

Distance-Profile Chart: a Novel Visual Representation of Mutual Location of 3D Objects

Magdalena A. Tkacz

Institute of Computer Science
 University of Silesia in Katowice
 ul. Będzińska 39, 41-200 Sosnowiec, Poland
 Email: magdalena.tkacz@us.edu.pl

Abstract—This document presents a novel method for visual representation of mutual objects location (relative to each other) in 3D. The motivation and inspiration for such a work come from chromosome territory (CT) adjacency analysis. This paper describes: the idea of the cone of sight (CoS), with an explanation of the origin of such approach; the way a mathematical model of CoS was build and a process of a space segmentation with CoSes. Next, the way how distance-profile chart (DPC) is designed and created was described and finally, DPCs on the exemplary dataset was presented. Finally, some conclusions were formulated.

I. INTRODUCTION

EVERY DAY we assess where different objects are situated around us, and in which distance they are. In fact, this is a basis that allows us to live and move – by perpetual environment observation, we can notice any change in surrounding world. Then, in dependency on the situation, we can take a proper, adequate to a situation action: either we can come closer (e.g. to take something) or make a step back to avoid injury. For orientation in an environment (navigation) – determination of direction and distance in which any object is situated humans for ages used rhumb (marine navigation) [1], the wind rose or – so do now we use the clock-related method. It is the easiest way to instruct someone saying "It is on your four, in 2 meters distance" or during outdoor activity – "It is to the south-east of you, in 15 meters distance". This works, but only in 2D. We do not have either a 3D wind rose or 3D clock to determine direction when we want to orient in a 3D space.

A. Motivation and background

When we started to work on chromosome territory (CT) [2] modeling algorithm and application for model visualization [3] we did not assume that it will result in further methodological and algorithmic problems (others that concerns only model itself). But, when finishing Chromosome Territory Modeler (ChroTeMo) and Chromosome Territory Viewer (ChroTeVi) [3] subsequent question appears: OK, so we have a probabilistic 3D model of chromosome territory for certain species. Such a model can be visually analyzed and compared by the human to the photos from FISH microscopy using chromosome paintings method, but ...how can we automate and

objectify adjacency analysis of certain chromosomes, pairs of chromosomes or chromosome arms? Although at first sight it seems to be practically impossible, or at least difficult – an author decided to try to cope with that problem, and results of the first attempt are described and shown in this paper.

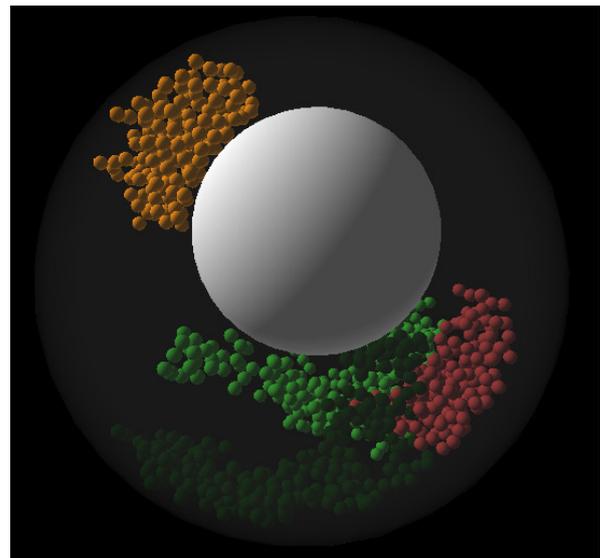


Fig. 1. Visualization of CT's with the use of ChroTeVi Model based on human species data.

This paper describes attempts of developing a method that allows automatically, in a reproducible and objective way get an answer to the question: which CT's (visualized as groups of spheres (see Fig. 1) are nearest to a given chromosome territory. How distant a surrounding objects are? Is a territory occupied by the cloud of a green spheres nearest to territory occupied by the red or rather by the orange ones (in 3D of course)? How near or how far they are? How they are situated in relation to each other? The first idea was to use some projection methods – but in that way, some information has been lost. There were also an attempt to make use of the Hausdorff distance to calculate distance among CT's. The

Hausdorff distance (HD) was computed according Eq. 1 ([5], p.150) (for the two sets, A and B , where $x \in A, y \in B$):

$$Hdist(A, B) = \max(\limsup(dist(x, B)), \limsup(dist(y, A))) \quad (1)$$

From my initial experiments, it looks that this method gives acceptable results with objects (approximated by the cloud of 3D points) with less complicated shape than CT's territories, so results of using HD were not satisfactory. I also take a look at octree [4] (quadtree in 3D): this approach gives detailed and complete 3D space segmentation, but does not give intuitively imaginable description of direction where other objects are located in relation to a given object. Moreover, method itself is not easy to follow by non-tech/IT person – CT analysis are to be developed for life scientists as a target group. This led to the necessity of rethinking this problem in a different way.

