-1, -1, 0)	GG	(0, 2, 0)
GT(TG)	(-1, 1, 0)	TT	(-2, 0, 0)

vector allocation model[3], we calculate the vector character of the sequence data and obtain sequence search positions to visualize the results. Table 1 summarizes the vector allocation method for each 2-mer. The base pairs AT and GC are represented on the z axis. The other base pairs are represented as the sum of two unit vectors for each base, as given by the WS-curve method. After the vector transition for DNA genome data information, those vectors are visualized in three-dimensional space. The method of visualization is the same as that of two-dimensional visualization, where the sum of vector values is computed according to the order of sequences and the results are connected with a line to provide the final visualization result. For the random walk plot R, the starting point is $R(0) = (X_0, Y_0, Z_0)$ $(X_0 = Y_0 = Z_0 = 0)$. $Unit^{3d}(i)$ is the converted value of the ith 2-mer of the unit vector. The ith point $R(i) = (X_i, Y_i, Z_i)$ of the random walk plot is computed as follows:

$$R(i) = R(i-1) + Unit^{3d}(i) = \sum_{0}^{i} Unit^{3d}(i) \quad (1)$$

Figure5 shows the direction of the walk plot for each 2-mer read. Since the first 2-mer read, 'AA,' is on the x-axis (+2), it can be confirmed from figure (a) that the positive x-axis moves from the origin O. Since the next 2-mer read is 'AT,' a movement in the z-axis by (-2) can be confirmed.

In case of the short genome sequence, it can be represented in a 500 × 500 × 500 grid easily. But the large size sequence needs space normalization to visualize the walk plot in limited space. When the vectors of the walk plot are calculated, the points that are farthest from the origin $O(0,0,0)$ to the X, Y, and Z axes are max_x, max_y, max_z, and the view size of visualization is V, the normalized ith point $R(i) = (X_i, Y_i, Z_i)$ can be expressed as:

$$Regular(R(i)) = (X_i \frac{V}{max_x}, Y_i \frac{V}{max_y}, Z_i \frac{V}{max_z}) \quad (2)$$

This visualization model is so useful to compare the huge whole genome. Figure 6 shows advantage of this works[3]. We have constructed the 3D random walk plots from two whole genomes such as Human Chromosome 1 and Chimpanzee Chromosome 1. In Figure 6, red walk plot represents the Human and green one represents the Chimpanzee. Red walk plots are up in the positive direction of the X and Y-axis than the green one. This visualization method directly make us to confirm that two genomes are quite similar and the Human chromosome has more 'G' and 'A' base compared to Chimpanzee.

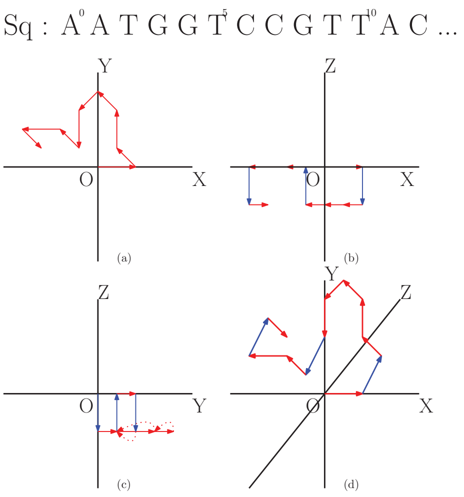

Figure 5: Movement of the walk plot for each 2-mer read. (a),(b),(c),and (d) show plots in the form of walks in the X-Y, X-Z, and Y-Z planes in three-dimensional space. From $O(0,0,0)$, the walk plot proceeds in accordance with the base assigned to 2-mer. The red walk plot represents movement on the X-Y plane, and the blue walk plot represents movement on the Z axis.

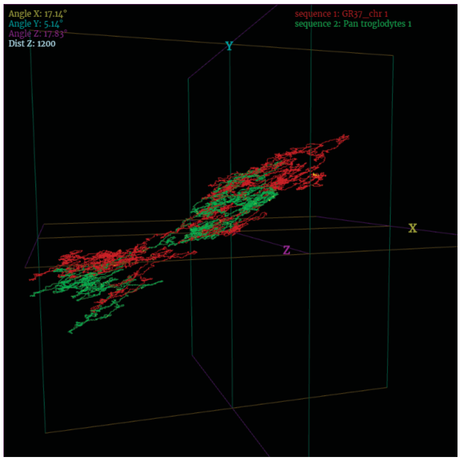

Figure 6: The visualization result of Human and Chimpanzee chromosome 1. Red plot is constructed from Human ch.1 and the green walk plot is constructed from the whole genome of Chimpanzee(Pan troglodytes) ch.1.

