7th European Conference on Computer Vision,
Copenhagen, May 28–31, 2002, © Springer, to appear.

Nonlinear Shape Statistics in Mumford–Shah Based Segmentation

Daniel Cremers, Timo Kohlberger, and Christoph Schnörr

Computer Vision, Graphics and Pattern Recognition Group
Department of Mathematics and Computer Science
University of Mannheim, D–68131 Mannheim, Germany
{cremers,tiko,schnoerr}@uni-mannheim.de
http://www.cvgpr.uni-mannheim.de

Abstract. We present a variational integration of nonlinear shape statistics into a Mumford–Shah based segmentation process. The nonlinear statistics are derived from a set of training silhouettes by a novel method of density estimation which can be considered as an extension of kernel PCA to a stochastic framework.
The idea is to assume that the training data forms a Gaussian distribution after a nonlinear mapping to a potentially higher–dimensional feature space. Due to the strong nonlinearity, the corresponding density estimate in the original space is highly non–Gaussian. It can capture essentially arbitrary data distributions (e.g. multiple clusters, ring– or banana–shaped manifolds).
Applications of the nonlinear shape statistics in segmentation and tracking of 2D and 3D objects demonstrate that the segmentation process can incorporate knowledge on a large variety of complex real–world shapes. It makes the segmentation process robust against misleading information due to noise, clutter and occlusion.

Keywords: Segmentation, shape learning, nonlinear statistics, density estimation, Mercer kernels, variational methods, probabilistic kernel PCA

1 Introduction

One of the challenges in the field of image segmentation is the incorporation of prior knowledge on the shape of the segmenting contour. The general idea is to learn the possible shape deformations of an object statistically from a set of training shapes, and to then restrict the contour deformation to the subspace of familiar shapes during the segmentation process. For the problem of segmenting a known object — such as an anatomical structure in a medical image — this approach has been shown to drastically improve segmentation results [15, 8].

Although the shape prior can be quite powerful in compensating for misleading information due to noise, clutter and occlusion in the input image, most approaches are limited in their applicability to more complicated shape variations of real–world objects. The permissible shapes are assumed to form a multivariate Gaussian distribution, which essentially means that all possible shape deformations correspond to linear combinations of a set of eigenmodes, such as those given by principal component analysis (cf. [14, 4, 15]). In particular, this means that for any two permissible shapes, the entire sequence of shapes obtained by a linear morphing of the two shapes is permissible as well.

Once the set of training shapes exhibits highly nonlinear shape deformations — such as different 2D views of a 3D object — one finds distinct clusters in shape space corresponding to the stable views of an object. Moreover, each of the clusters may by itself be quite non–Gaussian. The Gaussian hypothesis will then result in a mixing of the different views, and the space of accepted shapes will be far too large for the prior to sensibly restrict the contour deformation.

A number of models have been proposed to deal with nonlinear shape variation. However, they often suffer from certain drawbacks. Some involve a complicated model construction procedure [3]. Some are supervised in the sense that they assume prior knowledge on the structure of the nonlinearity [12]. Others require prior classification with the number of classes to be estimated or specified beforehand and each class being assumed Gaussian [13, 5]. And some cannot be easily extended to shape spaces of higher dimension [11].

In the present paper we present a density estimation approach which is based on Mercer kernels [6] and which does not suffer from any of the mentioned drawbacks. In Section 2 we review the variational integration of a linear shape prior into Mumford–Shah based segmentation. In Section 3 we present the nonlinear density estimate which was first introduced in [7]. We discuss its relation to kernel PCA and to the classical Parzen estimator, give estimates of the involved parameters and illustrate its application to artificial 2D data and to silhouettes of real objects. In Section 4 this nonlinear shape prior is integrated into segmentation. We propose a variational integration of similarity invariance. Numerous examples of segmentation with and without shape prior on static images and tracking sequences finally confirm the properties of the nonlinear shape prior: it can encode very different shapes and generalizes to novel views without blurring or mixing different views. Furhermore, it improves segmentation by reducing the dimension of the search space, by stabilizing with respect to clutter and noise and by reconstructing the contour in areas of occlusion.

2 Statistical Shape Prior in Mumford–Shah Segmentation

In [8] we presented a variational integration of statistical shape knowledge in a Mumford–Shah based segmentation. We suggested modifications of the Mumford–Shah functional and its cartoon limit [17] which facilitate the implementation of the segmenting contour as a parameterized spline curve:

$$C_{\boldsymbol{z}} : [0,1] \to \Omega \subset \mathbb{R}^2 \,, \qquad C_{\boldsymbol{z}}(s) = \sum_{n=1}^{N} \begin{pmatrix} x_n \\ y_n \end{pmatrix} B_n(s) \,, \tag{1}$$

where B_n are quadratic B–spline basis functions [10], and $\boldsymbol{z} = (x_1, y_1, \dots, x_N, y_N)^t$ denotes the control points. Shape statistics can then be obtained by estimating the distribution of the control point vectors corresponding to a set of contours which were extracted from binary training images.

In the present paper we focus on significantly improving the shape statistics. We will therefore restrict ourselves to the somewhat simpler cartoon limit of the modified Mumford–Shah functional. The segmentation of a given grey value input image $f : \Omega \rightarrow [0, 255]$ is obtained by minimizing the energy functional

$$E_{MS}(C, u_o, u_i) = \frac{1}{2} \int_{\Omega_i} (f - u_i)^2 dx + \frac{1}{2} \int_{\Omega_o} (f - u_o)^2 dx \ + \ \nu \, \mathcal{L}(C) \qquad (2)$$

with respect to u_o, u_i and the segmenting contour C. This enforces a segmentation into an inside region Ω_i and an outside region Ω_o with piecewise constant grey values u_i and u_o, such that the variation of the grey value is minimal within each region.[1]

In [8] we proposed to measure the length of the contour by the squared $\mathcal{L}_2$–norm $\mathcal{L}(C) = \int_0^1 \left(\frac{dC}{ds}\right)^2 ds$, which is more adapted to the implementation of the contour as a closed spline curve than the usual $\mathcal{L}_1$–norm, because it enforces an equidistant spacing of control points. Beyond just minimizing the length of the contour, one can minimize a shape energy $E_{shape}(C)$, which measures the dissimilarity of the given contour with respect to a set of training contours. Minimizing the total energy

$$E(C, u_o, u_i) = E_{MS}(C, u_o, u_i) \ + \ \alpha \, E_{shape}(C) \qquad (3)$$

will enforce a segmentation which is based on both the input image and the similarity to a set of training shapes.