B. Shaft of Light and The Cone of Sight idea

The idea of cone of sight was inspired by the shaft of light (a lighthouse or a torch). When you are in the dark and turn on the torch, pointing the light in a certain direction (for example to the left in relation to you), then you can see objects in the shaft of light. Then you can assess if at all, and how distant different objects are to the left of you. Directing the light in different directions (left, right, up, down, front, rear) allows you to explore entire space around you. This able you to make a kind of a "mental image" – the "map" of objects and their location in your surrounding.

To use this idea, developing a mathematical model of a shaft of light was necessary. The shaft of light has a shape that is similar to a cone (see Fig. 2): consists of a single point at the beginning and staying wider when far away.

The cone, as a solid block known from stereometry (solid geometry), cannot be described with the equation. The desirable solution for computational purposes would be the solution that allows writing conical-like shape in a form of equation. The solution came from analytical geometry: the conical surface [6] is one of the quadratic surfaces and can be described with a mathematical equation. Thus, the interior of space, limited by and located inside a conical surface could be considered as an appropriate model of the interior of the shaft of the light, suitable for computational purposes.

II. MATHEMATICAL MODEL OF THE CONE OF SIGHT

Conical surface is known as a family of lines, crossing at a certain point and tangent to the sphere (see Fig. 2). The equation of conical surface can be derived as a set of lines that are tangent to a certain shape (base – often circular) and crossing one point (the cone vertex). For the purpose of this paper, the base is circular, i.e., it is a circle made by a cross-section of the sphere by the plane that passes through a diameter.

So, when there is a need to cover 3D space around fixed object(point) with a certain set of cones, it is enough to have a set of spheres basing on which a certain cone equation could be derived. The problem of covering space with spheres is

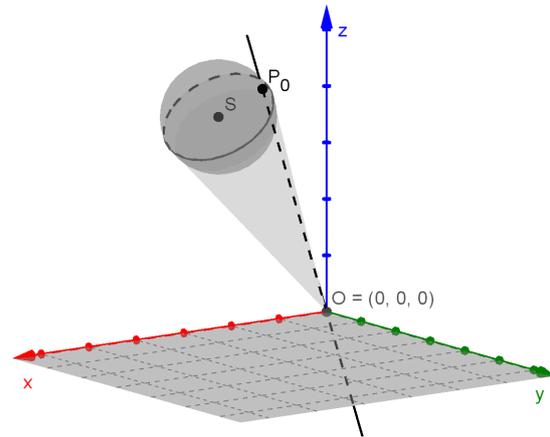


Fig. 2. **Conical surface construction.** Conical surface as a surface created by lines, crossing by a fixed point and tangent to the given sphere

known as "sphere packing" ([7], [8]) and will not be discussed here further.

To uniquely define a sphere in 3D you have to know the coordinates of the center of the sphere and the radius of this sphere. The coordinates of the additional point, that will become vertex should also be known. This point is also assumed to be the center of the given object, in relation to which we want to determine a relative position of other objects. For the purpose of this work, it is assumed that this point is a geometrical center of a given object – in general. In the case of CTs, the centromere is regarded as center of chromosome. To simplify calculation it is also assumed that the center of object is the vertex of conical surface and is the center of the coordinate system $O(0, 0, 0)$. The equation of any sphere, with the center in (a, b, c) and radius R is shown on Eq. 2:

$$(x - a)^2 + (y - b)^2 + (z - c)^2 = R^2 \quad (2)$$

For a given sphere (or set of spheres) with a center in a point (a, b, c) and a radius R , it is possible to derive the equation of a desirable conical surface (with a vertex in a given point).

A parametric equation of a line in 3D space, crossing a point $P_0(x_0, y_0, z_0)$ has the following form:

$$\begin{cases} x = x_0t \\ y = y_0t \\ z = z_0t \end{cases} \quad (3)$$

where t is a parameter.

