On Degeneracies Removable by Perspective Projections

F. Gómez * F. Hurtado † J.A. Sellarès ‡ G. Toussaint §

Abstract

In this paper we are concerned with computing non-degenerate *perspective* projections of sets of points and line segments in three-dimensional space. For sets of points we give algorithms for computing perspective projections such that (1) all points in the projection have distinct x-coordinates, (2) all points in the projection have both distinct x- and y-coordinates, (3) no three points in the projection are collinear, and (4) no four points in the projection are cocircular. For sets of line segments we present an algorithm for computing a perspective projection such that no two segments in the projection are parallel. All our algorithms have time and space complexities bounded by low degree polynomials. We also discuss the problem of removing some intrinsic degeneracies for point and line segment sets in the plane by using perspective projections.

Keywords: Degeneracies, general position, perspective projections, robustness of algorithms, visualization, computer graphics, computer vision, computational geometry.

1 Introduction

Algorithms in computational geometry are usually designed for the real RAM (random access machine) assuming that the data input is in *general position* in a large variety of senses that often depend on the nature of the problem and the algorithm designed to solve the problem. These assumptions are made in order to simplify the algorithm design process and often to obtain more efficient algorithms that do not need to check special degenerate cases. Yap [17] has distinguished between intrinsic or *problem-induced* degeneracies (such as three collinear points and four cocircular points) and extrinsic or *algorithm-induced* degeneracies (such as two points in the plane with the same x-coordinate). Due to the practical importance of having algorithms work correctly for degenerate input, there has recently been a flurry of activity on this problem in the computational geometry literature. There are at least two methods of coping with intrinsic degeneracies: approximation and perturbation [2, 7, 8, 9, 16, 17]. Gómez et al. [12] have studied several ways of removing algorithm-induced

*Departamento de Matemática Aplicada, U.P.M., Madrid, España (fmartin@eui.upm.es)
†Departament de Matemàtica Aplicada II, U.P.C, Barcelona, España (hurtado@ma2.upc.es)
‡Institut d'Informàtica i Aplicacions, Universitat de Girona, Girona, España (sellares@ima.udg.es)
§School of Computer Science, McGill University, Montreal, Quebec, Canada (godfried@cs.mcgill.ca)

degeneracies. Their method consists of performing a global rigid transformation on the input data that removes the algorithm-induced degeneracies. Once the solution to a problem has been obtained, they perform the inverse transformation on the result. Their methods make crucial use of *orthogonal* projections of the input data. Here we look at the problem of removing intrinsic degeneracies for point and line sets in the plane by using *perspective* projections as the global transformation. Not all intrinsic degeneracies can be removed with perspective projections. Intrinsic degeneracies that can be removed via perspective projection are called *quasi-intrinsic degeneracies.* For example, given a set of points on the xy-plane which has four or more cocircular points, we are interested in finding a center of projection and a plane such that the projection of that set of points does not contain four cocircular points. However, we are not interested in projecting on planes which are far away from the xy-plane since this could radically change the positions of the points. We must therefore look for planes that are close to the xy-plane. We consider the more general problem of computing a non-cocircular perspective projection of an arbitrary set of points in space. For other degeneracy assumptions we also solve the problem of finding a non-degenerate (non-collinear, non-parallel, etc) perspective projection in space.

In the scientific world we are frequently concerned with visualizing, describing and analyzing data consisting of three-dimensional (3-D) sets of points or line segments. We often have at our disposal only a 2-D medium, such as a computer-graphics screen, on which to display a necessarily incomplete representation of the data we are interested in. Therefore it is desirable to obtain 2-D representations of our data that make it easy for us to discover the structure in the real data in 3-D. Our methods for computing non-degenerate projections are also applicable to this family of problems that falls in the areas of scientific visualization [11, 14], computer graphics [3, 13] and computer vision [4, 15].

This paper is structured as follows. In Section 2 we give algorithms for computing perspective projections of a set of points such that all points in the projection have distinct x-coordinates; have both distinct x- and y-coordinates. In Section 3 we give an algorithm for computing a perspective projections of a set of points such that no three points in the projection are collinear. In Section 4 we give an algorithm for computing a perspective projections of a set of points such that no four points in the projection are cocircular. In section 5 we present an algorithm for computing a perspective projection of a set of line segments such that no two segments in the projection are parallel. In section 6 we discuss the problem of removing some quasi-intrinsic degeneracies for point and line segment sets in the plane by using perspective projections. Finally, in Section 7 we present conclusions and future work.

2 Projections with distinct x-coordinates

Motivated in part by the problem of visualizing a point set projected onto a plane, Gómez, Ramaswami and Toussaint [12] solve the problem of finding the *orthogonal* projection of a given point set that has the property that the projected points have distinct x and y coordinates. In this section we are concerned with the *perspective* version of that problem. Throughout this section we assume the plane of projection to be the xy-plane (Figure 1).

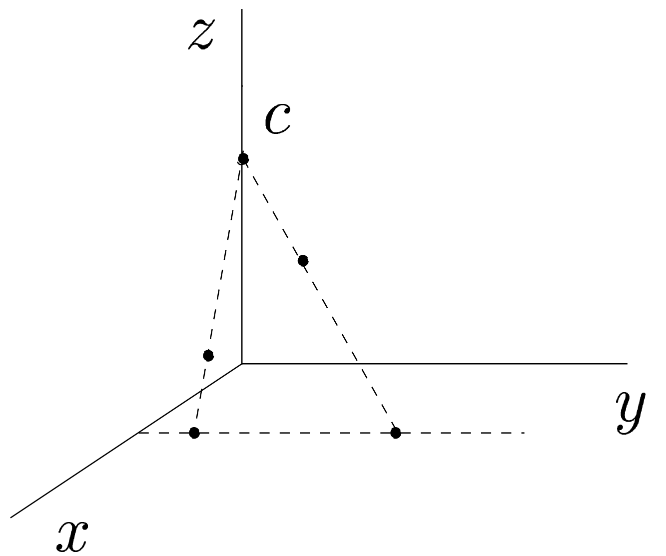

Figure 1: *Projected points with the same x-coordinate*

For the sake of simplicity of description we now introduce some terminology: let $P = \{p_1, \cdots, p_n\}$ be a set of n distinct points in space and c be a center of projection. We denote by $P^* = \{p_1^*, \cdots, p_n^*\}$ the perspective projection of P from c. When necessary, we will underline the dependency of P^* on c by writing $P^*(c)$ or $p_i^*(c)$.