In order to study the interaction between statistical shape knowledge and image grey value information we restricted the shape statistics in [8] to a common model by assuming the training shapes to form a multivariate Gaussian distribution in shape space. This corresponds to a quadratic shape energy on the spline control point vector $\boldsymbol{z}$:

$$E_{shape}\,(C_{\boldsymbol{z}}) \ = \ (\boldsymbol{z} - \boldsymbol{z}_0)^t \, \Sigma^{-1} \, (\boldsymbol{z} - \boldsymbol{z}_0)\,, \qquad (4)$$

where $\boldsymbol{z}_0$ denotes the mean control point vector and Σ the covariance matrix after appropriate regularization [8]. The effect of this shape energy in dealing with clutter and occlusion is exemplified in Figure 1. For the input image f of a partially occluded hand, we performed a gradient descent to minimize the total energy (3) without ($\alpha = 0$) and with ($\alpha > 0$) shape prior.

3 Density Estimation in Feature Space

Unfortunately, the linear shape statistics (4) are limited in their applicability to more complicated shape deformations. As soon as the training shapes form

[1] The underlying piecewise–constant image model can easily be generalized to incorporate higher–order grey value statistics [27] or edge information [18]. In this paper, however, we focus on modeling shape statistics and therefore do not consider these possibilities.

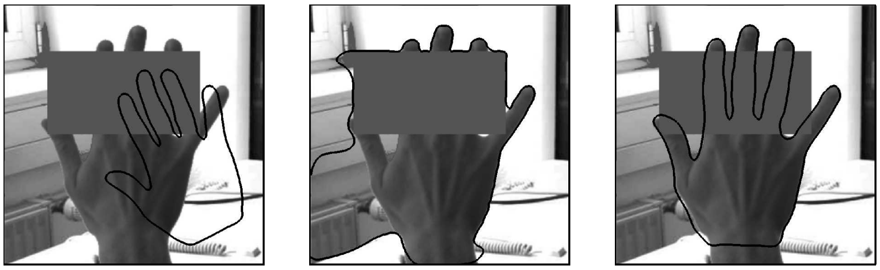


Fig. 1. Segmentation with **linear** shape prior on an image of a partially occluded hand: initial contour (left), segmentation without shape prior (center), and segmentation with shape prior (right). The statistical shape prior compensates for misleading information due to noise, clutter and occlusion. Integration into the variational framework effectively reduces the dimension of the search space and enlarges the region of convergence.

distinct clusters in shape space — such as those corresponding to the stable views of a 3D object — or the shapes of a given cluster are no longer distributed according to a hyperellipsoid, the Gaussian shape prior tends to mix classes and blur details of the shape information in such a way that the resulting shape prior is no longer able to effectively restrict the contour evolution to the space of familiar shapes.

In the following we present an extension of the above method which incorporates a strong nonlinearity at almost no additional effort. Essentially we propose to perform a density estimation not in the original space but in the feature space of nonlinearly transformed data. The nonlinearity enters in terms of Mercer kernels [6], which have been extensively used in the classification and support vector community [1, 2], but which have apparently been studied far less in the field of density estimation. In the present section we present the method of density estimation, discuss its relation to kernel principal component analysis (kernel PCA) [23] and to the Parzen estimator [20, 19], and propose estimates of the involved parameters. Finally we illustrate the density estimate in applications to artificial 2D data and to 200–dimensional data corresponding to silhouettes of real–world training shapes.

3.1 Gaussian Density in Kernel Space

Let $\boldsymbol{z}_1, \dots, \boldsymbol{z}_m \in \mathbb{R}^n$ be a given set of training data. We propose to map the data by a nonlinear function ϕ to a potentially higher–dimensional space Y. Denote a mapped point after centering with respect to the training points by

$$\tilde{\phi}(\boldsymbol{z}) \;:=\; \phi(\boldsymbol{z}) - \phi_0 \;=\; \phi(\boldsymbol{z}) - \frac{1}{m}\sum_{i=1}^{m} \phi(\boldsymbol{z}_i)\,, \tag{5}$$

and let the Mercer kernel [6] $k(\boldsymbol{x}, \boldsymbol{y}) := (\phi(\boldsymbol{x}), \phi(\boldsymbol{y}))$ denote the corresponding scalar product for $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$. Denote the centered kernels by

$$\tilde{k}(\boldsymbol{x},\boldsymbol{y}) := \left(\tilde{\phi}(\boldsymbol{x}), \tilde{\phi}(\boldsymbol{y})\right) = k(\boldsymbol{x},\boldsymbol{y}) - \frac{1}{m}\sum_{k=1}^{m}(k(\boldsymbol{x},\boldsymbol{z}_k)+k(\boldsymbol{y},\boldsymbol{z}_k)) + \frac{1}{m^2}\sum_{k,l=1}^{m} k(\boldsymbol{z}_k,\boldsymbol{z}_l). \tag{6}$$

We estimate the distribution of the *mapped* training data by a Gaussian probability density in the space Y — see Figure 2. The corresponding energy is given by the negative logarithm of the probability, and can be considered as a measure of the dissimilarity between a point $\boldsymbol{z}$ and the training data:

$$E_\phi(\boldsymbol{z}) \;=\; \tilde{\phi}(\boldsymbol{z})^t \; \Sigma_\phi^{-1} \tilde{\phi}(\boldsymbol{z}) \,. \tag{7}$$

