UNIVERSITÀ DEGLI STUDI FIRENZE

FLORE
Repository istituzionale dell'Università degli Studi di Firenze

From unordered point cloud to weighted B-Spline - a novel PCA-based method -

Questa è la Versione finale referata (Post print/Accepted manuscript) della seguente pubblicazione:

Original Citation:

From unordered point cloud to weighted B-Spline - a novel PCA-based method - / R.Furferi;L.Governi;M.Palai;Y.Volpe. - STAMPA. - Proceedings of: International Conference on COMPUTER ENGINEERING and APPLICATIONS (CEA '11):(2011), pp. 146-151. (Intervento presentato al convegno CEA '11 tenutosi a Puerto Morelos, Mexico nel January 29-31, 2011).

Availability:

This version is available at: 2158/500656 since:

Publisher:

World Scientific and Engineering Academy and Society

Terms of use:

Open Access

La pubblicazione è resa disponibile sotto le norme e i termini della licenza di deposito, secondo quanto stabilito dalla Policy per l'accesso aperto dell'Università degli Studi di Firenze (https://www.sba.unifi.it/upload/policy-oa-2016-1.pdf)

Publisher copyright claim:

(Article begins on next page)

12 October 2024

From unordered point cloud to weighted B-Spline - a novel PCA-based method -

ROCCO FURFERI, LAPO GOVERNI, MATTEO PALAI, YARY VOLPE
Department of Mechanics and Industrial Technologies
Università degli Studi di Firenze
Via Santa Marta, 3 - Firenze
ITALY
matteo.palai@unifi.it http://www.dmti.unifi.it

Abstract: - Digital applications such as CG, CAD and GIS are based on vectorial data since all the information about shape, size, topology etc. are provided in such kind of data representation rather than raster one. Turning raster images into vector ones is a key issue which has been addressed by a number of authors but still far to be exhaustively worked out. Especially in the case of 2D images representing technical drawings, fitting analytical curves to point clouds (pixel sets) is a critical matter.
The present paper provides a novel approach to fit unordered point cloud data. Such an approach integrates a PCA-based method, for detecting the main local directions of the point cloud and to order the points, with and a weighted approximation of a B-spline curve to the original data, based on pixel gray levels. The methodology, tested against alternative techniques based on Least Square (LS) B-spline approximation and on image thinning, proved to be effective in preserving the original shape according to human perception.

Key-Words: - image processing, curve reconstruction, PCA, weighted least-squares, unorganized points.

1 Introduction

During the last three decades, image processing techniques have been more and more applied in a wide number of research and industrial fields [1].
Independently from the kind of application, image processing is based on the acquisition and manipulation of digital images composed by data structures generally representing a rectangular grid of pixels with assigned color values. Such digital structures, known as "raster" data, provides a discrete representation of the acquired scene.
As widely known, digital applications such as CG, CAD and GIS seldom manipulate raster data since all the information about shape, size, topology etc. are not provided in such kind of representation. The only viable approach is to use vectorial data represented, for instance, by points, lines, curves, polygons and surfaces defined by a mathematical description geometrically related one with the other (e.g. points are represented using their coordinates and lines are stored as a series of point pairs, where each pair represents a straight line segment). Moreover, vector images are composed of "paths" representing individual objects that may be geometrically and morphologically manipulated without a degradation of the image quality since no sampling occurs.
Accordingly, turning raster images into vector ones is a key issue which has been addressed by a number of authors but still far to be exhaustively worked out.
One of the most common problem to be faced when vectorizing raster data (e.g. images obtained from blueprints scanning) is the fitting of analytical curves to pixel clusters representing drawing sketches.
Fitting vectorial curves to point datasets has been widely studied and many techniques, theories and commercial software tools have been developed. Depending on input data typology, the research approaches proposed in the scientific literature may be divided into two main families:
1) Ordered point cloud-based approaches; when the order of data points is assumed to be known, the fitting curve can be easily obtained by minimizing error functions or by means of computational geometry methods [2,3,4,5].
2) Unordered point cloud-based approaches. For many practical problems, point cloud data are, usually, unordered i.e. the order of data points is unknown. In this case the key problem is to transform the unordered point cloud into an ordered one. The order, of course, needs to be related to the geometry of the point cloud. When the order is known, curve reconstruction can be achieved by interpolating or fitting an ordered point set with a parametric curve.
A number of approaches have been developed for solving this issue; for instance, Pottmann [6] maps data points to binary images, then fits a curve to the image's medial axis. Clustering the point cloud data is another commonly used approach: Yan [7], for instance, presents a fuzzy curve-tracing algorithm that works by means of several clusters identified by fuzzy algorithms. In a more recent work, Liu et al. [8] devise a new algorithm based on the idea that a spline curve can be made to crawl and stretch along the curve shape defined by a point cloud.
Another typical issue for unordered data sets is the presence of noise. Levin [9] provides a Moving Least Squares (MLS) approach so as to clean noisy point cloud data. Lee [10] improves MLS by means of appropriate neighborhoods for