3.3 Vector Extraction from Random Walk Plot

For G, a genome sequence consisting of 4 DNA bases { a, g, t, c }, $ranwalk(G)$ represents a three-dimensional geometric object constructed by our proposed algorithm. Therefore, $ranwalk(G_i)$ consists of a list of linked pixels as follows:

Definition 3.1.

$$ranwalk(G) =< P_1, P_2, \ldots, P_l >$$

The position of a *ranwalk* pixel is denoted $P_i = (x_i, y_i, z_i)$ satisfying $|x_i - x_{i+1}| \leq 1$, $|y_i - y_{i+1}| \leq 1$ and $|z_i - z_{i+1}| \leq 1$, which means two pixels P_i and P_{i+1} are adjacent to each other, sharing a common face. We say P_i and P_{i+1} are 'adjacent' if they are within a distance of 1.

Now, we explain how to compute the distance between two *ranwalk* pixels obtained from two genomes G_a and G_b to be compared. Assume that we constructed two geometric objects, $R_a = ranwalk(G_a)$ and $R_b = ranwalk(G_b)$. The proposed distance measure, *random walk distance* (*Rdist*), is a vector with two components $\Delta Span$ and $\Delta Degree$. The proposed *Rdist*() measure has another parameter, depth k. The distance between two random walks R_a and R_b at depth k is defined recursively as follows. In this definition, R_a1 is the first half of R_a, and R_a2 is the last half of R_a. R_{b1} and R_{b2} are defined in a similar manner. Thus, $R_a = R_{a1} \odot R_{a_2}$, where $\odot$ denotes the geometric concatenation operation.

Definition 3.2.

$$Rdist(R_a, R_b, k) = Rdist(R_{a1}, R_{b1}, k+1) + Rdist(R_{a2}, R_{b2}, k+1)$$

Now, we explain how to compute $Rdist(R_a, R_b, k = 1)$ at the basic depth = 1 level. In Figure 7, the thick blue dotted curve represents the random walk for a genome sequence. Symbols $P_0(O)$ and P_1 denote the first and last pixel of a random walk plot, respectively. P_t denotes the first t-percentile pixel. Thus, $P_{0.5}$ denotes the exact middle pixel in the list of pixels generated by our transformation algorithm.

For an interval in a random walk, we obtain a parameter, the length of the direction vector (P_0, P_1). If two random walks to be compared start with the origin $(0, 0, 0)$, then we can obtain the lengths of two direction vectors from R_a and R_b and compute the angle difference between two vectors Pa_1 and Pb_1.

Assume the start and end points of R_a are P_{a0}, P_{a1}, and those of R_b are P_{b0}, P_{b1}. If $k = 1$ is, the comparison target is $\overrightarrow{P_{a0}P_{a1}}$ and $\overrightarrow{P_{b0}P_{b1}}$. If $k = 2$, further down one step,divided into two vectors are compared both front and rear vector. Therefore, the comparison target are $\overrightarrow{P_{a0}P_{a0.5}}$ and $\overrightarrow{P_{b0}P_{b0.5}}$, $\overrightarrow{P_{a0.5}P_{a1.0}}$ and $\overrightarrow{P_{b0.5}P_{b1.0}}$. If $k = 3$, by applying the same method, it performs a comparison of eight times (2^k).

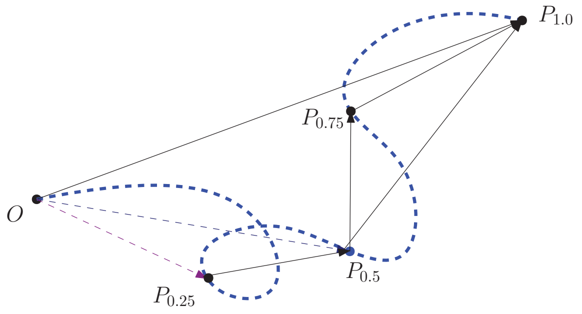

Figure 7: A geometric random walk plot(blue dotted line) and corresponding vectors.

