Approximation by Spline Curves: towards an Application to Cognitive Neuroscience

Maria-Laura Torrente, Stefano Anzellotti, Chiara Finocchiaro and Claudio Fontanari

Abstract—We present a procedure to approximate a plane contour by piecewise polynomial functions, depending on various parameters, such as degree, number of local patches, selection of knots. This procedure aims to be adopted to study how information about shape is represented.

Index Terms—Shape recognition, perceptual discrimination, spline curves, B-splines, least squares approximation.

I. INTRODUCTION

The shape of an object is a fundamental source of information for recognition. While color and texture information are also important (see [1]), line drawings, which discard texture and color, are usually sufficient to recognize an object. Furthermore, infants generalize object labels on the basis of shape similarity rather than color similarity (see [2]), suggesting that shape plays a key role in categorization. In order to understand how we recognize and categorize objects, therefore, it is crucial to study how information about shape is represented.

Since the space of shapes is infinite dimensional, the brain likely approximates it using a lower number of dimensions in order to make the problem more tractable. For comparison, it can be useful to consider the example of color. Like the space of shapes, the space of light spectra is infinite dimensional. However, with three dimensions it is possible to model accurately the space of perceptually discernible colors (see [3]). In the same vein, we can look for finite dimensional spaces that encompass the perceptually discernible shapes.

Recent work (see [4]) found that a model with approximately 47 dimensions explains accurately a large amount of the variance in the object recognition errors and similarity judgments of human participants. However, models of this type leave open the question of whether the dimensionality of shape space is the same in the neighborhoods of all shapes. In addition, the most relevant dimensions for recognition might differ in the neighborhood of different shapes. For example, the relevant dimensions for discriminating between individual faces and for discriminating between buildings might not be the same, and thus different low-dimensional approximations of the space of shapes could be used in the two neighborhoods. This possibility is compatible with the differential involvement of different brain regions within temporal cortex in the processing of different object categories (see [5]). Our interdisciplinary research project aims to introduce techniques to individuate lower dimensional subspaces that locally approximate the space of perceptually discernible shapes.

A couple of crucial remarks are in order here. First, our approach aims to model shape discriminability rather than object recognition accuracy. In other words, it is a model of perception rather than a model of perceptual categorization (in this respect, it is quite similar to the case of color space). The choice of investigating discriminability is based on the idea that a common representation of shape underlies both shape discrimination and judgements of perceptual similarity that are at the basis of perceptual categorization. This idea is supported by the evidence currently available (see [4]). Next, our goal is to represent shapes locally rather than constructing a single lower dimensional space underlying the representation of all shapes. This choice is a consequence of the aim to model perceptual discriminability: different local spaces seem necessary to account for the large variety of shapes we can perceptually discriminate.

For modeling shape space in the neighborhood of a fixed shape spline curves will be used. Spline curves can approximate contours closely, they enjoy elegant mathematical properties, and their computational complexity is relatively low. For all these reasons, splines are widely considered to be an ideal tool to approximate signals (see for instance [6]). In particular, the use of B-splines for digital signal processing has a long history (at least since [7]). In this paper we employ spline curves to address the following approximation problem: given a plane contour \( C \) construct a (parametric) spline curve \( C' \) of a fixed degree \( d \) well approximating \( C \). The procedure (described in Section II and based on Section I) is divided into three parts: use of strengthened edge detection techniques to extract the points of the contour \( C \); computation of the spline curve \( C' \), performed by solving a Least Squares Approximation problem (see Problem II.5 and subsequent discussion); plot of the computed spline curve (and, eventually, comparison with \( C \)). The spline approximation part (second part) is the core of the presented procedure and, as it is well known to the experts (see for instance [8]), an important issue for it is the choice of knots. At the end of Section II we recall three of the most used parametrization methods (uniform, chord length and centripetal). The chord length parametrization method is then chosen to show a concrete example (see figures I(a) I(f)).

This research project was partially supported by GNSAGA of INdAM, by PRIN 2010–2011 “Geometria delle varietà algebriche”, and by FIRB 2012 “Moduli spaces and Applications”.

M. Torrente is with the Dipartimento di Matematica, Università di Genova, 16146 Genova, Italy (e-mail: torrente@dima.unige.it).

S. Anzellotti is with the Department of Brain and Cognitive Sciences, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: anzellot@mit.edu).

