Int J Comput Vis (2009) 83: 12–29
DOI 10.1007/s11263-009-0208-2

Contour Grouping Based on Contour-Skeleton Duality

Nagesh Adluru · Longin Jan Latecki

Received: 30 May 2008 / Accepted: 6 January 2009 / Published online: 27 January 2009

Abstract In this paper we present a method for grouping relevant object contours in edge maps by taking advantage of contour-skeleton duality. Regularizing contours and skeletons *simultaneously* allows us to combine both low level perceptual constraints as well as higher level model constraints in a very effective way. The models are represented using paths in symmetry sets. Skeletons are treated as trajectories of an imaginary virtual robot in a discrete space of "symmetric points" obtained from pairs of edge segments. Boundaries are then defined as the maps obtained by grouping the associated pairs of edge segments along the trajectories. Casting the grouping problem in this manner makes it similar to the problem of *Simultaneous Localization and Mapping (SLAM)*. Hence we adapt the state-of-the-art probabilistic framework namely Rao-Blackwellized particle filtering that has been successfully applied to SLAM. We use the framework to maximize the joint posterior over skeletons and contours.

Keywords Contour grouping · Skeletons · Shape models · Rao-Blackwellized particle filters · SLAM

1 Background

Object boundaries, also known as contours, are very useful descriptors for object recognition. Extracting contours using only edge-detection algorithms without any regularization is an ill-posed problem due to ambiguities in the gradient space of real-world images. Hence *grouping* edge pixels into contours also known as contour grouping is a very popular approach. The problem even after decades of work still remains to be open and a very active research field as documented by recent papers (Zhu et al. 2007; Trinh and Kimia 2007; Tamrakar and Kimia 2007; Hoiem et al. 2007; Stein et al. 2007; Galun et al. 2007). Different perceptual grouping constraints motivated by Gestalt psychology (Wertheimer 1923) and cues like closure, good-continuity (Mohan and Nevatia 1992), minimal model theory (Feldman 1999, 2003) have been used to regularize contour growth. All such low-level grouping constraints can be used in extracting "smooth and compact" contours but not necessarily those useful for object recognition problem. Hence researchers started using higher level constraints which when used appropriately can result in "relevant contours". This is possible because high level constraints can capture both local and global characteristics of an image similar to how humans process images. For example Hoiem et al. (2007) uses "depth cues" to recover *occluding* contours. Stein et al. (2007) uses motion cues where the central idea is based on the fact that if the "object of interest" moves around in a sequence of image frames, its contours can be detected by capturing the dynamics of the occluding contours. Their system is based on the belief that the situation of having a sequence of frames is more natural than using a static image for object recognition. But it is clear that even though humans might exploit the motion cues they do not *depend* on those to detect objects. Also it is not always possible to have a sequence of frames for the task of recognition.

The use of symmetry as a key contour grouping cue has been studied in both human vision and computer vision. Among others, the results in Lowe (1985), Witkin and

This work was supported by the National Science Foundation under Grants No. IIS-0534929 and IIS-0812118 and by the Department Of Energy under Grant No. DE-FG52-06NA27508.

N. Adluru (✉) · L.J. Latecki
Temple University, Philadelphia, PA 19122, USA
e-mail: nagesh@temple.edu

Tenenbaum (1983), Leyton (1992), Zhu and Yuille (1995), Kimia and Tamrakar (2002) show that symmetry is non-accidental. Therefore symmetry can be expected to be useful in not only distinguishing salient contour structures from noisy background in the low-level processing but also in extracting *more relevant* contours for the task of object recognition. Relevant contours are closely related to *shape* of an object. Shapes of objects and their symmetries have several dual geometric properties (Giblin and Kimia 2003b; Kuijper and Olsen 2004). Shapes can be represented using signatures defined using locally symmetric curves (Kuijper et al. 2006b) and shapes can be reconstructed from symmetries (Giblin and Kimia 2003a). Symmetry for rigid and non-rigid objects can be captured in several ways: for example using ribbons (Mohan and Nevatia 1992), using planar reflections (Podolak et al. 2006), using intrinsic self-similarities (Raviv et al. 2007) or more classically using medial axes (Blum 1967, 1973) and symmetry sets (Bruce et al. 1985; Giblin 2000). A lot of work has been done in using symmetry for object recognition (Siddiqi et al. 1999; Sebastian et al. 2004; Kuijper and Olsen 2006; Kuijper et al. 2006a; Bai and Latecki 2008). Symmetry principle expressed as *global* contour symmetry has been used in contour grouping in various approaches. One of the more recent approaches that is based on global contour symmetry is presented in Stahl and Wang (2006). It is related to the grouping method developed in Mohan and Nevatia (1992), where symmetry is considered along with closure and proximity. Symmetry is applied as a cue to pair the extracted curves by producing a set of ribbons. These ribbons are then grouped into structures using heuristic algorithms. While different representations of symmetry have different advantages based on the context of application, those that can capture *local* symmetry play more important role in grouping contours, since global symmetry is usually not present in 2D images due to perspective distortion and nonrigid deformation.

Probabilistic reasoning is popular in processing noisy images. For e.g., a popular set of edge detectors, *pb*, is built using probabilistic representation of boundary (Martin et al. 2001). In contrast, our approach represents a joint-posterior of *both* contours and skeletons. Also a probabilistic method for tracking of motion boundaries for motion estimation is presented in Black and Fleet (2000). Bayesian reasoning is gaining popularity even in psychophysical theory of vision. For example, Feldman (2001) presents a Bayesian framework to probabilistically judge grouping hypotheses used by humans.

Bayesian approach for grouping contours using multiple hypotheses tracking was introduced in the seminal paper Cox et al. (1993) which is heavily based on another seminal work in Reid (1979). More recently, a particle filter based system called JetStream was applied to contour grouping in Perez et al. (2001). This work treats the grouping process as a dynamic process and the detection is performed on edge pixels with particles following the contour directly. In this paper a simple ribbon geometry is also used for road extraction. They introduced a novel and an unconventional version of a particle filter tracking algorithm, where temporal sequence is replaced with a sequence of growing contour points. JetStream often fails to track contours in complex images, since only low level features (image gradient and corner detection) are used as the basis for particle observations. Therefore, it requires user interaction during the tracking process.

Since particle filtering (also known as sequential Monte Carlo estimation) provides a strong framework with Bayesian reasoning to capture complex (non-linear and non-Gaussian) dynamics of probability density functions it has been extensively applied for robust object tracking. One of the best known approaches in computer vision is the Condensation algorithm (Isard and Blake 1998), which allows tracking object contours in the presence of background clutter. Particle filtering has become the standard approach for mobile-robot localization with the main application being SLAM (Thrun et al. 2005; Grisetti et al. 2007; Eliazar and Parr 2003), where probability distributions for the robot poses (position plus heading direction) and the possible maps are approximated and propagated by a set of particles. We first cast contour grouping as SLAM problem in Adluru et al. (2007). The presented work extends this idea in several ways as summarized in Sect. 2. Since we use a strong statistical framework with constraints that are not entirely low-level, our work has advantage of capturing relevant contours even in the presence of significant distractors and inner structures. While the techniques like Perez et al. (2001) use good statistical framework they do not exploit geometric information. The methods like Liu et al. (1998b), Ren et al. (2005) exploit geometry but are limited in noisy conditions.

2 Overview of Our Approach

Skeletons capture local symmetry in a very useful way. According to Blum's definition a skeleton S of a set of object boundaries D, is the locus of the centers of maximal disks. A maximal disk in D is a closed disk contained in D that is interiorly tangent to the boundary of D and that is not contained in any other disk in D. Each maximal disc must be tangent to the boundary in at least two different points. A set of skeleton points $s \in S$ and the radii $r(s)$ of their maximal disks can be used for reconstructing the boundary without any ambiguity. An important property is that skeletons can be computed for every planar shape, but computation of skeletons without boundary information is not possible. Thus, there is cyclic dependency between contours and

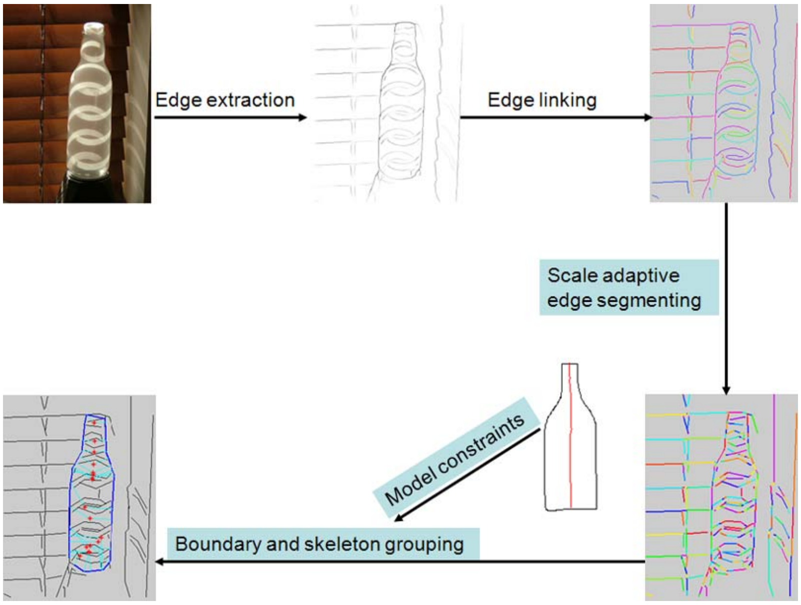

Edge extraction
Edge linking
Scale adaptive edge segmenting
Model constraints
Boundary and skeleton grouping

Fig. 1 Overview of our algorithm: (1) Edge extraction. (2) Edge linking. (3) Approximation by scale adaptive edge segments. (4) Grouping boundary and skeleton simultaneously using Rao-Blackwellized particle filter with symmetry based model constraints

skeletons, which is called contour-skeleton duality. The key idea of our approach is to exploit this dependency for grouping relevant contours by *simultaneous* estimation of skeletons and the contours in edge images. Thus we regularize both contours and skeletons. A detailed analysis of medial axis properties and algorithms is presented in a recent book, Siddiqi and Pizer (2007). A good mathematical introduction about medial axis transform can also be found in Choi et al. (1997). Skeletons can be used to represents shape models effectively (Trinh and Kimia 2007). This allows us to incorporate higher level model information effectively in a sequential way which significantly reduces the risk of accidental groupings of edge pixels. Since grouping is inherently a *sequential* process, we maximize a joint-posterior of contours and skeletons using Bayesian *filtering*. We adapt a practically very successful approach Rao-Blackwellized particle filtering which is used for the problem of Simultaneous Localization and Mapping (SLAM) in the field of robot mapping. We treat skeletons as trajectories of a virtual robot and the maps of associated edge segments as boundaries. The odometric and range constraints are replaced by perceptual grouping and model constraints. Perceptual constraints can be viewed as practical realization of Gestalt grouping principles (Wertheimer 1923).

Our work is related to the grouping method developed in Liu et al. (1998b) in that local symmetry axes are used. They identify segments along local symmetry-axis and apply a shortest-path algorithm to connect some of them into a complete symmetry axis. The grouping cost function is defined as the sum of local costs along the symmetry axis. In addition to using different measures, we have a more powerful computational framework in the proposed approach. However, the main difference is the usage of flexible shape models based on symmetry sets to guide contour grouping in our approach. Integer Quadratic Programming is used in Ren et al. (2005) to group contour segments based on constrained Delaunay triangulation. In contrast to our proposed approach, this approach fails in the presence of distractor edges induced by object inner structures. Moreover, grouping of only parts of contours is possible in our framework. This is also in contrast to active contour based methods (Blake and Isard 1997).

