Published in final edited form as:

IEEE Trans Image Process. 2011 October ; 20(10): 2912–2924. doi:10.1109/TIP.2011.2142002.

Object Tracking and Target Reacquisition Based on 3-D Range Data for Moving Vehicles

Jehoon Lee [Student Member, IEEE],

School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (jehoon.lee@gatech.edu)

Shawn Lankton [Member, IEEE], and

School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA. He is now with McKinsey & Company, New York, NY 10022 USA (shawn.lankton@gmail.com)

Allen Tannenbaum [Fellow, IEEE]

School of Electrical and Computer Engineering and the School of Biomedical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (tannenba@ece.gatech.edu)

Abstract

In this paper, we propose an approach for tracking an object of interest based on 3-D range data. We employ particle filtering and active contours to simultaneously estimate the global motion of the object and its local deformations. The proposed algorithm takes advantage of range information to deal with the challenging (but common) situation in which the tracked object disappears from the image domain entirely and reappears later. To cope with this problem, a method based on principle component analysis (PCA) of shape information is proposed. In the proposed method, if the target disappears out of frame, shape similarity energy is used to detect target candidates that match a template shape learned online from previously observed frames. Thus, we require no *a priori* knowledge of the target's shape. Experimental results show the practical applicability and robustness of the proposed algorithm in realistic tracking scenarios.

Index Terms

Geometric active contours; object tracking; particle filtering; target reacquisition; 3-D range data

I. Introduction

Tracking moving objects is an important task in computer vision. The motion of an object is observed using an image input device (e.g., camera) which generates a sequence of images at discrete time steps. The task of tracking is to locate and segment the object of interest from the background at each frame of the image sequence. A large number of tracking algorithms have inspired significant interest in recent years (see [1]–[4] and references therein).

Target reacquisition has been extensively studied in visual surveillance systems in order to maintain the identity of a moving object across cameras with overlapping [5] as well as nonoverlapping [6] fields of view. Much of the work adopted a probabilistic approach to

compute the correspondence matching of objects between multiple cameras [5], [6]. One usually integrates spatio-temporal relationships with appearance information of a desired target. The work in [7] constructs a graph-based track initialization for target matching, and the Kalman filter is utilized to predict the movement of a target in the blind region. In [8], a learning method based on the prior knowledge of camera network topology is employed to build a spatio-temporal link, and a normalized histogram is used for the appearance model. In comparison to a number of approaches for visual surveillance systems, we address the reacquisition problem in the context of object tracking with a moving camera. More specifically, the proposed method uses a single stereo camera to capture a moving vehicle in an uncontrolled environment. Thus, the spatio-temporal link between cameras is not applicable in our framework. In addition, we take advantage of shape information obtained from the proposed contour-based tracker to continuously track the target instead of a histogram-based appearance model many times adopted in the surveillance system.

We now briefly introduce several papers relevant to the proposed work. These papers address problems on partial or total occlusions of the tracked objects, by defining shape-based energy functionals based on prior shape knowledge. Yilmaz *et al.* [9] present a contour-based method for tracking nonrigid objects using a Bayesian framework in which the contour is implemented via level set methods and a shape energy term is added to handle occlusions. In [10] and [11], Rathi *et al.* propose a particle filtering algorithm, which incorporates geometric active contours for tracking deformable objects. They also embed shape information into the update step of particle filtering to cope with occlusions. Dambreville *et al.* in [12] combined the unscented Kalman filter and geometric active contours for dynamic tracking. By using a small number of deterministic sample points, they were able to reduce computational complexity in contrast to the approach in [10].

The works mentioned so far have only considered partial occlusion handling with prior knowledge based on a shape model. Severe or complete occlusions have also been explored in the following papers. In [13], Nguyen and Smeulders present a template matching method using an appearance model smoothed temporally by the Kalman filter. Here, they define a maximal bound on the length of recoverable occlusions. After a certain number of frames, the temporal filter slowly incorporates the occlusion into the object model. Similarly, in [14] a minimum cost computation-based template selection is introduced to handle occlusions assuming the object reappears within a certain time and spatial region. However, both algorithms fail track continuously when the object does not appear again within predefined time or when the object presents an appearance not accounted for in the template. Jackson *et al.* [15] propose an explicit model for the motion of contours that is enforced by higher-order motion models, such as inertia. In [16], Bartesaghi and Sapiro propose a tracking algorithm using spatio-temporal minimal surfaces in 3-D space-time domain in order to handle severe or total occlusions of an object.

The main objective of the present paper is to design a reliable algorithm to track deformable targets in a time-varying sequence of 3-D range data by combining particle filtering and geometric active contour models. These are employed to estimate the global motion and local deformation of objects. Specifically, the particle filtering is used for estimating the group motion of the object, and tracking of its deformations is achieved by active contours. In addition, this paper will present a method to reacquire track of an object in the event that it disappears from the field of view and reappears later.

For segmentation and tracking, we utilize 3-D range data computed by determining correspondences and disparity maps from stereoscopic image sequences. Depth information from disparity maps has been used in the literature to improve the performance of object tracking and segmentation tasks [17]–[19]. For the detection of moving objects, Talukder

and Matthies [17] introduced the 3-D measured flow field by combining the 2-D optical flow and the disparity flow from depth maps. Also, in [18], Alattar and Jang presented a segmentation method using edge maps for a stereo image as an external energy for active contours. The edge combination image is introduced in [19], which is obtained from both the original intensity image and stereo-driven depth maps for contour-based segmentation. Both approaches improve the capability in image segmentation but they have difficulties in dealing with weak edge levels or boundary gaps.

In this paper, we employ region-based active contours driven by the Bhattacharyya gradient flow [20], [21] for object segmentation due to its robustness against noise, and its ability to deal with cluttered environments. Depth information from disparity maps is also used to position the object in 3-D space for motion estimation in the filtering process. The scheme proposed in this paper provides the three contributions given here:

1. An active contour segmentation method that combines range information weighted by a Gaussian weighting scheme with the Bhattacharyya gradient flow.
2. A robust estimation scheme for global motion of the object using the proposed weighted depth maps in the particle filtering framework.
3. An on-line shape learning method based on principal component analysis (PCA) that allows the tracker to detect the disappearance and the reappearance of a target.

The remainder of this paper is organized as follows. In Section II, we give an overview of some fundamental theories used in the proposed algorithms including: particle filtering, curve evolution via level sets, and statistical shape representation. Section III describes the proposed algorithms for object tracking in stereo image sequences. First, a segmentation method that incorporates depth information is presented. Next, the filtering algorithm used to estimate global motion is discussed. Finally, disappearance and reappearance handling is presented. In Section IV, experimental results on a real image sequence are shown, and conclusions and directions for future work are given and some limitations of the proposed algorithm are discussed in Section V.