In general the covariance matrix Σ_ϕ is not invertible. We therefore regularize it by replacing the zero eigenvalues by a constant $\lambda_\perp$:

$$\Sigma_\phi = V \Lambda V^t \;+\; \lambda_\perp \left(I \;-\; V\,V^t\right) \,, \tag{8}$$

where Λ denotes the diagonal matrix of nonzero eigenvalues $\lambda_1 \leq \ldots \leq \lambda_r$ and V is the matrix of the corresponding eigenvectors $V_1, \ldots, V_r$. By definition of Σ_ϕ, these eigenvectors lie in the span of the mapped training data:

$$V_k = \sum_{i=1}^{m} \alpha_i^k \, \tilde{\phi}(\boldsymbol{z}_i) \,, \qquad 1 \leq k \leq r \,. \tag{9}$$

In [23] it is shown that the eigenvalues λ_k of the covariance matrix correspond (up to the factor m) to the nonzero eigenvalues of the $m \times m$–matrix K with entries $K_{ij} = \tilde{k}(\boldsymbol{z}_i, \boldsymbol{z}_j)$, and that the expansion coefficients $\{\alpha_i^k\}_{i=1,\ldots,m}$ in (9) form the components of the k–th eigenvector of K.

Inserting (8) splits energy (7) into two terms:

$$E_\phi(\boldsymbol{z}) \;=\; \sum_{k=1}^{r} \lambda_k^{-1} \left(V_k, \tilde{\phi}(\boldsymbol{z})\right)^2 \quad + \quad \lambda_\perp^{-1} \left(|\tilde{\phi}(\boldsymbol{z})|^2 \;-\; \sum_{k=1}^{r} \left(V_k, \tilde{\phi}(\boldsymbol{z})\right)^2 \right) \,. \tag{10}$$

With expansion (9), we obtain the final expression for our energy:

$$E_\phi(\boldsymbol{z}) = \sum_{k=1}^{r} \left(\sum_{i=1}^{m} \alpha_i^k \, \tilde{k}(\boldsymbol{z}_i, \boldsymbol{z}) \right)^2 \cdot \left(\lambda_k^{-1} - \lambda_\perp^{-1}\right) \;+\; \lambda_\perp^{-1} \cdot \tilde{k}(\boldsymbol{z}, \boldsymbol{z}) \,. \tag{11}$$

As in the case of kernel PCA, the nonlinearity ϕ only appears in terms of the kernel function. This allows to specify an entire family of possible nonlinearities by the choice of the associated kernel. For all our experiments we used the Gaussian kernel:

$$k(\boldsymbol{x},\boldsymbol{y}) = \frac{1}{(2\pi\sigma^2)^{\frac{n}{2}}} \exp\left(-\frac{||\boldsymbol{x}-\boldsymbol{y}||^2}{2\sigma^2} \right) \,, \quad \boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^n \,. \tag{12}$$

We refer to Section 3.4 for a justification of this choice.

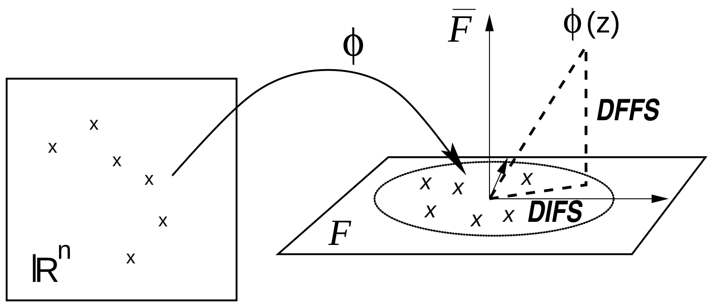


Fig. 2. Nonlinear mapping into $Y = F \bigoplus \overline{F}$ and the distances DIFS and DFFS.

3.2 Relation to Kernel PCA

Just as in the linear case (cf. [16]), the regularization (8) of the covariance matrix causes a splitting of the energy into two terms (10), which can be considered as a *distance in feature space* (DIFS) and a *distance from feature space* (DFFS) — see Figure 2. For the purpose of pattern reconstruction in the framework of kernel PCA, it was suggested to minimize a reconstruction error [22], which is identical with the DFFS. This procedure is based on the assumption that the entire plane spanned by the mapped training data corresponds to acceptable patterns. However, this is not a valid assumption: already in the linear case, moving too far along an eigenmode will produce patterns which have almost no similarity to the training data, although they are still accepted by the hypothesis. Moreover, the distance DFFS is not based on a probabilistic model. In contrast, energy (11) is derived from a Gaussian probability distribution. It minimizes both the DFFS and the DIFS; the latter can be considered a Mahalanobis distance in feature space.

3.3 On the Regularization of the Covariance Matrix

A regularization of the covariance matrix in the case of kernel PCA — as done in (8) — was first proposed in [7] and has also been suggested more recently in [24]. The choice of the parameter $\lambda_\perp$ is not a trivial issue. For the linear case, such regularizations of the covariance matrix have also been proposed [4, 16, 21, 25, 9]. There [16, 25], the constant $\lambda_\perp$ is estimated as the mean of the replaced eigenvalues by minimizing the Kullback–Leibler distance of the corresponding densities. However, we believe that this is not the appropriate regularization of the covariance matrix. The Kullback–Leibler distance is supposed to measure the error with respect to the correct density, which means that the covariance matrix calculated from the training data is assumed to be the correct one. But this is not the case because the number of training points is limited. For essentially the same reason this approach does not extend to the nonlinear case considered here: depending on the type of nonlinearity ϕ, the covariance matrix is potentially infinite–dimensional such that the mean over all replaced eigenvalues will be zero. As in the linear case [9], we therefore propose to choose $0 < \lambda_\perp < \lambda_r$, which means that unfamiliar variations from the mean are less probable than the smallest variation observed on the training set. In practice we fix $\lambda_\perp = \lambda_r/2$.