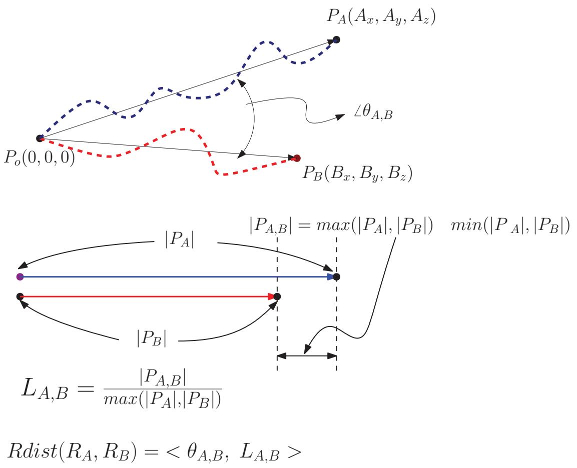
$P_A(A_x, A_y, A_z)$
$\angle\theta_{A,B}$
$P_o(0,0,0)$
$P_B(B_x, B_y, B_z)$
$|P_{A,B}| = max(|P_A|, |P_B|)$ $min(|P_A|, |P_B|)$
$|P_A|$
$|P_B|$
$L_{A,B} = \frac{|P_{A,B}|}{max(|P_A|,|P_B|)}$
$Rdist(R_A, R_B) = < \theta_{A,B},\ L_{A,B} >$

Figure 8: Two comparison parameters $\{\theta_{AB}, L_{AB}\}$.

3.4 Computing Similarity and Search on Walk Plot

Rdist refers to the similarity distance between the two vectors. Figure 8 shows that two parameters of $\theta_{A,B}, L_{A,B}$ for *Rdist*. $\theta_{A,B}$ refers to the angle between the two vectors, and $L_{A,B}$ refers to the ratio between the length of two vectors differ and from those of the longer vector. If the two vectors have the same orientation, $\theta_{A,B} = 0$, two vectors, if the length is equal to $L_{A,B} =$ is $0(0 \leq \theta_{A,B} \leq 180, 0 \leq L_{A,B} \leq 1)$.

To compare and visualize the walk plot in a limited space, compression is necessary, as described earlier formula 2. However, in the case of the reference sequence, to calculate the overall similarity of the two vectors, it maintains the two normalized values set. One is a normalized value that is used to process the query sequence, and the other is a normalized value of the calculated original reference sequence. When comparing the sequence to search when the use of normalized values of the query, and visualization uses the original normalized value. This is because it can not be an accurate comparison due to the size difference between the reference and the query, the normalized values differ.

After the normalization of the reference sequence and query sequence the normalized according to the normalization value of the query sequence, extend the depth to a predetermined level k to proceed comparison by dividing a walk plot as unit size. Compare all the pieces of the vector unit size extracted from the two walk plot by *Rdist*(). When processing the results meet the pre-determined reference range, the higher the degree of similarity($\theta_{A,B} \leq \theta_s$ and $L_{A,B} \leq L_s$). The ratio between the number of the unit vectors that meet the conditions and the total number of vector is similarity between two sequences.

4 EXPERIMENTS

4.1 Data Set

Actual biological sequence data were used for the searching experiment, and artificial data were used to validate the similarity

Algorithm 1 Comparison Algorithm

```
initialize $beg \leftarrow 0$
initialize $end \leftarrow$ len($R_a$)
initialize $O \leftarrow \{0, 0, 0\}$
initialize $D \leftarrow threshold\ lenth\ of\ vector$
procedure Compare($beg$, $end$ : index of vector list, $R_a$, $R_b$ : random walk plot of $G_a$, $G_b$, threshold $\theta_s$, $L_s$)
    $mid \leftarrow (end - beg)/2 + beg$
    $cnt \leftarrow 0$
    if $end - beg > D$ then
        $cnt+ = Compare(beg, mid, R_a, R_b)$
        $cnt+ = Compare(mid + 1, end, R_a, R_b)$
    else
        $V_a \leftarrow R_a[end] - R_a[beg]$
        $V_b \leftarrow R_b[end] - R_b[beg]$
        $Len_a \leftarrow$ euclideanDist($O, V_a$)
        $Len_b \leftarrow$ euclideanDist($O, V_b$)
        $\theta_{a,b} \leftarrow \mathbf{acos}(\frac{\mathbf{dotProduct}(V_a, V_b)}{Len_a \times Len_b}) \times 180$
        $Ł_{a,b} \leftarrow \frac{\mathbf{abs}(Len_a - Len_b)}{\mathbf{max}(Len_a, Len_b)}$
        if $\theta_{a,b} \leq \theta_s$ and $Ł_{a,b} \leq L_s$ then
            return 1
        else
            return 0
        end if
    end if
    return $cnt$
end procedure
```