If point P_0 belongs to the line that is also tangent to the sphere, then both equations those of the sphere and the line have to be fulfilled. So:

$$(x_0 - a)^2 + (y_0 - b)^2 + (z_0 - c)^2 = R^2 \quad (4)$$

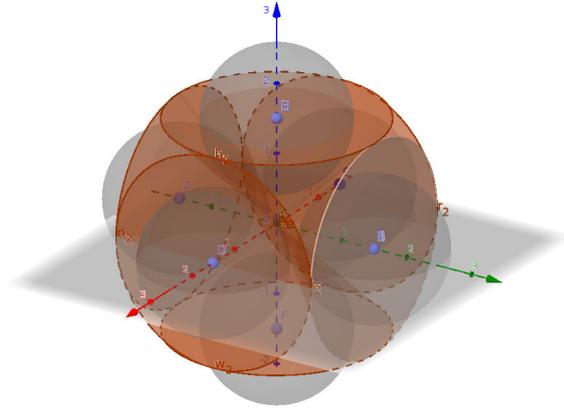


Fig. 3. **Cones of sight** that are located alongside axes.

Substituting equation 3 to the equation of the sphere 4, after making some transformation (according [9], p. 227) we have a conical surface equation derived (Eq. 5):

$$(-2ax - 2by - 2cz)^2 - 4(x^2 + y^2 + z^2)(a^2 + b^2 + c^2 - R^2) = 0 \quad (5)$$

With the inequality (Eq. 6) it is possible to determine which certain points in 3D space are outside or inside the conical surface. A certain point with coordinates (x, y, z) will be inside conical surface if and only if its coordinates will conform (satisfy) Eq. 6.

$$(-2ax - 2by - 2cz)^2 - 4(x^2 + y^2 + z^2)(a^2 + b^2 + c^2 - R^2) \geq 0 \quad (6)$$

This is all we need to make 3d space segmentation, and after segmentation we can check which objects are in which space segment represented by a (CoS).

This approach was used by the author to create a set of cones, that allows to segment 3D space into precisely defined sectors (representing directions). Those sectors will be further referred to as **Cone of Sight (CoS)**, and the process of dividing 3D space into cones will be referred to as **conification**. Conification itself does not ideally cover the space – there are "gaps" between conical surfaces. Additional procedure that allows to assign yet unassigned points to the certain CoS is described later in this paper altogether with discussion of coverage during conification. In this paper the most intuitive (and thus relatively easy to understand) conification was used: CoSes were placed alongside axes in 3D Cartesian coordinate system (see Fig.3), with top of all cones placed at the beginning of this system, in the point $O(0, 0, 0)$.

Then, there will be six CoSes (see also Fig. 3 and 5):

- OX+ (relative direction: right),

- OX- (relative direction: left),
- OY+ (relative direction: front),
- OY- (relative direction: rear),
- OZ+ (relative direction: up),
- OZ- (relative direction: down).

Each CoS represents direction in which someone can look at. To be able to uniquely determine to which CoS a certain point belongs the only thing is to substitute a given point coordinates to the equation of the conical surface (Eq. 6) and check whether it fulfills the inequality.

```
if CoS_eq(x_p, y_p, z_p) <= 0:
  p inside CoS
else:
  p outside CoS
```

The number of required comparison, at these stage of segmentation, equals (*number of points times number of CoSes*).

After this procedure, some of the points may still reside outside any CoS. As you can see on Fig. 3 CoSs itself does not fulfill space entirely, there is a little space in between them. Anyway, we have to have all points (of all objects) being allocated to CoS. To solve this problem, the topological measure (known as the distance from the point x to the set A ([5], p.140)) was used (Eq. 7):

$$dist(x, A) = \liminf |x - a|, a \in A \quad (7)$$

Points that remain unallocated after conification process are then allocated to the nearest CoS according mentioned measure (Eq. 7).

It was interesting how many points are conified and how many have to be assigned using additional, mentioned above measure. To check coverage of conification, a set of testing sets (with different number of points) were prepared. They were a spherical 3D sets generated with the normal or unified distribution. Percentage of points assigned in conification step and in the second (as distance between point and the set) step is shown in Tab. I.

It is easily to notice that during conification 76-81 % of points are assigned to CoSes, so 19-24 % have to be assigned using measure defined by the Eq. 7 to reach full conification. It can be also seen that most of the points were assigned during the first step. The assignment made during the second step assure us, that no point remains unallocated.

So, having uniquely segmented 3D space (or – thinking in categories of the shaft of light – everything visible *inside* a shaft) – we can go into checking what is inside this certain shaft (CoS) (what can be "seen" inside). After full conification (conification and assignment of remaining points to the nearest CoS) each point P_i is characterized by the set of parameters $P_i(x_i, y_i, z_i, obj_{id}, CoS_{id})$. Therefore we can subset all points grouping them by CoS_{id} .