3.4 Relation to Classical Density Estimation

Why should the training data after a nonlinear mapping corresponding to the kernel (12) be distributed according to a Gaussian density? The final expression of the density estimate (11) resembles the well–known Parzen estimator [20, 19], which estimates the density of a distribution of training data by summing up the data points after convolution with a Gaussian (or some other kernel function).

In fact, the energy associated with an *isotropic* (spherical) Gaussian distribution in feature space is (up to normalization) equivalent to a Parzen estimator in the original space. In the notations of (5) and (6), this energy is given by the Euclidean feature space distance

$$E_{sphere}(z) = |\tilde{\phi}(z)|^2 = \tilde{k}(z,z) = -\frac{2}{m}\sum_{i=1}^{m} k(\boldsymbol{z},\boldsymbol{z}_i) + \text{const.}$$

Up to scaling and a constant, this is the Parzen estimator.

Due to the regularization of the covariance matrix in (8), the energy associated with the more general anisotropic feature space Gaussian (7) contains a (dominant) isotropic component given by the last term in (11). We believe that this connection to the Parzen estimator justifies the assumption of a Gaussian in feature space and the choice of localized kernels such as (12).

Numerical simulations show that the remaining anisotropic component in (11) has an important influence. However, a further investigation of this influence is beyond the scope of this paper.

3.5 On the Choice of the Hyperparameter σ

The last parameter to be fixed in the proposed density estimate is the hyperparameter σ in (12). Let μ be the average distance between two neighboring data points:

$$\mu^2 := \frac{1}{m}\sum_{i=1}^{m} \min_{j \neq i} |\boldsymbol{z}_i - \boldsymbol{z}_j|^2 \,. \tag{13}$$

In order to get a smooth energy landscape, we propose to choose σ in the order of μ. In practice we used

$$\sigma = 1.5\,\mu \tag{14}$$

for most of our experiments. We chose this somewhat heuristic measure μ for the following favorable properties: μ is insensitive to the distance of clusters as long as each cluster contains more than one data point, μ scales linearly with the data points, and μ is robust with respect to the individual data points.

Given outliers in the training set, i.e. clusters with only one sample, one could refer to the more robust $\mathcal{L}_1$–norm or more elaborate robust estimators in (13). Since this is not the focus of our contribution, it will not be pursued here.

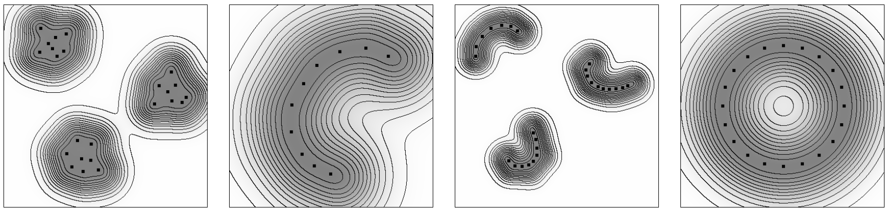


Fig. 3. Density estimate (7) for artificial 2D data. Distributions of variable shape are well estimated by the Gaussian hypothesis in feature space. We used the kernel (12) with $\sigma = 1.5\mu$ — see definition (13).

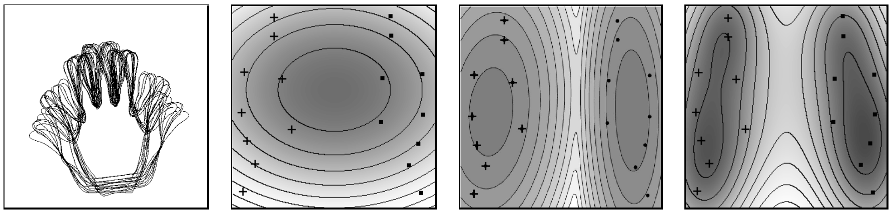
3.6 Density Estimate for Silhouettes of 2D and 3D Objects

Although energy (7) is quadratic in the space Y of mapped points, it is generally not convex in the original space, showing several minima and level lines of essentially arbitrary shape. Figure 3 shows artificial 2D data and the corresponding lines of constant energy $E_\phi(\boldsymbol{z})$ in the original space.

For a set of binarized views of objects we automatically fit a closed quadratic spline curve around each object. All spline curves have N=100 control points, set equidistantly. The polygons of control points $\boldsymbol{z} = (x_1, y_1, x_2, y_2, \ldots, x_N, y_N)$ are aligned with respect to translation, rotation, scaling and cyclic permutation. This data was used to determine the density estimate $E_\phi(\boldsymbol{z})$ in (11).

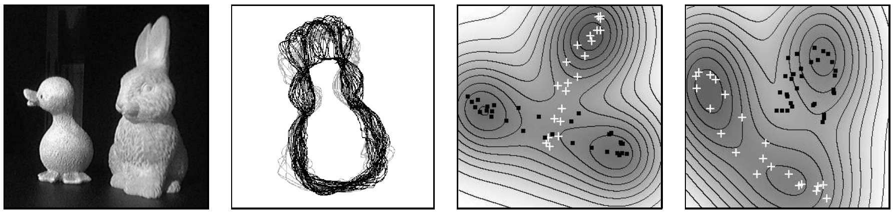
For the visualization of the density estimate and the training shapes, all data was projected onto two of the principal components of a linear PCA. Note that due to the projection, this visualization only gives a very rough sketch of the true distribution in the 200–dimensional shape space.

Figure 4 shows density estimates for a set of right hands and left hands. The estimates correspond to the hypotheses of a simple Gaussian in the original space, a mixture of Gaussians and a Gaussian in feature space. Although both



Aligned contours | Simple Gaussian | Mixture model | Feature space Gaussian

Fig. 4. Model comparison: density estimates for a set of left (+) and right (•) hands, projected onto the first two principal components. **From left to right:** aligned contours, simple Gaussian, mixture of Gaussians, Gaussian in feature space (7). Both the mixture model and the Gaussian in feature space capture the two–class structure of the data. However, the estimate in feature space is unsupervised and produces level lines which are not necessarily ellipses.



two objects | aligend contours | projection onto 1st and 2nd principal comp. | projection onto 2nd and 4th principal comp.