II. Background

Here, we briefly introduce some of the basic notions for particle filtering, geometric active contours driven by the Bhattacharyya gradient flow, and PCA-based shape representations.

A. Particle Filtering

Sequential Bayesian filtering estimation with Monte Carlo simulations, i.e., *particle filtering*, was first introduced by Gordon *et al.* [22]. In recent years, it has proven to be a powerful scheme for nonlinear and non-Gaussian estimation problems due to its simplicity and versatility. Assuming that X_t is a state vector and Y_t is a set of observations, we can model a state transition equation and an observation equation as

$$X_{t+1} = f_t(X_t) + u_t \quad Y_t = h_t(X_t) + v_t \quad (1)$$

where u_t and v_t are independent and identically distributed (iid) random state variables representing noise with known probability density functions (pdfs). f_t and h_t are known functions, and the initial state distribution $p(X_0)$ is also known. The basic idea of particle filtering is to produce a set of N particles $\{X_t^i\}_{i=1,\dots,N}$ at each time t , whose observed measurements closely approximate the posterior distribution of state, $p(X_t | Y_{1:t})$, based on a sequence of observations $Y_{1:t}$.

Particle filtering is carried out by sampling N times from initial state distribution and applying Bayesian recursive filtering. This is performed in two steps: the prediction step and the update step. In the prediction step, we use importance sampling [23] in which samples are drawn from an alternative *importance (or proposal)* distribution $q(\cdot)$ because we cannot sample directly from the true state distribution $p(\cdot)$. In the update step, each particle is weighted by the following equation:

$$w_t^i \propto \frac{p(X_t^i | Y_{1:t})}{q(X_t^i | X_{1:t-1}^i, Y_{1:t})} \quad (2)$$

where w_t^i is the normalized weight of the i th particle. From the above approach, the filtering posterior are represented by a set of particles (or samples) X_t^i and its associated weights w_t^i as

$$p(X_t | Y_{1:t}) \approx \sum_{i=1}^N w_t^i \delta(X_t - X_t^i). \quad (3)$$

Importance sampling causes most particles to have negligible weight after a few iterations, which is called the *sampling degeneracy* problem. To avoid this, one can apply a resampling scheme, which can generally be done by replicating particles in proportion to their weights. This process eliminates samples with low weights and chooses better particles [24], [25]. However, it produces the loss of diversity for a set of particles, i.e., particles with high weights are selected too much, and thus the others disappear as time goes on. Therefore, all of the particles will eventually collapse to the same value. To alleviate this *sample impoverishment*, several particle filtering variants are proposed in the literature (see [22] and [26]). In our proposed filtering framework, the careful choice of the number of curve evolutions, introduced in [11] and [27], is adopted to solve both problems. This method will be described in detail in Section III-C.

B. Geometric Active Contours

Since the introduction of deformable contours or snakes by Kass *et al.* [28], curve evolution has been widely used for segmentation in single images as well as object tracking. The basic idea of active contour models is to evolve or deform the closed curve so as to minimize an energy functional by gradient descent until it finds the outline of the object of interest (see [28] and [29] for details). Active contour models are often represented implicitly via the level set methods which offer a powerful representation tool for numerical implementation of curve evolution. This representation can automatically handle topological changes with which parametric representations have difficulty in handling; see [30] and [31] for details. In level set methods, a closed curve C is represented as the zero level set of a higher dimensional function Φ , which is typically chosen to be a signed distance function such that $\Phi < 0$ inside C and $\Phi > 0$ outside C . Therefore, the curve can be described by an implicit surface

$$C = \{x | \Phi(x) \equiv 0, x \in \Omega\} \quad (4)$$

where Ω is the domain of an image $I(x): \mathbb{R}^2 \rightarrow \mathbb{Z}$ which is mapping to the photometric variable $z \in \mathbb{Z}$.

Many active contour segmentation energies have been proposed to drive the segmenting curve toward the boundaries of an object [32]–[34]. In the present work, we incorporate an active contour model driven by maximizing a statistical measure of dissimilarity (known as the Bhattacharyya distance) between the curve's interior and exterior. We employ this

particular model because it is robust against noisy images and initial curve placement while retaining the ability to segment multimodal objects in cluttered scenes. Furthermore, it has been shown to be sufficiently computationally efficient for use in tracking applications [20], [21].

We now offer a brief summary of the Bhattacharyya distance and its use in active contour segmentation. The Bhattacharyya distance between two pdfs P_i and P_o is defined by

$$D_B = -\log(B)$$

where

$$B(\Phi) = \int_{\mathbb{Z}} \sqrt{P_i(z)P_o(z)} dz. \quad (5)$$

The Bhattacharyya coefficient B varies between 0 and 1 (0 indicates complete mismatch while 1 represents perfect similarity). P_i and P_o are pdfs defined inside and outside the curve C , respectively. Now, the active contour segmentation energy is defined as the combination of the Bhattacharyya coefficient and a regularizing term to constrain the curve length for smooth evolution

$$E_{\text{image}} = B(\Phi) + \alpha \int_{\Omega} \|\nabla H(\Phi)\| dx \quad (6)$$

where α is a user-defined regularization constant ($\alpha > 0$) and ∇ denotes the gradient. The curve evolution is obtained by minimizing the energy E_{image} via a gradient descent flow to maximize the Bhattacharyya distance between the inside and outside of the segmenting curve. Thus, the optimal level set function is given by

$$\Phi^* = \arg\inf_{\Phi} \{E_{\text{image}}\} \quad (7)$$

Each pdf can be defined by

$$P_i(z) = \frac{\int_{\Omega} K(z - I(x)) H(-\Phi(x)) dx}{\int_{\Omega} H(-\Phi(x)) dx} \quad P_o(z) = \frac{\int_{\Omega} K(z - I(x)) H(\Phi(x)) dx}{\int_{\Omega} H(\Phi(x)) dx} \quad (8)$$

where K is the kernel, and H is the Heaviside step function given by

$$H(\Phi) = \begin{cases} 1, & \text{for } \Phi \geq 0 \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Popular choices for the kernel K are either Gaussian Kernel or the Dirac delta function.