If there are two points in P that determine a line parallel to the y-axis, then the perspective projection of P will contain points with the same x-coordinate. Let us find how to detect this situation first.

Lemma 2.1 *Given a set of n distinct points, whether there are two points that determine a line parallel to the y-axis can be decided in $O(n \log n)$ time and linear space.*

Proof. It suffices to project P orthogonally onto the xz-plane and check the resulting projected points for duplicates. Using a lexicographic sort of the projected points followed by a scan of the sorted list, we can determine whether there exist any such duplicates in $O(n \log n)$ time. □

Theorem 2.1 *Given a set of n distinct points such that no two of them determine a line parallel to the y-axis, deciding whether a given projection has distinct x-coordinates can be done in $\Theta(n \log n)$ time and $\Theta(n)$ space in the algebraic decision tree model.*

Proof. It is sufficient to check for duplicate x-coordinates by sorting lexicographically the projected points on the xy-plane. This takes $O(n \log n)$ time.

The *element-uniqueness* problem is to determine, given an input set of real numbers, whether any two of them are equal. This decision problem was shown to have complexity $\Omega(n \log n)$, in the algebraic decision tree model, by Dobkin and Lipton [5]. Given a set of real numbers $A = \{a_1, \cdots, a_n\}$, let us consider the set of distinct points $P = \{(a_i, i, 1),\ i = 1, \cdots, n\}$. The perspective projection of center $(0, 0, 2)$ onto the xy-plane maps the set P onto the set $P^* = \{(2a_i, 2i),\ i = 1, \cdots, n\}$. Points of P^* have distinct x-coordinates if and only if there are not two identical numbers in A. Thus we have a linear time reduction from the element-uniqueness problem to the distinct projected x-coordinates decision problem. This proves that the latter is also $\Omega(n \log n)$.

□

2.1 Existence of projections with distinct x-coordinates

We now look at the existence problem, namely: does a projection with distinct x-coordinates always exist? A point is said to be a forbidden point if it produces a projection with non-distinct x-coordinates when projecting from it. A plane is called a forbidden plane if it contains a pair of points of P and intersects the xy-plane at a line parallel to the y-axis. Furthermore, every pair of points in P yields a forbidden plane and there are no others. Since all forbidden points are contained in $O(n^2)$ planes, and these planes have measure zero in space, we conclude that there always exists a perspective projection whose x-coordinates are all distinct.

2.2 Computation of projections with distinct x-coordinates

Let us now turn to the computation problem: given a set of n points, to find a center whose projection has distinct x-coordinates. As we saw before, the forbidden directions have measure zero and hence centers of projection are allowed in abundance. In particular, the following theorem shows how to find a valid projection center on the z-axis.

Theorem 2.2 *Given a set of n distinct points such that no two of them determine a line parallel to the y-axis, a projection with distinct x-coordinates can be computed in $O(n \log n)$ time and linear space.*

Proof 1. If necessary, we first change the coordinate system so that P lies in the first octant (that is, all points have positive coordinates). Next we project P orthogonally onto the xy-plane, and taking that projection as a planar set, compute the perspective projection from the origin onto some line. If such projection is not regular, then we will find a new center on the negative part of the x-axis which does produce a regular projection. (Recall that in a regular projection all projected points are distinct; see [1]). An algorithm whose running time is $O(n \log n)$ can be used in order to compute that new center. We note that after all these operations the z-axis cannot be a forbidden line. That is because no point of P belongs to the yz-plane and there is no plane containing the z-axis and more than one point of P.

The following step finds a valid projection center on the positive z-axis. Actually, the algorithm not only computes a valid center, but also obtains an open line segment of valid centers.

If we used a brute force algorithm, then we should take into account $\Omega(n^2)$ forbidden planes to be intersected with the z-axis and that would lead us to a quadratic complexity. However, we show that it is not necessary to examine all planes to obtain the desired center of projection.

Let z_0 be a positive number such that $z_0 > \max\{z(p_i) \mid i = 1, \cdots, n\}$, where $z(p_i)$ is the z-coordinate of p_i, and $c(z_0) = (0, 0, z_0)$ is a point on the z-axis. Let us consider $P^*(z)$ the perspective projection of P from $c(z) = (0, 0, z), z \geq z_0$ onto the xy-plane. If $P^*(z_0)$ turns out to have all its x-coordinates distinct, then the process is completed. Let us assume it is not so. In that case, we know that in $P^*(z_0)$ there are at least two points with the same x-coordinate. When varying the parameter $z, z \geq z_0$ in a continuous way, the points of $P^*(z)$ move in a continuous way too. Let $c(z_1) = (0, 0, z_1)$ be the nearest forbidden point to $c(z_0)$. In the set $P^*(z_1)$, there must exist two points, $p^*_{i_j}(z_1), p^*_{i_{j+1}}(z_1)$, whose x-coordinates when sorted are equal. Since the order of the projected points only changes at a forbidden point when $c(z)$ varies, it follows that points $p^*_{i_j}(z_1)$ and $p^*_{i_{j+1}}(z_1)$ were already consecutive at $c(z_0)$. Therefore, it suffices to consider only those planes passing through two consecutive points in the ordering by x-coordinate at z_0. For each one of those planes, we compute the intersection with the z-axis and select the nearest point $c(z_1)$ to $c(z_0)$. Any point of the open line segment $\overline{c(z_0)c(z_1)}$ is an allowed point. The algorithm runs in $O(n \log n)$ time and uses $O(n)$ space.

□

Next we describe an alternate second approach for computing perspective projections with distinct x-coordinates also in $O(n \log n)$ time and linear space. Although this algorithm is more complicated up to a constant factor, unlike the previous algorithm, it yields an unbounded interval of valid projection centers located on the z-axis. This may prove useful for stability purposes in some applications in computer vision [4, 15].

Proof 2. This second algorithm is based on the observation that when we project a point set P from a projection center c on the z-axis onto the xy-plane, two points p_i and p_j of P project to points with the same x-coordinate if and only if, in the orthogonal projection P' of P onto the xz-plane, we have that the projected points p'_i and p'_j are collinear with the projection center c. So, to compute a perspective projection of the point set P with distinct x-coordinates, it suffices to determine a point in the z-axis that is non collinear with any two points of P'. This is only possible when the points of P' are distinct.

The algorithm uses several results from [12]. In particular the algorithms there for computing regular orthogonal projections of planar points onto a line, and of 3-dimensional points onto a plane.