regressions; the main weakness of this method is represented by the computational cost of iterating MLS.

A common approach, which can also be found in commercial vectorization software packages, is to convert the original color or grayscale image into a binary ones and, subsequently, to perform a thinning procedure prior to vectorization step [11]. Such an approach, though usually quite effective, is unsuitable in case image gray levels convey relevant geometrical information.

The main idea of the present paper is to provide an approach to fit unordered point cloud data with a weighted B-spline curve, whose weights are computed on the basis of raster data gray levels.

The method involves a two-steps procedure. The first step, in order to preserve the geometry of the point cloud, consists of an iterative application of Principal Component Analysis (PCA) [12] to point clusters composing the entire dataset taking into account their "relevance" by means of their gray level (weighting procedure).

As described in the next section, the result of this first step is to detect both the main local directions of the point cloud and to order the point set. Once the ordered point cloud has been obtained, the second step allows to approximate the points with a parametric weighted B-spline whose degree of approximation is customized according to a user defined tolerance.

The paper is organized as follows: in section 2 the methodology is described; in section 3 a comparison between the proposed method and alternative ones is provided; finally, conclusions and future works are discussed.

2 Method

The proposed method can be synthesized in two principal tasks named as follows:

1. Image processing;
2. PCA based polyline extraction;
3. B-Spline based order determination;
4. B-Spline fitting.

2.1 Image processing

Let J be a digital raster image obtained, for instance, by scanning a sketch or a blueprint (Fig. 1). Generally speaking, the image is treated as a grid of discrete elements (pixels), ordered from top to bottom and left to right.

Sketch contours are represented by a sequence of pixels $j_{m,n}$ characterized by discrete couples of coordinates referred to an orthogonal coordinate system centered on the top left corner of the image. Each pixel is, on its side, described by a triplet of values in the RGB colour space [13] or, possibly, by a single luminance value. In the first case, a simple transformation from RGB to grayscale image is required. An adaptive thresholding method [14] is applied to determine the pixels composing the sketch contour; in particular the image is divided into 9 patches, and each patch is thresholded by a value that depends on the patch contents.

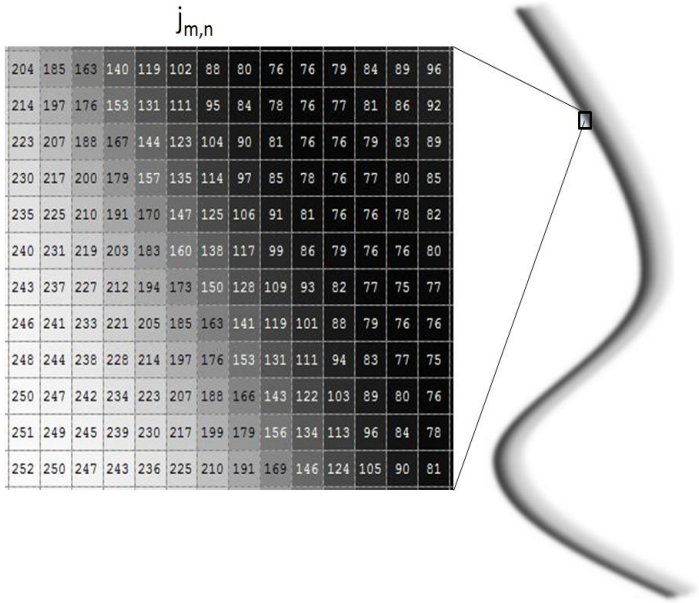

Fig. 1: Grayscale image detail.