comparison model. The biological sequences are human chromosome 1(246MB size) and the sequence of a 1M-10M size extracted from chromosome 1. Artificial sequence data are obtained by extracting a sequence of 1-10 MB length from the human chromosome 1 sequence at a random location and inserting noise in a predetermined ratio. A number of bases with different sizes are deleted, inserted, and replaced by a ratio of 1% to 50%. The artificial data information such as ratio and the b.p. size and number of pixels and compression ratio is shown in the Table 2, 3. 'A1-0'is meant that the artificial data of 1M size and 0% modified, namely it is just extracted from human sequence, not modified. But 'A10-25'is meant that the artificial data of 10M size and 25% modified. 'M.Rate' refers to the modified ratio of the number of B.P. on origin sequence. 'Ratio' refers to the compression ratio of the number of B.P. and pixels of the actual sequence to be converted to a walk plot. For example, in the Table 2, since A1-1 sequence has 1000.02K bases, and walk plot size consists of 36K pixel, the compression ratio is 3.58%. 'Sim' is meant that the similarity result of origin sequence and modified sequence and 'Comp.t' represents the comparison time.

4.2 Experiment:Similarity Comparison

Table 2 and Figure 11 show the result of similarity analysis of origin extracted sequence and modified sequences. In Table 2, 'Sim' is meant that the similarity result of origin sequence and modified sequence and 'Comp.t' represents the comparison time. As the modification ratio increases, the degree of similarity decreases. Thus, it can be confirmed that the similarity comparison model proposed in this study accurately reflects the similarity of the sequences. In addition, except for sequence generation, the time required for comparison is 0.02 seconds, which means that it can be processed at a very high speed.

Table 2: Specification of artificial data of 1M, 2M size extracted from Human chromosome 1 and comparison result

Sq N.	M.Rate (%)	Length (K B.P.)	Walk (K px)	Ratio (%)	Sim. (%)	Comp.t (s)
A1-0	0	1000.02	36.00	3.58	100.00	0
A1-1	1	999.93	35.79	3.58	99.59	0
A1-2	2	1000.01	36.17	3.62	99.45	0
A1-5	5	999.89	36.67	3.67	98.23	0
A1-8	8	999.97	37.74	3.77	96.06	0
A1-10	10	1000.49	38.05	3.80	91.73	0
A1-15	15	999.78	40.74	4.07	93.58	0.016
A1-20	20	1000.29	42.49	4.25	91.76	0
A1-25	25	999.92	44.2	4.42	86.14	0
A1-30	30	999.79	47.18	4.72	84.23	0.015
A1-40	40	1001.12	50.86	5.08	69.86	0.015
A1-50	50	999.47	58.36	5.84	63.53	0.016
A2-0	0	2000.04	67.09	3.35	100.00	0
A2-1	1	1999.96	66.89	3.34	98.03	0
A2-2	2	2000.15	67.27	3.36	95.85	0
A2-5	5	2000.26	68.99	3.45	94.65	0
A2-8	8	2000.2	70.4	3.52	90.5	0
A2-10	10	2000.14	69.64	3.48	91.2	0.016
A2-15	15	1999.94	70.84	3.54	85.71	0
A2-20	20	2000.18	77.56	3.88	83.62	0
A2-25	25	2000.66	79.97	4.00	72	0
A2-30	30	1999.85	89.15	4.46	73.37	0
A2-40	40	2001.5	88.54	4.42	63.34	0.016
A2-50	50	2000.62	104.11	5.20	54.91	0.016

