

A variational approach to recovering a manifold from sample points

José Gomes and Aleksandra Mojsilovic

IBM Watson Research Center
Route 134, Kitchawan Road
Yorktown Heights, N.Y. 10598 , U.S.A

Abstract. We present a novel algorithm for recovering a smooth manifold of unknown dimension and topology from a set of points known to belong to it. Numerous applications in computer vision can be naturally interpreted as instantiations of this fundamental problem. Recently, a non-iterative discrete approach, *tensor voting*, has been introduced to solve this problem and has been applied successfully to various applications. As an alternative, we propose a *variational formulation* of this problem in the *continuous* setting and derive an *iterative* algorithm which approximates its solutions. This method and tensor voting are somewhat the differential and integral form of one another. Although iterative methods are slower in general, the strength of the suggested method is that it can easily be applied when the ambient space is not Euclidean, which is important in many applications. The algorithm consists in solving a partial differential equation that performs a special anisotropic diffusion on an implicit representation of the known set of points. This results in connecting isolated neighbouring points. This approach is very simple, mathematically sound, robust and powerful since it handles in a homogeneous way manifolds of arbitrary dimension and topology, embedded in Euclidean or non-Euclidean spaces, with or without border. We shall present this approach and demonstrate both its benefits and shortcomings in two different contexts: (i) data visual analysis, (ii) skin detection in color images.

1 Introduction

In this section, we state the considered problem. Then, we give an overview of the corresponding state of the art. Finally, we present the organization of the paper.

1.1 Statement of the problem

Consider a set \mathcal{P} containing N points of a n -dimensional manifold Ω . In order to illustrate the ideas, we may think of \mathcal{P} as a data set of N measures performed on a stochastic (or deterministic) process whose state may be partially (or totally) described using n numeric values. Consequently, each point in \mathcal{P} is a sample, each coordinate is a parameter of the process under consideration, and Ω is the set of possible values these variables may take *a priori*. The paradigm of experimental disciplines is that each individual sample captures only a partial information about the state of the process and one hopes, by considering multiple samples,

to apprehend it in a more comprehensive way. Usually, this is motivated by the expectation that the variables, *i.e.* the coordinates of the points, may not be independent from one another and that the relationships between the variables capture the "essence" of the measured process.

In this paper, we propose a theoretical and practical framework for analysing such data sets when the measured variables are expected to have complex distributions, strong relationships and more specifically in the limit case of functionally related variables. Formally, this restriction is equivalent to saying that the points of \mathcal{P} are not distributed arbitrarily in Ω but, instead, up to some noise, belong to a submanifold of Ω of dimension smaller than n . Analysing the data set is then equivalent to recovering this submanifold. Let us make this idea clearer through the two following examples.

1.2 Examples

Stereo vision is perhaps the most straightforward example in computer vision. Pixel matching algorithms often output a cloud of points in \mathbb{R}^3 supposed to belong to the surfaces of the pictured objects. It remains then to find a surface "passing" through this points. Here, $\Omega = \mathbb{R}^3$, \mathcal{P} is the set of points and the searched surface is \mathcal{M} .

Image manifolds arise when considering a set of image features, like spectral, color or texture measures, for classification purposes. If n such numeric features can be extracted for each image then one may well associate a set of N images and a cloud of N points in \mathbb{R}^n , each point representing one image through its features vector. If complex relationships exist between different features within a certain class of images, one may ideally expect the corresponding points to form a submanifold of \mathbb{R}^n . A geometric representation of this submanifold may be used to distinguish this class of images from all the other image points. Here, $\Omega = \mathbb{R}^n$, \mathcal{P} is the set of features vectors and \mathcal{M} is the submanifold of \mathbb{R}^n , hopefully of low dimension, modeling the relations between the features.

1.3 State of the art

In this section, we present an overview of the state of the art on recovering inter-variables relationships and the reader will soon see that our motivation is actually an old one. We survey general statistical methods and also algorithms more specific to computer vision.

The case of variables that are well represented by their mean and variance is covered extensively in statistics. Under some normality and independence assumptions, the issue of deciding whether or not such relations exist is addressed by the *hypothesis testing* methodology [17,14]. In this paper, we are interested exclusively in data sets with more complex distributions where such assumptions are not relevant.

In certain situations, this can be addressed by using the statistical *non-parametric methods* because they do not rely on the normality assumption. There are several such methods for estimating a non-parametric correlation between

two or more variables. Well known such methods are *Spearman r* , *Kendall τ* and *coefficient G* (cf. [14]). Although these tests vary in their interpretation, they are all appropriate only when N is small because of the central limit theorem. Also, they are only interesting when the variables under consideration are well represented by their mean and variance. If higher order statistics are necessary, an alternative is to use *Pearson curves* [10] or *Johnson curves* [13]. These are families of simple distributions that can approximate more complex distributions up to their fourth moment. The original Pearson curves have been defined as the solutions to a differential equation and this is closely related to the method we are going to present. Though, neither Pearson curves nor Johnson curves are appropriate when both the distributions and the relations are expected to be even more complex.