Figure 1 shows the algorithmic overview of the proposed approach. (1) For a given input image, an edge image is computed. (2) The edge pixels are then linked to form chains which are approximated as edge segments. We currently use publicly available code (Kovesi 2008). Other sophisticated linking algorithms like Tamrakar and Kimia (2007), Zhu et al. (2007) could also be used. (3) The edge chains produced by low-level linking algorithms are often too long and run into noise and boundaries of different objects. Hence the edge chains are split into “scale adaptive edge segments”. (4) Boundary and skeleton of the object of interest are grouped using the scale adaptive edge segments in a probabilistic framework. A reference shape model and perceptual grouping constraints from contour-skeleton duality are used as constraints. The grouping process is based on Rao-Blackwellized particle filtering framework. The first three steps can be considered as low-level preprocessing steps. The last step, which is our main contribution, exploits mid-level (contour-skeleton properties) and high-level (shape properties) of model objects.

This work is an extension of Adluru et al. (2007). We made several improvements that make our system more robust and also work on even small scale images. The improvements are summarized below.

1. Design of proposal and importance weights are very crucial in using the particle filtering framework. A complex optimal proposal as introduced in Zaritskii et al. (1975) and presented in Grisetti et al. (2007) was used in our previous work (Adluru et al. 2007). Using optimal proposal gives us advantage when observation sensors are more accurate than motion sensors. More specifically when the modes in proposal function are very different from the modes in the likelihood function. Since we want to keep our models flexible, we do not treat the model sensor to be accurate. Therefore, in this paper we use "prior proposal" following Handschin and Mayne (1969), Handschin (1970), Tanizaki (1997). The prior proposal captures the dynamic of our system namely contour and skeleton growth. The growth is regularized using constraints from contour-skeleton duality. Thus the proposal chooses smooth extensions for both contours and skeletons. While the model helps us distinguish smooth contours and skeletons from those of interest, we are not forced to follow the model constraints closely, thus improving model deformability. Also to handle the multiple modes in our proposal we use prior boosting (Gordon et al. 1993; Carpenter et al. 1999). The details of simulating this proposal are explained in Sect. 4.2.

2. We extend our shape model to be based on symmetry sets (Bruce et al. 1985) instead of medial axes paths since symmetry sets capture local symmetry of even *partial* contours. We choose the longest path in the symmetry set that represents the significant pair of boundaries in the model shape. The model constraints encode not only the shape of the boundary pair but also the shape of the symmetric path itself. The details of these improvements are explained in Sect. 5.

3. Orientations of edge pixels play an important role in our framework. In Adluru et al. (2007), we used line fitting based on EM (Latecki et al. 2006) and sampled equidistant pixels from the segments with the orientations of the segments. These equidistantly sampled pixels formed the set of edgels (edge pixels with orientations) which were then grouped. In the current system we exploit the low-level edge linking code of Kovesi (2008) and obtain "scale adaptive edge segments" which are then grouped. The scale adaptive edge segmenting is similar in spirit to breaking of chains into linear segments as in Ren et al. (2008). In Ren et al. (2008) chains are broken at the sharp curvatures. Here in addition to breaking at the sharp curvatures we break them based on surrounding edge distribution as explained in Sect. 6. We would like to note that our framework does not *depend* on edge-linking. It uses the low-level information if available. If no linking is available, we can just group edge pixels with orientations for example those given by third order gradient introduced in Tamrakar and Kimia (2007). The better the low level edge linker is, the faster the virtual robot marches in the grouping process, because it groups longer edge segments. Thus, this is a system level improvement that is practically very relevant especially since lot of progress has been made in low-level linking, for example Tamrakar and Kimia (2007), Zhu et al. (2007). The details of these implementations are explained in Sect. 6.

The rest of the paper is organized based on the algorithmic flow in our system. Section 3 describes our main framework in which the virtual robot groups the contours and skeletons using Rao-Blackwellized particle filtering. Section 4 describes the details related to simulation of our proposal based on perceptual constraints from contour-skeleton duality. Section 5 describes the evaluation of importance weights for the particles based on reference shape model constraints. The implementation details are presented in Sect. 6. Section 7 presents our experimental results and then finally conclusions and discussions are presented. We would like to note that a lot more experimental results and demonstrative videos are presented in our supplementary material submitted with this paper. Also we use the terms skeletons and symmetry axes interchangeably and our skeletons do not necessarily mean traditional medial axes.

3 Probabilistic Grouping of Contours and Skeletons

In this section we relate simultaneous grouping of contour segments and skeleton points to the problem of simultaneous localization and mapping (SLAM). Then we describe the probabilistic framework used for our grouping task.

3.1 Mapping and Localizing Simultaneously

A robot needs to localize itself in an environment for autonomous navigation. For a robot to localize using local sensors, it is important to have a map of the environment. Typically the maps are not available up front and might be quite inaccurate so the robot has to build the map of the environment by itself. There is a cyclic dependency between the tasks of mapping and localizing: mapping requires knowing the position of the robot, and robot's position can only be recorded in a map. Since the sensors are usually noisy, probabilistic approaches are needed and have been successfully applied for the problem. For a comprehensive survey see Thrun et al. (2005). In such probabilistic approaches a joint-posterior representing trajectory, $x_{1:t}$, of the robot and map, m_t, of the environment is maximized. A sequence of sensor measurements namely range measurements, $z_{1:t}$, and odometry readings, $u_{1:t}$, are used as constraints. The goal is to find:

$$\operatorname*{argmax}_{x_{1:t}, m_t} p(x_{1:t}, m_t | z_{1:t}, u_{1:t}) \tag{1}$$

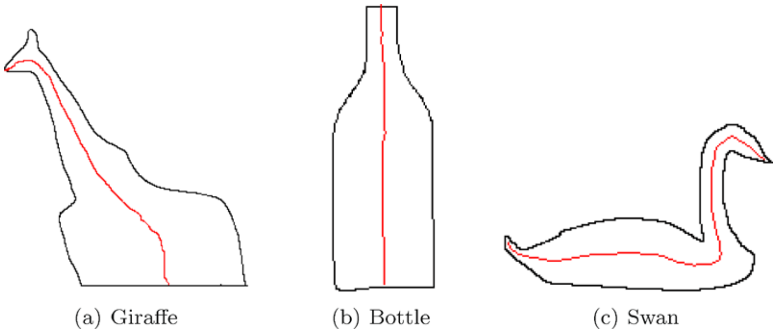


Fig. 2 The longest symmetric paths in the images form the bases of our shape models. The paths are subsets of the symmetry sets of the respective shapes. We choose the longest ones, since they capture the most significant boundary parts

Since both the trajectory and map are estimated simultaneously the problem is called *Simultaneous Localization and Mapping (SLAM)*. To allow for online optimization, *sequential* Monte Carlo estimation called *particle filters* have been successfully applied. The posterior is represented using a set of particles (random samples drawn from the posterior). Each particle, i, represents a trajectory $x_{1:t}^{(i)}$ up to time t and an associated map $m_t^{(i)}$ constructed up to time t. The dimensionality of the state being estimated plays a crucial role in efficient application of particle filters. Rao-Blackwellization allows to reduce the dimensionality by factorizing the states that are conditionally independent and can be estimated analytically. Since a map m_t can be analytically constructed for a given sequence of poses $x_{1:t}$ and observations $z_{1:t}$ (Moravec 1988), Rao-Blackwellization of the joint posterior results in

$$p(x_{1:t}, m_t | z_{1:t}, u_{1:t}) = p(m_t | x_{1:t}, z_{1:t}) p(x_{1:t} | z_{1:t}, u_{1:t}) \quad (2)$$

Once the state is decomposed any standard particle filter can be used, the most popular being Sampling Importance Resampling (SIR) filter (Grisetti et al. 2007). At every time step t, usually the most likely particle, i.e., the ith particle that maximizes $p(x_{1:t}^{(i)}, m_t^{(i)} | z_{1:t}, u_{1:t})$ is used for localization and navigation.

3.2 Grouping Contours and Skeletons Simultaneously

Our key idea is to maximize a *joint* posterior representing contours and skeletons. Intuitively we are trying to find the MAP (Maximum A Posteriori) estimate of:

$$p(\text{SKELETON, CONTOUR} \mid \text{MODEL,}$$
$$\text{GROUPING CONSTRAINTS}) \quad (3)$$

Bayesian estimation of skeletons given a complete contour i.e. $p(\text{SKELETON}|\text{CONTOUR})$ was presented in Feldman and Singh (2006). In our case we estimate both skeleton and contour simultaneously based on model and perceptual constraints from contour-skeleton duality. The model constraints are the high level constraints that actually help our robot distinguish the contours of object of interest from distractor contours that are "perceptually smooth" but do not belong to the boundary of object of interest. Our framework permits effective way of using model constraints for contour grouping. Our shape model is based on the symmetry set (Bruce et al. 1985) which can capture symmetry of even partial contours. Since the space of our symmetric points is a superset of the symmetry set (Sect. 4.1), by extending the model to be based on symmetry sets we can capture objects with partial contours. To keep our model simple we choose the longest path in symmetry set that captures the significant boundary parts of the model, see Fig. 2. Thus, we replace the set of medial axes paths used in Adluru et al. (2007) with a single path in the symmetry set. The details of the path information are explained in Sect. 5.

For ease of understanding the optimization problem, we cast the grouping problem as the mapping problem of an imaginary virtual robot. The virtual robot walks in the space of locally symmetric points (SPs) induced by pairs of edge segments. The construction of SPs, which generalizes medial axis points, is described in Sect. 4.1. The robot's trajectory composed of SPs represents a path in the space of locally symmetric points, $\mathbf{x}_{1:t}$ (SKELETON). The edge segments associated with the SPs form the contour map, c_t (CONTOUR). The robot's sensor information, Z_m, U_g (MODEL AND GROUPING CONSTRAINTS), is obtained from the reference shape model and grouping constraints based on contour-skeleton duality (Sect. 4.2). The goal is to maximize the joint posterior over the symmetric paths and contour maps:

$$\underset{\mathbf{x}_{1:t}, c_t}{\operatorname{argmax}} \, p(\mathbf{x}_{1:t}, c_t | Z_m, U_g) \quad (4)$$

Even though there are no explicit underlying dynamics based on real time, the grouping process is dynamic in the sense that the contours and skeletons *grow* in each step. Thus the dynamic system underlying the process is the "exploration". By inducing such virtual temporal information using the order of skeleton points ($\mathbf{x}_{1:t}$), we can produce *partial* skeletons and partial contours in case of occlusions of the objects of interest. The idea of inducing virtual temporal information for the grouping task is introduced in the seminal papers (Perez et al. 2001; Cox et al. 1993) which laid the foundation for Bayesian reasoning in the task of grouping. Since the contour c_t is conditionally independent and can be analytically computed given the sequence of skeleton points, $\mathbf{x}_{1:t}$, by grouping the corresponding contour segments of the symmetric point (Blum 1973), Rao-Blackwellization gives us the following equation in our case:

$$p(\mathbf{x}_{1:t}, c_t|Z_m, U_g) = p(c_t|\mathbf{x}_{1:t})p(\mathbf{x}_{1:t}|Z_m, U_g) \tag{5}$$

In the remainder of this section we provide the details of the Sampling Importance Resampling (SIR) particle filter for maximizing this posterior. Our goal is to find MAP estimate of the posterior $p(\mathbf{x}^{(i)}_{1:t}, c^{(i)}_t|Z_m, U_g)$. Thus a contour grouping decision is possible at every time step t. The contour $c^{(i)}_t$ and the skeleton $\mathbf{x}^{(i)}_{1:t}$ are determined by the most likely particles in the modes of the posterior $p(\mathbf{x}^{(i)}_{1:t}, c^{(i)}_t|Z_m, U_g)$. Clearly, at early grouping stages (for small t), we obtain only part of the contour and part of the skeleton of the object of interest. In each iteration, i.e., at every time step t, the following four steps are executed:

(1) *Sampling/Proposal*: The next generation of particles $\{\mathbf{x}^{(i)}_{1:t}\}$ is obtained from the current generation $\{\mathbf{x}^{(i)}_{1:t-1}\}$ by sampling from a proposal distribution $\pi(\mathbf{x}_{1:t}|Z_m, U_g)$ which is assumed to satisfy the following recursion:

$$\begin{aligned}&\pi(\mathbf{x}_{1:t}|Z_m, U_g)\\ &\quad = \pi(\mathbf{x}_t|\mathbf{x}_{1:t-1}, Z_m, U_g)\pi(\mathbf{x}_{1:t-1}|Z_m, U_g)\end{aligned} \tag{6}$$

Therefore, each particle is extended as $\mathbf{x}^{(i)}_{1:t} = \langle \mathbf{x}_t, \mathbf{x}^{(i)}_{1:t-1}\rangle$ where $\mathbf{x}_t \sim \pi(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)$.