As with the first algorithm we start by changing the coordinate system, if necessary, so that P lies strictly in the first octant of the coordinate space. Then we rotate P so that the *orthogonal* projection P' of P onto the xz-plane is a regular projection, i.e., the points of P' are distinct. This can be done in $O(n \log n)$ time with the algorithm in [12]. Let P'' be the projection of P' onto the x-axis. Note that P'' may have duplicates. Now perform a lexicographic sorting of P'' to first delete duplicates and then compute the smallest (non-zero) gap between an adjacent pair in the points of P'' that are not discarded. Call this gap G_{x-min}. This step is also $O(n \log n)$. Next compute the difference in z-coordinates between the highest and lowest (maximum and minimum z-coordinates) points in P'. Call this difference H_{z-max}. Let S_{max} equal the ratio of H_{z-max} over G_{x-min}. Note that S_{max} is an upper bound on the maximum slope (nearest to vertical) determined by any pair of points in P'. This follows from the fact that P' is a regular orthogonal projection of P.

Finally, let p_0 denote the point on the xz-plane whose x and z coordinates are equal to the maximum x and z coordinates, respectively, of the points in P'. We construct a line with slope equal to the negative of S_{max} such that it passes through p_0 and we compute the intersection point c_0 that this line makes with the z-axis. By this construction any point on the z-axis above c_0 cannot be collinear with two points of P'. Therefore the perspective projection of P from any such center above c_0 will not have two points with the same x-coordinate.

□

2.3 Perspective projections with distinct x and y coordinates

It is quite natural to ask for projections that not only have the x-coordinates distinct, but also the y-coordinates. This problem can be easily solved using the ideas put forward above.

To determine whether a given projection has distinct x- and y-coordinates it is enough to sort the points lexicographically and check for duplicates. The forbidden planes are those that pass through a pair of points of P and are parallel to either the x-axis or the y-axis. As a consequence, we can conclude that projections with distinct x- and y-coordinates are always possible to find.

Finding a projection with distinct x-coordinates has already been solved. However, such a center may not produce a projection with distinct y-coordinates and we need to do some extra work.

By using the algorithm presented in proof 1 of Theorem 2.2, we obtain an open line segment of valid centers. Let $c(z_2)$ be an interior point of that segment. If $c(z_2)$ is not a forbidden center, then we have found our center. If it is we adapt the algorithm presented in proof 1 of Theorem 2.2 so that it finds the point $c(z_3)$ closest to $c(z_2)$ such that it is a forbidden center for the y-coordinates. If we let $z_4 = \min\{z_1, z_3\}$, then any point of the open line segment $\overline{c(z_2)c(z_4)}$ is a valid projection center.

We can also use the algorithm presented in proof 2 of Theorem 2.2 to obtain an *unbounded* segment of projection centers. Initially we rotate P so that it yields regular orthogonal projections on both the xz-plane and the yz-plane. Then we proceed to compute c_0 based on the xz-plane. Finally, we compute a candidate c_1 based on the yz-plane and select the maximum of the two candidates for our z-value. Then any point on the z-axis above this z-value will serve as the center of projection.

3 Non-collinear perspective projections

In this section we turn our attention to computing non-collinear projections of a set of points in space, that is, a center of projection such that there are no three collinear points on the projection. Note that the condition of being collinear only depends on the position of the center with respect to the points in space, and hence, it is independent of the plane of projection. Because of this, we fix the xy-plane as the plane of projection throughout this section (Figure 2).

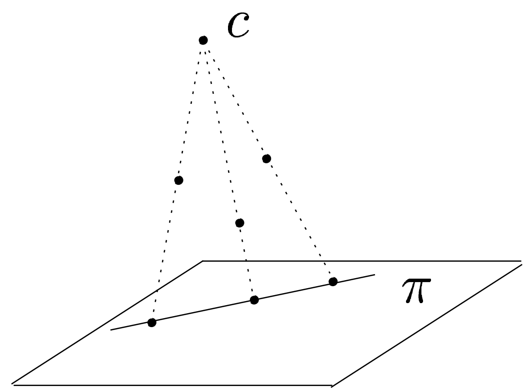

Figure 2: *Collinear projection*

If three or more points in space are collinear then they will also be collinear in any projection. In [12] Gómez et al. describe an algorithm that, in $O(n^2)$ time and space, decides if a set of points in space contains three or more collinear points. We will take their algorithm as a preliminary step for the decision and computation problems to be solved below.

3.1 Existence of non-collinear perspective projections

Theorem 3.1 *Given a set of n distinct points, deciding whether a given center produces a non-collinear projection can be done in $O(n^2)$ time and space.*

Proof. We first apply the algorithm in [12] to the point set in space before carrying on. If there are collinear points, then we conclude that the desired projection cannot exist. Otherwise, we project P onto the xy-plane. The projection is then transformed into a dual set of lines via the standard map $(a, b) \mapsto y = ax + b$. It is well known that there are three collinear points in the primal space if and only if there are vertices of degree six or more in the arrangement of lines in the dual space. Constructing such an arrangement and checking for those vertices takes $O(n^2)$ time and space (see [6]).

□

The decision problem admits a reduction from the 3-collinear-problem. We recall the latter problem belongs to the so-called 3-SUM-hard class (see [10]).

Theorem 3.2 *Given a set of n distinct points, to decide whether a given center produces a non-collinear projection is 3-SUM-hard.*

Proof. Let $P^* = \{p_1^*, \cdots, p_n^*\}$ be a set of points in the xy-plane and denote by $p_i^* = (x_i, y_i)$, $i = 1, \cdots, n$ its coordinates. Given a number $d > n$, for $i = 1, \cdots, n$ consider the point $p_i = \left(\frac{d-i}{d}x_i, \frac{d-i}{d}y_i, i\right)$ and the set $P = \{p_1, \cdots, p_n\}$ formed by those points. The perspective projection of the center $(0, 0, d)$ onto the xy-plane maps set P onto set P^*. Therefore, there exist three collinear points in P^* if, and only if, there exist three collinear points in P. □

A point in space is said to be a forbidden point (for non-collinear projections) if the projection of P from that point contains three or more collinear points. A plane determined by three points in P is then called a forbidden plane. Since all forbidden points belong to $O(n^3)$ forbidden planes, and planes have measure zero in space, non-collinear perspective projections always exist.