Fig. 5. Density estimate for views of two 3D objects: the training shapes of the duck (white +) and the rabbit (black •) form distinct clusters in shape space which are well captured by the energy level lines shown in apropriate 2D projections.

the mixture model and our estimate in feature space capture the two distinct clusters, there are several differences: firstly the mixture model is supervised — the number of classes and the class membership must be known — and secondly it only allows level lines of elliptical shape, corresponding to the hypothesis that each cluster by itself is a Gaussian distribution. The model of a Gaussian density in feature space does not assume any prior knowledge and produces level lines which capture the true distribution of the data even in the case that it does not correspond to a sum of hyperellipsoids.

This is demonstrated on a set of training shapes which correspond to different views of two 3D objects. Figure 5 shows the two objects, their contours after alignment and the level lines corresponding to the estimated energy density (7) in appropriate 2D projections.

4 Nonlinear Shape Statistics in Mumford–Shah Based Segmentation

4.1 Minimization by Gradient Descent

Energy (7) measures the similarity of a shape $C(\boldsymbol{z})$ parameterized by a control point vector $\boldsymbol{z}$ with respect to a set of training shapes. For the purpose of segmentation, we combine this energy as a shape energy E_{shape} with the Mumford–Shah energy (2) in the variational approach (3).

The total energy (3) must be simultaneously minimized with respect to the control points defining the contour and with respect to the segmenting grey values u_i and u_o. Minimizing the modified Mumford–Shah functional (2) with respect to the contour C (for fixed u_i and u_o) results in the evolution equation

$$\frac{\partial C(s,t)}{\partial t} = -\frac{dE_{MS}}{dC} = -(e_s^+ - e_s^-) \cdot \boldsymbol{n}_s + \nu \, \frac{d^2 C}{ds^2} \,, \tag{15}$$

where the terms e_s^+ and e_s^- denote the energy density $e_s^{+/-} = (f - u_{i/o})^2$, inside and outside the contour $C(s)$, respectively, and $\boldsymbol{n}_s$ denotes the outer normal vector on the contour. The two constants u_i and u_o are updated in alternation with the contour evolution to be the mean grey value of the adjoining regions Ω_i and Ω_o. The contour evolution equation (15) is transformed into an evolution equation for the control points $\boldsymbol{z}$ by introducing definition (1) of the contour as a spline curve. By discretizing on a set of nodes s_j along the contour we obtain a set of coupled linear differential equations. Solving for the x–coordinate of the i-th control point and including the term induced by the shape energy we obtain:

$$\frac{dx_i(t)}{dt} = \left(\mathbf{B}^{-1}\right)_{ij}\left[(e^+_{s_j} - e^-_{s_j})n_x - \nu(x_{j-1} - 2x_j + x_{j+1})\right] - \alpha\left[\frac{dE_{shape}(\boldsymbol{z})}{d\boldsymbol{z}}\right]_{2i-1}, \quad (16)$$

where summation over j is assumed. The cyclic tridiagonal matrix $\mathbf{B}$ contains the spline basis functions evaluated at these nodes, and n_x denotes the x–component of the normal vector on the contour. An expression similar to (16) holds for the y–coordinate of the i–th control point.

The three terms in the evolution equation (16) can be interpreted as follows: the first term pulls the contour towards the object in the image, thus minimizing the grey value variance in the adjoining regions. The second term pulls each control point towards its respective neighbors, thus minimizing the length of the contour. And the third term pulls the control point vector towards the nearest cluster of probable shapes, which minimizes the shape energy.

4.2 Invariance in the Variational Framework

By construction, the density estimate (7) is not invariant with respect to translation, scaling and rotation of the shape $C(\boldsymbol{z})$. We therefore propose to eliminate these degrees of freedom in the following way: since the training shapes were aligned to their mean shape $\boldsymbol{z}_0$ with respect to translation, rotation and scaling and then normalized to unit size, we shall do the same to the argument $\boldsymbol{z}$ of the shape energy before applying our density estimate E_ϕ.

We therefore define the shape energy by

$$E_{shape}(\boldsymbol{z}) = E_\phi(\tilde{\boldsymbol{z}}), \quad \text{with } \tilde{\boldsymbol{z}} = \frac{R_\theta\, \boldsymbol{z_c}}{|R_\theta\, \boldsymbol{z_c}|}, \quad (17)$$

where $\boldsymbol{z_c}$ denotes the control point vector after centering, and R_θ denotes the optimal rotation of the control point polygon $\boldsymbol{z_c}$ with respect to the mean shape $\boldsymbol{z}_0$. We will not go into details about the derivation of R_θ. A similar derivation can be found in [26]. The final result is given by the formula:

$$\tilde{\boldsymbol{z}} = \frac{M\,\boldsymbol{z_c}}{|M\,\boldsymbol{z_c}|}, \quad \text{with } M = I_n \otimes \begin{pmatrix} \boldsymbol{z}_0^t\,\boldsymbol{z_c} & -\boldsymbol{z}_0 \times \boldsymbol{z_c} \\ \boldsymbol{z}_0 \times \boldsymbol{z_c} & \boldsymbol{z}_0^t\,\boldsymbol{z_c} \end{pmatrix},$$

where $\otimes$ denotes the Kronecker product and $\boldsymbol{z}_0 \times \boldsymbol{z_c} = \boldsymbol{z}_0^t R_{\pi/2} \boldsymbol{z_c}$.