(2) *Importance weighting/Evaluation*: Since it is usually hard to design $\pi(\mathbf{x}_{1:t}|Z_m, U_g)$ that exactly simulates the true posterior $p(\mathbf{x}_{1:t}|Z_m, U_g)$. An individual importance weight $w(\mathbf{x}^{(i)}_{1:t})$ is assigned to each particle, according to:

$$w(\mathbf{x}^{(i)}_{1:t}) = \frac{p(\mathbf{x}^{(i)}_{1:t}|Z_m, U_g)}{\pi(\mathbf{x}^{(i)}_{1:t}|Z_m, U_g)} \tag{7}$$

The weights $w(\mathbf{x}^{(i)}_{1:t})$ account for the fact that the proposal distribution π in general is not equal to the true distribution of successor states. Under 1st order Markovian assumption and conditional independence the weights can be recursively estimated as:

$$\begin{aligned}w(\mathbf{x}^{(i)}_{1:t}) &= \frac{p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)\; p(\mathbf{x}^{(i)}_{1:t-1}|Z_m, U_g)}{\pi(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)\; \pi(\mathbf{x}^{(i)}_{1:t-1}|Z_m, U_g)}\\ &= w(\mathbf{x}^{(i)}_{1:t-1})\frac{p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)}{\pi(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)}\\ &\propto w(\mathbf{x}^{(i)}_{1:t-1})\frac{p(Z_m|\mathbf{x}^{(i)}_{1:t-1}, \mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)}{\pi(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)}\\ &\quad \text{(using Bayes rule)}\end{aligned} \tag{8}$$

The proportionality in (8) is from normalization constant in Bayesian decomposition of $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g)$.

The choice of proposal π is a very important design criterion for successful implementation of a particle filter. The closer it is to true posterior the better the filter converges, with finite number of particles. In Adluru et al. (2007) we used a complex optimal proposal according to Doucet (1998). This was originally introduced in Zaritskii et al. (1975). The basic idea of the optimal proposal is to use the model constraints (in our case Z_m) in designing π. We simulated the optimal proposal by approximating it with a Gaussian similar to the technique described in Grisetti et al. (2007). However, integrating Z_m into the proposal restricts the model deformability, because the particles are forced to closely represent the model. Therefore, we no longer use Z_m in the proposal. Instead we use the prior distribution, $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)$ as our proposal, following Handschin and Mayne (1969), Handschin (1970), Tanizaki (1997). This proposal captures the virtual dynamic underlying our system i.e. “growth” of contours and skeletons. Since we prefer smooth contours and skeletons the prior distribution is simulated using grouping constraints, U_g from contour-skeleton duality. Hence in our case $\pi(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, Z_m, U_g) = p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)$. Using this simple exploration based proposal the weight recursion in (8) becomes:

$$\begin{aligned}w(\mathbf{x}^{(i)}_{1:t}) &\propto w(\mathbf{x}^{(i)}_{1:t-1})\frac{p(Z_m|\mathbf{x}^{(i)}_{1:t-1}, \mathbf{x}_t)\cancel{p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)}}{\cancel{p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)}}\\ &= w(\mathbf{x}^{(i)}_{1:t-1})p(Z_m|\mathbf{x}^{(i)}_{1:t-1}, \mathbf{x}_t)\end{aligned} \tag{9}$$

$p(Z_m|\mathbf{x}^{(i)}_{1:t-1}, \mathbf{x}_t)$ represents particle evaluation with respect to the shape model. The details of computing this likelihood are explained in Sect. 5. The details of simulating the proposal are explained in Sect. 4. Intuitively, our proposal is “exploration based” and we explore all SPs in a certain region of interest around $\mathbf{x}^{(i)}_{t-1}$ and assign them probability masses based on the “quality of continuation” determined using contour-skeleton duality constraints.

One important point to note is that *all* the posteriors in our case are *discrete* since the space of SPs is discrete.

(3) *Resampling*: For the particle filter to converge the variance of the weights has to be low since ideally the particles are supposed to represent *random* samples. To avoid the problem of “weight degeneracy” (Kong et al. 1994) particles with low importance weights are replaced by those with higher weights. This step is necessary since only a finite number of particles are used. Resampling is a key step that

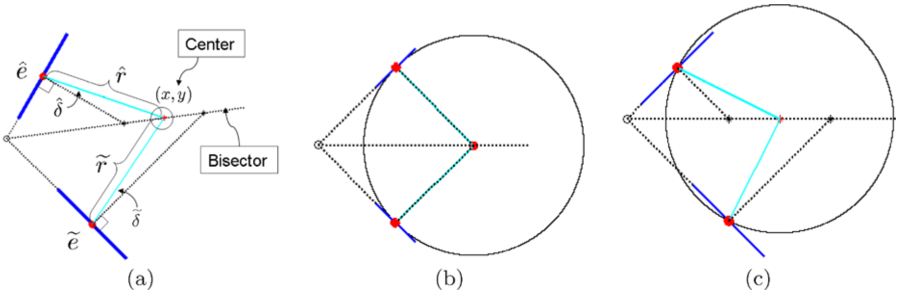


Fig. 3 (**a**) Construction and components of a symmetric point. (**b**) An SP with circle touching tangentially to the edgels. (**c**) To tolerate the inaccuracies in the edge orientations and the effects of discretization we allow SPs with approximately tangential circles

allows application of a particle filter in situations in which the true distribution differs from the proposal. We employ residual resampling (Liu et al. 1998a). Since resampling duplicates particles with higher weights there is a risk of "particle depletion". This occurs when all particles become identical which again means they are not random samples. To mitigate this effect an adaptive resampling schedule as proposed in Adluru et al. (2007) is used. The schedule is based on the measure

$$N_{eff}\left(=\frac{1}{\sum_{1=1}^{N_p}(w^{(i)})^2}\right)$$

introduced in Liu (1996), where N_p is the number of particles. This measure is related to the dispersion of weights and following Doucet et al. (2001), Grisetti et al. (2007), we resample only if $N_{eff} < N_p/2$.

(4) *Updating contour*: This step involves computing $p(c_t^{(i)}|\mathbf{x}_{1:t}^{(i)})$. The contour is represented as a discrete probability distribution of edge segments similar to occupancy maps (Moravec 1988). If an edge segment belongs to the contour its probability is one, otherwise it is zero. This is analytically computed because a contour can be reconstructed given a skeleton (Blum 1973). At each step it essentially involves connecting the edge segments in the image closest to those that are involved in generating $\mathbf{x}_{1:t}^{(i)}$. Since we store c_t per particle it essentially involves just appending the new edge segments to $c_{t-1}^{(i)}$. The edge segments grouped upto time t will have the probability one, while all other edge segments will have a zero probability of belonging to the contour. This is the step that allows us to "Rao-Blackwellize" the joint posterior.

4 Proposal Based on Regions of Interest

In this section we explain the details of simulating our proposal which we selected to be equal to the prior distribution, $p(\mathbf{x}_t|\mathbf{x}_{1:t-1}^{(i)}, U_g)$. Simulation of the prior distribution is based on the virtual dynamic of our system namely contour and skeleton growth. We first explain the space of locally symmetric points in which our robot walks and then explain the region of interest based simulation of the prior distribution.

4.1 Symmetric Points

Given a pair of edge pixels $(\widehat{e}, \widetilde{e})$ with orientations, called edgels, a symmetric point (SP) $\mathbf{x} = (x, y)$ is computed as the center of the projections of the edge normals onto the bisector as shown in Fig. 3(a). The SP is a generalization of the center of maximal disk in the definition of medial axis. By this construction an SP $\mathbf{x}$ is characterized by

1. Distances $\widehat{r} = \|\widehat{e} - \mathbf{x}\|$ and $\widetilde{r} = \|\widetilde{e} - \mathbf{x}\|$. Since we want the SP, $\mathbf{x}$ to remain a center of a disk, we require that $\widehat{r} = \widetilde{r}$ (with some very small tolerance).
2. Deviation angles $\widehat{\delta}$ and $\widetilde{\delta}$ are defined as the angles between the normal at $\widehat{e}$ and vector $\overrightarrow{\widehat{e}\mathbf{x}}$ and that between the normal at $\widetilde{e}$ and $\overrightarrow{\widetilde{e}\mathbf{x}}$ respectively.

If the deviation angles $\widehat{\delta} = \widetilde{\delta} = 0°$, the disk with center (x, y) and radius $r = \widehat{r} = \widetilde{r}$ is tangential to both edgels, see Fig. 3(b). However, we cannot require the deviation angles to be zero due to inaccuracies in the directions of edgels and the effects of discretization. Hence we must tolerate SPs for which $\widehat{\delta}, \widetilde{\delta} > 0°$, see Fig. 3(c). The smaller the angles the more tangential is the disk. Hence we measure the quality of an SP by the tangentiality of the disk. It is computed as:

$$\mathcal{N}(\delta, 0, S_\sigma), \quad \text{where } \delta = \frac{\widehat{\delta} + \widetilde{\delta}}{2} \tag{10}$$

where $\mathcal{N}(x, \mu, \sigma)$ is the Gaussian function with mean μ and standard deviation σ evaluated at x and S_σ is the tolerance parameter for skeleton regularization.

The space of SPs is related to the space of symmetry set (SS) and evolute of the object of a given contour. We obtain that the set of SPs is equal to SS if $\widehat{\delta} = \widetilde{\delta} = 0°$, since SS is the locus of centers of disks that are tangential in at least two distinct points on the contour (Bruce et al. 1985) i.e., the requirement of maximal disks is dropped. If we allow both $\widehat{\delta}, \widetilde{\delta}$ to be larger than 0°, the set of SPs is a superset of the union of SS and the evolute. For example in Fig. 4(c) we show SPs where the average of $\widehat{\delta}$ and $\widetilde{\delta}$ is $\leq 1°$. Evolute is the locus of the centers of curvature, i.e. centers of disks that osculate the object boundary (Lockwood 2007). Figure 4(b)

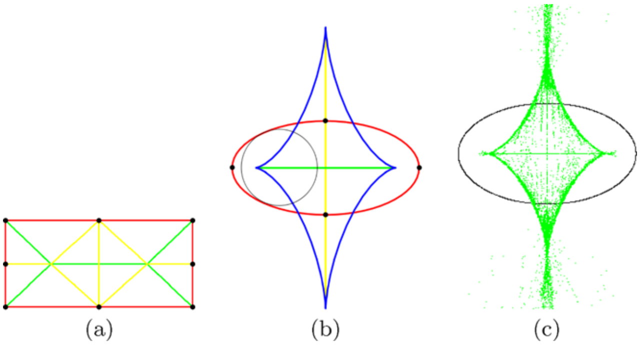


Fig. 4 (Color online) (**a**) Medial axis (*green line*), symmetry set (*green* and *yellow*) for a rectangle (*red*). The evolute for the rectangle is just at the corners since the boundary has curvature only at the corners. (**b**) Medial axis (*green line*), symmetry set (*green* and *yellow lines*) and evolute (*blue curve*) for an ellipse (*red*). (**c**) The space of symmetric points for an ellipse with the deviation angle threshold of 1°

shows the MA, SS and evolute of an ellipse[1] and Fig. 4(a) shows MA and SS of a rectangle.