3.2 Computation of non-collinear perspective projections

Theorem 3.3 *Given a set of n non-collinear points in space, computing a non-collinear projection can be done in $O(n^2)$ time and space.*

Proof. Let $P = \{p_1 = (x_1, y_1, z_1), \cdots, p_n = (x_n, y_n, z_n)\}$ be the set of non-collinear points. We begin by computing a perspective projection onto the xy-plane so that all x-coordinates are distinct. To do so, we use the first algorithm described in the previous section. We recall that this algorithm returns as output an open line segment of valid centers of projection. Furthermore, the projected points have positive x- and y-coordinates and $z_0 > \max\{z_i \mid i = 1, \cdots, n\} > 0$. As a first candidate we pick $c(z_1) = (0, 0, z_1)$; if $c(z_1)$ is not a forbidden center, then we are finished. If it is not the case, we keep looking for a valid center.

Let $c(z) = (0, 0, z)$ be a point on the z-axis with $z > z_1$. When point $p_i = (x_i, y_i, z_i)$ is projected from $c(z)$ onto the xy-plane, we obtain the point $\left(\frac{z}{z-z_i}x_i, \frac{z}{z-z_i}y_i\right)$. Note that since $z > z_i$ and $\frac{z}{z-z_i} > 1, \forall i = 1, \cdots, n$, the term $\frac{z}{z-z_i}$ tends to 1 as z goes to infinity and, therefore, $p_i^*(z)$ tends to (x_i, y_i).

Consider the projection $P^*(z)$ of P from $c(z) = (0, 0, z), z \geq z_1$. For the set $P^*(z_1)$, in particular, we construct its dual arrangement $\mathcal{A}(P^*(z_1))$ with the mapping $(a, b) \mapsto y = ax+b$. Since we have guaranteed that $P^*(z_1)$ does not have repeated x-coordinates, the arrangement does not contain parallel lines. For each $z > z_1$, we look at the dual arrangement $\mathcal{A}(P^*(z))$. As z increases from z_1, for $i = 1, \cdots, n$, the lines $r_i : y = \left(\frac{z}{z-z_i}x_i\right)x + \frac{z}{z-z_i}y_i$ of the arrangement $\mathcal{A}(P^*(z))$ tends to the lines $r : y = x_i x + y_i$. All lines pass through a fixed point $f_i = (-\frac{y_i}{x_i}, 0)$ and since $x_i > 0$ and $\frac{z}{z-z_i} > 1$, the slope $\frac{z}{z-z_i}x_i$ is kept positive, while decreasing continuously until reaching x_i (Figure 3).

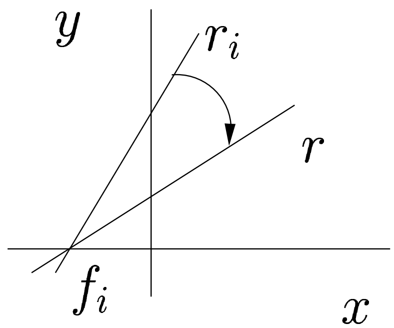

Figure 3: *Line transformation*

During this process, for $z > z_1$ the arrangement $\mathcal{A}(P^*(z_1))$ transforms into the arrangement $\mathcal{A}(P^*(z))$. The first intersection of three lines, by continuity, will take place among three lines that form a triangular cell in $\mathcal{A}(P^*(z_1))$. Thus, it suffices to compute the $O(n^2)$ planes determined by the triplets of points which give triangular cells in the dual space. Intersecting those planes with the z-axis produces a set of forbidden points; by selecting the closest one to $c(z_1)$, we obtain the desired open line segment of valid centers.

It only remains to establish the complexity. A projection with all x-coordinates can be computed in $O(n \log n)$ time. To determine whether a projection is collinear takes quadratic time by virtue of the results proved above. Constructing the arrangement of lines also takes quadratic time as pointed out before. Finally, computing the forbidden planes given by the triangular cells and intersecting them with the z-axis can be done in quadratic time also.

□

4 Non-cocircular perspective projections

A perspective projection of a point set in space is said to be non-cocircular if it contains no four cocircular points. Note that the property of being cocircular simultaneously depends on the position of the center and the plane of projection. Without loss of generality, we will assume the plane of projection to be the xy-plane (Figure 4).

Theorem 4.1 *Given a perspective projection of a set of n points in space, we can decide whether it is cocircular in $O(n^3)$ time and space.*

Proof. Let $P = \{p_1, \cdots, p_n\}$ be the set of points and $P^* = \{p_1^*, \cdots, p_n^*\}$ its projection from a given center. Consider the dual of P^* given in the following way. Each point in the xy-plane is lifted to the paraboloid $z = x^2 + y^2$ and associated to its tangent plane. The following two properties of the dual map allow us to solve the decision problem: (1) three planes intersect at a point if, and only if, the corresponding points are non-collinear and, (2) four planes are concurrent if and only if the corresponding points are cocircular. By constructing the arrangement of planes, we can determine if there are four or more concurrent planes in $O(n^3)$ time and space (see [6]).

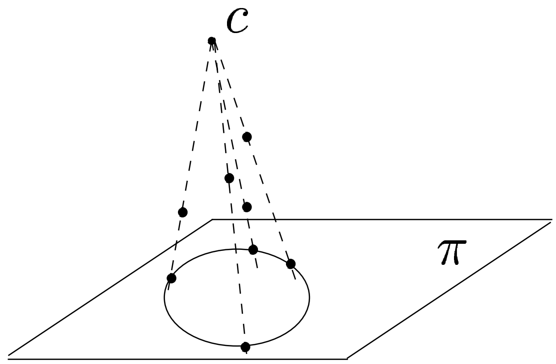

Figure 4: *Cocircular projection*

□

4.1 Existence of non-cocircular perspective projections

A center of projection is called a forbidden center if it produces a cocircular projection. As before, let us call $P^* = \{p_1^*, \cdots, p_n^*\}$ the set of projected points and, for i, j, k, l all distinct, let us denote by $R_{i,j,k,l}$ the region of forbidden centers for the four points $p_i^*, p_j^*, p_k^*, p_l^*$. If these points are cocircular, then their liftings to the paraboloid are coplanar. Four points are coplanar if, and only if, the volume of the tetrahedron formed by them is zero; this characterization leads to the following equation

$$\begin{vmatrix} x(p_i^*) & y(p_i^*) & x(p_i^*)^2 + y(p_i^*)^2 & 1 \\ x(p_j^*) & y(p_j^*) & x(p_j^*)^2 + y(p_j^*)^2 & 1 \\ x(p_k^*) & y(p_k^*) & x(p_k^*)^2 + y(p_k^*)^2 & 1 \\ x(p_l^*) & y(p_l^*) & x(p_l^*)^2 + y(p_l^*)^2 & 1 \end{vmatrix} = 0\,.$$

The region of forbidden points is given by the solution of the equation. The equation has degree 6 and the region has measure zero in space. Therefore, there always exists a non-cocircular projection.