The result of this thresholding is a binary image B where the sketch contour is represented by white pixels (i.e. $b_{m,n} = 0$) while the background is represented by black pixels (i.e. $b_{m,n} = 1$).

As described below, the proposed approach assign a weight to each contour pixel on the basis of its luminance. As a consequence, the Hadamard product of the 1-complement B' of B and the inverse of the original image J allows to obtain a new image W whose non-zero elements $w_{m,n}$ are the weight values of the contour pixels (Fig. 2):

$$w_{m,n} = b'_{m,n} \cdot \frac{1}{j_{m,n}} \quad \forall b'_{m,n} \in B', \forall j_{m,n} \in J \tag{1}$$

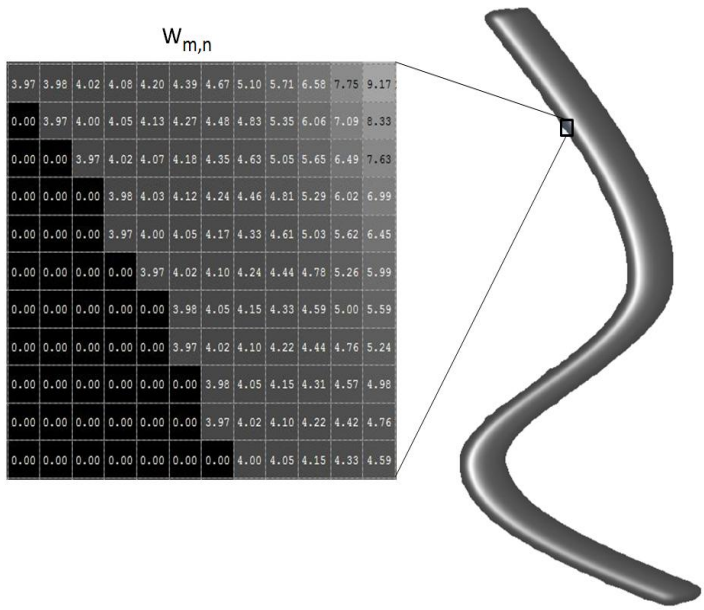

Fig. 2: Weight image detail.

2.2 PCA based polyline extraction

In order to extract the local geometric features of the point cloud, the Principal Component Analysis (PCA) has been used. In fact, PCA is a multivariate method that analyzes a dataset in which observations are described by several inter-correlated quantitative dependent variables [12].
Such a technique allows to extract the most relevant information from the dataset, to represent it as a set of new orthogonal variables called principal components, and to display the pattern of similarity of the observations and of the variables as points in a map. Given a blob of points (Fig. 3), PCA allows to determine, among other, the principal inertia axis and the centroid of the dataset (p_1 and p_2).

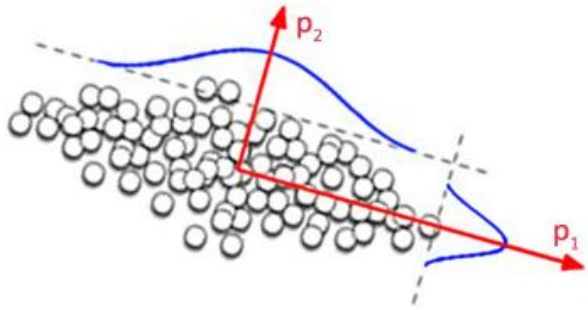

Fig. 3: PCA on a blob of points.