C. Finocchiaro is with the Dipartimento di Psicologia e Scienze Cognitive, Università degli Studi di Trento, 38068 Rovereto, Italy (e-mail: chiara.finocchiaro@unitn.it).

C. Fontanari is with the Dipartimento di Matematica and Centro Interdipartimentale Mente/Cervello-CIMEC, Università degli Studi di Trento, 38123 Trento, Italy (e-mail: fontanar@science.unitn.it).
II. BASICS OF SPLINE APPROXIMATION

The main reference for this section is [9].

Bézier curves and, after their introduction in 1946 (see [10]), spline curves have been widely used to construct smooth curves from a given set of points in an efficient and numerically stable way. Their geometrical construction is essentially based on recursive convex combinations of curves of smaller degrees. Namely:

Definition II.1. Let \( n, d \) be positive integers, with \( n \geq d + 1 \). Let \( p_1, \ldots, p_n \in \mathbb{R}^2 \) be \( n \) control points, and let \( \tau = (t_i)_{i=1}^{n+d+1} \) be the knot vector, which is assumed to be a nondecreasing sequence of \( n + d - 1 \) real numbers, that is, \( t_2 \leq \ldots \leq t_{n+d} \). The functions \( p_{i,j}(t) \) are recursively defined in the following way: we set \( p_{i,0}(t) = p_i \) for \( i = 1, \ldots, n \) and

\[
p_{i,j}(t) = \frac{t-t_i}{t_{i+j} - t_i} p_{i+1,j-1}(t) + \frac{t_{i+j+1} - t}{t_{i+j+1} - t_{i+1}} p_{i,j-1}(t)
\]

for \( j = 1, \ldots, d \) and \( i = j + 1, \ldots, n \), where possible division by zero is resolved by the convention that a division by zero is resolved by the convention that a division by zero is equal to zero. The (parametric) spline curve \( f(t) \) of degree \( d \) with control points \( (p_i)_{i=1}^n \) and knots \( \tau = (t_i)_{i=1}^{n+d+1} \) can be written as:

\[
f(t) = \sum_{i=1}^{n} p_i B_{i,d}(t)
\]

where \( B_{i,d}(t) \) is the \( i \)th B-spline of degree \( d \) with knots \( \tau \).

The notion of linear combination of B-splines, also called spline function, is formalized in the following definition.

Definition II.2. Let \( n, d \) be positive integers, with \( n \geq d + 1 \). Let \( \tau = (t_i)_{i=1}^{n+d+1} \) be a nondecreasing sequence of reals, and let \( B_1, \ldots, B_n \) be the \( n \) B-splines of degree \( d \) with knots \( \tau \). The linear space of all linear combinations of \( B_1, \ldots, B_n \) is called the spline space of degree \( d \) with knots \( \tau \) and denoted by:

\[
S_{d,\tau} = \left\{ \sum_{i=1}^{n} c_i B_{i,d} \mid c_i \in \mathbb{R}, i = 1, \ldots, n \right\}
\]

Now we consider the approximation problem. The method is based on the standard least squares approach, in which classically the function to minimize is the sum of squared errors committed at the given data points. The problem can be formulated as follows.

Problem II.5. (Least Squares Approximation Problem) Let \( m > n \); given data \( (x_k, y_k)_{k=1}^m \) and a spline space \( S_{d,\tau} \) whose knot vector \( \tau = (t_i)_{i=1}^{n+d+1} \) satisfies \( t_i \leq t_{i+1} \) for \( i = 1, \ldots, n \), find a spline function \( f = \sum_{i=1}^{n} c_i B_{i,d}(t) \in S_{d,\tau} \) which solves the minimization problem

\[
\sum_{k=1}^{m} (y_k - f(x_k))^2 = \sum_{k=1}^{m} \left( y_k - \sum_{i=1}^{n} c_i B_{i,d}(x_i) \right)^2 = \min_{g \in S_{d,\tau}} \sum_{k=1}^{m} (y_k - g(x_k))^2
\]

It is easy to see that this is a LS problem which can be expressed in matrix form as follows: find \( c = (c_1, \ldots, c_n)^t \) s.t.