4.2 Computation of non-cocircular perspective projections

From now on we assume the intersection of each forbidden region with the z-axis can be computed in $O(1)$ time.

Theorem 4.2 *Computing a center of a non-cocircular perspective projection can be done in $O(n^3)$ time and space.*

Proof. We begin by computing a non-collinear projection. This can be done with an algorithm put forward in the previous section. As we have already seen, that algorithm provides a line segment whose interior points are all allowed points. Furthermore, the line segment can be chosen so that the x- and y-coordinates are all positive and its endpoints have greater z-coordinates than any point in P. Note that the z-axis is not a line of forbidden points because the intersection of the forbidden regions with it consists of a finite set of points. Let $c(z_1) = (0, 0, z_1)$ be the center of a non-collinear projection. If $c(z_1)$ produces a non-cocircular projection we have finished. Otherwise, we continue looking for a valid center.

We recall from the previous section that a point $p_i = (x_i, y_i, z_i)$ is projected from center $(0, 0, z), z \geq z_1$ to point $p^*(z) = \left(\frac{z}{z - z_i}x_i, \frac{z}{z - z_i}y_i\right)$ and as z tends to infinity, $p^*(z)$ tends to point (x_i, y_i) continuously. Let $P^*(z_1)$ be the projection of P from $c(z_1)$ and let us consider the set of planes given by the dual of $P^*(z_1)$ via the map $z = x^2 + y^2$ as described above. For a projected point (x, y), the normal of its dual plane is $(2x, 2y, -1)$. Since there are no two points with the same x-coordinate, the normals are all distinct and therefore, no parallel planes exist. As in $P^*(z_1)$ there are no three collinear points, in the dual there will not be planes intersecting at a line. Let us then construct the arrangement of planes and call it $\mathcal{A}(P^*(z_1))$.

For each point $c(z') = (0, 0, z'), z' > z_1$, let $P^*(z')$ be the projection of P from $c(z')$ onto the xy-plane. Consider the dual planes corresponding to that set via the map $z = x^2 + y^2$ and let $\mathcal{A}(P^*(z'))$ be the arrangement given by those planes. As z' increases from z_1, for all $i = 1, \cdots, n$, planes

$$\pi_i : z = \frac{2z'}{z' - z(p_i)}x^*(p_i)x + \frac{2z'}{z' - z(p_i)}y^*(p_i)y - \left(\left(\frac{2z'}{z' - z(p_i)}x^*(p_i)\right)^2 + \left(\frac{2z'}{z' - z(p_i)}y^*(p_i)\right)^2\right)$$

are continuously transformed into the planes:

$$\pi : z = 2x^*(p_i)x + 2y^*(p_i)y - (x^*(p_i)^2 + y^*(p_i)^2).$$

All these planes pass through a fix line r_i in the plane $z = 0$ whose equation is

$$r_i : 2x^*(p_i)x + 2y^*(p_i)y - (x^*(p_i)^2 + y^*(p_i)^2) = 0$$

and whose slope is negative and equal to $-x^*(p_i)/y^*(p_i)$ (Figure 5).

Therefore, during that process the arrangement of planes $\mathcal{A}(P^*(z_1))$ is transformed into $\mathcal{A}(P^*(z'))$ in a continuous way. This is because the coefficients of the planes are continuous functions of z. The first intersection to be found, by continuity, must take place among four planes that form a tetrahedral cell in $\mathcal{A}(P^*(z_1))$. This implies that it is sufficient to compute the $O(n^3)$ forbidden regions identified by the vertices of tetrahedral cells. Once computed, we intersect them with the z-axis and select the closest forbidden point to $c(z_1)$, say $c(z_2) = (0, 0, z_2)$. The open line segment $\overline{c(z_1)c(z_2)}$ is the desired set of valid centers.

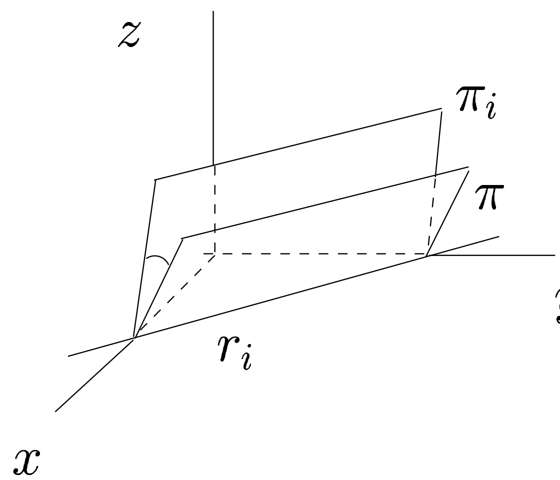

Figure 5: *Plane transformation*

It only remains to establish the complexity. A non-collinear perspective projection can be computed in quadratic time and space. To check if $c(z_1)$ is a non-cocircular projection and construct the arrangement $\mathcal{A}(P^*(z_1))$ takes $O(n^3)$ time. Finally computing the forbidden regions for the selected cells and finding $c(z_2)$ is also cubic time. Thus, the total time and space complexity is $O(n^3)$.

□

5 Non-parallel perspective projections of line segments

In this section we show how to compute, given a set of n disjoint line segments in space, a projection such that there are no parallel segments in the projection. Such projections receive the name of non-parallel projections. If there are two line segments which are parallel to each other and parallel to the xy-plane, then no center of projection will allow us to obtain a non-parallel projection. Let us find how to detect this situation first (Figure 6).

Lemma 5.1 *Given a set of n disjoint line segments, whether there exist two of them parallel to each other and parallel to the xy-plane can be determined in $O(n \log n)$ time and linear space.*

Proof. Let $S = \{s_1, \cdots, s_n\}$ be the set of line segments and $a_i = (x_{a_i}, y_{a_i}, z_{a_i}), b_i = (x_{b_i}, y_{b_i}, z_{b_i})$ the endpoints of $s_i, i = 1, \cdots, n$. Among all segments in S we select those parallel to the xy-plane, that is, those such that $z_{a_i} = z_{b_i}$. Next we detect if there are two segments parallel to each other in that subset. In order to do that, we project the line segments onto the xy-plane and translate them to the origin so that the line segments have a positive x-coordinate. Next we intersect the segments with the unit half-circle with positive x-coordinates and keep the intersection points. All that can be done by making the followings assignments