Having sets of points located in a certain CoS's (segment of space) we are able to state which objects can be seen in a predefined part of a space (CoS) around given object. When a set of points is limited only to points being inside CoS,

TABLE I

NUMBER OF CONIFIED POINTS (SPHERICAL SET WITH UNIFORM AND NORMAL DISTRIBUTION WITH DIFFERENT NUMBERS OF POINTS). THE LOWEST IS IN *italics*, THE HIGHEST IN **BOLD**

distribution	no of points	conified	remaining
normal	1000	77%	23%
	2000	76%	24%
	5000	77%	23%
	7500	76%	24%
	10000	76%	24%
unified	1000	81%	19%
	2000	81%	19%
	5000	81%	19%
	7500	80%	20%
	10000	80%	20%

then it is easily to compute the less and the most distant points of a given object inside certain CoS. It is also possible to state (e.g. in percent), which part of points representing object are inside. Therefore, all the data necessary to make "a mental map" (mentioned in section I-B) of any object (with its surrounding) is present. Those (often numerous) data, although contains all information necessary to make adjacency analysis can be stored in a tabular form, but this is not easy to review format by humans. Humans have visual perception stronger than any other sense, so decision to present those relationships in visual form was taken. However, in some applications (e.g. computer vision) storing and processing this data in a form of a table or array can be more useful.

III. CONSTRUCTING DISTANCE-PROFILE CHARTS (DPCs)

In this section the idea and details of distance-profile chart will be presented. After conification process, the following information concerning every given object is present: fraction of other objects that are located in the surrounding examined objects CoSes and the distances of the nearest and the outermost point for every other object. Basing on these information, a kind of "mental map" (that allows us to imagine what object, to which extent, and in which ranges of distance it spreads) can be prepared. This is a situation similar to those, when in a darkness a torch is used: the one can turn and look around to determine the location and distance of objects that are visible in torch's shaft of light (as mentioned previously).

The assumption for the chart was that it should show all the information necessary to create "a mental map" of all objects surrounding a given one. That means, it should reflect the range of distances and the fraction of all objects located inside each CoS (direction), because this information is necessary to asses mutual location of objects in 3D space. Such set of charts were named the **Distance-Profile Chart, (DPC)**. Such a name was given because this visualization shows the "profile of distances" of all other objects in relation to a given object.

Each DPC consists of a given number of sub-charts. Each sub-chart presents distance range, and a fraction of objects located in a certain CoS (so the number of sub-charts equals

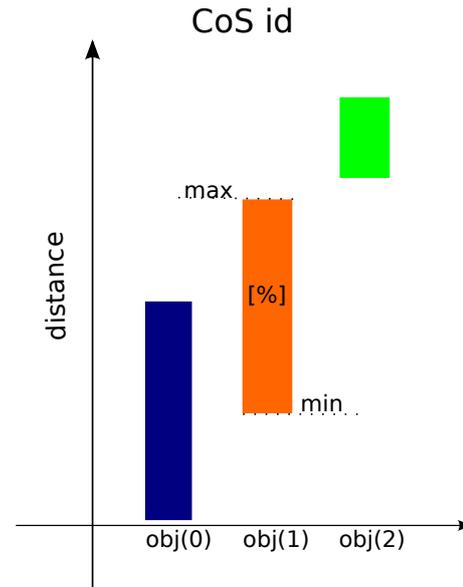


Fig. 4. Distance-profile chart construction schema.

to the number of CoS that divides the space, for every given object). The idea of creation each sub-chart and parameters is shown on Fig. 4.

It is easy to notice, that it is somehow inspired by a boxplot charts. The sub-charts are placed on a grid to present in a concise way all information about what is inside every CoS in relation to a given object. Each grid field represent one CoS, so this chart is "CoS-centered". The grid used to show arrangement of subcharts is presented in the Fig. 5.

The color of the bars corresponds to the scale that represents the fraction (percentage) of an object inside a given CoSs. The scale used in this paper is shown in Fig. 6.

A. DPC of the exemplary dataset

To show how CoS idea and DPC work in practice a special testing set (consists of three easy to show and imagine objects) was prepared. This set consists of three elliptical objects that are positioned in such a way that looks like the head of a mouse with ears (see Fig. 7).

This set was fully conified, and the DPCs were generated. Because this set of objects consists of three objects – also three DPCs were created (one for every object: head, left ear and right ear). All three DPCs are shown on Fig. 8–10.