The last term in the contour evolution equation (16) is now calculated by applying the chain rule:

$$\frac{dE_{shape}(\boldsymbol{z})}{d\boldsymbol{z}} = \frac{dE_\phi(\tilde{\boldsymbol{z}})}{d\tilde{\boldsymbol{z}}} \cdot \frac{d\tilde{\boldsymbol{z}}}{d\boldsymbol{z}}.$$

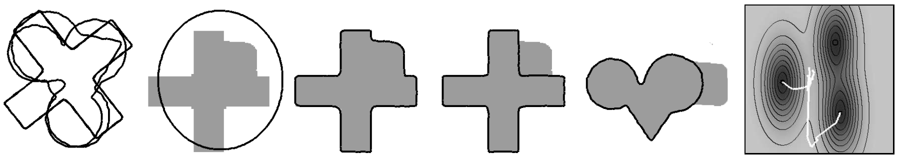


Fig. 6. Segmentation of artificial objects (left) with nonlinear shape prior: the *same* prior can encode very different shapes. Introduction of the shape prior upon stationarity of the contour causes the contour to evolve normal to the level lines of constant energy into the nearest local minimum, as indicated by the white curves in the projected density estimate (right).

Since this derivative can be calculated analytically, no additional parameters enter the above evolution equation to account for scale, rotation and translation.

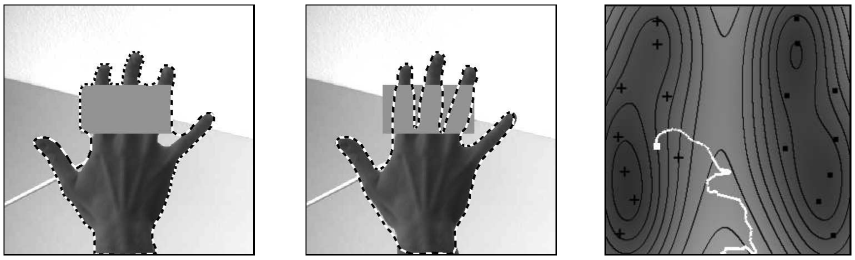
Other authors often propose to explicitly model a translation, an angle and a scale and minimize with respect to these quantities (e.g. by gradient descent). In our opinion this has several drawbacks: firstly it introduces four additional parameters, which makes numerical minimization more complicated — parameters to balance the gradient descent must be chosen. Secondly this approach mixes the degrees of freedom corresponding to scale, rotation and shape deformation. And thirdly potential local minima may be introduced by the additional parameters. On several segmentation tasks we were able to confirm these effects by comparing the two approaches.

Since there exists a similar closed form solution for the optimal alignment of two polygons with respect to the affine group [26], the above approach could be extended to define a shape prior which is invariant with respect to affine transformations. However, we do not elaborate this for the time being.

4.3 Coping with Multiple Objects and Occlusion

Compared to the linear case (4), the nonlinear shape energy (7) is no longer convex. In general it has several minima corresponding to different clusters of familiar contours. Minimization by gradient descent will end up in the nearest local minimum. In order to obtain a certain independence of the shape prior from the initial contour, we propose to first minimize the image energy E_{MS} by itself until stationarity and to then include the shape prior E_{shape}, after performing the cyclic permutation of control points which — given the optimal similarity transformation — best aligns the current contour with the mean of the training shapes. This approach guarantees that we will extract as much information as possible from the image before "deciding" which of the different clusters of accepted shapes the obtained contour resembles most.

Figure 6 shows a simple example of three artificial objects. The shape prior (17) was constructed on the three aligned silhouettes shown on the left. The next images show the initial contour for the segmentation of a partially occluded image of object 1, the final segmentation without prior knowledge, the final



Segmentation without prior Segmentation with prior Projected density estimate

Fig. 7. Segmentation with a nonlinear shape prior containing right (+) and left (•) hands — shown in the projected energy plot on the right. The input image is a right hand with an occlusion. After the Mumford–Shah segmentation becomes stationary (left image), the nonlinear shape prior is introduced, and the contour converges towards the final segmentation (center image). The contour evolution in its projection is visualized by the white curve in the energy density plot (right). Note that the final segmentation (white box) does not correspond to any of the training silhouettes, nor to the minimum (i.e. the most probable shape) of the respective cluster.

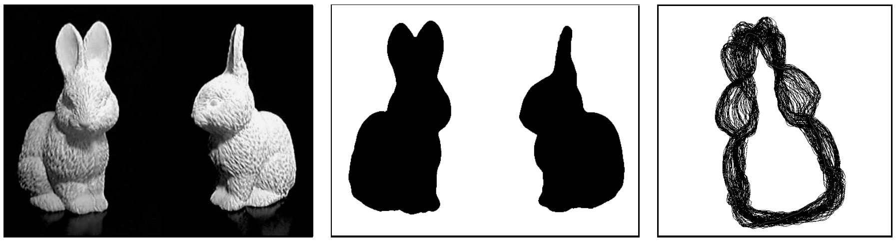
segmentation after introducing the prior, and a segmentation with *the same* prior for an occluded version of object 2.

The final image (Figure 6, right) shows the training shapes and the density estimate in a projection onto the first two axes of a PCA. The white curves correspond to the path of the segmenting contour from its initialization to its converged state for the two segmentation processes respectively. Note that upon introducing the shape prior the corresponding contour descends the energy landscape in direction of the negative gradient to end up in one of the minima. The example shows that the nonlinear shape prior can well separate different objects without mixing them as in the simple Gaussian hypothesis. Since each cluster in this example contains only one view for the purpose of illustration, the estimate (14) for the kernel width σ does not apply; instead we chose a smaller granularity of $\sigma = \mu/4$.

4.4 Segmentation of Real Objects

The following example is an application of the nonlinear shape statistics to silhouettes of real objects. The training set consisted of nine right and nine left hands, shown together with the estimated energy density in a projection onto the first two principal components in Figure 7, right side.