Another traditional approach, exploratory data analysis, consists in identifying relations between several variables by searching systematic patterns. Different methods have been proposed. Most of those assume one of the following. There are certain particular values of the variables that represent well the data: cluster analysis. There exists special linear or polynomial combinations of the variables that yield simple relations: *principal component analysis*, *discriminant function analysis*, *multi-dimensional scaling*, *canonical correlation*, *step-wise linear* and *nonlinear regression*, *projection pursuit*. All these techniques are nicely presented in this textbook [16]. When no distributional assumption is available, *neural networks* are often used for their flexibility, although they relate very closely to standard statistical regressions [5]. Actually, they are often equivalent and, even if it is not always explicit, rely on the same assumptions. In addition, although *neural networks* are very powerful, their actual process is difficult to interpret and this may be a drawback in computer vision where geometric considerations are often crucial.

It is also important to mention a special branch of exploratory data analysis, visual data analysis, which relies upon the visualization of data sets and the ability of humans to detect relevant patterns. The most popular technique is brushing [3, 24]. This is an interactive method which allows a user to select, *i.e.* brush with color, subsets of the data displayed with a certain representation (say, a scatter plot) and observe simultaneously the corresponding recolored subset in another one (say, a histogram). In this technique, the user is also allowed to manually fit models (curves, surfaces) to the observed distributions. This powerful technique can be further enhanced by the use of complementary data representations as well as animation. The weakness of brushing is that it is not automatic, not quantitative and not objective. Though, it is often the solution of last resort.

In computer vision, the very same “fitting” problem is particularly recurrent because typical detection algorithms output clouds of points that are then subject to high level processing.

Most of the methods in computer vision focus on reconstructing shapes in \mathbb{R}^2 or \mathbb{R}^3 and are motivated by geometric interpretations. As in the case of *neural networks*, they are often mathematically equivalent to known statistical methods but provide an effective way to achieve the result when *statistics* may only provide a mathematical interpretation of the underlying regression and optimality prior conditions. Three methodologies can be identified. The first one consist in using graph theory. For instance, one can consider \mathcal{P} as a set of graph

vertices [15, 6] or use normalized cuts [27] for partitioning the ambient space into regions. Another interesting approach, *alpha shapes* [8] defines meaningful subsets of the Delaunay triangulation of \mathcal{P} by balancing convexity and resolution scale measures. The second methodology consists in gathering local perceptual or regularity cues for grouping [34, 1], like good continuation [7], constant curvature [22] or local measures of confidence [26, 31, 25, 12, 9]. The third methodology also results in such local operations but is primarily motivated by mechanical or physical metaphors like in the *dynamic particles* [29]. They often use a variational formulation [18, 23] and explicit or implicit geometric models [30, 33].

However, there is also a more recent interest for methods meant to work in arbitrary dimension. For instance, *tensor voting* [21] has been introduced as a unified formalism for addressing the issues of grouping noisy sets of points into more regular features. It is founded on *tensor calculus*: the multi-dimensional data is first encoded into a tensor, then elements vote in their neighborhood through a convolution-like operation which is most appropriate in Euclidean ambient spaces. This results in a dense tensor map, containing both orientation and confidence informations, from which curves or surfaces can be extracted by a n -linear interpolation. This algorithm is not iterative because the smoothness is imposed by convolution. Note that *tensor voting* was inspired from *vector voting* [12].

We suggest another formalism, founded on *variational calculus*, that is interestingly related to almost all the previous methods. The remainder of the paper is as follows. In Section 2, we present the theoretical and practical aspects of the method. In Section 3, we discuss the benefits of this new approach in different applications. We conclude in Section 4.

2 Theory and implementation

As a preliminary, we provide the reader with some informal intuition about the various ingredients of the method. Then, we propose a mathematical formalization of it and discuss its implementation.

2.1 Some intuition

Suppose that \mathcal{P} is a sparse cloud of points in the plane ($\Omega = \mathbb{R}^2$) and that these points are roughly distributed along a smooth curve. The problem consists in recovering “the” smooth curve \mathcal{M} that “passes through” \mathcal{P} . We simply develop the idea that \mathcal{P} , seen as a subset of Ω , can be transformed continuously into the curve \mathcal{M} . This is achieved through an iterative process in which each point of \mathcal{P} spreads itself in the direction of its neighbouring points. Little by little, each point in \mathcal{P} transforms itself into a short piece of curve oriented toward other points of \mathcal{P} , and grows toward them. Eventually, all those pieces connect to one another so that the final curve is smooth and simply connected. At the same time, outliers are eliminated and the shape is regularized. This “spreading” process transforms continuously \mathcal{P} into the smooth manifold \mathcal{M} .