Reading of DPCs needs some training. On the presented Mouse 3D exemplary dataset we can see three object: "the head", and two "ears". "Ears" are located above "head". Moreover, "ears" are located on opposite sideways. So, placing center of coordinate system in the center of the "head" we can see, that (refer to Mouse 3D image on Fig. 7):

- "head", as a center of coordinate system lie on every ax (in every direction),

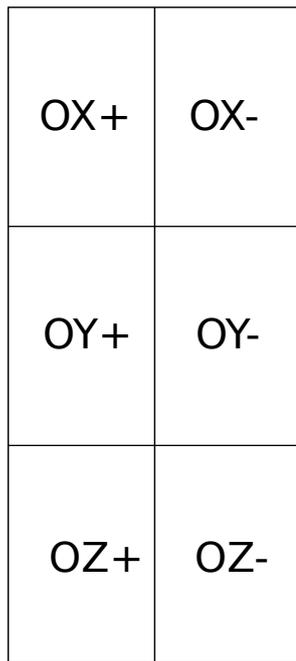


Fig. 5. Grid with fields for DPC subcharts. OX+ means that this is a CoS alongside OX+ axis, OY+ alongside OY axis and so on.

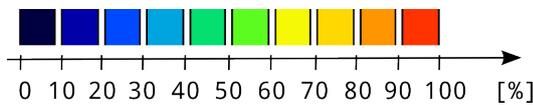


Fig. 6. Scale – percentage of object included in a certain CoS.

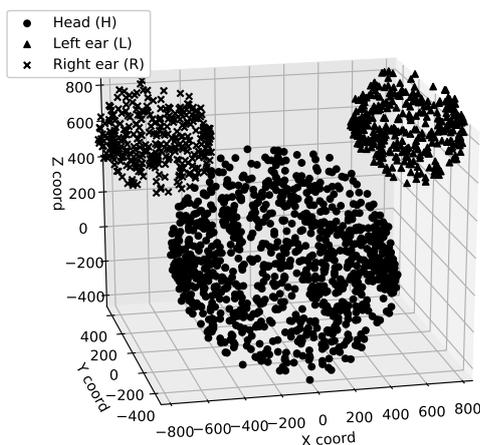


Fig. 7. Mouse 3D exemplary set.

- both ears lie in a "range" of OZ+ axis,
- one of the ear lies in a range of OX+ axis,
- the second ear lies in a range of OX- axis.

Let's take a look on the DPC "for" the "head" object (Fig. 8): it is the only object that has the smallest distance from the center of coordinate system (starting nearby zero) AND is in every CoS. It is depicted on a horizontal axis as second object. What does it mean? The second object is around center of coordinate system, alongside OY and OZ axes the most distant point is in the same distance, but alongside OX axis distance is greater than alongside OY and OZ. That means that the object being a point of reference (object in relation to which we assess mutual position of other objects) is longer alongside OX. The dark blue color on fields OY+, OY-, OZ+, OZ- means, that there is 10-20% of the object. The light blue on fields that represents OX+ and OX- means, that there is a 20-30% of the object. Putting together information about extent of the bar concerning the set and information about fraction of object coded as color, we can infer, that head of mouse is not symmetrical and is longer alongside OX axis.

Looking at length of bars, we can ascertain, that the part of the second object located in every CoS is more or less the same in size than entire objects one and three. On field that represent OX+, OX- and OZ+ we can see additional bars. They are related to the other objects. First object on horizontal axis of the DPC corresponds to the right ear, third object corresponds to the left ear

On OX+ DPC field we can see an object that is farther than the head. The same object appears also in OZ+. This means that this object spans between OX+ and OZ+. We can see that it is located through diagonal to the upper right. Reading fraction from colors, we can say that this object is more in OZ+ (50-60%) than in OX+, so is shifted more up than to the right. This is how the right ear is situated (this can be verified looking at Fig. 7)

Then, we can read information about third object. It appears on DPC in the field OX- and OZ+. Analyzing in the same way as in a case of the first object, we can infer that this object is located in the left upper "corner". Opposite to the first object, it is more "shifted" towards OX- (what can be read from color – contains more points).

The next DPC to analyze is presented on Fig. 9. On this figure we can see the bars in all CoS for object one – so this is our object of reference. We can also see one additional bar (object) on field OZ-, and two additional bars on OX- field. This means, that, in relation to this object only one object is located below. But the same object is also located in OX- field in proportion 50-60% in OZ- and 40-50% in OX- field. This means that bigger part of object two is located below. Looking at length of bars, we can also ascertain that the second object is bigger (in size) than object one and two (spans more widely). Looking at third object presented as red bar in OX- field of CoS we can see that it is about twice in size in comparison to the object one, begins somewhere in a middle of the object two, but ends further – the distance to its extreme point is