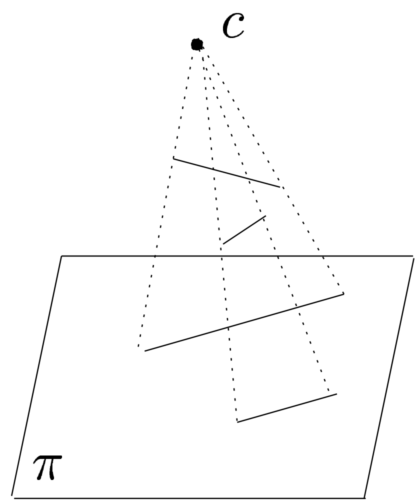

Figure 6: *Parallel projection*

1. if $x_{a_i} < x_{b_i}$, then $\alpha_i = x_{b_i} - x_{a_i}$ and $\beta_i = y_{b_i} - y_{a_i}$,
2. if $x_{a_i} = x_{b_i}$, then $\alpha_i = 0$ and $\beta_i = |y_{b_i} - y_{a_i}|$,
3. if $x_{a_i} > x_{b_i}$, then $\alpha_i = x_{a_i} - x_{b_i}$ and $\beta_i = y_{a_i} - y_{b_i}$.

Finally, we define the number γ_i as follows

$$\gamma_i = \left(\frac{\alpha_i}{\sqrt{\alpha_i^2 + \beta_i^2}}, \frac{\beta_i}{\sqrt{\alpha_i^2 + \beta_i^2}} \right), i = 1, \cdots, n.$$

Two segments $s_i, s_j, i \neq j$ are parallel if, and only if, this equality holds $\gamma_i = \gamma_j$. This condition can be checked in $O(n \log n)$ time by sorting the values γ_i's in lexicographic order on the half-circle and finding if there are duplicates.

□

From now on, we assume that the set S of disjoint line segments does not contain line segments parallel to each other and parallel to the xy-plane.

Theorem 5.1 *Given a center of projection, deciding whether the perspective projection of the set S is non-parallel can be done in $\Theta(n \log n)$ time and $\Theta(n)$ space in the algebraic decision tree model.*

Proof. In order to decide if there are parallel segments in the projection it suffices to check for duplicates by sorting the projected segments by their γ_i values as in Lemma 5.1. This takes $O(n \log n)$ time.

To show an $\Omega(n \log n)$ lower bound, we reduce the element-uniqueness problem to our decision problem in linear time. Given the set of real numbers $M = \{m_1, \cdots, m_n\}$, let us consider the set S of segments s_i , $i = 1, \cdots, n$, whose endpoints are

$$a_i = \left(i, \frac{i^2}{n}, \frac{n-i}{n}\right), \qquad b_i = \left(i, \frac{i(i+m_i)}{n+1}, \frac{n+1-i}{n+1}\right).$$

Observe that the segments of S are disjoint and no segment of S is parallel to the xy-plane. For $i = 1, \cdots, n$, the perspective projection of center $(0, 0, 1)$ onto the xy-plane maps the segment s_i of the set S onto the segment s_i^* of the set S^* whose endpoints are $a_i^* = (n, i), b_i^* = (n + 1, i + m_i)$. Since the slope of segment s_i^* is m_i, we can conclude that there are not two parallel segments in S^* if and only if there are not two identical numbers in M.

□

5.1 Existence of non-parallel projections

A center is called a forbidden center if the projection from it is parallel. The forbidden points for two segments s_i, s_j with respect to the xy-plane are those centers that project them onto parallel segments. We denote by $P_{i,j}$ such a region and present an algebraic characterization of it. Let $C = (x, y, z)$ be a center of projection; two projected segments are parallel if, and only if, the following relation holds

$$\left(\frac{x_{b_i}-x}{z-z_{b_i}} - \frac{x_{a_i}-x}{z-z_{a_i}}\right)\left(\frac{y_{b_j}-y}{z-z_{b_j}} - \frac{y_{a_j}-y}{z-z_{a_j}}\right) = \left(\frac{y_{b_i}-y}{z-z_{b_i}} - \frac{y_{a_i}-y}{z-z_{a_i}}\right)\left(\frac{x_{b_j}-x}{z-z_{b_j}} - \frac{x_{a_j}-x}{z-z_{a_j}}\right).$$

Working out the equations, we obtain the following quadric:

$$P_{i,j} : Az^2 + Bxz + Cyz + Dx + Ey + Fz + G = 0.$$

The term $A = (x_{b_i} - x_{a_i})(y_{b_j} - y_{a_j}) - (y_{b_i} - y_{a_i})(x_{b_j} - x_{a_j})$ is never zero since the segments s_i, s_j are non-parallel. The z-axis intersects each quadric at two points at most since the equation for points on the z-axis yields a degree-two polynomial. Therefore, the z-axis is not a line of forbidden centers. Since the quadrics have measure zero in space, there always exist non-parallel perspective projections.

5.2 Computation of non-parallel projections

Theorem 5.2 *Given a set of n disjoint line segments, a center of non-parallel projection can be computed in $O(n \log n)$ time and linear space.*

Proof. We will find an allowed center on the z-axis. We first pick a positive number z_0 such that $z_0 > \max\{z_{a_i}, z_{b_i}, i = 1, \cdots, n\}$ and check if the perspective projection of S from

$c(z_0) = (0,0,z_0)$ is non-parallel. If so, then process is finished. Otherwise, we consider $c(z_1) = (0,0,z_1)$, the closest forbidden point to $c(z_0)$ and show how to compute it in $O(n \log n)$ time and linear space. A straightforward approach to obtain $c(z_1)$ would be to compute all the forbidden regions $P_{i,j}$ and intersect them with the z-axis, and finally, select the closest forbidden point to $c(z_0)$. This approach takes quadratic time, and by using a continuity argument, we show how to lower the complexity to $O(n \log n)$.

Let $S^*(z)$ be the projection of S onto the xy-plane from $c(z) = (0,0,z), z \geq z_0$. Let us associate each projected segment $s_i^*(z)$ with its pair $\gamma_i(z)$ as defined above. These pairs allow us to sort the projected segments lexicographically. In particular, we know that in the set $\{\gamma_1(z_1), \cdots, \gamma_n(z_1)\}$ there must be two repeated points because $c(z_1)$ is also a forbidden point. Let $s_{i_k}^*(z_1)$ and $s_{i_{k+1}}^*(z_1)$ be the two consecutive segments such that $\gamma_{i_k}(z) = \gamma_{i_{k+1}}(z)$. From the continuity of function $\gamma_i(z)$ as z varies from z_0 to z_1, it follows that $\gamma_{i_k}(z_0)$ and $\gamma_{i_{k+1}}(z_0)$ were already consecutive in the sorting. Thus, it suffices to consider the $O(n)$ quadrics determined by consecutive segments in the sorting at z_0. Intersecting these quadrics yields a linear number of candidates among which $c(z_1)$ is found. The total complexity is $O(n \log n)$ because of the sorting.