\[
\|y - Bc\|_2^2 = \min_{\alpha \in \mathbb{R}^n} \|y - B\alpha\|_2^2
\]

where

\[
B = \begin{bmatrix} B_{1,d}(x_1) & \cdots & B_{n,d}(x_1) \\ \vdots & \ddots & \vdots \\ B_{1,d}(x_m) & \cdots & B_{n,d}(x_m) \end{bmatrix} \in \text{Mat}_{m \times n}(\mathbb{R})
\]

is the coefficient matrix, also called B-spline collocation matrix, and

\[
c = \begin{bmatrix} c_1 \\ \vdots \\ c_n \end{bmatrix} \in \text{Mat}_{n \times 1}(\mathbb{R}), \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} \in \text{Mat}_{m \times 1}(\mathbb{R})
\]

It is well-known that solving (1) is equivalent to solve the normal system:

\[
B^t Bc = B^t y
\]
Note that Problem [1.5] has always a solution. Moreover, there are explicit conditions under which the matrix $B^tB$ is invertible, and consequently system (3), as well as Problem [1.5], has exactly one solution (see for instance [9], Theorem 5.23).

Finally, we address the following problem: suppose we are given a set $X = \{p_1, \ldots, p_m\}$ of $m$ points in $\mathbb{R}^2$, and we want to construct a (parametric) spline curve $C$ of degree $d$ that approximates the points. From Problem [1.5] recalling Definition [1.4] in order to define a spline space $S_{d,\tau}$, it is clear that a knot vector $\tau$ is required. Obviously, different choices of the knot vector lead to different spline spaces and, consequently, to different approximating spline curves. In the literature (see for instance [11]), there are various competing parametrization methods, such as:

1. **Uniform:** $t_i = 0$ and $t_i = \frac{i}{m}$, for $i = 2, \ldots, m$.
2. **Chord length:** $t_1 = 0$ and $t_i = t_{i-1} + \frac{\|p_i - p_{i-1}\|_2}{\sum_{j=1}^{i-1} \|p_{j+1} - p_j\|_2}$, for $i = 2, \ldots, m$.
3. **Centripetal:** $t_1 = 0$ and $t_i = t_{i-1} + \frac{\|p_i - p_{i-1}\|_2^{1/2}}{\sum_{j=1}^{i-1} \|p_{j+1} - p_j\|_2^{1/2}}$, for $i = 2, \ldots, m$.

In this paper we mainly focused on the chord length method, though the procedure for approximating plane contours using spline curves (described in Section III) has been implemented with all three parametrization methods.

### III. Algorithm Description and Implementation

In this section we describe an algorithmic procedure for approximating a given plane contour using spline curves. The procedure can be divided into three parts:

I. acquisition of a digital image reproducing a silhouette;  
   extraction of the contour $C$ by strengthened edge detection techniques; 
II. computation of the spline curve, approximation of the contour $C$; 
III. plot of the computed spline curve (and eventually, comparison with $C$).

Accordingly, the implementation of the algorithm is divided into three steps: the first and the third step, corresponding to parts I and III, are implemented in MatLab, using well-established built-in functions; the second step, corresponding to part II, is implemented in C++ language, using routines both of CoCoALib [12], a GPL C++ library – the mathematical kernel for the computer algebra system CoCoA-5, and of the numerical library GSL - GNU Scientific Library [13]. Note that part II is the core of the algorithm: based on Section II and in particular on Problem [1.5], and the subsequent discussion, its implementation requires as input:

I. an affine set of points in $\mathbb{R}^2$ (resulting from the edge detection on the input image); 
II. a parameter $n$, corresponding to the number of polynomial patches; 
III. a parameter $d$, corresponding to the degree of polynomials; 
IV. a parametrization method (see Section II).

It returns as output:

O1. a coefficient vector defining the piecewise polynomial function made up of $n$ patches of degree $d$; 
O2. the corresponding Least Square Error.

Just to give an idea, we consider the silhouette of a horse, represented in Figure 1(a) Using classical edge detection techniques, the contour of the horse is extracted (see Figure 1(b)) and stored as a set of 2713 points in $\mathbb{R}^2$. The chord length parametrization method is chosen (see Section II), and contour approximations are obtained through computations with different values of the parameters, namely $n = 100$, $d = 2$ (see Figure 1(c)), $n = 200$, $d = 2$ (see Figure 1(d)), $n = 100$, $d = 3$ (see Figure 1(e)) and $n = 200$, $d = 3$ (see Figure 1(f)). We observe that there is a substantial difference in the approximation quality comparing the cases $n = 100$ and $n = 200$, whereas we notice a slight improvement passing from the cases $d = 2$ to $d = 3$.