The following equation is obtained from the computation of the first variation of (5) with respect to Φ :

$$\nabla_{\Phi} B = \frac{1}{2} \int \left(\frac{\partial P_i(z)}{\partial \Phi} \sqrt{\frac{P_o(z)}{P_i(z)}} + \frac{\partial P_o(z)}{\partial \Phi} \sqrt{\frac{P_i(z)}{P_o(z)}} \right) dz. \quad (10)$$

By differentiating P_i and P_o with respect to Φ and substituting them into (10), the gradient flow of Φ that minimizes (5) is given by

$$\frac{\partial \Phi}{\partial t} = -\nabla_{\Phi} B = -\delta(\Phi) S$$

where

$$S = \frac{B}{2} \left(\frac{1}{A_i} - \frac{1}{A_o} \right) + \frac{1}{2} \int_{\mathbb{Z}} K(z - I(x)) \times \left(\frac{1}{A_o} \sqrt{\frac{P_i(z)}{P_o(z)}} - \frac{1}{A_i} \sqrt{\frac{P_o(z)}{P_i(z)}} \right) dz. \quad (11)$$

Here, A_i and A_o are the areas inside and outside the segmenting curve, respectively. Now, we have the final equation of the gradient flow for the level set evolution minimizing (6) by adding the regularizing term

$$\frac{\partial \Phi}{\partial t} = \delta(\Phi)(\alpha \kappa - S) \quad (12)$$

where $\delta(\cdot)$ is the delta function and the curvature of the evolving curve κ is given by the divergence of the gradient of the level set function Φ : $\kappa = \text{div}\{\nabla \Phi / \|\nabla \Phi\|\}$. The gradient flow in (12) will converge when the curve's interior and exterior distributions are maximally different [21].

C. Shape Representation

Shape statistics are necessary to represent the modes and variations of shapes. In this paper, we employed the eigen-shape-based representation using PCA, introduced by [35]. The basic idea is that a given shape may be represented by a linear combination of a set of eigen-shapes after projecting the shape into the eigen-space. Shapes are represented by the signed distance function. In the level set framework, Φ_i is the signed distance function determined by the zero level set, corresponding to the segmenting curve C_i . The mean shape μ for n shapes is computed as $\mu = (1/n) \sum \Phi_i$. The mean offset $\Phi_i - \mu$ is placed as a column vector in an matrix $M(r \times n)$, where $\Phi_i \in \mathbb{R}^r$ and r is the number of dimensions. The covariance matrix in shape $(1/n)MM^T$ is decomposed by the singular value decomposition (SVD)

$$U \Sigma U^T = \frac{1}{n} M M^T \quad (13)$$

where U is a matrix whose column vectors represent the set of orthogonal modes of shape variations and Σ is a diagonal matrix of corresponding singular values. The estimated shape for Φ is given by

$$\tilde{\Phi} = U_k U_k^T (\Phi - \mu) + \mu \quad (14)$$

where U_k is a matrix consisting of the first k columns of U .

III. Proposed Algorithm

The overall motion of an object can be described by the combination of a global motion and local deformations, the so-called *deformation* [37]. Therefore, the objective of object tracking is to estimate an object's group displacement and its shape deformations in terms of a temporal function. We separate the object tracking problem into two parts: the object segmentation for each frame and the prediction of temporal change of the object's position.

More specifically, the proposed algorithm is comprised of the segmentation process (for tracking local deformations) and the filtering process (for tracking the global motion).

As part of the segmentation approach, a weighted depth map is defined, and the curve evolution for object extraction is performed via the Bhattacharyya gradient flow. In the filtering process, particle filtering and active contours are employed in conjunction to estimate the global position of the object in the weighted image space. For motion estimation, comparable approaches using the filtering schemes in conjunction with active contour models may be found in [10]–[12], [27].

Lastly, we define a certain similarity shape energy based on a statistical shape model in order to achieve continuous tracking without the prior shape model for an uncertain period of time even if the object is not observable, i.e., the object being tracked is fully occluded or leaves the scene completely.

A. Segmentation Using 3-D Range Data

There are many algorithms used to find stereo correspondence from a pair of rectified stereo images (see [38]–[41] and references therein). The output of these algorithms is a disparity map showing the relative displacement of a pixel from one image to the other. In the proposed algorithm we compute disparity maps using a naive sum of squared difference stereo matching scheme that yields much lower quality matching results than available algorithms. This is done to show the robustness of proposed algorithm against poor-quality stereo reconstruction. Better results could be expected if more accurate stereo correspondences were used.

The disparity space is defined as a 3-D projective transformation of 3-D space $[x, y, z]^T$ for a given pair of rectified stereo images [41]. The correspondence between a pixel (x, y) in a reference image and a pixel (x', y') in a matching image is given by

$$x' = x + D(x, y), y' = y \quad (15)$$

where $D(x, y)$ is a univalued disparity function (or map) with respect to a reference image. This disparity map can yield a 3-D range value (or a depth value) for each pixel in the scene if the cameras' calibration information is known [42]. The depth map $D_p(x): \mathbb{R}^2 \rightarrow \mathbb{D}$ which maps the depth value $d \in \mathbb{D}$ of a pair of images is represented in terms of disparity function $D(x)$. Thus, given the position vector $[x, y, z]^T$ of an object in the 3-D image space, z can be substituted with the depth value d : $[x, y, d]^T$.

Depth information allows simplification and improvement of the segmentation task on a single image as well as a sequence of images for the following reasons. First, the object's 3-D range values will be quasi-homogenous even though intensities inside the object have different photometric intensities. Furthermore, the size of the object being tracked can be approximated with the associated depth values. This allows us to estimate parameters for active contour models such as the maximum iteration number for curve evolution or the size of initial contours. Thus, if intensity distribution of an object is a multimodal and non-homogenous, the segmentation using 3-D range data can be a useful scheme to handle the following problems:

1. Extracting the outlines of an object in a highly textured image.
2. Object segmentation in scenes where the foreground and background have a significant depth difference (e.g., moving vehicles, such as an aircraft in the sky or a car on a road).

Some results of segmentation using active contours driven by the Bhattacharyya gradient flow with and without using the disparity map are shown in Fig. 1. The cone of interest is located in the rear center of the given image. In Fig. 1(b), the cone is a highly textured object, thus the segmenting curve did not enclose the whole outline of the cone. Instead, it converged to edges inside the cone. When using the disparity information, an acceptable segmentation is achieved.

B. Weighted Depth Maps

The basic idea of the proposed segmentation scheme is to weight depth values in proportion to the probability of the appearance of the object of interest prior to curve evolution. By applying a Gaussian weighting filter to the range data, unimportant noise and faraway structures are suppressed or eliminated. This allows for improved segmentation results and faster convergence. In addition, it restricts the image information according to the depth value of the object and weights the most probable regions highly.