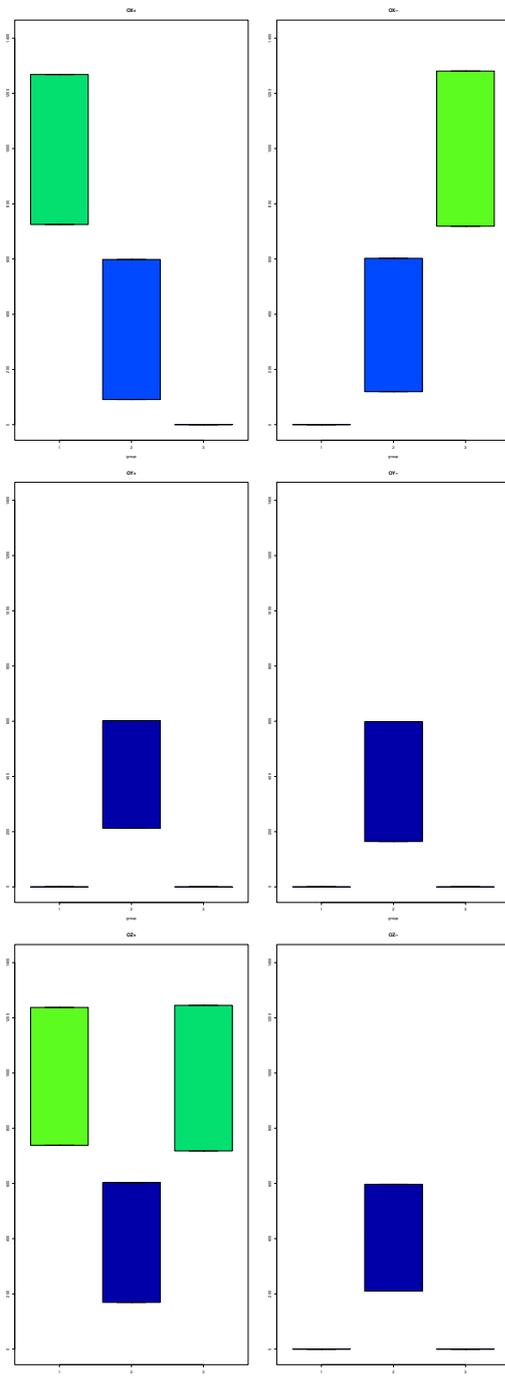


Fig. 8. Distance-Profile Chart for the Mouse 3D testing set. CoSes defined in relation to 3D Mouse's "head".

greater than in a case of object two. The red color means that this third object is situated in OX- almost in full.

This description is adequate as description of a surrounding of the object being the left mouse "ear": the third object (right "ear") is entirely located to the right of the left "ear", on the bottom-right direction the "head" is located. Moreover, basing on bar length, we know that the "head" is bigger than the "ear".

The last *DPC* for the Mouse 3D set is presented on Fig. 10. This *DPC* is symmetric to those presented on Fig. 9 and can be read in a similar way as presented above.

As you can see, (comparing 3D set visualization (Fig. 7) and *DPC*s (Fig. 8 – 10), after a little training (to accustom to such charts and learn how to read them) you can recreate the most important features of objects positioning, their distance and distribution in relation to each other basing only on *DPC*s.

On Fig. 11 an exemplary *DPC* for the CT model is presented. This is *DPC* generated only for one object, as the number of pages in this paper is limited. This chart can be read in a similar way that charts for 3D Mouse exemplary set.

Data for this *DPC* charts was taken from *Brachypodium Distachyon* chromosome territory model, generated with ChroTeMo [3]. This certain *DPC* shows a mutual location of CTs of the other chromosome arms for the first arm of the fifth chromosome as an object of reference.

IV. CONCLUSION AND FUTURE WORK

In this paper, a novel methodology for assessing mutual 3D objects location in a 3D space was presented with the use of the cone of sight idea. For the ease of perception for humans the novel method of visual presentation of 3D object adjacency – the distance-profile chart *DPC* was developed.

The advantage of such approach is, that there is no need to have a special tool for visualization of 3D objects (which is computationally demanding and requires relatively powerful hardware). Also, the one examining the problem of the proximity of the objects in 3D does not have to spend a lot of time manually zooming in, zooming out and rotating scene to determine which objects are closer to the given object of interest, and how the objects are situated in relation to each other. The researcher can print *DPC*, share tis in such form and also analyze offline. Additionally, *DPC* approach allows to asses the mutual proximity of 3D objects by providing the exact and objective measure of proximity and thus made mutual proximity assessment of 3D objects reproducible, comparable and objective.