4.3 Experiment:Sequence Search over whole genome sequence

Table 4 is the result of sequence searching process for extracted original sequence from human chromosome 1 and the modified sequences. 'Unit B. P. ' is the size of B.P. as a unit of search,' Unit Vector' refers to the size of the vector to consider when comparing a time. 'Error Dist.' is the distance between the actual sequence position and the result of search position. 'Find.t' shows the amount of time spent on search. The original sequence(0% modified sequence) search, as well as about the modified sequence of up to 20% are also searched in a short time. The difference between the actual position and the search result is relatively accurate, as the query size is less than 200 B.P. when the query size is 1M, and only about 2000 B.P. when the query is 10M. Figure 12,13 are the visualization result of search for the query sequence of 1MB, 10MB in the chromosome 1 of the human. Red random walk plot is a visualization of human chromosome 1, and blue point is the location where the query was searched. Through the visualization results, we can see that a query of 1MB size was found at a relatively early stage of the reference sequence, and a query of 10MB size was at the end of the sequence.

Table 3: Specification of artificial data of 4M, 10M size

Sq	M.Rate	Length	Walk	Ratio	Sim.	Comp.t
N.	(%)	(K B.P.)	(K px)	(%)	(%)	(s)
A4-0	0	4000.09	42.62	1.07	100.00	0
A4-1	1	4000.18	42.69	1.07	99.3	0
A4-2	2	3999.71	42.15	1.05	98.93	0
A4-5	5	3999.51	44.13	1.10	98.18	0
A4-8	8	3999.36	44.08	1.10	96.03	0
A4-10	10	4000.1	45.95	1.15	96.27	0
A4-15	15	3999.75	45.69	1.14	94.63	0
A4-20	20	4000.23	49.33	1.23	91.33	0
A4-25	25	3999.7	49.78	1.24	90.93	0
A4-30	30	4001.21	53.79	1.34	84.36	0.016
A4-40	40	3999.59	57.16	1.43	76.82	0.015
A4-50	50	4000.14	64.1	1.60	66.87	0
A10-0	0	10000.05	65.26	0.65	100.00	0
A10-1	1	10000.03	65	0.65	98.08	0
A10-2	2	10000.13	64.81	0.65	97.29	0
A10-5	5	9999.47	66.32	0.66	96.76	0.015
A10-8	8	9999.74	68.75	0.69	95.12	0
A10-10	10	10000.71	67.93	0.68	94.9	0.015
A10-15	15	9999.97	75.13	0.75	91.18	0
A10-20	20	9998.82	74.38	0.74	90.24	0
A10-25	25	9999.4	78.34	0.78	87.68	0.016
A10-30	30	9999.24	82.29	0.82	82.49	0
A10-40	40	9999.82	87.51	0.88	78.48	0
A10-50	50	10001.48	94.45	0.94	66.47	0

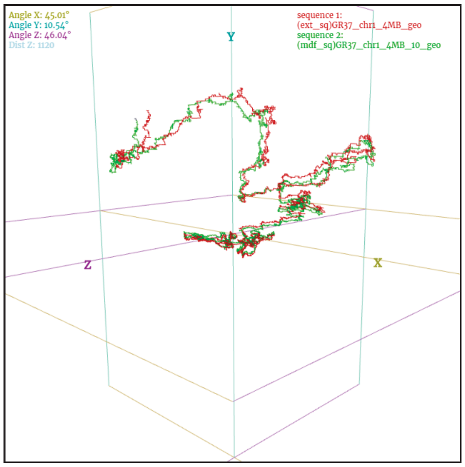

Figure 9: Red random plot represents one part of human chromosome 1, the length of which is 4 MB, in terms of nucleotide bases. Green random plot represents the 10% distorted sequence of the red one, human chromosome 1.

This is consistent with the position in the actual sequence, and represents a search result in a more intuitive.

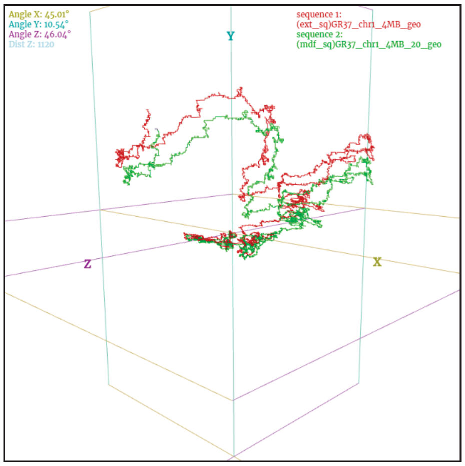

Figure 10: Red random plot represents one part of human chromosome 1, the length of which is 4 MB, in terms of nucleotide bases. Green random plot represents the 30% distorted sequence of the red one, human chromosome 1.