In the next section, we design an objective cost, or energy, associated with this spreading shape at each time instant of its deformation. The iterative minimization of this energy results in the “right” transformation because the energy is designed for that very purpose. Like in any variational method, the final shape

of \mathcal{M} will correspond to a minimum of the energy. Now, there are obviously two main issues here. The first one is “How to define a proper energy ?” and the second one is “How to represent the evolving shape ?”.

As far as the energy is concerned, it has to reflect the relevant properties of the desired final shape. In variational methods, the energy is often a weighted sum of several energies. Each term contributes to favor or penalize a certain shape property. The contributions are often contradictory and they compete by summation, like kinetic and potential energy do in mechanics. Hopefully, the achieved minimum of the total energy yields a satisfactory balance between each effect. In our problem, we need at least two energy terms. There should definitely be a *data attachment term*. In effect, shapes which contain a lot of points not belonging to \mathcal{P} should be penalized. This prevents the spreading process to add too many points and make \mathcal{M} too “fat”. Symmetrically, shapes which do not contain all the points of \mathcal{P} should be penalized as well. This is because the final result should not miss parts of \mathcal{P} . But this data attachment alone is useless because its minimization results in nothing but $\mathcal{M} = \mathcal{P}$ and that is why it is called the data attachment term. We definitely need a *regularization term* as well. This one should favor better connected and smoother shapes. It will of course be in competition with the data attachment term because \mathcal{P} is not smooth and is not well connected at all. We do not discuss smoothness further for now since the issue is so well known. A way to obtain well connected shapes is to favor convexity. In effect, it is well known that convex shapes are simply connected, *i.e.* contain only one connected component. So, the regularization term favors convexity. Once again, this term alone is useless because minimizing it would really connect all the points of \mathcal{P} . The final result would then be the convex hull of \mathcal{P} and this is not desirable in general. To summarize this paragraph, the total energy will only favor spreading toward other neighbouring points of \mathcal{P} because this is the only way the two terms may actually reach a satisfying agreement.

The representation of the evolving shape is an important issue as well. In effect, *a priori*, no hypothesis is made neither about the dimension of the final shape \mathcal{M} nor its topology. This is a domain where implicit representations are usually superior to shape explicit parametrizations. In solving this issue, we were mostly inspired by [2,20] where a curve in \mathbb{R}^3 is represented by a one-parameter family of concentric tubes of increasing radii. The represented curve is the medial axis of the tubes. If the radius r of the tubes is the parameter of the family then the tubes converge toward the curve when r tends to 0. This is actually a very general approach which is valid regardless of the dimension and topology. The key is to consider neighborhoods (or approximations) of the represented object with increasing tolerance. Note that those neighborhoods are always hyper-surfaces of the ambient spaces, *i.e.* manifolds of dimension $n-1$. For instance, in \mathbb{R}^3 , both concentric tubes and concentric spheres are bi-dimensional although their limits are curves and points. Finally, those hypersurfaces are conveniently encoded as the iso-hypersurfaces of a scalar function defined on Ω . This implicit representation makes it not too hard to formulate the problem as a variational one.

2.2 Formalization

Let $u : \Omega \rightarrow [0,1]$ be a smooth function to be constructed so that the family of hypersurfaces $\mathcal{S}_\alpha = u^{-1}(\alpha)$, $0 < \alpha < 1$, tends to a submanifold \mathcal{M} of Ω when α

tends to 0. In that sense, u can be interpreted as a *weak implicit representation* since

$$\mathcal{M} = \lim_{\alpha \rightarrow 0^+} u^{-1}(\alpha).$$

In particular, the value of u far from \mathcal{P} is 1 and tends to 0 as one approaches \mathcal{P} . We propose to define u as the solution to a variational problem that is naturally related to the reconstruction one. As a preliminary, we shall present some useful integral criteria, along with their Euler-Lagrange equations. Then, we shall see how to combine them and finally, since variational methods are iterative, we shall describe how to initialize u . We start with the case $\Omega = \mathbb{R}^n$.

Ingredients For the moment, consider independently the four non-negative integrals

$$\int_{\mathcal{P}} u^2, \quad \int_{\Omega/\mathcal{P}} (u-1)^2, \quad \int_{\Omega} \nabla^2 u, \quad \text{and} \quad \int_{\Omega} \nabla u \cdot Q_u \nabla u,$$

where Q_u is the projector onto the sub tangent space of Ω corresponding to negative eigenvalues of the Hessian of u . In other words, if $D^2u = P^T D P$, with $D = \text{Diag}(\lambda_i)$ and $P^T P = I$, then $Q_u = P^T G P$ with $G = \text{Diag}(\nu(\lambda_i))$ where $\nu(\mathbb{R}^-) = \{1\}$ and $\nu(\mathbb{R}^{*+}) = \{0\}$. Note that Q_u is symmetric positive.

