Forschungsschwerpunkt S92

Industrial Geometry

http://www.ig.jku.at

FSP Report No. 66

Energy Training for Variational Shape Detection

M. Fuchs and S. Gerber

March 2008

ENERGY TRAINING FOR VARIATIONAL SHAPE DETECTION

MATTHIAS FUCHS AND SAMUEL GERBER

ABSTRACT. This paper presents a novel variational formulation incorporating statistical knowledge to detect shapes in images. We propose to train an energy based on joint shape and feature statistics inferred from training data. Variational approaches to shape detection traditionally involve energies consisting of a feature term and a regularization term. The feature term forces the detected object to be optimal with respect to image properties such as contrast, pattern or edges whereas the regularization term stabilizes the shape of the object. Our trained energy does not rely on these two separate terms, hence avoids the non-trivial task of balancing them properly. This enables us to incorporate more complex image features while still relying on a moderate number of training samples. Shape detection in microscope images and tracking of moving objects on cluttered background illustrate the capability of the proposed method to automatically adapt itself to different image features.

1. INTRODUCTION

Variational approaches to detect shapes in images are based on functionals which map shape geometries to an energy that reflects how well the given shape corresponds to the image features. Mumford and Shah [12] proposed to use the mean intensity of the region defined by the shape compared to the intensity of the background as such a feature. This idea can be extended to regions of homogeneous patterns as in Chan and Vese [1]. A second important feature are the edges in images. Kass et al. [9] proposed the Snakes approach to fit curves to the edges of an image. Both formulations require an additional regularization term in the energy functional to ensure that the corresponding variational problem is well-posed. This term measures the regularity of the boundary of the detected region.

These traditional methods have difficulties to correctly detect shapes that are partially occluded, on cluttered image background, or on images corrupted by too much noise. A common solution to this problem is the use of shape priors as in Chen et al. [2], Cremers et al. [4], Fang and Chan [5], Gastaud et al. [6], Leventon et al. [11], Rousson and Paragios [14] and Tsai et al. [16]. Shape prior methods use training data to compute shape statistics. These statistics define a likelihood functional that maps a shape to its probability w.r.t.the shape statistics and replaces the regularization term in the traditional variational formulation. This regularization ensures that only shapes which seem to be "reasonable" with respect to the training statistics are detected.

A common property of approaches based on shape priors is that they require the choice of a weighting parameter which determines the influence of the shape statistics. A high weight stabilizes the shape detection but might render it impossible to detect shapes which are very different from the training shapes (but still correct), whereas a too low weight introduces the danger of getting wrong results in case of noisy, cluttered or occluded image data. The correct choice of the parameter is not trivial and application dependent. Multiple (possibly time consuming) tests are often necessary to validate the weighting parameter.

The cited approaches further limit themselves to the use of only one kind of image feature, e.g.image contrast *or* edges. A combination of multiple features would again require each of them to be weighted with respect to the other and thus introduce even more parameters.

In this paper we propose an approach which does not require the explicit choice of a regularization parameter. Similar to the above works on shape priors, we train statistics on annotated data. In contrast of limiting the statistics to shapes only, we incorporate the corresponding image features from the training data. This allows us to learn the full-fledged segmentation energy and not only a regularization term. Furthermore, we avoid the choice of regularization parameters. Our method is capable of incorporating an arbitrary number of different kinds of image features. The relative importance of the various features is automatically learned from the training data. I.e.the trained energy gives high weight to combinations of features it learned to be representative for a class of objects and does not consider features which vary a lot across the training data. The computational effort to evaluate the resulting energy for a given shape is comparable to the methods mentioned above and the number of required training samples is very moderate.

The idea of learning an energy from multi-dimensional training data and leave the work of selecting important features to the learning process is similar to machine learning approaches acting on raw pixel values of image data (cf. LeCun et al. [10, and references therein]). In comparison to these methods our approach requires significantly less training because we use shape knowledge and intelligently computed image features. Another approach related to ours was proposed by Cremers et al. [3]. There, the authors learn a kernel density based on shape and image features. In contrast to our work, they focus on level set representations of shapes and the intensity distribution within shapes. They also consider the distributions of the shapes and the image features separately, whereas we treat them jointly.

The outline of this paper is as follows. In the next section we introduce the *shape-to-feature map* which, for a given image, maps a shape to a vector of image features determined by this shape. The shape-to-feature map is used to learn an energy based on shape and image feature statistics. For the results in this paper we concentrated on features which can be expressed as boundary integrals along the shape outline. Section 3 is dedicated to the training of an energy for a given image, a given set of training samples and a given shape-to-feature map. We present two different kinds of energies which are based on estimating the parameters of a normal distribution and kernel density estimation respectively. In addition, we state the variational problem based on these energies.