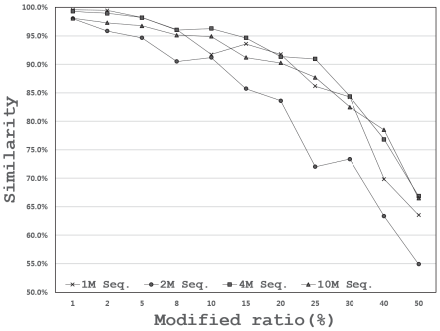

Figure 11: Similarity between origin sequence and modified sequences in each size 1-10MB.

5 CONCLUSION

Most genome sequence analyses proceed through comparative analysis by finding similar sequence data. Therefore, there is a need for a technique to quickly compare and search for large amounts of sequence data. The alignment technique is a very accurate method to compare sequences, but its high time and space complexity is inadequate to handle large sequences. To overcome these disadvantage, we suggest a new method for comparison and finding for Mega size sequence. Converts the genome sequence as a random walk on the three-dimensional, followed by replacing the sequence comparison problem with geometric object comparison problem. As a result of experiments, similarity precessed by our comparison model accurately reflects the modified ratio between the modified sequence and the original sequence. Most analytical studies based on visualization derive only a single result because they derive a numerical value based on the final result of the visualization. The search and comparison method based on the sequence visualization proposed in this study has high value of utilization of information

Table 4: The result of sequence search for origin sequence and modified sequence in Human chromosome 1

Q sq.	Unit sz. (B.P)	Unit Vec.sz (px)	error Dist.	Sim. (%)	Find.t (s)
A1-0	28	11200	0	99.29	17.269
A1-5	27	10800	150	97.27	21.341
A1-10	26	10400	840	91.34	23.213
A1-20	23	9200	120	88.75	22.514
A4-0	92	36800	1160	92.81	6.537
A4-5	90	36000	160	98.41	6.896
A4-10	88	35200	1040	92.68	7.678
A4-20	80	32000	1040	86.3	9.132
A10-0	154	61600	1120	93.88	13.665
A10-5	150	60000	560	97.21	16.065
A10-10	148	59200	280	95.09	14.245
A10-20	134	53600	2020	81.95	22.241

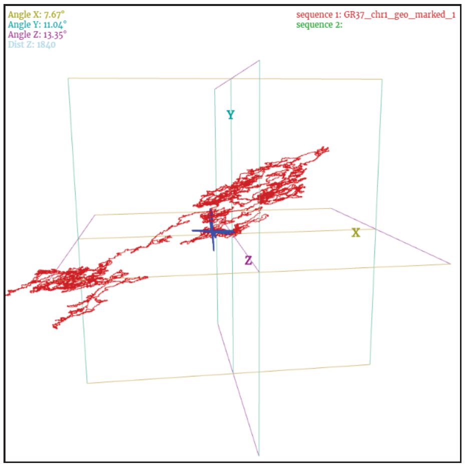

Figure 12: The visualization of search result for query sequence of 1M size in human chromosome 1. Red random walk plot represents human chromosome 1, and blue point represents the location where the query was searched.

because all compressed partial visualization information is used for searching sequence. It is useful in that the partial similarity of the sequence can be measured. In addition, a query sequence of size 1-10M was searched in a whole genome sequence of 200M or more, and a relatively precise position was found for the original sequence as well as the modified sequence up to 20%. Also the search time 25 seconds or less, was confirmed handled in a very improved speed compared to the alignment algorithm.

ACKNOWLEDGMENT

[redacted]

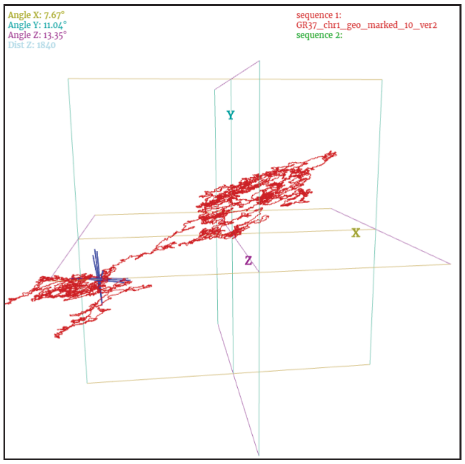

Figure 13: The visualization of search result for query sequence of 10M size in human chromosome 1.