The future work is:

- refactor the code and provide it as a library for wide use,
- check different arrangement of spheres (basing on which conical surface equation is derived) in the context of space coverage, to maximize coverage during conification
- made benchmarks of the performance of this approach covering computational time, memory load in a context

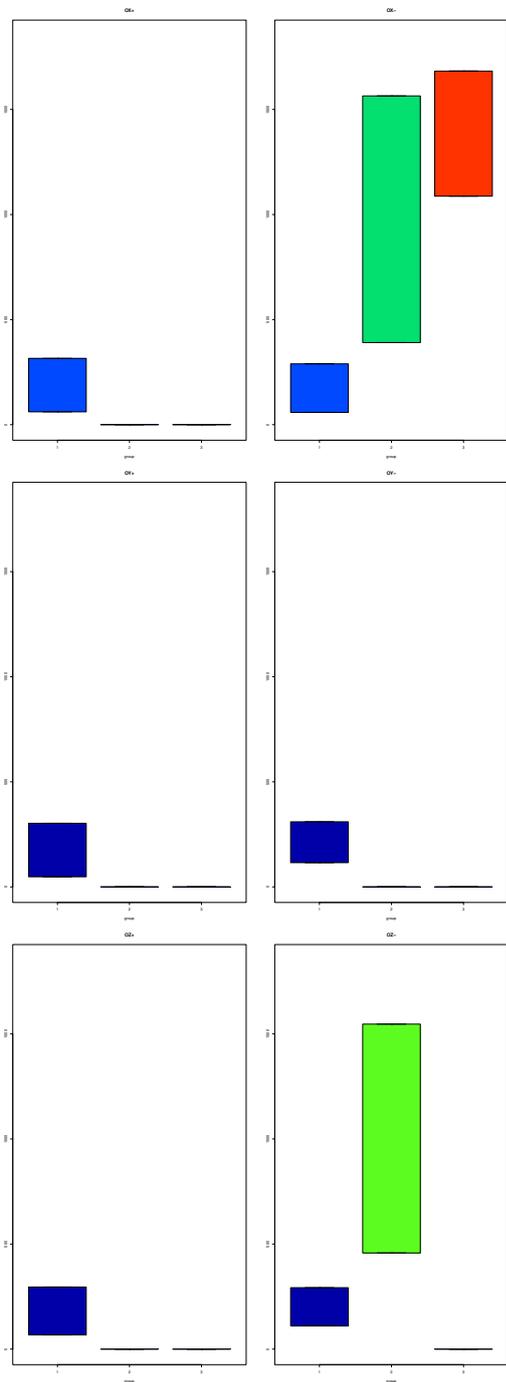


Fig. 9. Distance-profile chart for the Mouse 3D testing set. CoSes defined in relation to mouse's left ear.

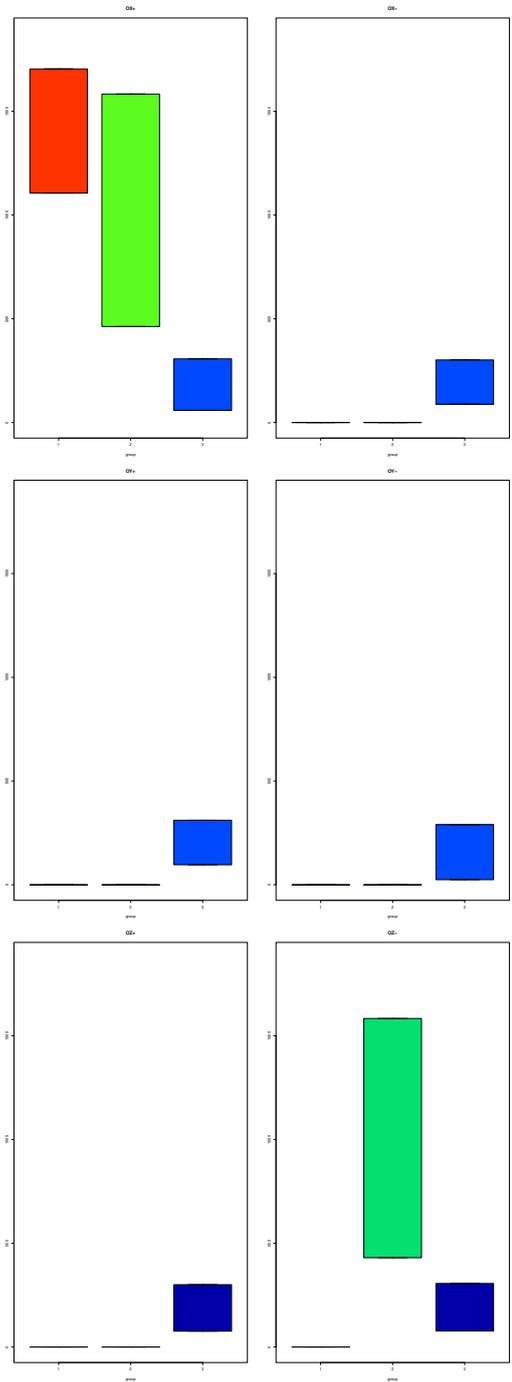


Fig. 10. Distance-profile chart for the Mouse 3D testing set. CoSes defined in relation to mouse's right ear.

of the type of sphere arrangement, the quantity of conified dataset and number of objects in a dataset.