Even though MA completely represents a shape it cannot be used to capture symmetries of partial shapes. SS forms a superset of MA because it is similar to MA except that it does not require the circles to be maximally inscribing. The main difference between the space of SPs and SS and evolute is that we do not need the knowledge of true object boundary in the construction of SPs. Kuijper and Olsen (2005) presents a technique for extracting skeletons using the symmetry set when boundaries are clearly defined. In contrast we try to extract skeletons and boundary simultaneously using the set of SPs.

4.2 Simulation of Prior Distribution

Now we explain the details of simulating our proposal distribution, $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)$. There are two main steps in simulating our proposal namely defining the region of interest for $\mathbf{x}_t$ and assigning probability masses to the symmetric points in that region of interest. The region of interest is based on $\mathbf{x}^{(i)}_{1:t-1}$ and the probability masses are computed according to the grouping constraints from contour-skeleton duality, U_g. Thus we obtain a discrete probability distribution approximating $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{1:t-1}, U_g)$ from which we sample the followers. Note that this discrete distribution is computed per particle.

The first step of designing region of interest is motivated by the recent work in *active* SLAM (Stachniss et al. 2005; Chang et al. 2007). The basic idea in *active* SLAM is to plan robot's action to optimize its exploration of the environment by choosing from a set of actions weighed by the information gain. Since the goal of our virtual robot is to group the contour and skeleton of the object of interest we distinguish explored and unexplored regions and make the robot move into unexplored regions of interest.

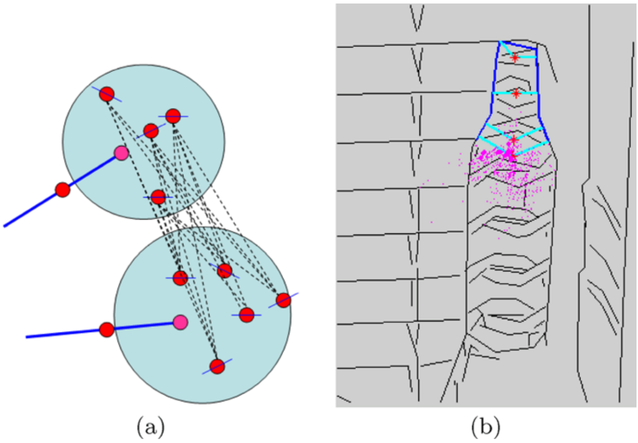


Fig. 5 (Color online) (**a**) The circular look ahead regions for possible contour extensions are shown as *cyan circles*. The possible extending edge segments in these regions are shown as *red dots* along with directions shown with *short lines*. All possible pairings of these edge segments generate the discrete space of $\mathbf{x}_t$s. (**b**) The contour grouped up to time $t-1$ i.e. $c^{(i)}_{t-1}$, is shown in *blue*. The region enclosed by this contour is designated as *explored region* for the virtual robot. We do not allow the look ahead segments to be in the *explored region*. The discrete space of $\mathbf{x}_t$s is shown with *magenta points*. The size of the *dots* is proportional to the probability mass computed according to $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{t-1}, U_g)$

Using $\mathbf{x}^{(i)}_{1:t-1}$ we can compute the explored region which is the interior of the contour, $c^{(i)}_{t-1}$ grouped up to time $t-1$. Since we group pairs of contours along with skeleton points the interior of the boundary is known. We use a circular neighborhood of the endpoints of edge segments in unexplored regions (i.e., regions that are not enclosed by $c^{(i)}_{t-1}$) for possible extensions to the boundary. This can be seen in Fig. 5(a). The edge segments in the look ahead region are paired and then some pairs are culled in similar spirit to the culling used in DP-SLAM (Eliazar and Parr 2004). We cull the pairs that extend boundary or skeleton with self-intersections, those that can not be used for inducing an SP, and those that induce an SP with radius out of range of the model radii. After culling, each pair of edge segments induces a possible follower SP, $\mathbf{x}_t$. Thus, at each step the virtual robot tries to extend the skeleton by expanding the contour. The process of looking ahead and the discrete space of $\mathbf{x}_t$s are shown in Fig. 5. The $\mathbf{x}_t$s in the region of interest are assigned probability masses according to $p(\mathbf{x}_t|\mathbf{x}^{(i)}_{t-1}, U_g)$. Intuitively this means computing the likelihood of $\mathbf{x}_t$ being a follower of $\mathbf{x}^{(i)}_{t-1}$ given the perceptual grouping constraints U_g. The perceptual grouping constraints, U_g can be understood as a practical realization of the Gestalt psychology principles in perception (Wertheimer 1958). In comparison to typical grouping approaches where a single contour is grown, we grow a pair of contours and their symmetry axis

[1] Figure taken from Wikipedia: http://en.wikipedia.org/wiki/.

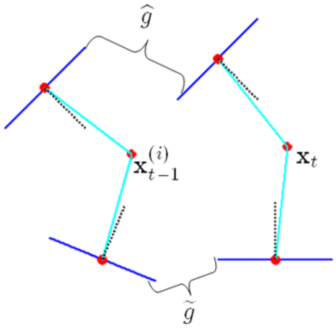


Fig. 6 The average gap $g_t = (\widehat{g} + \widetilde{g})/2$ introduced by appending edge segments of SP $\mathbf{x}_t$ to those of SP $\mathbf{x}_{t-1}^{(i)}$

simultaneously. Hence our perceptual grouping constraints are motivated from contour-skeleton duality.

The skeleton growth constraint prefers robot's trajectory along good symmetric points (SPs). The quality of a symmetric point, $\mathbf{x}$ is measured as the average of its deviation angles, $\widehat{\delta}, \widetilde{\delta}$ as shown in Fig. 3(a). The smaller the average deviation the higher the quality of the SP. Figure 3(b, c) show examples of a tangential (good) and a nearly tangential (not so good) SP. Both SPs are allowed but the SP in (c) is of lower quality than that in (b). It is not possible for the robot to always be able to walk along good symmetric points because of noise in the boundary orientations and gaps in the boundary. The gaps are present because of discretization and missing parts of true edges. Hence we need a tolerance measure that restricts the drop in quality of SP. Since we regularize the skeleton *growth* the likelihood is computed based on the *change* in deviation angles of consecutive SPs. Therefore, we only penalize the drop in the quality of SP, $\mathbf{x}_t$ from that of the previous SP, $\mathbf{x}_{t-1}^{(i)}$. The likelihood of $\mathbf{x}_t$ as a follower SP according to this constraint is computed as $\mathcal{N}(s_t, 0, S_\sigma)$ which is a function of $s_t = \max(\delta_t - \delta_{t-1}^{(i)}, 0)$. δ_t and $\delta_{t-1}^{(i)}$ are the average deviation angles of $\mathbf{x}_t$ and $\mathbf{x}_{t-1}^{(i)}$ respectively. S_σ represents the tolerance for the constraint violation.

The contour growth constraint favors boundaries with smaller gaps. Let $g_t = (\widehat{g} + \widetilde{g})/2$ be the average gap between the edge segments of $\mathbf{x}_{t-1}^{(i)}$ and those of $\mathbf{x}_t$ as shown in Fig. 6. We use Gaussian $\mathcal{N}(g_t, 0, G_\sigma)$ to express the likelihood of $\mathbf{x}_t$ according to this constraint. Since contour gaps are unavoidable in real images because of missing gradient information, we have a certain tolerance of G_σ units in terms of pixels or sub-pixels.

Using the above perceptually motivated constraints, we compute the probability mass of $\mathbf{x}_t$ as:

$$p(\mathbf{x}_t|\mathbf{x}_{1:t-1}^{(i)}, U_g) = \mathcal{N}(g_t, 0, G_\sigma) \cdot \mathcal{N}(s_t, 0, S_\sigma). \tag{11}$$

Usually, the discrete distribution of $p(\mathbf{x}_t|\mathbf{x}_{1:t-1}^{(i)}, U_g)$ is multi-modal because of distractor segments and interior structures. It is important to have particles representing all meaningful regions of the posterior to be able to recover from distractions by noise. We use "prior boosting" (Gordon et al. 1993; Carpenter et al. 1999) so as to capture multimodal likelihood regions. In prior boosting more than one follower is sampled from $p(\mathbf{x}_t|\mathbf{x}_{1:t-1}^{(i)}, U_g)$, for each particle so that different followers can capture different modes of the likelihood of the posterior. The number of followers sampled typically depends on the number of modes which in turn depends on the local geometric conditions: if there are many distractor segments, then lot of followers are sampled and if there are less distractor segments then fewer followers are sampled. Thus after each iteration of Sampling Importance Resampling we might have more number of particles $N'_p > N_p$ but we retain only N_p *with their weights*. An important difference between resampling a fixed number of particles from large number of particles is that we retain their weights while after the adaptive resampling step the weights of all the particles are set to be equal.

An important parameter in the proposal is the size of the circular neighborhood L, which decides how far the virtual robot can "look ahead" to define the region of interest that can capture the likelihood regions of the posterior. If the robot looks too far ahead, then it will be less sure if the boundary and skeleton extensions are true. On the other hand if it does not look far enough it might be distracted by noise and gaps in the true contour. The details of choosing the value of this and other parameters are explained in Sect. 6.

5 Evaluation Based on Shape Model Constraints

The importance weights for the particles are computed recursively according to (9), where $p(Z_m|\mathbf{x}_{1:t-1}^{(i)}, \mathbf{x}_t)$ measures how likely the particle $\mathbf{x}_{1:t}^{(i)}$ is according to the sensor information Z_m. In this section we explain the sensor modeling for our virtual robot, which is based on the constraints from shape model.

As described in Sect. 3.2 our shape model is composed of the longest path in its symmetry set. The path is represented using a sequence of radii of the disks and *displacement vectors* of sample points on the path. The sequence of radii captures the shape of the contour generating the skeleton path, while the sequence of displacement vectors captures the shape of the skeleton path. Thus Z_m captures the shapes of both contour and major symmetric axis of the model image. Formally, if there are N sample points, $Z_m = \langle\{\overrightarrow{D_1}, R_1\}, \{\overrightarrow{D_2}, R_2\}, \ldots, \{\overrightarrow{D_N}, R_N\}\rangle$, where $\overrightarrow{D_k}, R_k$ are the displacement vector and the radius of the kth sample point respectively. A sample swan model is illustrated in Fig. 7.

Even though we have the *sequence* information for the path, the robot has *all* the information in each iteration.