The motivation for considering the first three integrals is obvious. The minimization of the first one enforces that \mathcal{M} passes through \mathcal{P} . In effect it is null if and only if $u(\mathcal{P}) = \{0\}$. Minimizing the second one prevents \mathcal{M} from passing through other points than those of \mathcal{P} . In effect, it is null if and only if $u(\mathcal{M}/\mathcal{P}) = \{1\}$ a.e. in Ω/\mathcal{P} . Finally, minimizing the third one enforces the smoothness of u hence, to some extent¹, that of \mathcal{S}_α . The Euler-Lagrange equations of these three first integrals are respectively

$$2u, \quad 2(u-1), \quad \text{and} \quad -2\Delta u$$

on the domains where the corresponding integrals are defined and $\mathbf{0}$, *i.e.* the null function, elsewhere in Ω .

As for the fourth integral, it is minimized by non-concave functions but this is not as straightforward and we are going to develop this point further. Note that the motivation for considering convex u 's is that it implies the convexity of its iso-hypersurfaces \mathcal{S}_α (*cf.* previous section) and this is how the connection between neighbouring points is favored. Here is how we minimize this integral. Although Q_u depends upon D^2u , we consider only the first order term in the Euler-Lagrange equation of the integral². It is equal to

$$-2\Delta^- u,$$

¹ *i.e.* to the extent that α is not a *singular* value of u .

² Observe that the dependence of Q_u with respect to D^2u can be thought of as having a contribution due to its eigenvalues λ_i and another one due to the isometry P . The first contribution is null a.e. because the derivative of ν is null a.e.. As for the other dependence, we neglect it here because it introduces third order derivatives in the Euler-Lagrange equation and those are too difficult to evaluate numerically. We have to admit that it is a shortcoming of this method.

where Δ^-u is the sum of the negative eigenvalues of the symmetric matrix D^2u , *i.e.*

$$\Delta^-u = \sum_{i=1, \dots, n} \nu(\lambda_i) \lambda_i \quad (1)$$

This result is obtained by writing

$$\frac{\partial (\nabla u \cdot Q_u \nabla u)}{\partial \nabla u} = (Q_u + Q_u^T) \nabla u = 2Q_u \nabla u$$

and then

$$\begin{aligned} -\nabla \cdot \frac{\partial (\nabla u \cdot Q_u \nabla u)}{\partial \nabla u} &= -2\nabla \cdot Q_u \nabla u \\ &= -2\nabla \cdot Q_u^2 \nabla u && Q_u \text{ is a projector} \\ &= -2Q_u \nabla \cdot Q_u \nabla u && Q_u \text{ is self-adjoint} \\ &= -2P^T G P \nabla \cdot P^T G P \nabla u && \text{by definition} \\ &= -2G P \nabla \cdot G P \nabla u && P^T \text{ is an isometry} \\ &= -2G P \nabla \cdot P \nabla u && G \text{ is a self-adjoint projector} \\ &= -2 \sum_{i=1, \dots, n} \nu(\lambda_i) \frac{\partial^2 u}{\partial p_i^2} \\ &= -2\Delta^-u, \end{aligned}$$

where the p_i 's are the eigenvector of D^2u and Δ^-u is defined in Eq. 1.

The quantity Δ^-u is the negative part of the Laplacian of u or “the negative Laplacian”. Obviously, it vanishes when u is a non-concave function. Hence again, this enforces the convexity of the hypersurfaces of u and consequently the convexity of \mathcal{M} . Of course, the final \mathcal{M} *will not* be globally convex because the four criteria are actually going to compete.

Putting them all together Following, it is important to combine these integrals properly in order to achieve the right balance between each effect. We are going to form a weighted sum of those four integrals depending only upon one parameter, the scale σ . We define the scale as the critical distance between two *just distinguishable points* of \mathcal{P} . This concept from observation theory will serve as a yard stick to “calibrate” our linear combination of contributions.

First, remind that our integrals are not all defined in Ω , so we use an indicator function of \mathcal{P} , *i.e.* $\mathbb{I}_{\mathcal{P}}(\mathcal{P}) = \{1\}$ and $\mathbb{I}_{\mathcal{P}}(\Omega/\mathcal{P}) = \{0\}$. The two first integrals can then be rewritten

$$\int_{\Omega} \mathbb{I}_{\mathcal{P}}(\mathbf{x}) u(\mathbf{x})^2 d\mathbf{x} \quad \text{and} \quad \int_{\Omega} (1 - \mathbb{I}_{\mathcal{P}}(\mathbf{x})) (u(\mathbf{x}) - 1)^2 d\mathbf{x}.$$

Now, we form

$$E = c_1 \int_{\Omega} \mathbb{I}_{\mathcal{P}} u^2 + c_2 \int_{\Omega} (1 - \mathbb{I}_{\mathcal{P}}) (u - 1)^2 + c_3 \int_{\Omega} \nabla^2 u + \int_{\Omega} \nabla u \cdot Q_u \nabla u, \quad (2)$$

and the corresponding Euler-Lagrange equation (up to a factor $\frac{1}{2}$)

$$\partial_u E \equiv c_1 \mathbb{I}_{\mathcal{P}} u + c_2 (1 - \mathbb{I}_{\mathcal{P}})(u - 1) - c_3 \Delta u - \Delta^- u, \quad (3)$$

where the three c_i 's are to be expressed in terms of σ .