By definition, point cloud consists of the only pixels of image *B'* satisfying the condition $b_{m,n}$ = 1. Let $b_{h,k}$ be a "seed point" randomly extracted from the point cloud. Let us define the first cluster ***C*** as the locus of points $c_{m,n}$ whose coordinates *(m, n)* satisfy the following condition:

$$(h-m)^2+(k-n)^2 \le r^2 \tag{2}$$

where *r* is a user defined cluster radius (customized tolerance). Let us assume that *C* contains *t* points (Fig. 4).
The cluster centroid χ is a vector whose elements are the weighted mean values of all the points $c_{m,n}$:

$$\chi = [\bar{m}, \bar{n}] \tag{3}$$

Where:

$$\begin{cases} \bar{m} = \dfrac{\sum_{c \in C} m \cdot w_{m,n}}{\sum_{c \in C} w_{m,n}} \\ \bar{n} = \dfrac{\sum_{c \in C} n \cdot w_{m,n}}{\sum_{c \in C} w_{m,n}} \end{cases} \tag{4}$$

Expressing the coordinates of the pixels belonging to the cluster ***C*** in a new coordinate system whose origin coincides with χ and whose axis are parallel to the original ones it is possible to define a matrix ψ of the new pixel coordinates. Such a matrix can be processed by means of PCA.
First, according to the well known Singular Value Decomposition (SVD) technique, matrix ψ is expressed as follows [15]:

$$\psi = \mathrm{M} \cdot \mathrm{S} \cdot \mathrm{V} \tag{5}$$

Where ***M*** is the 2 x 2 matrix of eigenvectors of $\psi\psi^T$, the matrix ***S*** is a rectangular diagonal *t* x 2 matrix with nonnegative real values (singular values), and ***V*** is a *t* x *t* square matrix. The first column of ***M*** identifies the first principal component (i.e. the principal inertia axes of the cluster).

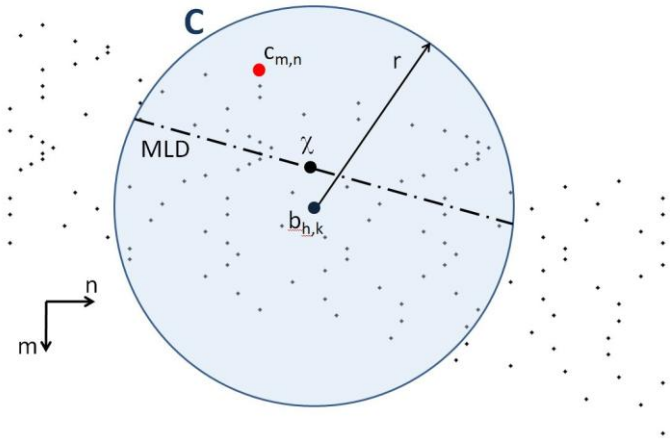

Fig. 4: First step of the polyline construction procedure.

From matrix ***S*** the 2 x 2 eigenvalues matrix Λ of $\psi\psi^T$ can be computed:

$$\Lambda = \frac{1}{t-1} \mathrm{S} \cdot \mathrm{S} \tag{6}$$

If the ratio between the smallest and the greatest eigenvalue tends to 1 (e.g. is greater than 0.9), the cluster is considered to have no "preferred" orientation; thus the cluster radius has to be increased. Otherwise the cluster radius is considered to be "correct" and the principal inertia axes determine the Main Local Direction (MLD) vector of the cluster itself. With this approach it is possible to define a Size Variable Cluster (SVC) whose MLD is determined (Fig. 4).
The two intersection points between the MLD and the circumference with radius *r* define the geometric centres of two new SVCs (C_{p1} and C_{f1}) to be processed as shown in Fig 5.