□

6 Removal of quasi-intrinsic degeneracies

If a perspective projection is applied to a set of points or a set of lines, only the projective properties are preserved, but not the affine and metric ones. For example, properties such as incidence, intersection and collinearity are invariant under projections while cocircularity, parallelism and perpendicularity are not.

Some degeneracies in the plane considered as intrinsic such as four cocircular points, two parallel lines or two perpendicular lines, can be removed through perspective projections. Intrinsic degeneracies that can be removed via perspective projections are referred as *quasi-intrinsic degeneracies* or *degeneracies removable by perspective projections.* Traditionally, such intrinsic degeneracies are removed with perturbation methods such as those in [6]. The perspective projection methods proposed here offer a new approach to handling intrinsic degeneracies in planar problems.

We study two typical inputs, point and line sets in the plane, and show how to eliminate some of their quasi-intrinsic degeneracies. In case the input is composed of lines, we need to carry out a previous step in which vertical lines are removed (this is because many of our methods imply the use of arrangements of lines.) This step consists of a a rotation of the lines such that vertical lines disappear. The procedure to remove quasi-intrinsic degeneracies is the following (Figure 7):

1. We assume that the input I is embedded in space

2. We pick a line e in the xy-plane and an angle α and rotate the xy-plane around e. The criterion used to choose the values of e and α will be specified later. The new plane π_α

also contains a new input I_α.

3. A center $c = (0, 0, d), d > 0$ of projection on the z-axis is chosen and I_α is projected back onto the xy-plane from c, giving place to $I_\alpha(c)$, the final perturbed input. The way of choosing c will be also made concrete below.

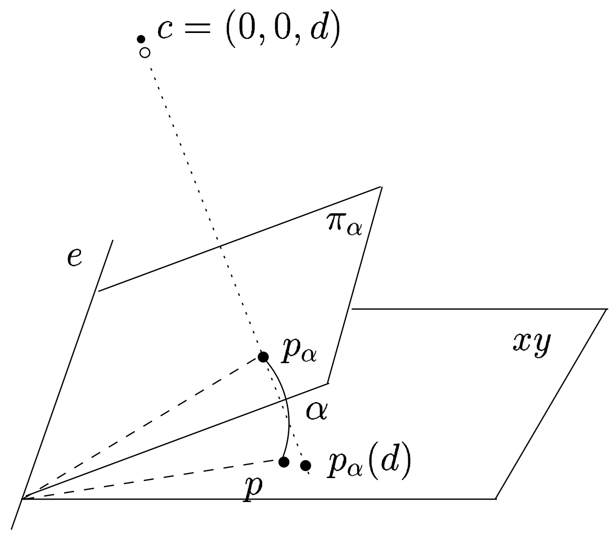

Figure 7: *Approximation by rotation and projection*

We now specify the criterion for choosing the line e, the angle α and the center c so that the input $I_\alpha(c)$ does not have quasi-intrinsic degeneracies and is still a good approximation of the original input I.

Let $e : y = mx + n$ be a line on the xy-plane. When applying a rotation in space of axis e and angle α, a point $p = (x, y)$ is mapped to point p_α

$$p_\alpha = \left(\frac{1+m^2 \cos\alpha}{1+m^2} x + \frac{m(1-\cos\alpha)}{1+m^2}(y-n),\ \frac{m(1-\cos\alpha)}{1+m^2} x + \frac{m^2+\cos\alpha}{1+m^2}(y-n) + n,\ -\frac{m \sin\alpha}{\sqrt{1+m^2}} x + \frac{\sin\alpha}{\sqrt{1+m^2}}(y-n) \right).$$

Point p_α belongs to π_α. Let $c = (0, 0, d), d > 0$ be a center on the z-axis. We compute the projection of p_α from c onto the xy-plane; the projected point $p_\alpha(d)$ has coordinates

$$p_\alpha(d) = \left(\frac{d\left(\frac{1+m^2\cos\alpha}{1+m^2} x + \frac{m(1-\cos\alpha)}{1+m^2}(y-n)\right)}{d + \frac{\sin\alpha}{\sqrt{1+m^2}}(mx - y + n)},\ \frac{d\left(\frac{m(1-\cos\alpha)}{1+m^2} x + \frac{m^2+\cos\alpha}{1+m^2}(y-n)\right) + n}{d + \frac{\sin\alpha}{\sqrt{1+m^2}}(mx - y + n)} \right).$$

It is straightforward to show that

$$\lim_{\alpha \to 0} p_\alpha(d) = p.$$

This expression implies that, by choosing α small enough, there always exists a good approximation between the original input and the perturbed input which is independent of the axis e and center c. For practical purposes, we recommend choosing an angle α close to zero; in particular, if we want to use rational values we can work with α such that $\sin\alpha = \frac{115}{6613}$ and $\cos\alpha = \frac{6612}{6613}$.

In the two remaining subsections we describe the scheme to set e for inputs composed of points and of lines. Once e and α are determined, we apply to I_α the algorithms described in the first few sections of this paper in order to obtain a center of projection $c = (0, 0, d)$ such that the perspective projection of I_α from c removes the original quasi-intrinsic degeneracies.

6.1 Scheme for point sets

If we make d to tend to infinity, we obtain

$$\lim_{d\to\infty} p_\alpha(d) = \left(\frac{1+m^2\cos\alpha}{1+m^2}x + \frac{m(1-\cos\alpha)}{1+m^2}(y-n), \frac{m(1-\cos\alpha)}{1+m^2}x + \frac{m^2+\cos\alpha}{1+m^2}(y-n)+n\right) = p'_\alpha.$$

We call p'_α the result of calculating that limit. On the other hand, the square of the distance from $p = (x, y)$ to the axis of rotation is

$$d(p,e)^2 = \frac{(mx - y + n)^2}{1+m^2},$$

and, therefore, for the distance from p to p'_α, we have the following relation

$$d(p,p'_\alpha)^2 = \frac{(mx-y+n)^2(1-\cos\alpha)^2}{1+m^2} = d(p,e)^2(1-\cos\alpha)^2\,.$$

This nice equation provides us with the desired criterion for choosing e. Since α is already fixed, the term we should minimize is $d(p, e)$. Therefore, if $P = \{p_1, \cdots, p_n\}$ are the points of the input, we choose e as the line given by the least-square fit, that is, the line that minimizes the expression $\sum_{i=1}^{n} d(p_i, e)^2$.