For symmetry reasons, it is natural to choose $c_1 = c_2$. Furthermore, consider the case of two isolated just distinguishable points in \mathbb{R} located at coordinates 0 and σ . Setting $c_3 = 0$, because the smoothing must be negligible with respect to the other effects, the equilibrium condition writes $c_2(u - 1) - u'' = 0$, which is a linear differential equation of the second order. It can be integrated by quadrature, supposing that $u(0) = u(\sigma) = 0$. Then, c_2 can be determined thanks to the additional equation $u(\frac{1}{2}\sigma) = \frac{1}{2}$: in effect, the fact that the two points are just distinguishable means in particular that their middle point can be assigned neither to \mathcal{M} nor to Ω/\mathcal{M} .

Finally, we find

$$c_1 = c_2 = \left(\frac{\ln(7 + \sqrt{48})}{\sigma} \right)^2 \approx \frac{6.938}{\sigma^2} \quad \text{and} \quad c_3 = \varepsilon \ll 1,$$

hence the partial differential equation to be solved $\frac{\partial u}{\partial t} = -\partial_u E$ or

$$\boxed{\frac{\partial u}{\partial t} = \frac{\beta^2}{\sigma^2} (-\mathbb{I}_{\mathcal{P}} u + (1 - \mathbb{I}_{\mathcal{P}})(1 - u)) + \varepsilon \Delta u + \Delta^- u,} \quad (4)$$

where $\beta = \ln(7 + \sqrt{48})$ and $\varepsilon \ll 1$.

Non-Euclidean ambient spaces The case where Ω is not Euclidean is important in practice. For instance, curve normal vectors live in \mathcal{S}^1 , surface normal vectors in \mathcal{S}^2 , line directions in \mathcal{P}^1 , color hue in \mathcal{S}^1 and k -uplets of such variables live in products of those spaces. Non-Euclidean variables occur quite often in *computer vision* and, in general, it is not accurate to consider them as taking values in Euclidean spaces. Fortunately, it is rather straightforward to generalize the presented method when Ω can be embedded in a Euclidean space. In that case, one has just to rewrite the previous equations in the tangent plane of Ω as introduced in [4]. Practically, one still solves a PDE using a regular grid and Ω is represented implicitly by its distance function.

Initialization of u One can simply initialize u as follows: $u_0(\mathcal{P}) = \{0\}$ and $u_0(\Omega/\mathcal{P}) = \{1\}$. This has the advantage of being simple and fast but it does not account for repetitions in \mathcal{P} and, since the grid has integer coordinates, rounding effects are important.

Another way that is more robust to outliers and behaves better regarding redundant samples and rounding effects is to set:

$$u_0(\mathbf{x}) = \prod_{\mathbf{p} \in \mathcal{P}} \left(1 - \varepsilon e^{-\frac{(\mathbf{x}-\mathbf{p})^2}{\sigma^2}} \right) \quad (5)$$

Of course, this can be implemented efficiently by considering an approximation of the exponential having a compact support.

2.3 Implementation issues

The function u is sampled over a regular grid of Ω if it is Euclidean or over the Euclidean space of higher dimension in which Ω has been embedded (*cf.* Section. 2.2) otherwise. The equation (4) is discretized using the standard explicit forward scheme for the time derivative (*i.e.* $u_{t+dt} = u_t + (\dots)dt$) and the standard explicit centered schemes for the spacial derivatives (*i.e.* the Hessian of u). The hypersurfaces S_α are extracted by n -linear interpolation. As far as the indicatrix function is concerned, it can be set to

$$\mathbb{I}_{\mathcal{P}}(\mathbf{x}) = e^{-\frac{\text{dist}^2(\mathbf{x}, \mathcal{P})}{\sigma^2}}$$

or any other reasonable approximation. Naturally, the finest possible scale, σ , is determined by the resolution of the grid. It is achieved by taking $\sigma = 1$ (*i.e.* two grid nodes are two distinguishable points) but it can be set to a lower value if needed. The most natural way to work with it is to set $\alpha = 1$ and choose an appropriate grid size. An approximation to the “negative Laplacian” may be computed as

$$\Delta^-u \approx \text{Trace}(N(D^2u))$$

where N is an appropriate polynomial. In effect, we have $\text{Trace}(N(D^2u)) = \sum_{i=1, \dots, n} N(\lambda_i)$ because D^2u is symmetric and thus, D^2u and $N(D^2u)$ share the same eigenvectors. So, the polynomial N has just to “turn off” positive λ_i ’s for a reasonable range of values. This idea has been suggested to us by [30]. Of course, Δ^-u may also be simply computed by diagonalizing D^2u . These approximations of Δ^-u are only valid if Ω is Euclidean, however it is straightforward to extend them if Ω is only embedded in a Euclidean space. Finally, in practice, one can really set $\varepsilon = 0$ because the other spatial centered schemes are already diffusive enough for assuring a regular solution.