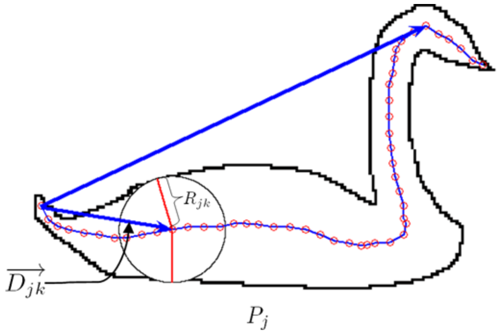


Fig. 7 (Color online) The swan model has one symmetric path: from tail to the beak. The path is shown in *blue*. The sample points are shown as *red circles*. Displacement vectors of two sample points are shown as *blue arrows* and the radius of one of the sample points is shown using the maximal disk

Hence this model is "static". At every time step (iteration), the robot computes its displacement vector since it knows its starting pose. Then it finds the closest displacement vector in the model and uses the expected radius associated with that sample point. Thus, the robot uses global information of the shape model to make a local decision in the grouping process. Formally, if the robot's current displacement vector is $\overrightarrow{(\mathbf{x}_1, \mathbf{x}_t)}$ which is a function of $\mathbf{x}_t$ and $\mathbf{x}_1$ (the starting pose), we first obtain:

$$\hat{k} = \operatorname*{argmin}_k d_s(\overrightarrow{D_k} - \overrightarrow{(\mathbf{x}_1, \mathbf{x}_t)}), \quad k = \{1 \dots N\} \tag{12}$$

where d_s measures the dissimilarity between two vectors, i.e., the distance between their orientations and magnitudes. Then using the index $\hat{k}$ we obtain the expected radius, $\mathcal{R}(\mathbf{x}_1, \mathbf{x}_t) = R_{\hat{k}}$ and the deviation from the closest displacement vector, $\mathcal{D}(\mathbf{x}_1, \mathbf{x}_t) = d_s(\overrightarrow{D_{\hat{k}}} - \overrightarrow{(\mathbf{x}_1, \mathbf{x}_t)})$. The values $\mathcal{R}(\mathbf{x}_1, \mathbf{x}_t)$ and $\mathcal{D}(\mathbf{x}_1, \mathbf{x}_t)$ constitute the sensor readings from the shape model for the robot at $\mathbf{x}_t$, whose trajectory started at $\mathbf{x}_1$. Using these sensor readings we compute

$$p(Z_m|\mathbf{x}^{(i)}_{1:t-1}, \mathbf{x}_t) = 0.5 \cdot \mathcal{N}(\mathcal{D}(\mathbf{x}^{(i)}_1, \mathbf{x}_t), 0, D_\sigma) + 0.5 \cdot \mathcal{N}(r(\mathbf{x}_t), \mathcal{R}(\mathbf{x}^{(i)}_1, \mathbf{x}_t), R_\sigma) \tag{13}$$

where D_σ and R_σ are the tolerance parameters for deformability in shape of the skeleton path and shape of the contour from model, respectively. $r(\mathbf{x}_t)$ is the radius of the SP, $\mathbf{x}_t$. The use of displacement vectors to obtain the expected radius at each step requires that one of the ends of the shape is present in the image. This is not a serious limitation because we can have two-way path for each model. The intuition behind this reasoning is that if neither ends of the path is present then the shape itself might be not be recognizable. Different parts of the object have different saliencies. Usually the more salient parts can be expected to be associated with the ends of symmetric paths.

The main advantages from the model proposed in this paper over the model in Adluru et al. (2007) are:

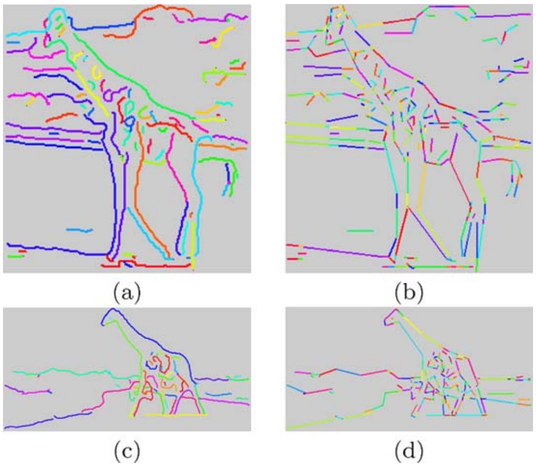


Fig. 8 (**a**, **c**) Chains of edge pixels extracted using Kovesi (2008). (**b**, **d**) The edge segments obtained by adaptively segmenting the chains. The chains and segments are shown in different random colors using the visualization tool of Kovesi (2008)

- By using the longest symmetric path as the model, we can capture the most significant pair of contours even if the complete contour is not available.
- By using the path information in a static way the robot uses global information from the model to make local decisions. This allows us to ignore minor shape details which can be easily confused with noise in real images.
- By using displacement vectors to obtain the expected radius at a time step (instead of using a sequence of radii), we have decoupled the parameters for controlling model deformability in terms of contour and skeleton. The deformability is captured by the deformability in the trajectory of the robot. The larger the tolerance, D_σ for $\mathcal{D}(\mathbf{x}^{(i)}_1, \mathbf{x}_t)$ is, the more deformable the trajectory (shape of skeleton) can be. The larger the R_σ i.e. tolerance for deviation from $\mathcal{R}(\mathbf{x}^{(i)}_1, \mathbf{x}_t)$ is, the more deformable the contour can be.
- The use of displacement vectors allows us to group larger pieces of contour segments by exploiting low-level linking algorithms without loosening model constraints.

6 Implementation Details

In this section we explain the details of using the low-level edge linking to reduce the space of SPs for faster grouping and the values of the parameters used. To take advantage of the progress made in low-level linking of edge pixels we first obtain edge chains. We use publicly available code (Kovesi 2008) but any other low-level linker can be used, for e.g. Zhu et al. (2007). The edge linking process generally produces smooth chains of edge pixels as shown in Fig. 8(a), but the edge chains often are too long and run into noise. Hence

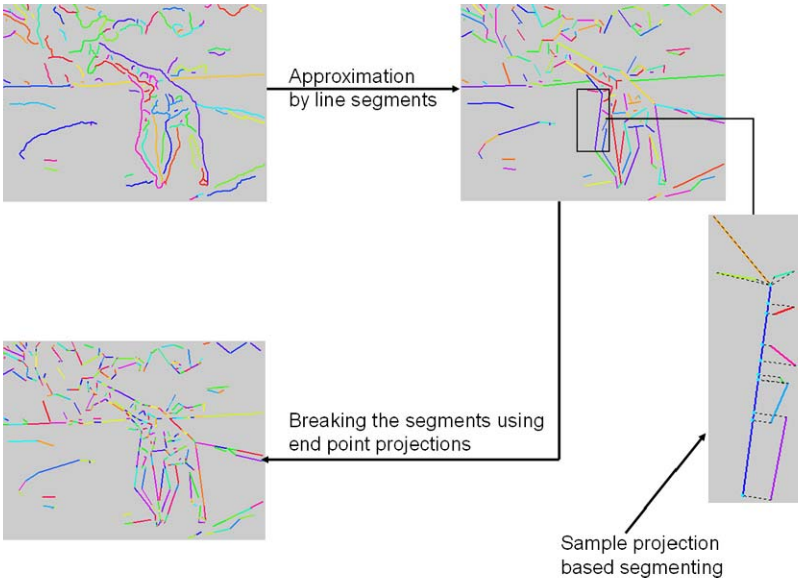


Fig. 9 (Color online) Process of splitting edge links based on spatial distribution of chains. The chains are approximated by *line segments*. Each *segment* is then broken by the projections of the end points of the neighboring *segments*. In the sample, the *long blue segment* is broken by the projections from its neighboring *segments*. The projections are shown as *dotted lines* and the *break points* in *cyan*

we split these chains into small linear segments adaptively based on the distribution of surrounding chains and then perform grouping on these small linear segments which we call *scale adaptive edge segments*. The scale adaptive edge segmenting is similar in spirit to the curvature based splitting of curves into linear segments used in Ren et al. (2008). A sample set of scale adaptive edge segments can be seen in Fig. 8(b). The higher the noise the shorter the edge segments are. In the worst case the chains are broken down all the way to edgels (edge pixels with orientations). The smaller the number of surrounding chains, the longer the resulting edge segments are. For example the neck of the giraffe in Fig. 8(a) is surrounded by more chains compared to that in (c). Hence the scale adaptive edge segments of neck of the giraffe in (b) are shorter than those of the giraffe in (d).

Scale adaptive edge segments are generated as follows: First, the chains are approximated by polylines using the publicly available code (Kovesi 2008). Then each segment is split into smaller segments using the projections of the end points of neighboring segments. Only the end points within twice the range of the look ahead parameter, L (used in simulating prior (Sect. 4.2)) are used. Thus the number of pieces a segment is split into depends on the spatial distribution of segments around it. Figure 9 demonstrates these steps.

To mitigate the effect of discretization on the turn angles at the junction points, we introduce virtual edge segments at those points. They are used to generate a set of SPs that can captures symmetries involving junctions in a better way. The orientation of a virtual segment at a junction is set equal to the mean of the orientations of the segments incident at the junction point. A sample set of virtual segments can be seen in Fig. 10.

Now we explain the state space of symmetric points (SPs) which captures local symmetries of our *scale adaptive edge segments*. Since the number of such edge segments is much smaller than the edge pixels, the number of SPs is significantly reduced in comparison to the full space of SPs generated by all edgel pairs. This reduced space of SPs forms the locally symmetric space for our virtual robot, and hence our robot marches faster, which improves the computation speed. For each pair of the edge segments we compute an SP as follows. We first sample equidistant points on each of the edge segment. Orientations of the points are set to the orientations of the respective edge segments. Then we compute the SPs between all pairs of the sampled points. Only the best SP according to the quality measure in (10) is selected as the SP induced by the two edge segments. The reduced and the full space of SPs are shown in Fig. 11. The reduced space is shown as red dots while the full space is shown as black dots.

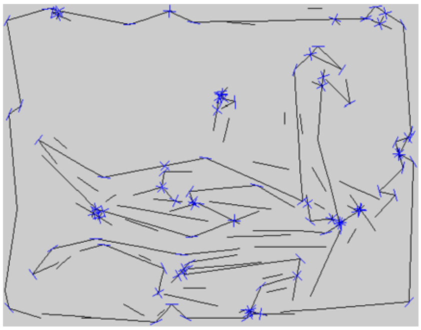


Fig. 10 (Color online) Virtual segments are introduced at the junction points to mitigate the effect of discretization in generation of symmetric points. The junction points are identified by intersections of the *line segments* within certain range. The orientation of a virtual segment at a junction is the average of the orientations of the segments inducing the junction point. The virtual *segments* are shown in *blue*. Their lengths are magnified for visualization

Now we explain the values of the parameters used in our system. G_σ, S_σ, L are used in simulating the prior based proposal distribution, and D_σ, R_σ are used in computing the importance weights of the particles. We set $D_\sigma = 0.7$, $R_\sigma = 0.2$ since the readings from the model are normalized using scale factor. That is we are more tolerant to the deformability in the shape of the skeleton but stricter with the deviation in the radii. The heuristic behind such choice is that under non-rigid transformations the shape of a skeleton path changes but the radii along the path do not. Hence we can be more tolerant to the skeleton paths. G_σ is set equal to the average length of the scale adaptive edge segments. The heuristic behind this is that the longer the scale adaptive segments the less noisy the image is and hence we can be more tolerant to gaps in the boundary. We set $S_\sigma = 45°$ in order to tolerate possible large inaccuracies in the directions of edges. L is set to be equal to $e_\mu + 0.5e_\sigma$ where e_μ is the average length of the edge segments approximating the edge chains before being split into scale adaptive segments. e_σ is the standard deviation of the lengths of the segments. The heuristic is that the longer the edge segments are the further the robot should look ahead to have meaningful followers.