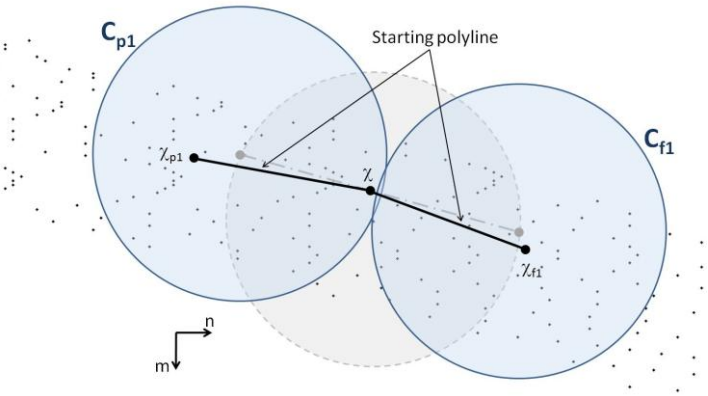

Fig. 5: Second step of the polyline construction procedure.

Analogously to the first cluster, two new centroids χ_{f1} and χ_{p1} are determined. The segments from χ_{p1} to χ and from χ to χ_{f1} define a starting polyline.
Starting from χ_{f1} the procedure is iteratively carried out until, at the ith iteration, one of the following two conditions occurs:

1. the cluster at the ith iteration C_i contains no point that are not contained in C_{i-1} (i.e. an endpoint of the point cloud has been reached); in this case, since the drafted entity is "open", the iterative process has to be continued starting from χ_{p1}.
2. the C_i region contains χ_{p1} (i.e. one of the polyline points belongs to C_i region); in this case the drafted entity is "close" so no further iteration is required. The last segment of the closed polyline is the one defined by the centroid of C_i and χ_{p1}.

2.3 B-Spline based order determination

Once the polyline has been built, a B-Spline called "ordering curve" (OC), intepolating only the polyline points, is traced (Fig. 6). The purpose of such a B-Spline consists of sorting the point cloud points, according to the order of their projection on the OC.
The point projection problem is to find the closest point on the curve to a given point. In this work this problem has been solved by a numerical, approximated approach as follows:
1. given a cluster C_u selected among the ones defined in section 2.2, the clusters C_{u-1} and C_{u+1} are considered along with their centroids χ_{u-1} and χ_{u+1}.
2. the OC, which by definition interpolates all the centroids χ_i, is uniformly sampled in the interval delimited by χ_{u-1} and χ_{u+1}. The number of OC samples is set equal to three times the global numerosity of the above considered clusters.
3. the projection of each point p belonging to C_u on the OC is the point $p_{s'}$ that is its nearest among the OC sample points p_s:

$$p_{s'} : \| p - p_{s'} \| = \min\{\| p - p_s \|\} \qquad (7)$$

4. the procedure carries on iteratively for each cluster. Obviously, the points belonging to previously processed clusters are not further considered.

The final result of this procedure is an implicit ordered point cloud. In order to obtain an explicit ordered point cloud, a normalized curvilinear coordinate s on the OC has to be defined. Using one of the OC endpoints as the coordinate origin, it is possible to assign a value s for each point projection.

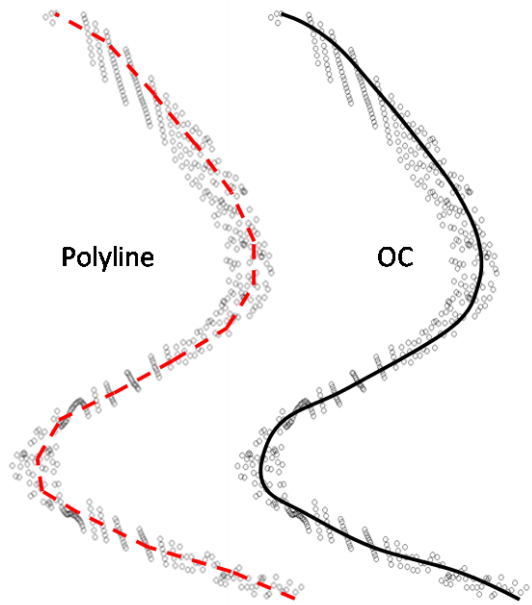

Fig. 6: Polyline curve and OC.