REFERENCES

- [1] Units in maritime navigation <http://www.siranah.de/html/sail020e.htm>
- [2] Cremer T, Cremer C. Chromosome territories, nuclear architecture and gene regulation in mammalian cells. *Nat Rev Genet.* 2001;2(4):292-301. Epub 2001/04/03. pmid:11283701.
- [3] Magdalena Tkacz, Kornel Chromiński, Dominika Idziak-Helmcke, Ewa Robaszekiewicz, Robert Hasterok: Chromosome Territory Modeller and Viewer. *PLoS ONE* 11(8): e0160303. <https://doi.org/10.1371/journal.pone.0160303>
- [4] Mathworks: octree – partitioning 3D points into spatial subvolumes. <https://www.mathworks.com/matlabcentral/fileexchange/40732-octree-partitioning-3d-points-into-spatial-subvolumes>
- [5] Kuratowski Kazimierz: Wstęp do teorii mnogości i topologii, PWN, Warszawa 1980 (Introduction to the Set Theory and Topology (in Polish))
- [6] Conical surface. *Encyclopedia of Mathematics.* http://www.encyclopediaofmath.org/index.php?title=Conical_surface&oldid=31530
- [7] Sphere packing – Wolfram Alpha: <http://mathworld.wolfram.com/SpherePacking.html>
- [8] Conway, J. H. and Sloane, N. J. A. *Sphere Packings, Lattices, and Groups*, 2nd ed. New York: Springer-Verlag, 1993.
- [9] Stark Marcełi: Geometria analityczna z wstępem do geometrii wielowymiarowej, PWN, Warszawa 1974 (Analytical Geometry with introduction to multidimensional geometry (in Polish))

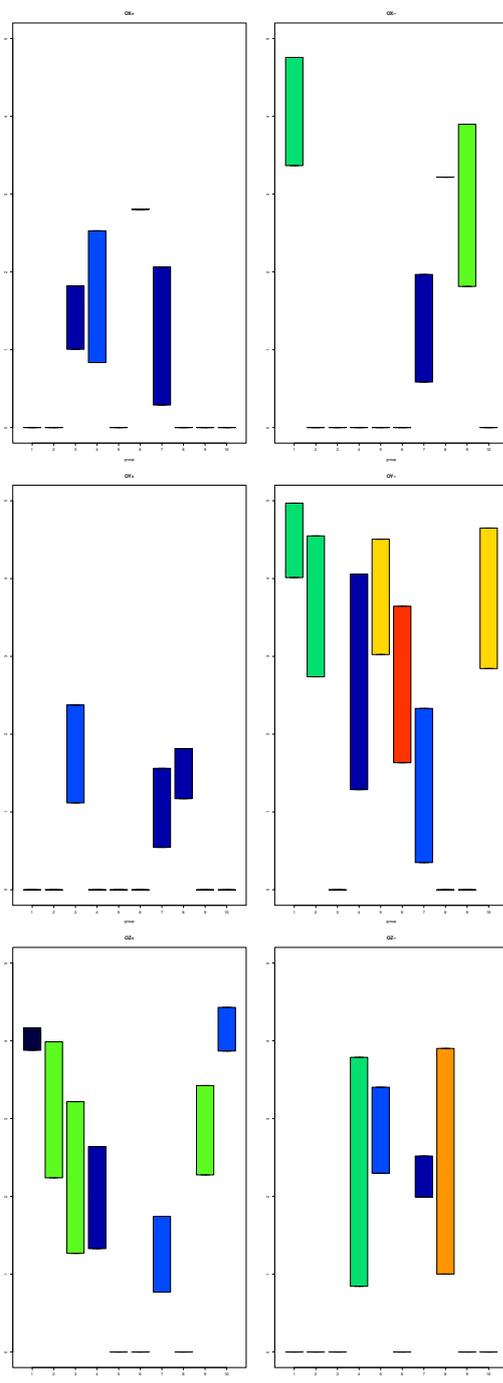


Fig. 11. Distance-profile chart for exemplary CTs model. Charts in relation to the 1st arm of the 5th chromosome in *Brachypodium Distachyon*.