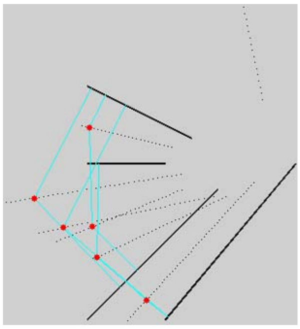
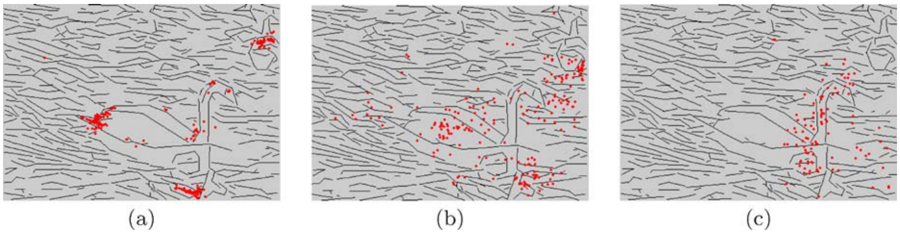


Fig. 11 (Color online) SPs (*black dots*) are computed between pairs of sampled points from different segments. The best SPs according to the quality measure based on the deviation angles are shown as *red dots* and the *corresponding intending lines* are shown in *cyan*

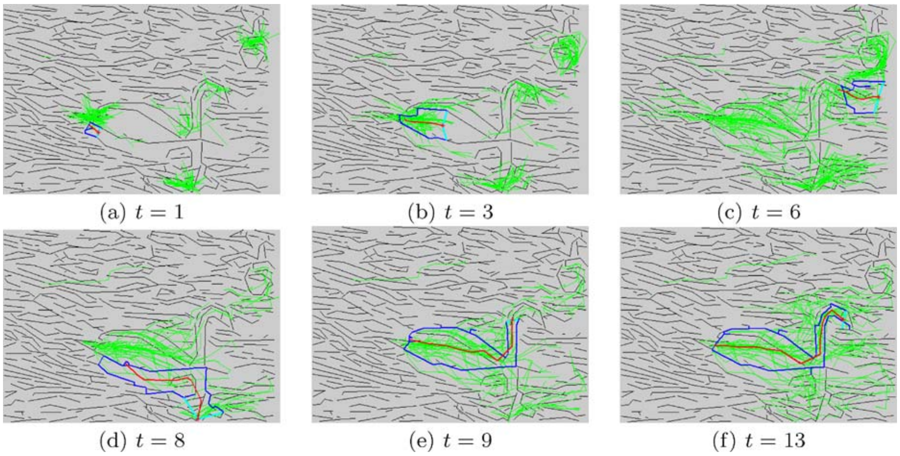
7 Experimental Results

We demonstrate the results of our technique using ETHZ dataset (Ferrari et al. 2006). We selected this data-set since it includes pre-computed edge images and model shapes for each class. The edge chains are obtained using Kovesi (2008). For initialization a threshold on the gradient strength is applied. The robot is initialized around end points of all strong chains. An example initial set of poses for the robot can be seen in Fig. 12(a). After several iterations the robot localizes itself on to the object of interest because of model constraints. The process can be seen in the Fig. 12(a)–(c). Also a video showing the process is attached as a supplementary material to this paper. This is similar to global localization in robot mapping domain where the initial position of the robot is unknown (Fox 2003). The difference between localization and *global localization* is that in localization, the robot's initial pose is known while in global localization its initial pose is unknown. We would like to note that a threshold on gradient strength is used only for initialization of particles not when the filtering is in progress. The



(a) (b) (c)

Fig. 12 (**a**) The initial set of particles, $\{\mathbf{x}_1^{(i)}\}_{i=1}^{N_p}$ sampled around the end points of thresholded edge chains. (**b**) and (**c**) The distribution of $\mathbf{x}_t$s at $t = 5$ and 10. As the filter progresses the particles are concentrated around the object of interest, because of the model constraints



(a) $t = 1$ (b) $t = 3$ (c) $t = 6$ (d) $t = 8$ (e) $t = 9$ (f) $t = 13$

Fig. 13 (Color online) The evolution of the particle filter. The multiple hypotheses for the *skeletons* are shown in *green*. The most likely *skeleton* is shown in *red* and the most likely *boundary* is shown in *blue*. The space of SPs is shown as *gray* points

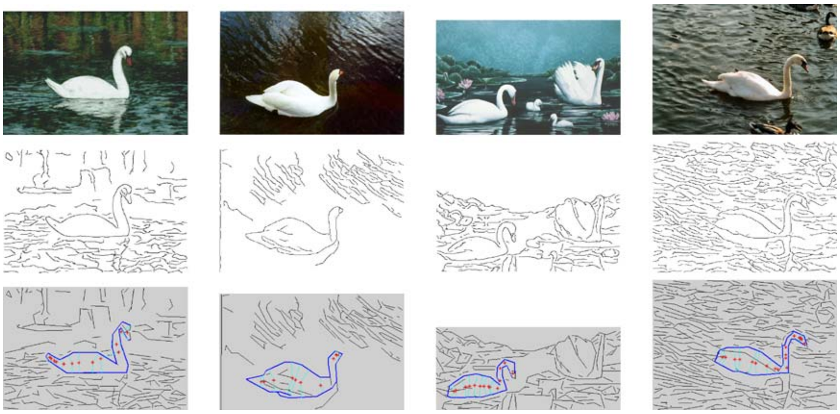


Fig. 14 Grouping results on swan images. *Top row*: original images; *middle row*: edge images; *bottom row*: grouping results

filtering process can be viewed as multi-hypotheses tracking where several hypotheses for skeletons and contours are maintained to avoid getting stuck in local optima of the grouping process. A sample evolution of the filtering process can be seen in Fig. 13. The green curves are the multiple hypotheses for the skeletons while the most likely one is shown in red. The most likely boundary is shown in blue. A sample recovery from local optima can be seen from Fig. 13(c), (d) and (e).

The filtering stops when the MAP estimate of the posterior $p(\mathbf{x}_{1:t}, c_t|Z_m, U_g)$ drops below a certain threshold Δp. This stop criterion is needed so as to make partial grouping possible. Also it is important to note that this posterior is different from the prior probabilities, $p(\mathbf{x}_t|\mathbf{x}_{1:t-1}, U_g)$ of the follower symmetric points ($\mathbf{x}_t$s). With enough particles this allows for both grouping "non-smooth" contours that confirm the model and distinguishing smooth contours that do not confirm the model. The best particles in all the modes of the posterior are selected as the skeletons, which induce the target contours. Typically the number of modes in the posterior is equal to the number of objects satisfying the model.

We present results on three classes of objects swans, bottles and giraffes whose model shapes are shown in Fig. 2. A sample of our results can be seen in Figs. 14, 15 and 16. Many more results are included in the supplementary material submitted with this paper. The large number of edges in the test images that do not belong to target contours demonstrates that we are able to group contour segments in the presence of distractor segments between locally symmetric contour pieces. Thus the proposed approach has strong potential for applications on real images.

The shape variability of grouped contours in Fig. 14, which were all obtain with a single shape model of a swan, demonstrates the flexibility of the proposed shape model. The second column of Fig. 15 shows an example of grouping multiple objects, while the fourth column shows an example of grouping partial contours when occluded by other objects. The generality of our symmetry set based model can be seen in Fig. 16, where we are able to ignore the shape details of legs of the giraffes and capture the significant parts of the giraffes namely the neck and the body.

It is important to note we are not able to find objects whose orientation is very different from the orientation of the model image. For example, in the third column of Fig. 14 we are able to group only one swan. Since we normalize the model readings to the size of the model shape we expect a scaling factor to be known for the object we are trying to group. The scaling factor is currently set manually.

Our framework seems similar to shape-based object detection. But there is a key fundamental difference to the other object detection techniques (like Zhu et al. 2008; Ferrari et al. 2008): we build matching tokens and perform matching *simultaneously*. For e.g. our grouping tokens can be edgels while the other approaches *require* a preprocessing step of grouping contour pieces using low-level techniques. Hence our experimental set-up is quite different and is similar to that in shape-prior based segmentation in Kumar et al. (2005), for example, we run the swan images with swan model. Further the detection success is usually measured using bounding-box overlap. In our case success is measured by object boundary overlap which tends to be more precise detection. Therefore an exact and fair comparative evaluation of our system is arguably not possible. However we perform quantitative and comparative evaluation of our system as described below.

For each class we run our system with the class specific model. We report the retrieval rates based on the number of instances whose contours were grouped successfully. Even though a fundamentally different paradigm we ran the publicly available system of Ferrari et al. (2008) in a similar fashion i.e. each class with the specific class model. The key and fundamental differences between our system and Ferrari et al. (2008) are:

- We consider the object is successfully retrieved if we group at least 80% of the boundary of the target object present in the edge image. The detection success in Ferrari et al. (2008) is determined by a bounding box overlap percentage. We report the results for overlap percentages

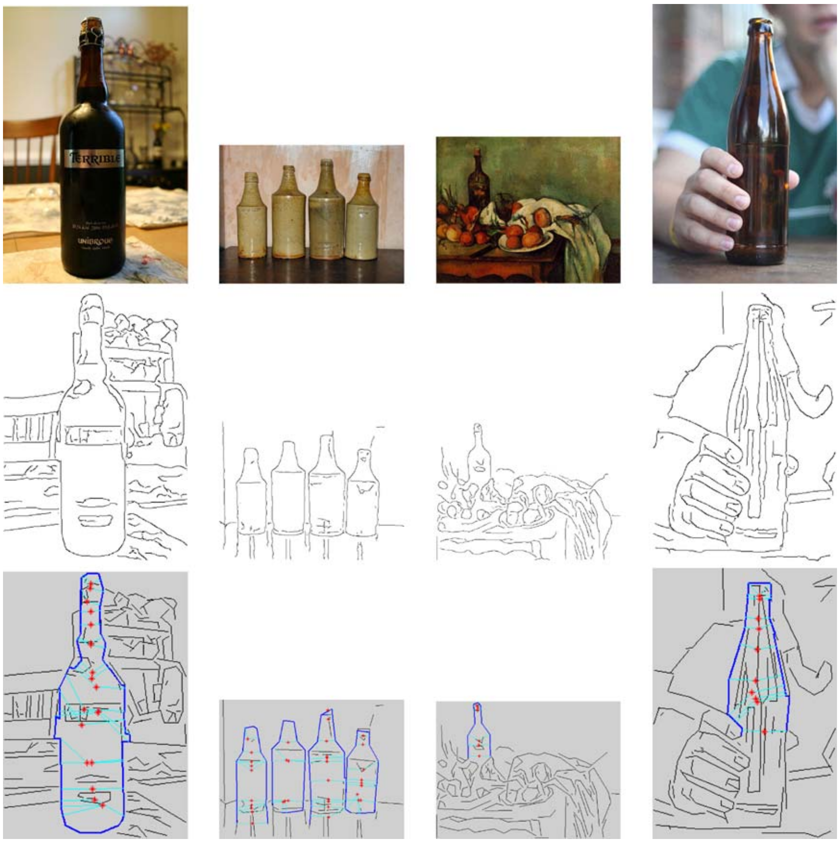


Fig. 15 Grouping results on bottle images. *Top row*: original images; *middle row*: edge images; *bottom row*: grouping results

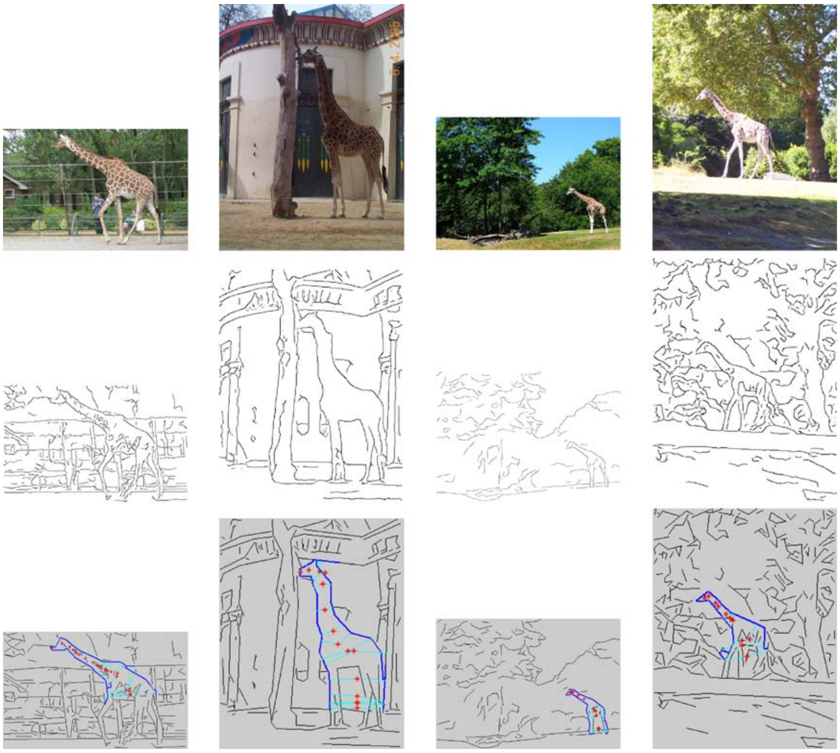


Fig. 16 Grouping results on giraffe images. *Top row*: original images; *middle row*: edge images; *bottom row*: grouping results

of 50%, 60%, 70% and 80%. The bounding box overlap of 80% appears to be most comparable to our definition of successful retrieval.