A kernel f_w is a nonnegative real-valued integrable weighting function. For example, the Gaussian function or a uniform function may be used. Specifically, we chose a Gaussian function whose mode is the certain depth value d_p and its variance is $\sigma_w^2: \mathcal{N}(d_p, \sigma_w^2)$. The new weighted depth map is given by

$$D_w(x) = f_w(D_p(x)) = \gamma \cdot \mathcal{N}(d_p, \sigma_w^2) \otimes f_s(\cdot) = \frac{\gamma}{\sqrt{2\pi}} \exp \left\{ -\frac{(D_p(x) - d_p)^2}{2\sigma_w^2} \right\} \otimes f_s(\cdot) \quad (16)$$

where γ is a weight parameter ($\gamma > 0$) and $f_s(\cdot)$ is a smoothness regularization filter. The weight parameter γ is assigned as 255 in our case, but this can be any positive value without loss of generality. The typical choice for $f_s(\cdot)$ is a morphological smoothing filter or an isotropic Gaussian filter. The size of filter depends on the noise present in the sequence. Fig. 2 shows the concept of the proposed weighting on range data, where the probable region is defined as the limited or sliced area within one standard deviation of the given depth value d_p .

The variance of the kernel f_w is chosen based on how confidently one can estimate the depth of the target. A large variance should be chosen if the image data is noisy or has poor contrast. Alternatively, if the variance is very small, the target will appear more prominent, but loss of the object information can occur. In addition, this value should be adjusted online according to the depth and size of the object. Fig. 3 shows the segmentation results using Gaussian function with different variances. Note that good results are obtained when σ_w^2 is chosen so that the weighting function accurately reflects the variation of depth values seen in the target. In practice, we have found that $\sigma_w^2 = [1^2, 3^2]$ works well in most cases.

In this work, segmentation is achieved with the active contours driven by the Bhattacharyya flow introduced in Section II-B. The kernel K in (11) is defined by delta function $\delta(\cdot)$. By substituting the weighted depth map $D_w(x, y)$ in (16) for an image space $I(x)$, we have the speed term S for (11) as

$$S = \frac{B}{2} \left(\frac{1}{A_i} - \frac{1}{A_o} \right) + \frac{1}{2} \int_{\mathbb{Z}} \delta(z - D_w(x)) \times \left(\frac{1}{A_o} \sqrt{\frac{P_i(z)}{P_o(z)}} - \frac{1}{A_i} \sqrt{\frac{P_o(z)}{P_i(z)}} \right) dz. \quad (17)$$

Fig. 4 shows a robust segmentation result in a highly cluttered environment. As can be seen in the results, since the backgrounds (e.g., the ground, skies, and trees) are far from the trucks, the proposed algorithm is well suited for this type of segmentation. Note that the range information in Fig. 4 is scattered and the trucks are not easily distinguished from the backgrounds. However, the proposed weighting scheme allows the trucks to be segmented without loss of information and prevents leaks into nearby structures, as shown in Fig. 4.

C. Filtering for Motion Estimation

We assume that the location of an object is within the probable region defined in Section III-B, and the difference of the object's location between consecutive frames is small: $[x \pm \delta x, y \pm \delta y, d \pm \delta d]^T$ given the position vector of the object is $[x, y, d]^T$ in 3-D image space, where δx , δy , and δd are the small perturbations of each coordinate. The global motion of an object is described by the transition of the coordinates of its centroid and depth value $[x_c, y_c, d_p]^T$, and the local motion is represented by the segmenting curve C evolved by the gradient flow in (12) and (17). Thus, the state vector is given by

$$X(t) = \begin{pmatrix} x_c \\ y_c \\ d_p \\ C \end{pmatrix} (t) = \begin{pmatrix} \rho \\ d_p \\ C \end{pmatrix} (t) \quad (18)$$

where $\rho = (x_c, y_c)^T$ is the centroid of an object and C denotes the contour coordinates represented by the zero level set of Φ . When a new image pair arrives, we have an observation $Y(t)$ which is the available weighted depth map $D_w(t)$ at time t : $Y(t) = D_w(t)$. Let a prior density be the proposal distribution to simplify the dynamic model and let resampling be applied at every time step [24], [25]. We then have the weight update equation from (2) as

$$w_t^i = p(Y_t | X_t^i)$$

where the likelihood of the observation is given by

$$p(Y_t | X_t^i) = \exp\{-E_{\text{image}}(X_t^i, Y_t)\}. \quad (19)$$

The proposed estimation scheme using particle filtering is described as follows:

1. Prediction Step:

- a. Generate the N particles, $\{X_t^i\}_{i=1 \dots N}$ around X_{t-1} in the following manner:

$$\rho_t^i = \rho_{t-1}^i + u_t, u_t \sim \mathcal{N}(0, \sigma_u^2) \quad (d_p)_t^i = (D_p)_{t-1}(\rho_t^i) \quad C_t^i = C_{t-1} + \rho_t^i.$$

2. Update Step:

- a. Make the weighted depth map based on $(d_p)_t^i$ from (16):

$$(D_w)_t = (f_w)_t((D_p)_t)$$

where

$$(f_w)_t \sim \mathcal{N} \left(\frac{1}{N} \sum_{i=1}^N (d_p)_t^i, \sigma_w^2 \right).$$

- b. Evolve the curve over the weighted depth map for each X_t^i for I iterations. I is generally a small number, which is carefully selected in consideration of the degree of trust of the system and measurement models to avoid *sample degeneracy* and *sample impoverishment*. More specifically, if I is chosen to be too large, this can lead to *sample degeneracy* due to the loss of temporal coherency. Likewise, choosing I to be too small results in *sample impoverishment* because all particles would not move to the region of the high likelihood [11], [27]. $I=3$ is selected experimentally for robust results of our experiments in Section IV.
- c. Compute the importance weights using (19)

$$\tilde{w}_t^i = \exp\{-E_{\text{image}}((D_w)_t, X_t^i)\}$$

and normalize

$$w_t^i = \frac{\tilde{w}_t^i}{\sum_{i=1}^N \tilde{w}_t^i}.$$

- d. The posterior distribution of the system is represented by a set of weighted particles

$$p(X_t | Y_{1:t}) = \sum_{i=1}^N w_t^i \delta(X_t^i).$$

- e. Resample N particles according to $p(X_t | Y_{1:t})$ by using the generic resampling scheme introduced in [24] and [25]: $\{X_t^i, 1/N\}_{i=1}^N$
- f. The best fitting curve, which maximizes the dissimilarity between two distributions for inside and outside the curve in (5) and minimizes the segmentation energy in (6), is selected for the measurement model.
- g. The centroid of the selected curve is taken as the centroid of the system and its depth value is given by

$$(d_p)_t = \frac{\int_{\Omega} (D_p)_t(x) H(-\Phi_t(x)) dx}{\int_{\Omega} H(-\Phi_t(x)) dx}.$$

D. Disappearance and Reappearance Handling

Here, we address the estimation problem of continuing to track an object of interest even when it is partially or fully occluded during the course of tracking. This naturally leads to the questions of how to detect the disappearance of an object and how to recognize the reappearance of an object and to reacquire track.