Rather than mixing the two classes of right and left hands, the shape prior clearly separates several clusters in shape space. The final segmentations without (left) and with (center) prior shape knowledge show that the shape prior compensates for occlusion by filling up information where it is missing. Moreover, the statistical nature of the prior is demonstrated by the fact that the hand in the image is not part of the training set. This can be seen in the projection (Figure 7, right side), where the final segmentation (white box) does not correspond to any of the training contours (black crosses).



2 views of a rabbit | binarized | aligned contours

Fig. 8. Example views and binarization used for estimating the shape density.

4.5 Tracking 3D Objects with Changing Viewpoint

In the following we present results of applying the nonlinear shape statistics for an example of tracking an object in 3D with a prior constructed from a large set of 2D views. We binarized 100 views of a rabbit — two of them and the respective binarizations are shown in Figure 8. For each of the 100 views we automatically extracted the contours and aligned them with respect to translation, rotation, scaling and cyclic reparameterization of the control points. We calculated the density estimate (7) and the induced shape energy (17).

In a film sequence we moved and rotated the rabbit in front of a cluttered background. Moreover, we artificially introduced an occlusion afterwards. We segmented the first image by the modified Mumford–Shah model until convergence before the shape prior was introduced. The initial contour and the segmentations without and with prior are shown in Figure 9. Afterwards we iterated 15 steps in the gradient descent on the full energy for each frame in the sequence.[2]

Some sample screen shots of the sequence are shown in Figure 10. Note that the viewpoint changes continuously.

The training silhouettes are shown in 2D projections with the estimated shape energy in Figure 11. The path of the evolving contour during the entire sequence corresponds to the white curve. The curve follows the distribution of training data well, interpolating in areas where there are no training silhouettes. Note that the intersections of the curve and of the training data in the center (Figure 11, left side) are only due to the projection on 2D. The results show that — given sufficient training data — the shape prior is able to capture fine details such as the ear positions of the rabbit in the various views. Moreover, it generalizes well to novel views not included in the training set and permits a reconstruction of the occluded section throughout the entire sequence.

[2] The gradient of the shape prior in (16) has a complexity of $O(rmn)$, where n is the number of control points, m is the number of training silhouettes and r is the eigenvalue cutoff. For input images of 83 kpixels and m=100, we measured an average runtime per iteration step of 96ms for the prior, and 11ms for the cartoon motion on a 1.2 GHz AMD Athlon. This permitted to do 6 iterations per second. Note, however, that the relative weight of the cartoon motion increases with the size of the image: for an image of 307 kpixels the cartoon motion took 100ms per step.

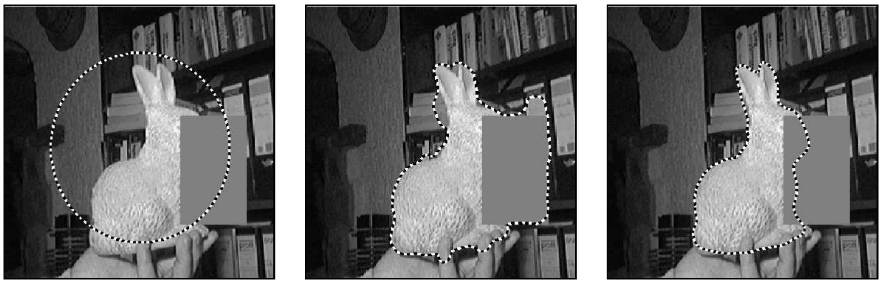


Fig. 9. Begin of the tracking sequence: initial contour, segmentation without prior, segmentation upon introducing the nonlinear prior on the contour.

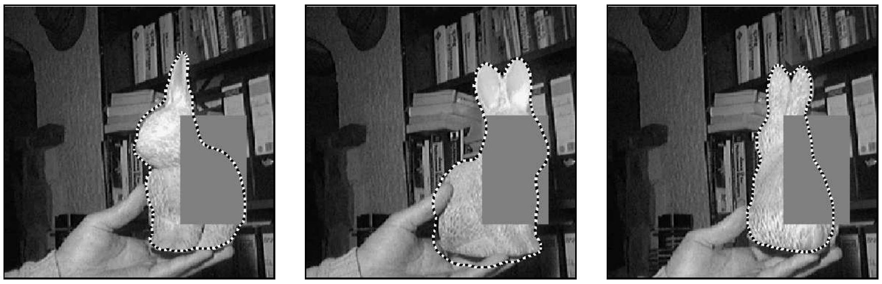


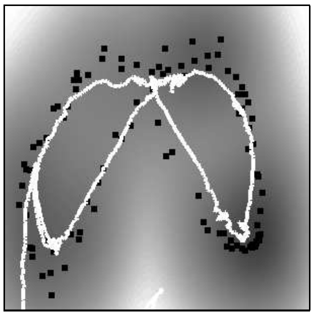
Fig. 10. Sample screen shots from the tracking sequence.

5 Conclusion

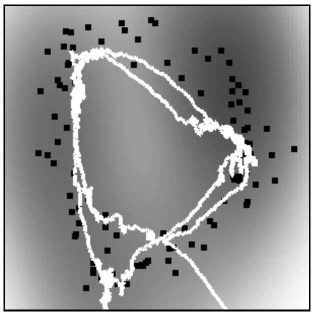
We presented a variational integration of nonlinear shape statistics into a Mumford–Shah based segmentation process. The statistics are derived from a novel method of density estimation which can be considered as an extension of the kernel PCA approach to a probabilistic framework. The original training data is nonlinearly transformed to a feature space. In this higher dimensional space the distribution of the mapped data is estimated by a Gaussian density. Due to the strong nonlinearity, the corresponding density estimate in the original space is highly non–Gaussian, allowing several shape clusters and banana– or ring–shaped data distributions.

We integrated the nonlinear statistics as a shape prior in a variational approach to segmentation. We gave details on appropriate estimations of the involved parameters. Based on the explicit representation of the contour, we proposed a closed–form, parameter–free solution for the integration of invariance with respect to similarity transformations in the variational framework.