The last part is devoted to experimental results. We first applied our method to the detection of objects in biologic microscope images. These results were obtained by gradient and intensity based image features together with normal density estimates. In a second application we tracked moving objects in a movie sequence. In these experiments we use gradient information and intensity histogram data of each color channel in combination with the kernel density energy. We conclude with a summary of our key contributions and an outlook on future work.

1.1. Notation and preliminaries.

In the following we always assume $u:\Omega\to\mathbb{R}^d$ to be a (possibly vector valued) image defined on a 2-dimensional domain $\Omega\subseteq\mathbb{R}^2$.

If $d = 1$ then u can be interpreted as a gray-scale image. Let γ be a closed planar curve in Ω without self-intersections, i.e.$\gamma : S^1 \to \Omega$, injective. We assume γ to be piecewise differentiable. We refer to γ as a *shape*. The shape γ is completely determined by a *shape parameter* $p \in \mathbb{R}^m$. This is denoted by $\gamma(p)$. In our case p parametrizes the medial axis of the shapes, but it might as well be a list of the coefficients of a B-spline curve or any other kind of shape parametrization.

Finally note that our work is presented for the planar case only, but generalizes to higher dimension very easily.

2. Shape-to-feature map

We call a map F,

$$\mathbf{F} : \mathbb{R}^m \to \mathbb{R}^n , \quad p \to \mathbf{F}(\gamma(p)) , \tag{1}$$

which maps a shape parameter to a vector of features of the image u a *shape-to-feature map*. I.e.$\mathbf{F}$ depends only on the shape $\gamma(p)$ but not on p itself. It is important to note, though, that $\mathbf{F}$ does depend on the image u. To simplify notation and because we chose u to be fixed throughout the paper we do not denote this dependence.

In this paper we concentrate on a subclass of shape-to-feature maps, which is characterized by a special form of $\mathbf{F}$. In particular, we consider $\mathbf{F}$ to be the composition $\mathbf{F} = \mathbf{G} \circ \mathbf{H}$, where $\mathbf{G} : \mathbb{R}^k \to \mathbb{R}^n$ and $\mathbf{H} : \mathbb{R}^m \to \mathbb{R}^k$. We assume that only $\mathbf{H}$ depends on the image u whereas $\mathbf{G}$ is independent of the image and the shape. Each of the components H_i, $1 \leq i \leq k$, of $\mathbf{H}$ should have one of the following forms:

$$H_i(p) = \int\limits_{\gamma(p)} a_i(u) ds \quad \text{or} \quad H_i(p) = \int\limits_{\gamma(p)} b_i(u) \cdot dn , \tag{2}$$

where $a_i(u) : \Omega \to \mathbb{R}$, $b_i(u) : \Omega \to \mathbb{R}^2$ and n denotes the outer unit normal of $\gamma(p)$. I.e.we assume each component of $\mathbf{H}$ to be either the integral of a scalar function along the shape boundary or the integral of a vector field along the same boundary. This construction enables us to evaluate complex image features and still give good estimates on the complexity of an evaluation of $\mathbf{F}$ for a given shape $\gamma(p)$. Because $a_i(u)$ and $b_i(u)$ depend only on the image u but not on the shape $\gamma(p)$ we can precompute a discretized version of them. The computation of $H_i(p)$ then involves

- the computation of $\gamma(p)$, and
- the evaluation of a 1-dimensional boundary integral.

For accordingly chosen functions $a_i(u)$ it is possible to evaluate a wide range of features such as intensity, histogram data and gradient information along the shape boundary. By the use of the divergence theorem, the integral over $b_i(u)$ enables us to compute the same values over the region $\Gamma(p) \subseteq \Omega$ inside a given shape $\gamma(p)$. Assume a scalar function $c_i(u)$ which we want to integrate over $\Gamma(p)$. We first compute $b_i(u)$ such that $\nabla \cdot b_i(u) = c_i(u)$. This equation constitutes an underdetermined system of partial differential equations for b_i which is trivial to solve for a given image u. Then

$$\int\limits_{\Gamma(p)} c_i(u) dx = \int\limits_{\gamma(p)} b_i(u) \cdot dn . \tag{3}$$

For $a_i(u) = 1$ or $c_i(u) = 1$ the integrals (2) evaluate to the length of the boundary of $\gamma(p)$ and its volume, respectively.

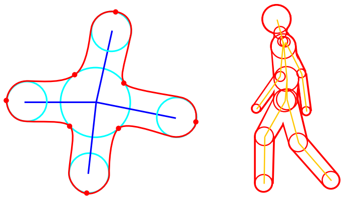

FIGURE 1. *Left:* The skeleton and the maximal circles of this cross are parametrized by p. The outline $\gamma(p)$ is computed by interpolating the points on the circles. *Right:* Medial axis shape model for a human silhouette.