In order to solve these problems, we analyze the characteristics of disappearance and reappearance in dynamic imagery. For example, the size of an object will decrease as it disappears and increase as it reappears. Thus, disappearance detection may be achieved by comparing the average size of the tracked object from past frames with the object's current size. Reappearance detection may be accomplished by finding an object that is most similar to the tracked object before it was lost. Some assumptions about the location at which the object will reappear are also necessary to avoid the computational complexity of searching the entire domain and increase the possibility of reacquisition by reducing the probability of unexpected detection of a noise. In this work, we assume that the object being tracked will appear near the last known position of the object (e.g., if the object disappears on left side of the image, it will also reappear on the left side), that is, the object's centroid when it reappears ρ_r is constrained to be near the position of its centroid before disappearance, ρ_d . Also, the probable region when the object reappears is assumed to be the last probable region before its disappearance. Thus, the searching region for reappearance detection is described by

$$\rho_r = [\rho_d - \varphi, \rho_d + \varphi](d_p)_r = (d_p)_d \quad (20)$$

where φ is the longest value between the average lengths of the major and minor axes of the object, an $(d_p)_d$ and $(d_p)_r$ are the depth value when the object disappeared and will reappear, respectively. Note that these assumptions may restrict the applicability of the proposed algorithms for certain very important applications such as video surveillance in which it would be quite difficult to estimate reappearance using the preceding setup. Thus, in this paper, the proposed algorithms are tested on appropriate scenarios for these assumptions, such as tracking moving vehicles. Indeed, the proposed algorithms show the best performance in following a moving car on a road with various cluttered backgrounds as shown in our experiments in Section IV.

Shape analysis based on multiple feature correspondences is an indispensable tool to cope with disappearance and reappearance. The contour tracker, proposed in the previous sections, provides useful feature information of the segmented object in terms of the level set function Φ , such as outline, position, size, curvature, and aspect ratio of the object. From this information, we propose the following shape energy-based disappearance and reappearance handling scheme. The basic idea of the proposed algorithm is to define the shape similarity energy based on the feature information of past frames and then detect the object's return by comparing this feature information with the current frame and finding a candidate region that closely matches the template shape. In order to create the template shape, we assume that the object being tracked deforms gradually from frame to frame, and its shape remains similar while out of view. If all sets of shapes lie on a linear manifold and their deformations are small, PCA can be used to provide a shape representation [43]. Thus, the template shape of an object is obtained using PCA-based shape statistics described in Section II-C.

The template shape is based on the history of the segmented object over previous frames up to the current, rather than learned priors. More specifically, the shape of an object at time step t is represented as a binary mask selecting regions inside the closed curve C_t . Thus, we can write this using the Heaviside function as $H(-\Phi_t)$. Finally, the template shape at time step t is defined as the mean shape over all previous frames

$$\Phi_{TS} = \frac{1}{n-1} \sum_{t=1}^{n-1} H(-\Phi_t), n > 1. \quad (21)$$

If the new shape Φ_{new} of the same class of the object is received online at time step t , the shape dissimilarity function for Φ_{new} can be obtained from (14) and (21) as

$$\Gamma_1^2(\Phi_{\text{new}}, \Phi_{\text{TS}}) = \frac{\|\log(1 + |U_k^T(H(-\Phi_{\text{new}}) - \Phi_{\text{TS}})|)\|}{1 + \|\log(1 + |U_k^T(H(-\Phi_{\text{new}}) - \Phi_{\text{TS}})|)\|} \quad (22)$$

where $|\cdot|$ and $\|\cdot\|$ denote the absolute value and the Euclidean norm, respectively.

The size and curvature differences between Φ_{new} and Φ_{TS} are considered as a feature distinction function described by

$$\Gamma_2^2(\Phi_{\text{new}}, \Phi_{\text{TS}}) = \left[\log \frac{\int_{\Omega} H(-\Phi_{\text{new}}(x)) dx}{\int_{\Omega} H(-\Phi_{\text{TS}}(x)) dx} \right]^2 + [\kappa(H(-\Phi_{\text{new}})) - \kappa(\Phi_{\text{TS}})]^2. \quad (23)$$

Other shape descriptors can be added to regulate the feature distinction, such as the aspect ratio or the degree of symmetry. Here, the aspect ratio of Φ is defined as the ratio between the lengths of the major and the minor axes, where these lengths are obtained from eigenvalues of the covariance matrix of Φ . Also, the symmetry degree can be estimated by computing a difference between the original shape and its reflected superimposition. One can find several relevant shape descriptors in [44]. Now, the similarity shape energy is obtained from:

$$E_{\text{shape}} = \exp \left\{ -\frac{1}{2}(\Gamma_1^2 + \Gamma_2^2) \right\}. \quad (24)$$

From (24), disappearance occurs when the shape similarity energy is sufficiently small. Indeed, E_{shape} will typically decrease gradually as the object disappears from the scene. Reacquisition of the tracked object is completed by evolving the curve in the search region until a candidate with sufficiently high shape similarity energy is detected. Thus, we determine disappearance or reappearance if E_{shape} in (24) satisfies certain criteria. To detect disappearance, we have

$$E_{\text{shape}}(\Gamma_1^2, \Gamma_2^2, (\Phi_{\text{TS}})_{t-1}, \Phi_t | Y_{1:t}) < \eta_d < \overline{E_{\text{shape}}} \quad (25)$$

and, to detect reappearance, the condition is given by

$$E_{\text{shape}}(\Gamma_1^2, \Gamma_2^2, (\Phi_{\text{TS}})_{t_d-1}, \Phi_t | Y_{t_d:t}) > \eta_r > \overline{E_{\text{shape}}} \quad (26)$$

where $\overline{E_{\text{shape}}}$ is the arithmetic mean of E_{shape} up to $t-1$. η_d and η_r are positive thresholds between 0 and 1 for each case of detection. The value t_d is the time when the object disappears. If the condition (25) is *true* during tracking, then one considers target as having disappeared, and similarly for the reappearance condition (26). Note that the proposed approach is performed without the prior training sets and puts no time limitation on the object's absence.

In Fig. 5, a synthetic image sequence is used to test the algorithms for disappearance and reappearance handling based on the proposed similarity shape energy in (24). The sequence includes several different shapes and is generated with Gaussian white noise of zero mean and 0.01 variance. Fig. 6 shows the template shape as well as the shape energies corresponding to each shape of the shape sequence. Some different shapes' energies are also

given in Fig. 6 to show the variation of the shape energy with respect to the template shape. The change of $\overline{E_{\text{shape}}}$ and E_{shape} for this sequence is shown in Fig. 7. Note that none of the shape energies satisfied the detection condition in (26) except the square. Thus, only the square is acquired again during the course of tracking.