– In our system we use the *single* hand-drawn models (shown in Fig. 2) for each class. In Ferrari et al. (2008) each class has multiple models learnt from a training sub-

Table 1 Comparison between Ferrari et al. (2008) and our method. F-x% denotes the retrieval result of Ferrari et al. (2008) using x% of bounding-box overlap. See text for details

Class	Total instances	F-50%	F-60%	F-70%	F-80%	**Our method**
Swans	16	93.75%	93.75%	81.25%	62.5%	**87.5%**
Bottles	30	91.667%	87.5%	83.333%	54.84%	**80%**
Giraffes	44	82.222%	75%	59.09%	15.91%	**45.45%**

Table 2 Retrieval results using our method on the ETHZ dataset

Class	Total instances	% Retrieved
Swans	33	81.82%
Bottles	55	85.45%
Giraffes	91	51.65%

set of the instances. Therefore, we report only the retrieval rate of the best performing model for their system. This gives their system a clear advantage.

In Table 1 we report comparisons of the two methods on the same test images and test instances of target objects. We would also like to note that the bounding box accuracy does not necessarily imply accuracy in *boundary* detection. In Ferrari et al. (2008) they use a subset of the dataset to learn the model and hence their test instances are only a subset of the dataset. Since we do not learn any model we can test our system on all instances. In Table 2 we present quantitative evaluation of our system on all instances for each class.

The number of particles needed for our experiments are determined empirically. On average we need about 200 to 500 particles. In general the number of particles required depends on the amount of clutter in an image. This is because particles can be viewed as multiple hypotheses needed to make delayed decisions after enough evidence has been accumulated.

The most computational burden of our system in simulating the prior distribution (described in Sect. 4.2). Prior has to be simulated for every unique particle. The average grouping time for noisy images is 5–8 minutes while for less noisy images it is about 2–4 minutes on a computer with Intel(R) Pentium(R) D CPU with 3.40 GHz and 0.99 GB of RAM. We would like to note that our system combines local information sequentially to make global optimal decisions so the speed of our system is quite independent of the resolution of the image unlike some methods where they build the global information of the image and then optimize it using graph-based techniques (for e.g. Stahl and Wang 2006).

8 Conclusion and Discussions

Object detection and recognition by shape has a long history in computer vision. The basic idea is to extract some shape features and define distances between shapes using those features. Many heuristics are used in designing the features and distances so that the distances can be used to label or classify objects in accord with human perception. Some examples include Siddiqi et al. (1999), Belongie et al. (2002), Ling and Jacobs (2007), Bai and Latecki (2008), Yang et al. (2008). Though such methods have some invariance capabilities to handle some deformations they cannot handle clutter and artifacts which commonly occur in real images. It is important to note that this is not their limitation per se because they are designed assuming the objects are already separated from the clutter or background. Separating objects from background has been a low-level vision problem for many years but for fully automatic segmentation it has to be combined with a recognition process. Hence researchers have been focussing on unified approaches for object recognition and segmentation (Borenstein and Ullman 2002; Leibe and Schiele 2003; Kumar et al. 2005; Tu et al. 2005; Wang et al. 2007). Unifying object detection/recognition and segmentation/grouping typically involves adding high-level prior about shapes of objects to the low-level discriminative approaches used for segmentation and/or grouping (Kumar et al. 2005; Wang and Oliensis 2006; Veksler 2008). It may also involve quantifying low-level tasks for improved detection/recognition (Tu and Yuille 2004; Meilă 2005; Wang and Oliensis 2008).

Detecting objects can be at various precision levels starting from roughly detecting bounding boxes around the objects to finding the boundaries of the objects and even to finding the full interior contours of the objects. Object detection involving contours can provide a basis for better high-level applications which require more precision like medical imaging, human robot interactions etc. The first step in detecting object boundaries is extracting contour parts from images. Contour parts are then grouped to form object boundaries using methods that are essentially trying to perform shape matching in real images. For example, Shotton et al. (2005), Opelt et al. (2006) use Chamfer distance (Borgefors 1988) to match fragments of contours learnt from training images to edge images, but their results are typically grassy contours. Ferrari et al. (2008) uses a network of nearly straight contour fragments and sliding window search. More recently Zhu et al. (2008) formulated the shape matching of contours (identified using Zhu et al. (2007)) in clutter as a set-set matching problem.

They present an approximate solution to the hard combinatorial problem by using a voting scheme (Wang et al. 2007; Leibe et al. 2005) and a relaxed context selection scheme using linear programming. They use shape context (Belongie et al. 2002) as shape descriptor.

Our direction of research is to group more primitive tokens (very small linear segments) instead of parts (chains of pixels encoding some shape) using local symmetry based shape constraints *sequentially* to group object boundaries directly. The main reasoning underlying such approach is that parts produced using low-level cues alone are prone to unrecoverable errors. It is similar to the reasoning behind incorporating shape prior in segmentation (Kumar et al. 2005). We map the problem of contour grouping to a SLAM problem as it is stated in the field of robot mapping. We extend the particle filter based approach used in SLAM so that statistical inference based on a reference model is possible. In comparison to previous approaches our work has at least two serious advantages that are demonstrated in our experimental results. We are able to group contour segments in the presence of distractor segments between locally symmetric contour pieces. Even if we our shape models are derived from complete contours, grouping of only parts of contours is possible.

Contour grouping is a very hard problem that requires low-level, mid-level and high-level constraints. The proposed approach provides a versatile and robust framework for integrating all the three types of constraints. The shape constraints currently are based on single paths in symmetry sets that capture major parts of the object. For some complex objects shape constraints cannot be represented using single paths. Multiple paths in a symmetry set can be viewed as a collection of single paths in a "shape-tree". Different paths capture different parts of the shape. These parts can be connected using a tree structure. Our system can be extended to accommodate multiple paths in at least two different ways:

- Our system can be run for each path separately to identify all major parts and then part-matching can be performed using techniques like Zhu et al. (2008), Latecki et al. (2008). This requires integration of tree-based shape matching systems on top of our system.
- Perform multi-robot mapping: different virtual robots can explore different single paths and can collaboratively combine the parts together. This requires new implementation for *joint* posterior for different paths and contour maps of parts.

The former approach can also be viewed as map merging problem studied in multi-robot mapping as in Carpin et al. (2005), Adluru et al. (2008) while the later approach as traditional collaborative mapping as in Howard (2005), Fox et al. (1999). We intend to develop a system that can handle multiple path based representation as part of our future work.

Acknowledgements We thank the anonymous reviewers for very insightful and constructive comments that helped to improve several aspects of our paper. We thank ChengEn Lu for obtaining the retrieval rates of Ferrari et al. (2008).

References

Adluru, N., Latecki, L. J., Lakamper, R., Young, T., Bai, X., & Gross, A. (2007). Contour grouping based on local symmetry. In *ICCV '07: Proceedings of the eleventh IEEE international conference on computer vision*. IEEE Computer Society.

Adluru, N., Latecki, L. J., Sobel, M., & Lakaemper, R. (2008). Merging maps of multiple robots. In *IAPR international conference on pattern recognition (ICPR)*.

Bai, X., & Latecki, L. J. (2008). Path similarity skeleton graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(7), 1283–1292.

Belongie, S., Malik, J., & Puzicha, J. (2002). Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 705–522.

Black, M. J., & Fleet, D. J. (2000). Probabilistic detection and tracking of motion boundaries. *International Journal of Computer Vision*, *38*(3), 231–245.

Blake, A., & Isard, M. (1997). *Active contours*. Berlin: Springer.

Blum, H. (1967). A transformation for extracting new descriptors of shape. In W. Wathen-Dunn (Ed.), *Models for the perception of speech and visual form*. Cambridge: MIT.

Blum, H. (1973). Biological shape and visual science. *Journal of Theoretical Biology*, *38*, 205–287.

Borenstein, E., & Ullman, S. (2002). Class-specific, top-down segmentation. In *ECCV '02: Proceedings of the 7th European conference on computer vision—Part II* (pp. 109–124). London: Springer.

Borgefors, G. (1988). Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *10*(6), 849–865. doi:10.1109/34.9107.

Bruce, J., Giblin, P., & Gibson, C. (1985). Symmetry sets. *Proceedings Royal Society of Edinburgh*, *101*(A), 163–186.

Carpenter, J., Clifford, P., & Fearnhead, P. (1999). *Building robust simulation-based filters for evolving data sets* (Tech. Rep.). Dept. of Statistics, University of Oxford.

Carpin, S., Birk, A., & Jucikas, V. (2005). On map merging. *Robotics and Autonomous Systems*, *53*(1), 1–14.

Chang, H. J., Lee, C. S. G., Lu, Y. H., & Hu, Y. C. (2007). P-SLAM: Simultaneous localization and mapping with environmental-structure prediction. *IEEE Transactions on Robotics*, *23*(2), 281–293.

Choi, H. I., Choi, S. W., & Moon, H. P. (1997). Mathematical theory of medial axis transform. *Pacific Journal of Mathematics*, *181*(1), 57–88.

Cox, I., Rehg, J., & Hingorani, S. (1993). A Bayesian multiple-hypothesis approach to edge grouping and contour segmentation. *International Journal of Computer Vision*, *11*(1), 5–24.

Doucet, A. (1998). *On sequential simulation-based methods for Bayesian filtering* (Tech. Rep.). Cambridge University Department of Engineering.

Doucet, A., de Freitas, N., & Gordon, N. (2001). *Sequential Monte Carlo methods in practice*. Berlin: Springer.

Eliazar, A., & Parr, R. (2003). DP-SLAM: Fast, robust simulataneous localization and mapping without predetermined landmarks. In *Int. joint conf. on artificial intelligence (IJCAI)*.

Eliazar, A., & Parr, R. (2004). DP-SLAM 2.0. In *IEEE int. conf. on robotics and automation (ICRA)*.

Feldman, J. (1999). The role of objects in perceptual grouping. *Acta Psychologica*, *102*(2–3), 137–163.

Feldman, J. (2001). Bayesian contour integration. *Perception & Psychophysics*, *63*(7), 1171–1182.

Feldman, J. (2003). Perceptual grouping by selection of a logically minimal model. *International Journal of Computer Vision*, *55*(1), 5–25.

Feldman, J., & Singh, M. (2006). Bayesian estimation of the shape skeleton. *Proceedings of the National Academy of Sciences*, *103*(47), 18,014–18,019.

Ferrari, V., Tuytelaars, T., & Gool, L. J. V. (2006). Object detection by contour segment networks. In *ECCV* (pp. 14–28).