The map **G** is used to combine the values of the integrals H_i to get more meaningful features. In our examples the function **G** normalizes the integrals along the boundary $\gamma(p)$ and the region $\Gamma(p)$ w.r.t.the length of $\gamma(p)$ and area of $\Gamma(p)$, respectively. It is further possible to write the simplified Mumford-Shah functional [12, 1] and the Snakes [9] functional in the form $\mathbf{F} = \mathbf{G} \circ \mathbf{H}$ with **H** being an expression of integrals as in (2).

2.1. **Shape representation.** The shape model we use is based on the idea of parameterizing a shape by constructing a discrete approximation of its medial axis as proposed by Joshi et al. [8]. In our case, a shape parameter p parametrizes a tree-like skeleton consisting of nodes, edges connecting the nodes, and circles at the nodes. These circles are supposed to be maximal circles within the shape, i.e.they touch the shape in at least two points. In more detail, the components of p determine

- the position and rotation of the skeleton,
- the lengths and the angles of the edges of the skeleton, and
- the radii of maximal circles centered at the nodes of the skeleton.

From this data we compute points on the maximal circles and interpolate them tangentially with a B-spline curve. Two different implementations of this idea (corresponding to the two applications in this paper) are illustrated in Figure 1. Note that the skeleton does not necessarily correspond to the real medial axis of the resulting shape. Still, the underlying idea of the medial axis results in a much more natural interpretation of the different components of a shape parameter than it is e.g.the case for a vector of B-spline coefficients.

Because we will explicitly refer to the position and rotation of shapes later on, we decide that the first three components of p determine these properties, i.e.

$$p = (\underbrace{p^1, p^2, p^3}_{\text{position, rotation}}, \underbrace{p^4, \ldots, p^m}_{\text{skeleton, radii}})^T . \tag{4}$$

3. ENERGY TRAINING

The main contribution of this paper is the computation of an energy E for given training shape parameters $p_1, \ldots, p_N$ and a shape-to-feature map **F**. In this section we will introduce the variational form of the shape detection problem based on this energy and explain two different ways to train energies based on normal density estimation and kernel density estimation.

The energy $E : \mathbb{R}^m \to [0,\infty)$ maps an unseen shape parameter p to a non-negative value which determines how well $\gamma(p)$ fits on the image considering shape and image properties learned from the training shapes. Small values $E(p)$ correspond to a good match. Hence, the shape detection problem of single shape can be written as

$$p = \operatorname{argmin}_{p \in D} E(p)\,, \tag{5}$$

where $D \subseteq \mathbb{R}^m$. The domain D constrains the above variational problem. In all applications we chose D such that only shapes on the image domain Ω are considered. We further can adapt D such that shapes close to training shapes or already detected shapes on the same image are not considered in the minimization problem.

In the following we explain two different approaches to compute E. As mentioned before we assume u to be a fixed image. Furthermore, $p_1, \dots, p_N$ are the parameters of manually determined training shapes on this image. This means that we expect the shapes $\gamma(p_1), \dots, \gamma(p_N)$ to match objects on u. Finally let $\mathbf{F}$ be a feature map for u. Then, for a given shape parameter p, we define its *shape-feature vector* $q(p)$ by setting

$$q(p) = (p^4, \dots, p^m, F_1(p), \dots, F_n(p))^T \in \mathbb{R}^M\,, \tag{6}$$

where $M := m + n - 3$. In other words, $q(p)$ consists of the features for the shape determined by p and the shape parameter p excluding position and rotation. We further denote the shape-feature vectors of the training data p_i as $q_i := q(p_i)$, $1 \leq i \leq N$.

In this paper we consider energies of the form

$$E(p) = -\log f(q(p))\,, \tag{7}$$

where f is a probability density on $\mathbb{R}^M$ and depends on the training data $p_1, \dots, p_N$. This formulation translates the energy learning into a density estimation problem. For our experiments we consider a parametric and a non-parametric approach to estimate the density $f(q)$.

It is worth noting that both versions are invariant to scaling of single components of the shape-feature vector. This is important as any dependence on scale would implicitly introduce a regularization parameter which is exactly what we want to avoid.