E. Discussion of the Reacquisition Method

Measuring the similarity between two objects is an important task in many areas, such as image retrieval, object identification and visual tracking. A comprehensive review of such similarity measures is beyond the scope of this paper; see [45] and references therein. Accordingly, here, the shape similarity-based reacquisition method proposed in this paper is discussed and compared with other similarity measures in the context of visual tracking.

It is quite common in object tracking that an image region that includes the object of interest is represented by a template model of the pixel intensities or other relevant feature information, and then is compared to candidate regions to determine the displacement of the object over consecutive frames. To compare or evaluate similarities between the template and an observed image (or a region), the Bhattacharyya distance [46], [47], Kullback–Leibler divergence [48], and the normalized mutual information [49] have been proposed to evaluate the degree of a similarity between intensity distributions. These probability-distance measures usually use histogram-based distributions, which describe the image region of interest. However, such histogram-based approaches are insensitive to the possible deformations of an object due to the lack of spatial information. Taking into consideration the invariance to translation and robustness to lighting changes, some edge information and the orientation histogram may be employed to measure the similarity with respect to the Euclidean distance as in [50], [51]. In addition, Birchfield and Rangarajan [52] introduced a spatial histogram, or *spatiogram*, in which some information of the geometry of an object feature distribution is involved. This showed improved performance of tracking for a mean-shift-based tracker. On the other hand, the similarity measure used in [52] is not suited for small spatial changes of the given object's features. To overcome this, the work of [53] formulates a similarity measure based on the spatiogram and the Bhattacharyya coefficient for robust object localization. Recently, Ling and Okada [54] exploited the diffusion process to derive a cross-bin histogram distance, called *diffusion distance*, which is robust to the object's deformation and a lighting change in histogram-based local descriptors.

We have tested the similarity measures discussed above in the real sequence of Fig. 12 of the experimental Section IV to comparatively evaluate the proposed shape-similarity energy. Fig. 8 shows the graph of the degree of similarity for similarity measures, such as the Bhattacharyya distance, the diffusion distance, Kullback–Leibler divergence, the normalized mutual information, the spatiogram, and the edge-orientation histogram. In Fig. 8, we see that most measures moderately catch the similarities between objects over the consecutive frames in the presence of the object but their similarity values are severely unstable during the disappearance portion of the video. To reacquire the tracked target, a similarity measure should provide the discriminating threshold to separate the state of the target's presence from the state of the target's absence. To evaluate this, statistical information, such as a maximum value and a variance of similarity values during the disappearance section of the tested sequence are computed and recorded in Fig. 9. From the results of Fig. 9, other measures' variances and maximum values in the disappearance section are too large to allow the distinguishable detection of the target's reappearance. However, the proposed shape-similarity energy gradually decreases while the target is turning to the left or right and its variance and maximum value in the disappearance section are of a size such that the detection threshold can be properly selected in the detection conditions in (25) and (26).

From the comparative study with other similarity measures, the present shape-based similarity measure seems to be more discriminative and shows the applicability for target reacquisition of a moving vehicle. However, since the tolerance of the template shape depends on the variation of the shape learned online from previously tracked object, the proposed method can fail to reacquire the track of the object due to deformations and shapes that have not been learned in our scheme. Theoretically, one can increase the number of prior training sets to increase the possibility of reacquisition of the track, but it does not always guarantee reacquisition of the object with an unregistered shape. Thus, as mentioned in the previous Section III-D, it must be pointed out that it is necessary to assume that the tracked object maintains its shape similarly while it has been out of view regardless of the number of training shapes before disappearance.

Finally, the robustness of the proposed shape energy to noise was tested as seen in Fig. 10. The shapes tested were generated with diverse noise levels of Gaussian noise whose variance ranges from $\sigma_n^2=1\%$ to $\sigma_n^2=100\%$. Note that, even though all similarity-shape energies of the shapes decrease as the noise level increases, the rectangular shape has the biggest shape energy and its energy difference between noises (or decreasing rate) is the smallest because it is the most similar shape to the template.

F. Framework for the Proposed Algorithms

The diagram of the entire tracking procedure is illustrated in Fig. 11. The various algorithms are shown in the order in which they are performed, and the loop indicates iteration to the next frame. In our applications, depth maps of pairs of stereo images are constructed from a stereo matching system at each time step. The initialization process to identify the position of the object of interest is only performed for the first frame. The diagram shows that the procedure starts with global and local estimation of the object, and finishes with shape analysis for disappearance and reappearance handling.

IV. Experiments

The proposed algorithm was tested on three different videos of moving vehicles obtained from stereoscopic image sequences. While the experimental sequences are composed of numerous stereo image pairs, results are shown only on the left image for simplicity. Additionally, the computed depth images from all sequences have low contrast, high noise, and cluttered backgrounds. In the experiments most of the parameters were held fixed across all trials. Specifically, the number of particles $N=40$ was chosen empirically to give good coverage without adding significant computational burden. Overall, the tracker produced accurate track signals on the three sequences with an acceptable computation time of approximately 20 seconds per frame on a 3.6-GHz Windows machine with 2 GB of RAM. The value of σ_w^2 , which is used to control the Gaussian weighting on depth information, is set automatically in our experiments as described in Section III-B.

To deal with disappearance and reappearance of the object, two thresholds η_d and η_r were selected based on the expected shape variation within a sequence and observations of how this affects tracking performance. For example, η_d should not be too small or the object may not be identified as “disappeared” before it disappears from the view. Similarly, the object may not be robustly reacquired if η_r is too large. A detailed analysis of these parameters is given in the sections below and in Figs. 13 and 16. However, for our experiments, choosing parameters in the ranges $\eta_d = [0.2, 0.3]$ and $\eta_r = [0.5, 0.6]$ gave robust detection.

A. Truck Sequence I

This sequence is taken from a moving camera following a truck while trying to keep a constant distance of approximately 20 m. It is comprised of 1168 frames in which the truck moves in and out of the field of view several times: once while turning left (frames 278 to 328), and once while turning right (frames 950 to 1020). In addition, the sequence shows changing illumination conditions and significant clutter. Weighted depth maps of each frame are obtained using the Gaussian function with $\sigma_w^2 = [1^2, 1.2^2]$ in (16). Tracking results with the proposed algorithms are shown in Fig. 12. We can see from the results that even though the truck has very weak edges and the background is quite cluttered, the truck is robustly tracked. Furthermore, the results demonstrate the capability of the proposed scheme to handle disappearance and reappearance. For disappearance and reappearance detection, η_d and η_r are selected as 0.25 and 0.55, respectively. The change of $\overline{E_{\text{shape}}}$ and E_{shape} during frames 190 through 390 is shown in Fig. 13. The shape energy of the truck decreases as the truck turns to the left and eventually drops below η_d as it disappears from the image domain. After ≈ 50 frames, the truck's reappearance is detected when it becomes visible and its shape-similarity energy satisfies the reappearance condition in (26).