REFERENCES

[1] Stephen F Altschul, Warren Gish, Webb Miller, Eugene W Myers, and David J Lipman. 1990. Basic local alignment search tool. *Journal of molecular biology* 215, 3 (1990), 403–410.

[2] Jiyuan An, John Lai, Atul Sajjanhar, Jyotsna Batra, Chenwei Wang, and Colleen C Nelson. 2015. J-Circos: an interactive Circos plotter. *Bioinformatics* 31, 9 (2015), 1463–1465.

[3] Lee Da-Young, Kim Kyung-Rim, Kim Taeyong, and Cho Hwan-Gue. 2016. Comparison-specialized visualization model for whole genome sequences. *Journal of WSCG* 24, 2 (2016), 43–52.

[4] Hwan-gue Cho Dayoung Lee, Daegeon Kwon. 2016. Web-GL based Visualization System for Whole Genomes. In *Proceedings of KIISE*. KOREA INFORMATION SCIENCE SOCIETY, 1414–1416.

[5] Min-Ah Kim, Eun-Jeong Lee, Hwan-Gue Cho, and Kie-Jung Park. 1997. A visualization technique for DNA walk plot using k-convex hull. *Journal of WSCG* 5, 1-3 (1997), 212–221.

[6] Martin Krzywinski, Jacqueline Schein, Inanc Birol, Joseph Connors, Randy Gascoyne, Doug Horsman, Steven J Jones, and Marco A Marra. 2009. Circos: an information aesthetic for comparative genomics. *Genome research* 19, 9 (2009), 1639–1645.

[7] Daegeon Kwon. 2015. *Whole Genome Data Visualization and Analysis Using 3D Random Walk Plot.* Master’s thesis. Pusan National University.

[8] Yongfan Li, Guohua Huang, Bo Liao, and Zanbo Liu. 2009. H-L curve: a novel 2D graphical representation of protein sequences. *MATCH-COMMUNICATIONS IN MATHEMATICAL AND IN COMPUTER CHEMISTRY* 61, 2 (2009), 519–532.

[9] Bo Liao and Kequan Ding. 2006. A 3D graphical representation of DNA sequences and its application. *Theoretical Computer Science* 358, 1 (2006), 56–64.

[10] Tetsuya Nakamura, Keishi Taki, Hiroki Nomiya, Kazuhiro Seki, and Kuniaki Uehara. 2013. A shape-based similarity measure for time series data with ensemble learning. *Pattern Analysis and Applications* 16, 4 (2013), 535–548.

[11] Alexey Pasechnik, Aleksandr Mylläri, Tapio Salakoski, A Mylläri, T Salakoski, and T Salakoski. 2005. Dynamical visualization of the DNA sequence and its nucleotide content. *Proceedings of KRBIO* 5 (2005), 47–50.

[12] Milan Randić. 2004. Graphical representations of DNA as 2-D map. *Chemical Physics Letters* 386, 4 (2004), 468–471.

[13] Milan Randić, Nella Lerš, Dejan Plavšić, Subhash C Basak, and Alexandru T Balaban. 2005. Four-color map representation of DNA or RNA sequences and their numerical characterization. *Chemical physics letters* 407, 1 (2005), 205–208.

[14] Milan Randić, Marjan Vračko, Jure Zupan, and Marjana Novič. 2003. Compact 2-D graphical representation of DNA. *Chemical physics letters* 373, 5 (2003), 558–562.

[15] Milan Randić, Jure Zupan, Dražen Vikić-Topić, and Dejan Plavšić. 2006. A novel unexpected use of a graphical representation of DNA: Graphical alignment of DNA sequences. *Chemical Physics Letters* 431, 4 (2006), 375–379.

[16] Yonghui Wu, Alan Wee-Chung Liew, Hong Yan, and Mengsu Yang. 2003. DB-Curve: a novel 2D method of DNA sequence visualization and representation. *Chemical Physics Letters* 367, 1 (2003), 170–176.