3.1. **Normal density estimation.**

In this section we assume q to be normally distributed with density function

$$f(q) = (2\pi)^{-M/2} \det(\Sigma)^{-1/2}\ \mathrm{e}^{-\frac{1}{2}(q-\mu)^T \Sigma^{-1} (q-\mu)}\,. \tag{8}$$

with $\mu \in \mathbb{R}^M$ and Σ a symmetric and positive definite $M \times M$-matrix. Assuming the shape-feature vectors of the training data to be independently and identically distributed w.r.t. f we compute maximum-likelihood estimators of the parameters μ and Σ:

$$\mu = \frac{1}{N} \sum_{i=1}^{N} q_i\,, \tag{9}$$

$$\Sigma = \frac{1}{N-1} \sum_{i=1}^{N} (q_i - \mu)(q_i - \mu)^T\,. \tag{10}$$

By (7) the energy E of a given shape parameter p is then

$$E(p) \propto (q(p)-\mu)^T\Sigma^{-1}(q(p)-\mu)\,. \tag{11}$$

There exist several interpretations of the above expression. For one, it equals the Mahalanobis distance between q and μ w.r.t.the covariance Σ. Also, it can be interpreted as the squared norm of the coefficients of q w.r.t.the principal component analysis of the training shape-feature vectors. It is further important to note that (11) with μ and Σ as in (9) and (10) is invariant under linear transformation of q. In particular, rescaling single components of the shape-feature vector does not change E.

3.2. **Kernel density estimation.** The parametric approach in Section 3.1 puts limitations on the range of probability densities we are able to estimate properly and therefore on the type of shape distributions we can successfully detect. We use a kernel density estimator with a Gaussian kernel yielding the density function:

$$f(q) = (2\pi)^{-M/2}\det(\Sigma)^{-1/2}\frac{1}{N}\sum_{i=0}^{N}\mathrm{e}^{-\frac{1}{2}(q-q_i)^T\Sigma^{-1}(q-q_i)}\,. \tag{12}$$

This leaves the problem of selecting an appropriate kernel width, a task which is also known as bandwidth or window width selection [15]. Various approaches have been proposed and there is no single optimal solution in practice. For our experiments we set the variance to the diagonal matrix $\Sigma = \operatorname{diag}(\sigma_1^2,\ldots,\sigma_M^2) \in \mathbb{R}^{M\times M}$. We chose the diagonal entries of Σ to be the average of the squared pairwise distances of the corresponding components of the training shape-feature vectors and then scaled the whole matrix by a constant $\alpha > 0$:

$$\sigma_k^2 = \frac{2\alpha}{N(N-1)}\sum_{i=1}^{N}\sum_{j=i+1}^{N}(q_i^k - q_j^k)^2\,,\ 1 \le k \le M\,. \tag{13}$$

Since we have a small number of training samples we set $\alpha = 10$ to avoid a highly peaked energy at the training locations. The resulting energy E of a shape parameter p is then

$$E(p) \propto -\log\sum_{i=0}^{N}\mathrm{e}^{-(q(p)-q_i)^T\Sigma^{-1}(q(p)-q_i)}\,. \tag{14}$$

Due to the specific choice of the covariance matrix Σ in (13), the energy (14) is invariant to scaling of the individual components of the shape-feature vector. The energy as formulated in (14) allows to model complex energies at the cost of increased computational effort (for large amounts of training data) as well as the problem of selecting an appropriate kernel width.

4. Results

We applied the proposed method to two kinds of image data. The first application is concerned with the detection of shapes in microscope image data. We manually annotated some of the objects on a given image and automatically detected the remaining ones by minimizing the energy learned from the annotated data. In the second application, we annotated the shape of a walking human on a couple of consecutive frames of a movie and tracked the same human silhouette in the subsequent frames. In the first example we used an energy based on the normal

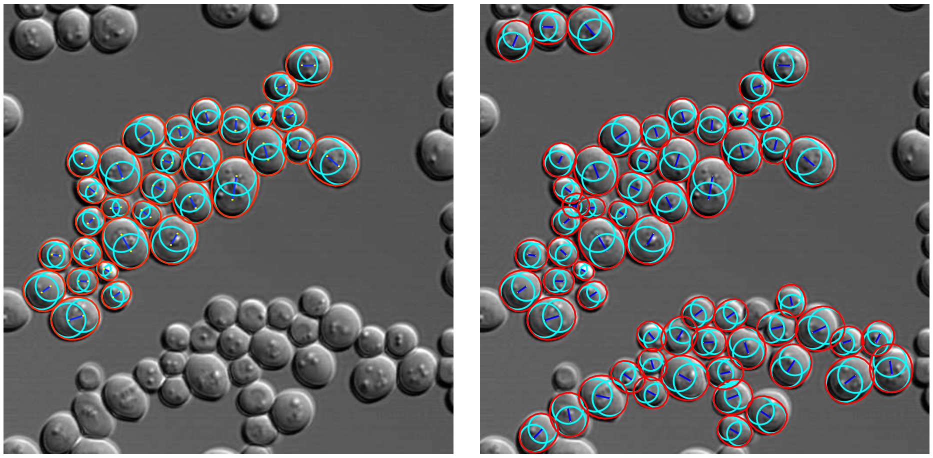

FIGURE 2. Microscope image (512 × 512 pixels) of yeast cells. *Left:* Manually annotated cells. *Right:* The training shapes and the detected cells.

distribution estimation as detailed in Section 3.1, whereas in the tracking example we used kernel density estimation (Section 3.2).