B. Truck Sequence II

In this sequence, the truck moves randomly in clockwise and counterclockwise loops. Note that the shape of the truck deforms significantly whenever it turns left or right. In addition, 3-D range values (i.e., depth values) decrease and increase markedly as it moves back and forth. The sequence includes 866 frames with a cluttered background. The parameter σ_w^2 is chosen to be $[1^2, 3^2]$. It has a larger range than in other experiments because the change of the object's size is much more dramatic. Fig. 14 shows several frames that demonstrate the robust tracking results. Note that no disappearance or reappearance of the truck took place because it was visible throughout the sequence.

C. Van Sequence

Fig. 15 shows the proposed algorithm robustly tracking a van in a cluttered urban environment throughout a 45-frame sequence. Initially, the van takes a turn to the left causing significant shape deformation. Then, it gradually moves away from the camera until it vanishes into the distance. Weighted depth maps are created with $\sigma_w^2 = [1^2, 2^2]$. The disappearance of the van is successfully detected with $\eta_d = 0.2$. This smaller value is chosen because of the drastic change in perceived shape energy of the target (side of the van to back of the van) in just 10 frames. The change of $\overline{E_{\text{shape}}}$ and E_{shape} for this sequence is shown in Fig. 16. The shape energy of the van decreases gradually and eventually drops below the detection threshold η_d as it disappears into the distance.

V. Conclusion and Limitations

In this paper, we described a straightforward approach for tracking moving objects using 3-D range data by incorporating a contour tracker in conjunction with PCA-based shape analysis. Region-based active contours are employed to track deformations of an object using the Bhattacharyya gradient flow on the Gaussian weighted depth map. A particle filtering scheme combined with the active contour model is utilized for global position estimation of the target. In addition, we proposed similarity shape energy in order to handle the possible disappearances and reappearances of the tracked object. We demonstrated the ability of the proposed method to track targets in cluttered environments and to detect automatically the target after it has gone out of frame and then reappeared.

The proposed algorithm has some limitations which we intend to overcome in our future work. First, the proposed algorithm is limited in that it will fail if the object reappears with an unexpected shape that is unlike those seen previously. Second, the proposed algorithm will lose the target if the unexpected object occludes the target or appears in a similar depth position with a similar shape to the tracked object in the course of tracking.

It is a possible solution for some limitations to incorporate the information of color and shape of an object from an original sequence into the proposed algorithms. Also, our future research will include the challenging problem of tracking multiple objects with varying shapes. To improve reliability and accuracy of tracking, more sophisticated shape analysis will be needed.

Acknowledgments

The authors would like to thank all of the reviewers for their valuable comments and suggestions.

This work was supported in part by grants from the National Science Foundation, the Air Force Office of Scientific Research, and the Army Research Office and the National Institute of Health under Grant NAC P41 RR-13218 through Brigham and Women's Hospital. This work is part of the National Alliance for Medical Image Computing (NAMIC), funded by the National Institutes of Health through the NIH Roadmap for Medical Research, Grant U54 EB005149. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Theo Gevers.

References

1. Blake, A.; Isard, M., et al. Active Contours. New York: Springer; 1998.
2. Bishop G, Welch G, Allen B. Tracking: Beyond 15 minutes of thought. SIGGRAPH Course. 2001; 11
3. Hu W, Tan T, Wang L, Maybank S. A survey on visual surveillance of object motion and behaviors. IEEE Trans. Syst., Man, Cybern. C. 2004 Mar; vol. 34(no. 3):334–352.
4. Yilmaz A, Javed O, Shah M. Object tracking: A survey. ACM Computing Surveys (CSUR). 2006; vol. 38(no. 4):1–45.
5. Cai Q, Aggarwal J. Tracking human motion in structured environments using a distributed-camera system. IEEE Trans. Pattern Anal. Mach. Intell. 2002 Nov; vol. 21(no. 11):1241–1247.
6. Huang T, Russell S. Object identification in a Bayesian context. Proc. 15th Int. Joint Conf. Artif. Intell. 1997:1276–1282.
7. Chilgunde A, Kumar P, Ranganath S, Huang W. Multi-camera target tracking in blind regions of cameras with non-overlapping fields of view. Proc. Brit. Mach. Vision Conf. 2004:397–406.
8. Chen K, Lai C, Hung Y, Chen C. An adaptive learning method for target tracking across multiple cameras. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 2008:1–8.
9. Yilmaz A, Li X, Shah M. Contour-based object tracking with occlusion handling in video acquired using mobile cameras. IEEE Trans. Pattern Anal. Mach. Intell. 2004 Nov; vol. 26(no. 11):1531–1536. [PubMed: 15521500]
10. Rath Y, Vaswani N, Tannenbaum A, Yezzi A. Particle filtering for geometric active contours with application to tracking moving and deforming objects. Proc. CVPR. 2005; vol. 2:2–9.
11. Rath Y, Vaswani N, Tannenbaum A, Yezzi A. Tracking deforming objects using particle filtering for geometric active contours. IEEE Trans. Pattern Anal. Mach. Intell. 2007 Aug.vol. 29(no. 8): 1470. [PubMed: 17568149]
12. Dambreville S, Rath Y, Tannenbaum A. Tracking deformable objects with unscented Kalman filtering and geometric active contours. Proc. Amer. Control Conf. 2006; vol. 6:2856–2861.
13. Nguyen H, Smeulders A. Fast occluded object tracking by a robust appearance filter. IEEE Trans. Pattern Analysis and Machine Intelligence. 2004; vol. 26(no. 8):1099–1104.
14. Latecki L, Miezianko R. Object tracking with dynamic template update and occlusion detection. Proc. 18th Int. Conf. Pattern Recognit. 2006:556–560.