6.2 Scheme for sets of lines

Given any line $r : y = ax + b$, let r_α be the rotated line on plane π_α and $r_\alpha(d)$ be the projection of π_α from $c = (0, 0, d), d > 0$ onto the xy-plane, as described above. As we did before, we take limits and find the line r'_α

$$\lim_{d\to\infty} r_\alpha(d) = r'_\alpha,$$

whose equation is

$$r'_\alpha : y = \frac{a(m^2 + \cos\alpha) + (1 - \cos\alpha)m}{(1 + m^2\cos\alpha) + am(1 - \cos\alpha)} x + \frac{amn(1 - \cos\alpha) + b\cos\alpha(1 + m^2) + n(1 - \cos\alpha)}{(1 + m^2\cos\alpha) + am(1 - \cos\alpha)}.$$

If we associate point $E = (m, n)$ with the axis $e : y = mx + n$ and associate point $R = (a, b)$ with the line $r : y = ax + b$, it can be verified that, as d tends to infinity, the less the distance $d(R, E)$ is, the better is the approximation between lines r and r'_α. This remark suggests a criterion for picking an optimal axis of rotation e. Let $r_i : y = a_i x + b_i, i = 1, \cdots, n$ be n non-vertical lines. We define $R_i = (a_i, b_i), i = 1, \cdots, n$ and choose axis e so that the function $\sum_{i=1}^{n} d(R_i, E)^2$ is a minimum. Note that we are performing a least-squares fit in the dual space.

6.3 The complexity of removing quasi-intrinsic degeneracies

Theorem 6.1 *The complexity of removing quasi-intrinsic degeneracies is the same as that of computing the projection center required to remove them.*

Proof. The axis of rotation can be computed in linear time. Rotating and projecting the input also takes linear time. Since the complexity of computing the projection center required to remove the quasi-intrinsic degeneracies is higher than linear, the total complexity of the removal process is the same as that of computing the projection center. Naturally, this complexity depends on the class of quasi-intrinsic degeneracies removed by the process. □

7 Conclusions and future work

In this paper we have considered several problems that are relevant to a variety of non-degeneracy assumptions and resulting robustness of geometric algorithms. Algorithms have been presented to compute a variety of non-degenerate projections: with distinct x-coordinates, distinct x- and y-coordinates, no three points collinear, no four points cocircular, and with non-parallel line segments. We also showed how to remove quasi-intrinsic degeneracies for point and line segment sets in the plane by using perspective projections as a way of perturbing the input. The results presented here are immediately applicable to graphics visualization problems such as those investigated by Kamada and Kaway [13] as well as computer vision problems such as those in [4] and [15].

Several avenues for further research remain unexplored. For handling degeneracies in computational geometry, dimensions higher than three are also important. The results presented here should extend to higher dimensions as well but the details remain to be worked out. Finally, no lower bounds on the complexity of the computation problems considered here have been obtained and so the optimality of our algorithms is not yet settled.

Acknowledgements:. The research of the first author was supported by MEC-DGES-SEUID PB98-0933. The second author was supported by MEC-DGES-SEUID PB98-0933 and DURSI 1999SGR00356. Research of the third author was supported by MEC-DGES-SEUID PB98-0933 and DURSI 1999SGR00162. The fourth author was supported by NSERC-A9293 and FCAR-EQ1678.

References

[1] P. Bose, M. F. Gómez, P. Ramos and G. Toussaint, "Drawing Nice Projections of Objects in Space", *Journal of Visual Communication and Image Representation* 10(2), pp 155-172,1999.

[2] C. Burnikel, K. Mehlhorn, and S. Schirra, "On degeneracy in geometric computation", *Proc. 5th ACM SIAM symposium on Discrete Algorithms* , pp 16-23, 1994.

[3] S.E Chen and L. Williams, "View interpolation for image synthesis", *Computer Graphics Proceedings, Annual Conference Series (SIGGRAPH'93)*, pp 279-288, 1993.

[4] S.J. Dickinson and D. Wilkes and J.K. Tsotsos, "A computational model of view degeneracy", *IEEE Transactions on Pattern Analysis and Machine Intelligence* 21, pp 673-689, 1999.

[5] D. Dobkin and R. Lipton, "On the complexity of computation under varying sets of primitives", *J. Comput. Syst. Sci.* 18, pp 86-91, 1979.

[6] H. Edelsbrunner, J. O'Rourke and R. Seidel, "Constructing arrangements of lines and hyperplanes with applications", *SIAM J. Computing* 15(2), pp 341-363, 1986.

[7] H. Edelsbrunner, E.P. Mücke, "Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms", *ACM Trans. Graphics* (9), pp 67-104, 1990.

[8] I.Z. Emiris and J.F. Canny, "An efficient approach to removing geometric degeneracies", *Proc. 8th ACM Symposium on Computational Geometry* , pp 74-82, 1991.

[9] I.Z. Emiris and J.F. Canny, "A general approach to removing degeneracies", *SIAM Journal of Computing* 24(3), pp 650-664, 1995.

[10] A. Gajentaan and M.H. Overmars, "On a class of $O(n^2)$ problems in computational geometry", *Computational Geometry. Theory and Applications*, 5(3), pp 165-185, 1995.

[11] R.S. Gallagher, "*Computer Visualization: Graphics Techniques for Engineering and Scientific Analysis*", IEEE Computer Society Press, 1995.

[12] F. Gómez, S. Ramaswami and G. Toussaint, "On removing non-degeneracy assumptions in computational geometry", *Proc. of the 3rd Italian Conference on Algorithms and Complexity*, 1997.

[13] T. Kamada and S. Kawai, "A simple method for computing general position in displaying three-dimensional objects", *Computer Vision, Graphics and Image Processing*, 41, pp 43-56, 1988.

[14] P.R. Keller and M.M. Keller, "*Visual Cues: Practical Data Visualization*", IEEE Computer Society Press, 1993.

[15] J. Kender and D. Freudenstein, "What is a degenerate view?", *Proc. 10th International Joint Conference on Artificial Intelligence*, pp 801-804, 1987.

[16] R. Seidel, "The nature and meaning of perturbations in geometric computing", *Discrete & Computational Geometry* 19(1), pp 1-17, 1998.

[17] C.K. Yap, "Symbolic treatment of geometric degeneracies", *Journal of Symbolic Computation* 10, pp 349-370, 1990.