4.1. **Shape detection in microscope images.** This section is concerned with the detection of similar objects in gray-scale microscope images. The data in Figure 2 involves two major challenges. The cells on the image form a huge cluster and it is difficult to separate them with traditional methods. In addition, the shadow-like features on each cell cause extra edge information in the cells which can not be removed by smoothing. The objects in Figure 3 are more clearly set apart from each other, but the quality of their appearance varies more than in the first examples.

We computed the same feature map for both images. First, we smoothed the images with a 2-pixel-wide Gaussian kernel, denoting the result as u_σ. Then we defined **F** by

$$\mathbf{F}(p) = \begin{pmatrix} ⨏_{\gamma(p)} |\nabla u_\sigma| \, ds \\ ⨏_{\gamma(p)} u \, ds \\ ⨏_{\gamma(p)} u^2 \, ds \\ ⨏_{\Gamma(p)} u \, dx \\ ⨏_{\Gamma(p)} u^2 \, dx \end{pmatrix}. \tag{15}$$

Here, $⨏_{\gamma(p)} ds$ and $⨏_{\Gamma(p)} dx$ denote the integrals over $\gamma(p)$ and $\Gamma(p)$ normalized by the length of $\gamma(p)$ and the area of $\Gamma(p)$ respectively. In a nutshell, we compute the normalized values of the absolute values of the image gradients along the boundary and the normalized values of the intensities and their first moment along the boundary and inside the shape. Note that as in (3) the latter two integrals can be transformed into a boundary integral of a vector field. This also holds for the computation of the area of $\Gamma(p)$.

We computed the training shape-feature vectors $q_1, \ldots, q_N$ from the manually annotated objects, $N = 31$ and $N = 75$ in Figure 2 and Figure 3, respectively. From these training sets we estimated μ and Σ as in (9) and (10) to define the energy (11).

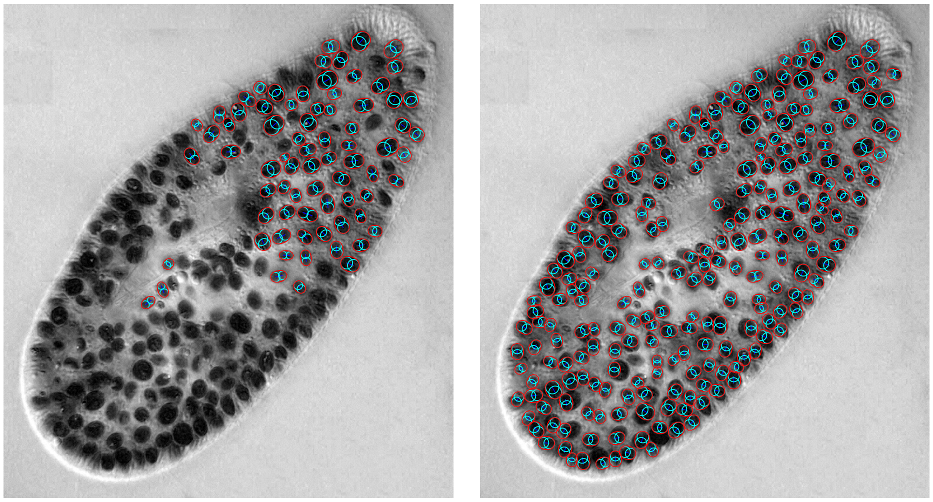

FIGURE 3. Microscope image (1536 × 1686 pixels) of a paramecium with cilia (a type of cell organelles). *Left:* Manually annotated cilia. *Right:* The training shapes and the detected cilia.

The solution of (5) was done iteratively. We chose the problem domain D such that only shapes on the image Ω are considered and further removed all parameters from D which corresponded to shapes which overlap with the training shapes. By ”overlapping shapes” we mean shapes whose common area is above some threshold (50 common pixels in Figure 2 and 150 pixels in Figure 3).

To find multiple shapes in the image we used Algorithm 4.1. In simple words, the algorithm generates random shapes and tries to improve the current detection result $p_1, \dots, p_M$ by successively replacing previously detected shapes with new ones.