3 Applications

In this section, we discuss the benefits and limitations of the proposed technique in two different contexts: *multi-dimensional data analysis* and *skin detection in color images*.

3.1 Application to multi-dimensional data analysis

A simple example will demonstrate how this technique can be helpful for the understanding of multi-dimensional data. When the number of variables is high, *i.e.* greater than 2 or 3, a traditional approach is to study slices or projections of the data with lower dimensions. Typically this approach yields some partial structural information. One limitation of analysing slices of the data is that one may fail to detect existing structural information. This is due in part to the fact that samples that where “almost” lying on the selected slice will be totally invisible. Fig. 2 depicts this situation on a real data set. Let us see how one may solve this with our method. Fig. 1 shows the iterative reconstruction of this data set using Eq.4 and Fig. 3 shows the resulting slice from which patterns can be detected more confidently. This is a rather trivial example since there are only three variables and a three-dimensional scatterplot yields the structural

information. However, if there are more than 3 variables, one may still perform exactly the same reconstruction in the whole dimensionality of the data set. This process cannot be easily visualized. But, once the reconstruction is done, standard analysis tools can be used to detect more reliably structural information in slices, scatterplots or other low-dimensional representations. The higher the number of variables, the more likely the reconstruction will yield information that would otherwise stay invisible. In practice, this technique would be most appropriate when the number of variables is smaller than 10 due to important memory requirements. This point is detailed in Section 4.



Fig. 1. This figure shows the iterative resolution of Eq. 4. The curve and the surface are reconstructed to a certain extent.

3.2 Application to skin detection in color images

Typical algorithms for detecting skin in color images proceed in two steps. First, a local skin color and/or skin texture detection is performed. Then, the geometry of the detected regions is regularized using morphological operators. This section concerns only the first step and deals with the local detection of skin color using the approach described in this paper: 200 images depicting people have been segmented manually and the set \mathcal{P} is formed by the colors, in the *CIE Lab* system, of the millions of pixels corresponding to human skin. Ω is the set of existing colors in the same color system and \mathcal{M} is supposed to approximate the set of colors corresponding only to human skin. After reconstructing \mathcal{M} (cf. Fig. 4), it is then possible to test whether the condition $u(L, a, b) < u_0$ is satisfied, where (L, a, b) is the color of a tested pixel and u_0 is a selected threshold related to the probability of a color to belong to \mathcal{M} . Consequently, this test may be used as a *local skin color detector*. Of course, it is important in this learning approach to eliminate the differences in lighting and camera responses within the learning set and we did this partially. To compensate for differences in lighting conditions, we have applied a simple model of the Von Kries adaptation [11]. The algorithm searches the image for likely representatives of white and black, and uses these values to compute a modified Von Kries adaptation. Although the spectrum of the light source cannot be completely recovered from the image, this model provides good results, as long as the spectrum of the light source is not too wildly skewed or irregular [28]. Fig. 5 shows a comparison of our detection method with two others on both images containing skin and images



Fig. 2. *The origin of this real data set is not mentioned for now otherwise it would clearly identify the authors.* On the left, a cloud of points in \mathbb{R}^3 distributed along a one-dimensional curve and a connected bi-dimensional surface. On the right, the trace of the cloud of points on a selected slice. Although it is clear from the three-dimensional scatterplot that this trace is distributed along a curve, this is not visible on the slice alone.



Fig. 3. The cloud of points of Fig. 2 has been reconstructed, *cf.* also Fig. 1, and its trace over the same slice is shown. By connecting points in the full dimensionality of the data set, one makes it possible to detect structure more reliably. Of course, if the reconstruction is performed on the initial slice, *i.e.* in two dimensions, this information cannot be recovered hence the importance of reconstructing in the full dimensionality, especially when considering more than 2 or 3 variables.

not containing skin. Of course, one of the strength of the presented method is that it can be applied with more than three variables and this can be used in order to further enhance the segmentation. For instance, in Fig. 6, we added a fourth variable which is the variance of $|(L, a, b) - (L_0, a_0, b_0)|$ in a small neighborhood of each pixel, where (L_0, a_0, b_0) is the color of the considered pixel and $|\cdot|$ denotes the Euclidean distance. This measure, noted $v(L, a, b)$ accounts somehow to the presence of texture and this enhances the results in particular on images which do not contain human skin. It is also fair to say that the value of the threshold u_0 is rather arbitrary and no physical interpretation has been attached to it, although it should not be difficult to relate it to a probability.