![Image](a) Original image.  
![Image](b) Points of the edge detection. 
![Image](c) Values $n = 100, d = 2$.  
![Image](d) Values $n = 200, d = 2$. 
![Image](e) Values $n = 100, d = 3$.  
![Image](f) Values $n = 200, d = 3$. 

Fig. 1. Silhouette of a horse

### IV. Conclusion

We have presented a novel procedure to approximate a plane contour by piecewise polynomial functions, depending on various parameters (degree, number of local patches, selection of knots). In order to optimize our choice of parameters, we plan to perform a series of behavioural experiments by adopting the following procedure. First, a set of contours will be selected, representing both shapes of daily life objects and
meaningless curves. Next, every contour will be approximated by different splines, obtained by varying parameters in such a way that each approximation is perceptually closed to the original shape. Then, participants will be asked to perform a same/different task. Finally, the experimental results shall allow us to individuate a lower dimensional spline space approximating the space of perceptually discriminable shapes in a neighborhood of a fixed shape. At this point, it will be possible to ask questions about shape representations such as:

1) Is the dimensionality of the space of perceptually discriminable shapes the same for neighborhoods of different shapes?
2) In particular, does the dimensionality of the space of perceptually discriminable shapes change as a function of experience?
3) As a consequence, do neighborhoods of real objects have higher dimensionality than neighborhoods of meaningless curves?

REFERENCES

[1] A. W. Yip and P. Sinha, “Contribution of color to face recognition”, *Perception*-London 31.8, pp. 995–1004, 2002.
[2] S. A. Graham and D. Poulin-Dubois, “Infants’ reliance on shape to generalize novel labels to animate and inanimate objects”, *Journal of Child Language* 26.02, pp. 295–320, 1999.
[3] J. Krauskopf, D. R. Williams, and D. W. Heeley, “Cardinal Directions of Color Space”, *Vision Res.*, vol. 22, pp. 1123–1131, 1982.
[4] H. Hong et al, “Large-scale Characterization of a Universal and Compact Visual Perceptual Space”, *Dim* 1501, 2014.
[5] L. Reddy and N. Kanwisher, “Coding of visual objects in the ventral stream”, *Current opinion in neurobiology* 16.4, pp. 408–414, 2006.
[6] M. Unser, “Splines: a Perfect Fit for Signal and Image Processing”, *IEEE Signal Processing Magazine*, vol. 16, no. 6, pp. 22–38, Nov. 1999.
[7] H. S. Hou and H. C. Andrews, “Cubic Splines for Image Interpolation and Digital Filtering”, *IEEE Trans. Acoust., Speech, Signal Process.,* vol. ASSP-26, no. 6, pp. 508–517, Dec. 1978.
[8] X. He, L. Shen, and Z. Shen: “A Data-Adaptive Knot Selection Scheme for Fitting Splines”, *IEEE Signal Processing Letters*, vol. 8, no. 5, pp. 137–139, May 2001.
[9] T. Lyche and K Morken, *Spline Methods Draft*, 2008, available at [http://www.uio.no/studier/emner/matnat/ifi/INF-MAT5340/v09/undervisningsmateriale/book.pdf](http://www.uio.no/studier/emner/matnat/ifi/INF-MAT5340/v09/undervisningsmateriale/book.pdf).
[10] I. Schoenberg, “Contributions to the Problem of Approximation of Equidistant Data by Analytic Functions”, *Quart. Appl. Math.*, vol. 4, pp. 45–99, 1946.
[11] E. T. Y. Lee, “Choosing Nodes in Parametric Curve Interpolation”, *Computer-Aided Design*, vol. 21, no. 6, pp. 363–370, 1989.
[12] J. Abbott and A. M. Bigatti, *CoCoALib: a C++ library for doing Computations in Commutative Algebra*, available at [http://cocoa.dima.unige.it/CocoaLib](http://cocoa.dima.unige.it/CocoaLib).
[13] GSL - GNU Scientific Library, available at [https://www.gnu.org/software/gsl/](https://www.gnu.org/software/gsl/).