Algorithm 1 Detection of multiple shapes

```
choose $M$ random shape parameters $(p_1, \dots, p_M)$
$c_i := E(p_i)$, $1 \leq i \leq M$
**repeat**
  choose a random shape parameter $p'$
  **if** $E(p') < c_i$ for some $1 \leq i \leq M$ **then**
    **if** $\gamma(p')$ does not overlap with any of the shapes $\gamma(p_1), \dots, \gamma(p_M)$ **then**
      $p_i := p'$ and $c_i := E(p')$
    **else if** $\gamma(p')$ overlaps with $\gamma(p_{i_1}), \dots, \gamma(p_{i_k})$ and $E(p') < \min(c_{i_1}, \dots c_{i_k})$
    **then**
      $p_{i_1} := p'$ and $c_{i_1} := E(p')$
      choose $p_{i_2}, \dots p_{i_k}$ randomly
      $c_{i_j} := E(p_{i_j})$, $2 \leq j \leq k$
    **end if**
  **end if**
**until** $(p_1, \dots, p_M)$ stop improving significantly
```

In the algorithm, by the random choice of a shape parameter p we mean selecting $p \in D$ as follows:

- The position (p^1, p^2) and the rotation p^3 are uniformly sampled on the image domain Ω and in the interval $[-\pi, \pi]$, respectively.
- We compute the mean $\mu_{4:M}$ and the covariance matrix $\Sigma_{4:M}$ of the components $(p_i^4, \dots, p_i^M)$, $1 \leq i \leq N$, of the training data. Then we sample $(p^4, \dots, p^M)$ from the multivariate normal distribution with mean $\mu_{4:M}$ and variance $\Sigma_{4:M}$.

The result of the above selection is accepted if $p \in D$. Otherwise a new candidate p is sampled.

We chose M larger than the expected number of shapes in the image. After stopping the algorithm we manually estimated a threshold $c_0 > 0$ such that the shapes $\gamma(p_i)$ with $E(p_i) \leq c_0$, $1 \leq i \leq M$, represented usable results. In many applications, the manual selection of c_0 does not really pose a problem, because it is done *after* the algorithm is run. I.e.changes of c_0 can be visualized in real-time. Furthermore, techniques to estimate c_0 from the distribution of the final energies c_i, $1 \leq i \leq M$, could be employed.

We chose this approach because it effectively demonstrates the capability of the E to detect shapes from learned shape and image features. Genetic algorithms or the combination of genetic and gradient based approaches might significantly speed up the minimization process.

For this work we did not do any further investigations on alternative stopping criteria for the algorithm but ran it until the result stopped to improve. An analysis of how the number of random samples, i.e.the number of iterations, compares to the quality of the result requires a meaningful way to measure the usefulness of a detection result and is beyond the scope of this work.

4.2. **Tracking.** The tracking example uses frames 33 through 100 from the *walking straight* sequence[1]. We annotated the walking person from frame 33 to frame 78. The task is to track the target person through the remaining frames. To avoid the technicalities of handling multiple shape-to-feature maps we define the image u as the horizontal concatenation of the 125 frames into a single image. In this example we are dealing with color images and refer with u^j to the j channel of u.

Note that the method is not specifically tuned for tracking but uses only statistical knowledge from the training shapes. The silhouette of the person to be tracked varies significantly over one walking cycle.

We defined the shape-to-feature map $\mathbf{F}(p)$ for u as the values of the normalized K-bin histograms over the shape boundary $\gamma(p)$ and the area $\Gamma(p)$ inside the shape w.r.t.intensity and gradient magnitude for each channel j of u. Formally, we denote for a set A, a function $v : A \to \mathbb{R}$ and $1 \leq k \leq K$

$$h_A(v,k) = \text{value of the } k\text{-th bin of the normalized } K\text{-bin histogram of } v \text{ over } A. \tag{16}$$

Then we set

$$F_i(p) = \begin{cases} h_{\gamma(p)}(u^j, k) & \text{for } i = 4(3k + j - 4) + 1 \\ h_{\gamma(p)}(|\nabla u^j_\sigma|, k) & \text{for } i = 4(3k + j - 4) + 2 \\ h_{\Gamma(p)}(u^j, k) & \text{for } i = 4(3k + j - 4) + 3 \\ h_{\Gamma(p)}(|\nabla u^j_\sigma|, k) & \text{for } i = 4(3k + j - 4) + 4 \end{cases} \tag{17}$$