2.4. B-Spline fitting.

Given an ordered set of points, it is straightforward to fit an approximating B-Spline curve called Fitting Curve (FC). The B-Spline which is used to fit the original points is built so that: its parameter t varies between 1 and a with unitary step, where a is the total number of the points belonging to the original point cloud; for each integer value t_i of t, the spline point of parameter t_i is biunivocally correspondent to the ith point of the ordered cloud. Thanks to this correspondence, it is immediate to evaluate the Maximum Fitting Error (MFE) as:

$$MFE = \max_{t=1 \to a}\{W(p_{t,1}, p_{t,2}) \| FC(t) - p_t \|\} \qquad (8)$$

The fitting task starts with the generation of a two-knot first approximation FC; the initial knots, which have a multiplicity of 4, are coincident with the two OC endpoints (Fig 7).
This FC, along with the ones subsequently computed, is generated according to Weighted Least Squares (WLS)

technique [16] where the weighting coefficient for each point of coordinates (m,n) is $w_{m,n}$.

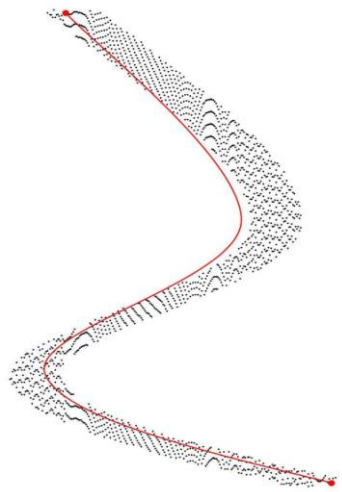

Fig. 7: First attempt FC.

By means of an iterative procedure (Fig 8), FC is updated by adding new knots with the aim of obtaining the necessary degree of approximation; in order to guarantee the desired accuracy, a user defined tolerance value (*tol*) is required.

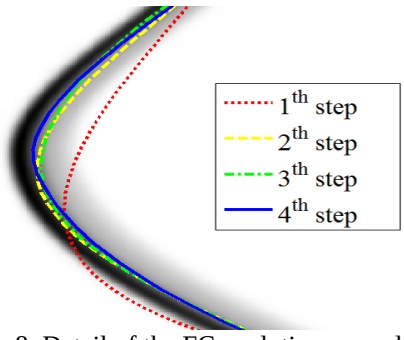

Fig. 8: Detail of the FC evolution procedure.

3 Results

In Figure 9 the result of the proposed method on a case study image is compared to two alternative B-spline fitting techniques.

The first technique (denoted with LS1) performs LS approximation obtained by using the same algorithm described in this work, considering all the pixels whose gray level is lower than 250 (background color is assumed to be represented by values greater or equal to 250) but disabling any kind of weighting (both for MLD detection and for B-spline curve computation). The second technique (denoted with LS2), implemented by a commercial vectorization software package, performs an automatic thresholding of the image and a subsequent thinning process. A B-spline is, then, fitted to the thinned image according to a LS method.

More in detail the LS1 technique roughly reproduces the shape presented in the image though, due to the lack of weighting, the fitting curve approximates the medial curve of the "non-white" region.

LS2 technique provides a more accurate fitting curve which approximates more closely the darker region due to the preliminary thresholding step eroding mainly the lighter areas. Nevertheless, due to the thinning step, spurious segments are generated.

The fitting curve obtained by means of the proposed solution proves to be the most effective in preserving the original shape according to human perception.