Ferrari, V., Fevrier, L., Jurie, F., & Schmid, C. (2008). Groups of adjacent contour segments for object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*(1), 36–51.

Fox, D. (2003). Adapting the sample size in particle filters through kld-sampling. *International Journal of Robotics Research (IJRR)*, *22*(12), 985–1003.

Fox, D., Burgard, W., Kruppa, H., & Thrun, S. (1999). Efficient multi-robot localization based on Monte Carlo approximation. citeseer.ist.psu.edu/article/fox99efficient.html.

Galun, M., Basri, R., & Brandt, A. (2007). Multiscale edge detection and fiber enhancement using differences of oriented means. In *ICCV '07: Proceedings of the eleventh IEEE international conference on computer vision*. IEEE Computer Society.

Giblin, P. J. (2000). Symmetry sets and medial axes in two and three dimensions. In *Proceedings of the 9th IMA conference on the mathematics of surfaces* (pp. 306–321). London: Springer.

Giblin, P. J., & Kimia, B. B. (2003a). On the intrinsic reconstruction of shape from its symmetries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *25*(7), 895–911.

Giblin, P. J., & Kimia, B. B. (2003b). On the local form and transitions of symmetry sets, medial axes, and shocks. *International Journal of Computer Vision*, *54*(1–3), 143–156.

Gordon, N., Salmond, D., & Smith, A. (1993). Novel approach to nonlinear/non-Gaussian Bayesian state estimation. In *Radar and signal processing, IEE proceedings* (Vol. 140, pp. 107–113).

Grisetti, G., Stachniss, C., & Burgard, W. (2007). Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, *23*, 34–46.

Handschin, J. E. (1970). Monte Carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*, *6*, 555–563.

Handschin, J. E., & Mayne, D. Q. (1969). Monte Carlo techniques to estimate the conditional expectation in multi-stage non-linear filtering. *International Journal of Controls*, *9*(5), 353–358.

Hoiem, D., Stein, A., Efros, A. A., & Hebert, M. (2007). Recovering occlusion boundaries from a single image. In *International conference on computer vision (ICCV)*.

Howard, A. (2005). Multi-robot simultaneous localization and mapping using particle filters. In *Proceedings of robotics: science and systems*. Cambridge, USA.

Isard, M., & Blake, A. (1998). Condensation—conditional density propagation for visual tracking. *International Journal of Computer Vision*, *29*(1), 5–28.

Kimia, B. B., & Tamrakar, A. (2002). The role of propagation and medial geometry in human vision. In *BMCV '02: Proceedings of the second international workshop on biologically motivated computer vision* (pp. 219–229). London: Springer.

Kong, A., Liu, J., & Wong, W. (1994). Sequential imputations and Bayesian missing data problems. *The American Statistical Association*, *89*, 278–288.

Kovesi, P. D. (2008). *MATLAB and Octave functions for computer vision and image processing*. School of Computer Science & Software Engineering, The University of Western Australia, available from: http://www.csse.uwa.edu.au/~pk/research/matlabfns/.

Kuijper, A., & Olsen, O. F. (2004). Transitions of the pre-symmetry set. In *ICPR '04: Proceedings of the pattern recognition, 17th international conference on (ICPR'04)* (Vol. 3, pp. 190–193). Washington: IEEE Computer Society.

Kuijper, A., & Olsen, O. F. (2005). Geometric skeletonization using the symmetry set. In *Image processing, 2005. ICIP 2005* (pp. 497–500). IEEE International Conference.

Kuijper, A., & Olsen, O. (2006). Describing and matching 2d shapes by their points of mutual symmetry. In *ECCV06* (pp. III: 213–225).

Kuijper, A., Olsen, O. F., Bille, P., & Giblin, P. (2006a). Matching 2d shapes using their symmetry sets. In *ICPR '06: Proceedings of the 18th international conference on pattern recognition* (pp. 179–182). Washington: IEEE Computer Society.

Kuijper, A., Olsen, O. F., Giblin, P., & Nielsen, M. (2006b). Alternative 2d shape representations using the symmetry set. *Journal of Mathematical Imaging and Vision*, *26*(1–2), 127–147.

Kumar, M. P., Torr, P. H. S., & Zisserman, A. (2005). OBJ CUT. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (Vol. 1, pp. 18–25). San Diego.

Latecki, L. J., Sobel, M., & Lakaemper, R. (2006). New EM derived from Kullback-Leibler divergence. In *ACM SIGKDD int. conf. on knowledge discovery and data mining*.

Latecki, L. J., Lu, C., Sobel, M., & Bai, X. (2008). Multiscale random fields with application to contour grouping. In *Neural information processing systems conf. (NIPS)*. Vancouver.

Leibe, B., & Schiele, B. (2003). Interleaved object categorization and segmentation. In *BMVC03*.

Leibe, B., Seemann, E., & Schiele, B. (2005). Pedestrian detection in crowded scenes. In *Proceedings of the 2005 IEEE computer society conference on computer vision and pattern recognition* (Vol. 1, pp. 878–885). Washington: IEEE Computer Society.

Leyton, M. (1992). *Symmetry, causality, mind*. Cambridge: MIT.

Ling, H., & Jacobs, D. W. (2007). Shape classification using inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *29*, 286–299.

Liu, J. S. (1996). Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, *6*, 113–119.

Liu, J. S., Chen, R., & Wong, W. H. (1998a). Rejection control and sequential importance sampling. *Journal of the American Statistical Association*, *93*(443), 1022–1031.

Liu, T., Geiger, D., & Yuille, A. L. (1998b). Segmenting by seeking the symmetry axis. In *Proc. CVPR* (pp. 994–998).

Lockwood, E. H. (2007). *A book of curves*. Cambridge: Cambridge University Press.

Lowe, D. G. (1985). *Perceptual organization and visual recognition*. Boston: Kluwer Academic.

Martin, D., Fowlkes, C., Tal, D., & Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. *Proceedings of the 8th International Conference on Computer Vision*, *2*, 416–423.

Meilă, M. (2005). Comparing clusterings: an axiomatic view. In *ICML '05: Proceedings of the 22nd international conference on machine learning* (pp. 577–584). New York: ACM.

Mohan, R., & Nevatia, R. (1992). Perceptual organization for scene segmentation and description. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *14*(6), 616–635.

Moravec, H. (1988). Sensor fusion in certainty grids for mobile robots. *AI Magazine*, *9*(2), 61–74.

Opelt, A., Pinz, A., & Zisserman, A. (2006). A boundary-fragment-model for object detection. In *Proceedings of the European conference on computer vision*.

Perez, P., Blake, A., & Gangnet, M. (2001). Rjetstream: Probabilistic contour extraction with particles. In *Proc. ICCV* (pp. 524–531).

Podolak, J., Shilane, P., Golovinskiy, A., Rusinkiewicz, S., & Funkhouser, T. (2006). A planar-reflective symmetry transform for 3D shapes. *ACM Transactions on Graphics (Proceedings SIGGRAPH) 25*(3).

Raviv, D., Bronstein, A., Bronstein, M., & Kimmel, R. (2007). Symmetries of non-rigid shapes. In *ICCV*.

Reid, D. B. (1979). An algorithm for tracking multiple targets. *IEEE Transactions on Automatic Control 24*(6), 843–854.

Ren, X., Berg, A., & Malik, J. (2005). Recovering human body configurations using pairwise constraints between parts. In *Proc. ICCV*.

Ren, X., Fowlkes, C. C., & Malik, J. (2008). Learning probabilistic models for contour completion in natural images. *International Journal of Computer Vision*, *77*(1–3), 47–63. doi:10.1007/s11263-007-0092-6.

Sebastian, T. B., Klein, P. N., & Kimia, B. B. (2004). Recognition of shapes by editing their shock graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(5), 550–571.

Shotton, J., Blake, A., & Cipolla, R. (2005). Contour-based learning for object detection. In *ICCV '05: Proceedings of the tenth IEEE international conference on computer vision (ICCV'05)* (Vol. 1, pp. 503–510). Washington: IEEE Computer Society.

Siddiqi, K., & Pizer, S. M. (2007). *Medial representations: mathematics, algorithms and applications*. Berlin: Springer.

Siddiqi, K., Shokoufandeh, A., Dickinson, S. J., & Zucker, S. W. (1999). Shock graphs and shape matching. *International Journal of Computer Vision*, *35*, 13–32.

Stachniss, C., Grisetti, G., & Burgard, W. (2005). Information gain-based exploration using rao-blackwellized particle filters. In *Proc. of robotics: science and systems (RSS)* (pp. 65–72). Cambridge, MA, USA.

Stahl, J. S., & Wang, S. (2006). Globally optimal grouping for symmetric boundaries. In *Proc. CVPR*.

Stein, A., Hoiem, D., & Hebert, M. (2007). Learning to find object boundaries using motion cues. In *IEEE international conference on computer vision (ICCV)*.

Tamrakar, A., & Kimia, B. B. (2007). No grouping left behind: From edges to curve fragments. In *ICCV '07: Proceedings of the eleventh IEEE international conference on computer vision*. IEEE Computer Society.

Tanizaki, H. (1997). Nonlinear and nonnormal filters using Monte Carlo methods. *Computational Statistics and Data Analysis*, *25*(4), 417–439.

Thrun, S., Burgard, W., & Fox, D. (2005). *Probabilistic robotics*. Cambridge: MIT.

Trinh, N., & Kimia, B. B. (2007). A symmetry-based generative model for shape. In *ICCV '07: Proceedings of the eleventh IEEE international conference on computer vision*. IEEE Computer Society.

Tu, Z., & Yuille, A. (2004). Shape matching and recognition: using generative models and informative features. In *ECCV04* (Vol. 3, pp. 195–209).

Tu, Z., Chen, X., Yuille, A., & Zhu, S. C. (2005). Image parsing: Unifying segmentation, detection, and recognition. *International Journal of Computer Vision*, *63*, 113–140.

Veksler, O. (2008). Star shape prior for graph-cut image segmentation. In *ECCV08*.

Wang, H., & Oliensis, J. (2006). A global contour measure for image segmentation. In *POCV06*.

Wang, H., & Oliensis, J. (2008). Shape matching by segmentation expectation. In *ECCV08*.

Wang, L., Shi, J., Song, G., & Shen, I. (2007). Object detection combining recognition and segmentation. In *ACCV07* (pp. 189–199).

Wertheimer, M. (1923). Untersuchungen zur lehre von der gestalt II. *Psycologische Forschung*, *4*, 301–350.

Wertheimer, M. (1958). Principles of perceptual organization. In D. Beardslee & M. Wertheimer (Eds.), *Readings in perception*. Princeton: Princeton University Press.

Witkin, A. P., & Tenenbaum, J. M. (1983). On the role of structure in vision. In J. Beck, B. Hope, & A. Rosenfeld (Eds.), *Human and Machine Vision*. New York: Academic Press.

Yang, X., Bai, X., Latecki, L. J., & Tu, Z. (2008). Improving shape retrieval by learning graph transduction. In *ECCV08*.

Zaritskii, V., Svetnik, V., & Shimelevich, L. (1975). Monte Carlo technique in problems of optimal data processing. *Automation and Remote Control*, *12*, 95–103.

Zhu, Q., Song, G., & Shi, J. (2007). Untangling cycles for contour grouping. In *ICCV '07: Proceedings of the eleventh IEEE international conference on computer vision*. IEEE Computer Society.

Zhu, Q., Wang, L., Wu, Y., & Shi, J. (2008). Contour context selection for object detection: A set-to-set contour matching approach. In *ECCV08*.

Zhu, S. C., & Yuille, A. (1995). Forms: a flexible object recognition and modelling system. In *Proc. ICCV* (pp. 465–472).