[1] http://www.nada.kth.se/ hedvig/data.html

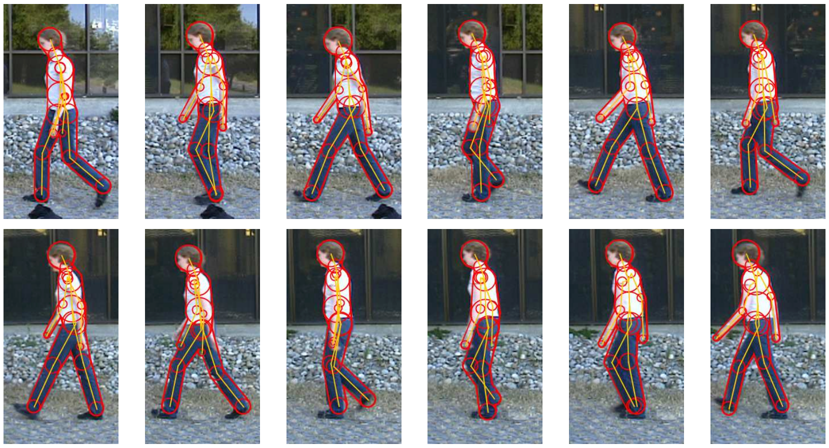

FIGURE 4. *Top:* 6 frames with manually annotated shapes from the training set. *Bottom:* Detected shapes in 6 of the tracked frames.

for $1 \leq k \leq K$, $1 \leq j \leq 3$ and $1 \leq i \leq 12K$. Here, u_σ denotes the image u after convolution with a 2-pixel-wide Gaussian kernel. In our experiments we set $K = 5$.

We then defined the energy E as in (14) using the shape-feature vectors $q_1, \ldots, q_{36}$ of the annotated shapes in the frames 33 through 78. For the tracking we proceeded as in Algorithm 4.2.

Algorithm 2 Tracking

```
$p_0 =$ shape parameters of the last annotated frame
for $i = 1$ to number of unseen frames do
    $p_i = p_{i-1}$ shifted to the next frame
    $c = E(p_i)$
    for number of samples do
        choose $p'$ randomly at location $p_{i-1}$ shifted to next frame
        if $E(p') \leq c$ then
            $p_i = p'$
            $c = E(p_i)$
        end if
    end for
end for
```

In the algorithm, the random selection of the sample p' in the i-th frame is done as follows:

- We set p' to p_{i-1} and then offset the position of p' by the frame width to move it to the next frame. We did not sample the rotational component but kept it fixed during training and tracking.
- The remaining components of p' were chosen as in Section 4.1, i.e.we sampled them from the $(\mu_{4:M}, \Sigma_{4:M})$-normal distribution, where $\mu_{4:M}$ and $\Sigma_{4:M}$ are the mean and the covariance of the upper components of the training shape parameters.

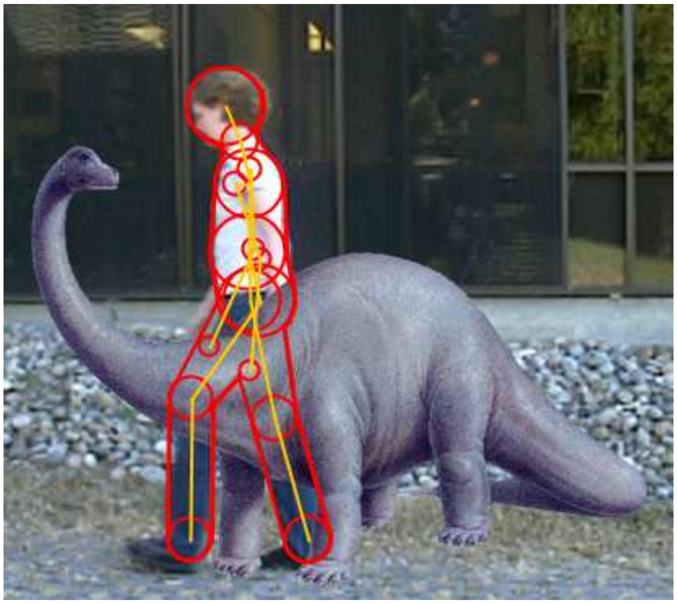

FIGURE 5. Tracked shape with a Diplodocus occluding the tracking target. The visible parts of the human are accurately detected.

Figure 4 shows 6 of the 36 manually annotated frames from the training set and 6 frames with tracked shapes. The proposed method is able to perform tracking with a small amount of training data (36 samples).

In Figure 5 we manually overlaid an image of a dinosaur on the frames of the movie sequence. We trained the energy as before on the original training set (frames 33 through 78 without the added dinosaur) and then tracked in the images with the dinosaur. This demonstrates the robustness of the proposed method to occlusions.