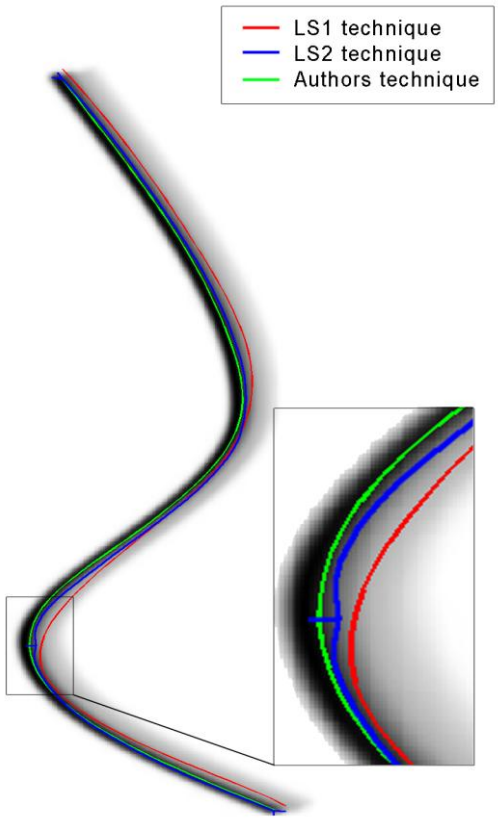

Fig. 9: Comparison between the proposed method and the classic one (based on simple LS fitting splines) .

4 Conclusion

In the present work an approach to fit unordered point cloud data with a weighted B-spline curve, whose weights are computed on the basis of raster data gray levels is presented.

Such an approach may be particularly suitable for the B-Spline vectorization of raster images (for instance representing handmade drawings or sketches) where the line fuzziness cannot be neglected. Furthermore, even assuming that a carefully selected threshold value may compensate the line fuzziness, the proposed method performance overcomes the ones of alternative techniques making use of image thinning (which is often responsible of spurious segments generation). Future work will be addressed to a 3D generalization of the proposed method and to the implementation of variable cluster dimension in order to take into account variable line thickness.

References:

[1] I. Pitas. Digital Image Processing Algorithms and Applications (1st ed.). *John Wiley & Sons, Inc.*, 2000, New York, NY, USA.

[2] N. Amenta, M. Bern, and D. Eppstein. The crust and the beta-skeleton: combinatorial curve reconstruction. *Graphical Models and Image Processing*, 60:125–135, 1998.

[3] A. Atieg and G. A. Watson. A class of methods for fitting a curve or surface to data by minimizing the sum of squares of orthogonal distances. *Journal of Computational and Applied Mathematics*, 158:227–296, 2003.

[4] A. Bjorck. Numerical Methods for Least Squares Problems. *Mathematics Society for Industrial and Applied Mathematics*, Philadelphia, 1996.

[5] T. K. Dey, K. Mehlhorn, and E. Ramos. Curve reconstruction: connecting dots with good reason. *Comput. Geom. Theiry & Appl.*, 15:229–244, 2000.

[6] H. Pottmann and T. Randrup. Rotational and helical surface approximation for reverse engineering. *Computing*, 60(4):307–322, 1998.

[7] H. Yan. Fuzzy curve-tracing algorithm. *IEEE transactions on Systems, Man,and Cybernetics–Part B: Cybernetics*, 31(5):768–780, 2001.

[8] Y. Liu, H. Yang, W. Wang, Reconstructing B-spline curves from point clouds—a tangential flow approach using least squares minimization, *Proceedings of SMI*, 2005, pp. 4–12.

[9] D. Levin. The approximation power of moving least-squares. *Mathematics of Computation*, 67(224):1517–1531, 1998.

[10] I.-K. Lee. Curve reconstruction from unorganized points. *Computer Aided Geometric Design*, 17:161–177, 2000.

[11] URL: http://www.scan2cad.com/

[12] J. Shlens. A Tutorial on Principal Component Analysis. *Institute for Nonlinear Science*, UCSD, 2005.

[13] S. Süsstrunk, R. Buckley, and S. Swen, "Standard RGB color spaces", *Proc. IS T/SID 7th Color Imaging Conf.*, 1999, pp. 127 - 134.

[14] Shapiro, L.G., Stockman, G.C. "Computer Vision". *Prentice Hall*, 2002.

[15] Klema, V. C. and Laub, A. J. The singular value decomposition: its computation and some applications. *IEEE Trans. Automat. Control*, AC-25, 1980, 164--176.

[16] Andrew Nealen, “An as-short-as-possible introduction to the least squares, weighted least squares and moving least squares methods for scattered data approximation and interpolation,”, May 2004, URL: http://www.nealen.com/projects/.