Applications to the segmentation of static images and image sequences show, that the nonlinear prior can capture even small details of shape variation without mixing different views. It copes for misleading information due to noise and clutter, and it enables the reconstruction of occluded parts of the object silhouette. Due to the statistical nature of the prior, a generalization to novel views



Projection onto 1st and 2nd principal component



Projection onto 2nd and 4th principal component

Fig. 11. Tracking sequence visualized: Training data (•), estimated energy density and the contour evolution (white curve) in apropriate 2D projections. The contour evolution is restricted to the valleys of low energy induced by the training data.

not included in the training set is possible. Finally we showed examples where the 3D structure of an object is encoded through a training set of 2D projections.

By projecting onto the first principal components of the data, we managed to visualize the training data and the estimated shape density. The evolution of the contour during the segmentation of static images and image sequences can be visualized by a projection into this density plot and by animations. In this way we verified that the shape prior effectively restricts the contour evolution to the submanifold of familiar shapes.

Acknowledgments

We thank P. Bouthemy and his group, C. Kervrann and A. Trubuil for stimulating discussions and hospitality.

References

1. M.A. Aizerman, E.M. Braverman, and L.I. Rozonoer. Theoretical foundations of the potential function method in pattern recognition learning. *Automation and Remote Control*, 25:821–837, 1964.
2. B. E. Boser, I. M. Guyon, and V. N. Vapnik. A training algorithm for optimal margin classifiers. In D. Haussler, editor, *Proc. of the 5th Annual ACM Workshop on Comput. Learning Theory*, pages 144–152, Pittsburgh, PA, 1992. ACM Press.
3. B. Chalmond and S. C. Girard. Nonlinear modeling of scattered multivariate data and its application to shape change. *IEEE PAMI*, 21(5):422–432, 1999.
4. T. Cootes and C. Taylor. Active shape model search using local grey-level models: A quantitative evaluation. In J. Illingworth, editor, *BMVC*, pages 639–648, 1993.
5. T.F. Cootes and C.J. Taylor. A mixture model for representing shape variation. *Image and Vis. Comp.*, 17(8):567–574, 1999.
6. R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume 1. Interscience Publishers, Inc., New York, 1953.

7. D. Cremers, T. Kohlberger, and C. Schnörr. Nonlinear shape statistics via kernel spaces. In B. Radig and S. Florczyk, editors, *Pattern Recognition*, volume 2191 of *LNCS*, pages 269–276, Munich, Germany, Sept. 2001. Springer.
8. D. Cremers, C. Schnörr, and J. Weickert. Diffusion–snakes: Combining statistical shape knowledge and image information in a variational framework. In *IEEE First Workshop on Variational and Level Set Methods*, pages 137–144, Vancouver, 2001.
9. D. Cremers, C. Schnörr, J. Weickert, and C. Schellewald. Diffusion–snakes using statistical shape knowledge. In G. Sommer and Y.Y. Zeevi, editors, *Algebraic Frames for the Perception-Action Cycle*, volume 1888 of *LNCS*, pages 164–174, Kiel, Germany, Sept. 10–11, 2000. Springer.
10. G. Farin. *Curves and Surfaces for Computer–Aided Geometric Design*. Academic Press, San Diego, 1997.
11. D. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.
12. T. Heap and D. Hogg. Automated pivot location for the cartesian-polar hybrid point distribution model. In *BMVC*, pages 97–106, Edinburgh, UK, Sept. 1996.
13. T. Heap and D. Hogg. Improving specificity in pdms using a hierarchical approach. In *BMVC*, Colchester, UK, 1997.
14. C. Kervrann and F. Heitz. A hierarchical markov modeling approach for the segmentation and tracking of deformable shapes. *Graphical Models and Image Processing*, 60:173–195, 5 1998.
15. M.E. Leventon, W.E.L. Grimson, and O. Faugeras. Statistical shape influence in geodesic active contours. In *Proc. Conf. Computer Vis. and Pattern Recog.*, volume 1, pages 316–323, Hilton Head Island, SC, June 13–15, 2000.
16. B. Moghaddam and A. Pentland. Probabilistic visual learning for object detection. In *Proc. IEEE Internat. Conf. on Comp. Vis.*, pages 786–793, 1995.
17. D. Mumford and J. Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Comm. Pure Appl. Math.*, 42:577–685, 1989.
18. N. Paragios and R. Deriche. Coupled geodesic active regions for image segmentation: a level set approach. In D. Vernon, editor, *ECCV*, volume 1843 of *LNCS*, pages 224–240. Springer, 2000.
19. E. Parzen. On the estimation of a probability density function and the mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.
20. F. Rosenblatt. Remarks on some nonparametric estimates of a density function. *Annals of Mathematical Statistics*, 27:832–837, 1956.
21. S. Roweis. Em algorithms for PCA and SPCA. In M. Jordan, M. Kearns, and S. Solla, editors, *Advances in Neural Information Processing Systems 10*, pages 626–632, Cambridge, MA, 1998. MIT Press.
22. B. Schölkopf, S. Mika, Smola A., G. Rätsch, and Müller K.-R. Kernel PCA pattern reconstruction via approximate pre-images. In L. Niklasson, M. Boden, and T. Ziemke, editors, *ICANN*, pages 147–152, Berlin, Germany, 1998. Springer.
23. B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10:1299–1319, 1998.
24. M. Tipping. Sparse kernel principal component analysis. In *Advances in Neural Information Processing Systems 13*, Vancouver, Dec. 2001.
25. M.E. Tipping and C.M. Bishop. Probabilistic principal component analysis. Technical Report Woe-19, Neural Computing Research Group, Aston University, 1997.
26. M. Werman and D. Weinshall. Similarity and affine invariant distances between 2d point sets. *IEEE PAMI*, 17(8):810–814, 1995.
27. S.C. Zhu and A. Yuille. Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE PAMI*, 18(9):884–900, 1996.