5. CONCLUSION AND FUTURE DIRECTIONS

We suggest a novel variational formulation to shape detection based on training a task-specific segmentation energy. The underlying mathematical model of our method is very general and can be easily used for a wide range of applications. The proposed energy learns the significant shape and image features from training shapes and is able to distinguish them from non-relevant features. This is illustrated in two different applications.

A major advantage over existing approaches is the absence of a regularization term. This avoids the often difficult task of choosing the optimal regularization for a given application.

On the other hand, because we incorporate shape priors and rely on a meaningful selection of image features, our approach requires far less training samples than methods solely relying on learning pixel values [13] and the training is computationally cheap.

In future we would like to investigate different energies. E.g.learning a kernel based on positive and negative training samples is considered. A Bayesian approach to the parametric density estimation could help mitigate difficulties due to small training sets (over fitting). For the nonparametric approach, techniques such as adaptive kernel density estimation or projection pursuit density estimation could further improve results [7]. Furthermore, the development of efficient algorithms to minimize the learned energy is subject of ongoing research.

REFERENCES

[1] Tony F. Chan and Luminita A. Vese. Active contours without edges. *IEEE Transactions on Image Processing*, 10(2):266–277, 2001.

[2] Yunmei Chen, Hemant D. Tagare, Sheshadri Thiruvenkadam, Feng Huang, David Wilson, Kaundinya S. Gopinath, Richard W. Briggs, and Edward A. Geiser. Using prior shapes in geometric active contours in a variational framework. *International Journal of Computer Vision*, 50(3):315–328, 2002.

[3] D. Cremers and M. Rousson. Efficient kernel density estimation of shape and intensity priors for level set segmentation. In J.S. Suri and A.A. Farag, editors, *Deformable Models*, pages 447–460, 2007.

[4] Daniel Cremers, Florian Tischhäuser, Joachim Weickert, and Christoph Schnörr. Diffusion snakes: Introducing statistical shape knowledge into the mumford-shah functional. *International Journal of Computer Vision*, 50(3):295–313, 2002.

[5] Weng Fang and Kap Luk Chan. Incorporating shape prior into geodesic active contours for detecting partially occluded object. *Pattern Recognition*, 40(7):2163–2172, 2007.

[6] Muriel Gastaud, Michel Barlaut, and Gilles Aubert. Combining shape prior and statistical features for active contour segmentation. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(5):726–734, 2004.

[7] J. Hwang, S. Lay, and A. Lippman. Nonparametric multivariate density estimation: a comparative study. *IEEE Transactions on Signal Processing*, 42(10):2795–2810, 1994.

[8] Sarang Joshi, Stephen Pizer, P. Thomas Fletcher, Paul Yushkevich, Andrew Thall, and J. S. Marron. Multiscale deformable model segmentation and statistical shape analysis using medial descriptions. *IEEE Transactions on Medical Imaging*, 21(5):538–550, 2002.

[9] Micheal Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[10] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F.-J. Huang. A tutorial on energy-based learning. In G. Bakir, T. Hofman, B. Schölkopf, A. Smola, and B. Taskar, editors, *Predicting Structured Data*. MIT Press, 2006.

[11] Micheal E. Leventon, W. Eric L. Grimson, and Olivier Faugeras. Statistical shape influence in geodesic active contours. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 316–323, June 2001.

[12] David Mumford and Jayant Shah. Optimal approximations by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics*, 42(4):577–684, 1989.

[13] M. Ranzato, P.E. Taylor, J.M. House, R.C. Flagan, Y. LeCun, and P. Perona. Automatic recognition of biological particles in microscope images. *Pattern Recognition Letters*, 28:31–39, 2007.

[14] Mikael Rousson and Nikos Paragios. Shape priors for level set representations. In Anders Heyden, Gunnar Sparr, Mads Nielsen, and Peter Johansen, editors, *ECCV 2002 Proceedings, Part II*, volume 2351 of *Lecture Notes in Computer Science*, pages 78–92. Springer, 2002.

[15] Bernard W. Silvermann. *Density Estimation for Statistics and Data Analysis*. Chapmann and Hall, New York, 1986.

[16] Andy Tsai, Anthony Yezzi, Clare Tempany, Dewey Tucker, Ayres Fan, W. Eric L. Grimson, and Alan Willsky. A shape-based approach to the segmentation of medical imagery using level sets. *IEEE Transactions on Medical Imaging*, 22(2):137–154, 2003.

INFMATH IMAGING, UNIVERSITY OF INNSBRUCK, 6020 INNSBRUCK, AUSTRIA
E-mail address: matz.fuchs@uibk.ac.at

SCI INSTITUTE, UNIVERSITY OF UTAH, SALT LAKE CITY, UT 84112
E-mail address: sgerber@cs.utah.edu