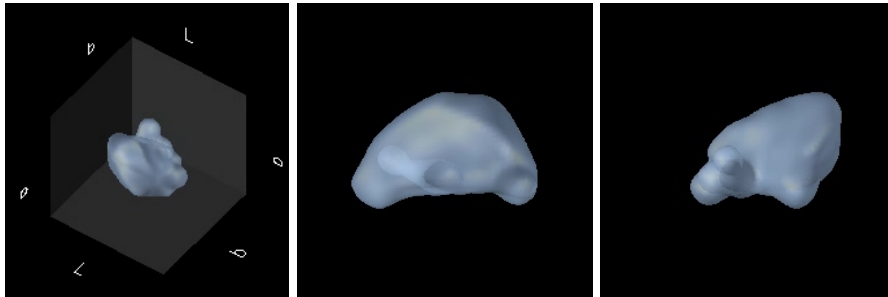


Fig. 4. This figure shows the learned shape of the set of human skin colors in the *CIE Lab* system from different standpoints.

4 Conclusion

In this paper, we have presented a new solution to the fundamental problem of recovering a manifold from a set of points known to belong to it. It is founded on variational calculus and results in a partial differential equation which can be interpreted explicitly as an anisotropic diffusion that “connects” neighbouring points. This is particularly interesting when dealing with non-Euclidean ambient spaces, where convolutions are not easy to implement. Although this technique is surprisingly simple, it gathers various good ideas from the three classes of existing methods surveyed in Section 1. In particular, the variational formulation of *geodesic snakes* [18], the fact that the initial solution is the set of points itself like in *dynamic particles* [29], the balance between convexity and scale like in *alpha shapes* [8], and finally the geometric implicit representation like in the level set methods [33]. All this makes it a very powerful tool in the most difficult situations as shown by our experiments. As for the limitations of the method, one has to be aware that, in practice, it is limited to dimensions smaller than about 10 because of the inherent computational complexity. If a narrow band of “voxels” is used, then the complexity is linear in the size of the reconstructed objects, both in time and memory. But, the involved constant may be very important due to the voxel-based representation. Another difficulty that has not been addressed is the choice of the metric to use when the variables are



Fig. 5. Examples of results of skin color detection using (from the left to the right) the methods proposed in [32], in [19] and in this paper with the variables (L, a, b) .

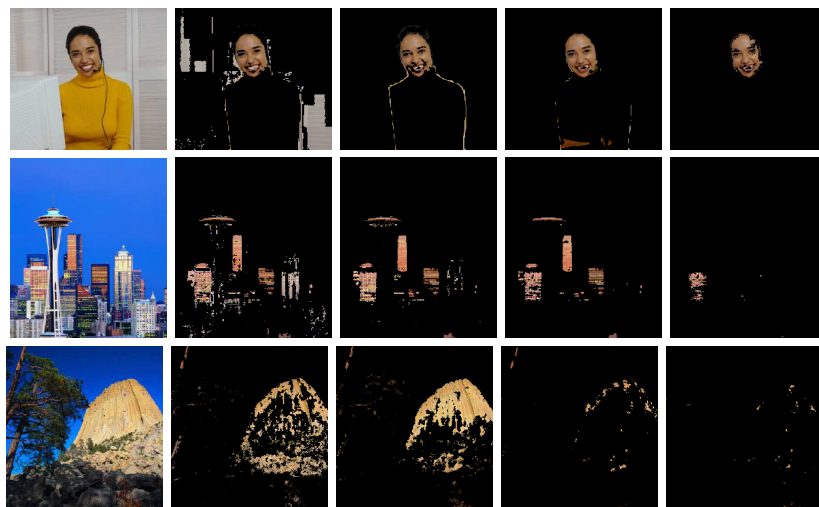


Fig. 6. Examples of results of skin color detection using (from the left to the right) the methods proposed in [32], in [19] and in this paper with the variables (L, a, b) and, in last column, with the variables $(L, a, b, v(L, a, b))$. Notice how the additional variable, related to texture, enhances the results in images that do not contain skin.

of different nature. However, the framework we have presented is certainly an appropriate one to introduce these considerations. One has also to be aware of the two layers of approximations introduced by the weak implicit representation (*cf.* Section 2).

We would like to end this paper with the following general notes. Although the method performs extremely well in practice, there are several theoretical issues which remain to be investigated. Like in any other variational method, the Euler-Lagrange equations provide only necessary conditions to the minimization of the energy and it should be clarified in which cases this is actually not sufficient. Further more, the link to standard statistical methods should be clarified as well. In particular, is the performed reconstruction equivalent to any known statistical regression ? Future work on this technique will focus on answering these questions.

References

1. Narendra Ahuja and Mihran Tuceryan. Extraction of early perceptual structure in dot patterns. integrating region, boundary, and component Gestalt. *CVGIP*, 48(3):304–356, December 1989.
2. Luigi Ambrosio and Halil M. Soner. Level set approach to mean curvature flow in arbitrary codimension. *J. of Diff. Geom.*, 43:693–737, 1996.
3. A. Becker and W. Cleveland. Brushing scatterplots. *Technometrics*, 29(2):127–142, 1987.
4. Marcelo Bertalmio, Li-Tien Cheng, Stanley Osher, and Guillermo Sapiro. Variational problems and partial differential equations on implicit surfaces: The framework and examples in image processing and pattern formation. *UCLA Research Report*, June 2000.
5. C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, Oxford, 1995.
6. J.D. Boissonnat. Representation of objects by triangulating points in 3-d space. In *Proceedings of ICPR*, pages 830–832, 1982.
7. J. Dolan and R. Weiss. Perceptual grouping of curved lines. In *Image Understanding Workshop*, pages 1135–1145, 1989.
8. H. Edelsbrunner and E. P. Mücke. Three-dimensional alpha shapes. *ACM Transactions on Graphics*, 13(1):43–72, 1994.
9. P. Fua and P. Sander. Segmenting unstructured 3d points into surfaces. In *Proceedings of the European Conference on Computer Vision*, pages 676–680, Santa Margherita Ligure, Italy, 1992.
10. L. R. Lamberson G. F. Gruska, K. Mirkhani. *Non-Normal data Analysis*. Garden City, MI: Multiface Publishing., 1967.
11. Tziritas G. Garcia C. Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Transactions on Multimedia*, 1(3):264–277, 1999.
12. G. Guy. Inference of multiple curves and surfaces from sparse data. Technical Report 96-345, USC IRIS, Ph.D. thesis, 1996.
13. G. Hahn and S. Shapiro. *Statistical Models in Engineering*. John Wiley & Sons, 1967.
14. S. Hays, W. Statistics, and Y. Holt. *Statistics*. Holt, Rinehart and Winston, New York, 1981.
15. Hugues Hoppe, Tony DeRose, Tom Duchamp, John McDonald, and Werner Stuetzle. Surface reconstruction from unorganized points. *Computer Graphics*, 26(2):71–78, 1992.
16. StatSoft Inc. *Electronic Statistics Textbook*, volume <http://www.statsoftinc.com/textbook/stathome.html>. Tulsa, StatSoft, 2001.

17. S. Kachigan. *Statistical Analysis*. Radius Press, 1986.
18. M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. In *Proceedings of the First International Conference on Computer Vision*, pages 259–268, London, June 1987.
19. J. M. G. Lammens. *A computational model of color perception and color naming (Ph.D. thesis)*. University of New York, Buffalo, 1994.
20. L. Lorigo, O. Faugeras, W.E.L. Grimson, R. Keriven, R. Kikinis, and C-F. Westin. Co-dimension 2 geodesic active contours for mra segmentation. In *Proceedings of the International Conference on Information Processing in Medical Imaging*, pages 126–139, June 1999.
21. Gérard Medioni, Mi-Suen Lee, and Chi-Keung Tang. *A Computational Framework for Segmentation and Grouping*. Elsevier, 2000.
22. P. Parent and S. Zucker. Trace inference, curvature consistency and curve detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2(8), 1989.
23. Tomaso Poggio and Federico Girosi. A theory of networks for approximation and learning. Technical Report 1140, AIM, 1989.
24. David. A. Rabenhorst. Interactive exploration of multidimensional data. In *Proceedings of the SPIE Symposium on Electronic Imaging*, volume 2179, pages 277–286, 1994.
25. S. Sarkar, K. Boyer, and I. inference. and management of spatial information using bayesian networks: perceptual organization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(3), 1993.
26. A. Shashua and S. Ullman. Structural saliency: the detection of globally salient structures using a locally connected network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 7(1):90–94, 1988.
27. J. Shi and J. Malik. Normalized cuts and image segmentation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, 1997.
28. K. Sobottka and I. Pitas. Extraction of facial regions and features using color and shape information. In *Proceedings of the International Conference on Pattern Recognition*, volume 3, pages C421–C425, Vienna, Austria, 1996.
29. R. Szelisky, D. Tonnesen, and D. Terzopoulos. Modelling surfaces of arbitrary topology with dynamic particles. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 82–87, New York, June 1993.
30. Gabriel Taubin. A signal processing approach to fair surface design. *Computer Graphics*, 29(Annual Conference Series):351–358, 1995.
31. K.K. Thornber and R.L. Williams. Analytic solution of stochastic completion fields. In *Proceedings of SCV*, page 11B Segmentation and Grouping II, 1995.
32. G. Wyszecky and W. S. Stiles. *Color science: Concepts and methods, quantitative data and formulae*. John Wiley and Sons, New York, 1982.
33. H. Zhao, S. Osher, B. Merriman, and M. Kang. Implicit and non-parametric shape reconstruction from unorganized points using variational level set method. *Computer Vision and Image Understanding*, 80(3):295–319, 2000.
34. S.W. Zucker and R.A. Hummel. Toward a low-level description of dot clusters: Labeling edge, interior, and noise points. In *Proceedings of CGIP*, volume 9, pages 213–233, 1979.