15. Jackson J, Yezzi A, Soatto S. Tracking deformable moving objects under severe occlusions. *Proc. IEEE Conf. Decision Control*. 2004;2990–2995.
16. Bartesaghi A, Sapiro G. Tracking of moving objects under severe and total occlusions. *Proc. IEEE Int. Conf. Image Process*. 2005; vol. 1:301–304.
17. Talukder A, Matthies L. Real-time detection of moving objects from moving vehicles using dense stereo and optical flow. *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst*. 2004; vol. 4:3718–3725.
18. Alattar A, Jang J. A newstereo correspondence method for snake-based object segmentation. *Proc. IEEE Int. Conf. Image Process*. 2007; vol. 3:381–384.
19. Gelautz M, Markovic D. Recognition of object contours from stereo images: An edge combination approach. *Proc. 2nd Int. Symp. 3D Data Process., Visualization Transmission*. 2004;774–780.
20. Freedman D, Zhang T. Active contours for tracking distributions. *IEEE Trans. Image Process*. 2004 Apr; vol. 13(no. 4):518–526. [PubMed: 15376586]
21. Michailovich O, Rathi Y, Tannenbaum A. Image segmentation using active contours driven by the Bhattacharyya gradient flow. *IEEE Trans. Image Process*. 2007 Nov; vol. 16(no. 11):2787–2801. [PubMed: 17990755]
22. Gordon N, Salmond D, Smith A. Novel approach to non-linear/non-Gaussian Bayesian state estimation. *Proc. Inst. Electr. Eng.—Radar Signal Process*. 1993; vol. 140(no. 2):107–113.
23. Doucet A, Godsill S, Andrieu C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Computing*. 2000; vol. 10(no. 3):197–208.
24. Ristic, B.; Arulampalam, S.; Gordon, N. *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Norwood, MA: Artech House; 2004.
25. Simon, D. *Optimal State Estimation: Kalman, H Infinity, and Non-linear Approaches*. New York: Wiley-Interscience; 2006.
26. Arulampalam M, Maskell S, Gordon N, Clapp T, Sci D, Organ T, Adelaide S. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process*. 2002 Feb; vol. 50(no. 2):174–188.
27. Sandhu R, Georgiou T, Tannenbaum A. Tracking with a new distribution metric in a particle filtering framework. *Proc. IST/SPIE Symp. Electron. Imaging*. 2008; vol. 6813:68310N.
28. Kass M, Witkin A, Terzopoulos D. Snakes: Active contour models. *Int. J. Comput. Vis*. 1988; vol. 1(no. 4):321–331.
29. Tannenbaum A. Three snippets of curve evolution theory in computer vision. *Math. Comput. Modelling*. 1996; vol. 24:103–120.
30. Osher S, Sethian J. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. *J. Computational Phys*. 1988:12–49.
31. Sethian, J. *Level Set Methods and Fast Marching Methods*. Cambridge, U.K.: Cambridge Univ.; 1999.
32. Caselles V, Kimmel R, Sapiro G. Geodesic active contours. *Int. J. Comput. Vis*. 1997; vol. 22(no. 1):61–79.
33. Chan T, Vese L. Active contours without edges. *IEEE Trans. Image Process*. 2001 Feb; vol. 10(no. 2):266–277. [PubMed: 18249617]
34. Cremers D, Rousson M, Deriche R. A Review of statistical approaches to level set segmentation: Integrating color, texture, motion, and shape. *Int. J. Comput. Vis*. 2007; vol. 72(no. 2):195–215.
35. Leventon M, Grimson W, Faugeras O. Statistical shape influence in geodesic active contours. *Proc. IEEE Conf. Comput. Vis. Pattern Recognit*. 2000; vol. 1:316–323.
36. Scharstein D, Szeliski R. High-accuracy stereo depth maps using structured light. *Proc. IEEE CVPR*. 2003; vol. 1:195–202.
37. Yezzi A, Soatto S. Deformation: Deforming motion, shape average and the joint registration and approximation of structures in images. *Int. J. Comput. Vis*. 2003; vol. 53(no. 2):153–167.
38. Marr D, Poggio T. Cooperative computation of stereo disparity. *Science*. 1976; vol. 194(no. 4262):283. [PubMed: 968482]
39. Kanade T, Okutomi M. A stereo matching algorithm with an adaptive window: Theory and experiment. *Proc. IEEE Int. Conf. Robot. Autom*. 1991:1088–1095.
40. Horn, B. *Robot Vision*. Cambridge, MA: MIT Press; 1986.

41. Scharstein D, Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *Int. J. Comput. Vis.* 2002; vol. 47(no. 1/2/3):7–42.
42. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ.; 2003.
43. Rathi Y, Vaswani N, Tannenbaum A. A generic framework for tracking using particle filter with dynamic shape prior. *IEEE Trans. Image Process.* 2007 May; vol. 16(no. 5):1370. [PubMed: 17491466]
44. da Fontoura Costa, L. *Shape Analysis and Classification: Theory and Practice*. Boca Raton, FL: CRC; 2001.
45. Duda, R.; Hart, P.; Stork, D. *Pattern Classification*. New York: Wiley-Interscience; 2000.
46. Comaniciu D, Ramesh V, Meer P. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* 2003 Apr; vol. 25(no. 4):564–575.
47. Vermaak J, Lawrence N, Perez P. Variational inference for visual tracking. *Proc. IEEE CVPR.* 2003; vol. 1:773–780.
48. Elgammal A, Duraiswami R, Davis L. Probabilistic tracking in joint feature-spatial spaces. *Proc. IEEE CVPR.* 2003; vol. 1:781–788.
49. Loutas E, Pitas I, Nikou C. Probabilistic multiple face detection and tracking using entropy measures. *IEEE Trans. Circuits Syst. Video Technol.* 2004 Jan; vol. 14(no. 1):128–135.
50. Yang C, Duraiswami R, Davis L. Fast multiple object tracking via a hierarchical particle filter. *Proc. IEEE Int. Conf. Comput. Vis.* 2005; vol. 1:212–219.
51. Lowe D. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* 2004; vol. 60(no. 2):91–110.
52. Birchfield S, Rangarajan S. Spatiograms versus histograms for region-based tracking. *Proc. IEEE CVPR.* 2005:1158–1163.
53. Conaire C, O'Connor N, Smeaton A. An improved spatiogram similarity measure for robust object localization. *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.* 2007; vol. 1:1069–1072.
54. Ling H, Okada K. Diffusion distance for histogram comparison. *Proc. IEEE CVPR.* 2006; vol. 1:246–253.

Biographies

Jehoon Lee (S'05) is currently working toward the Ph.D. degree from the Georgia Institute of Technology, Atlanta.

His research interests include the usage of linear or nonlinear filters and geometric active contours to solve various problems of segmentation, visual tracking, and pose estimation in computer vision.

Shawn Lankton (M'09) received the Ph.D. degree from the Georgia Institute of Technology, Atlanta.

While with the Georgia Institute of Technology, he researched variational optimization to solve segmentation and visual tracking problems. He is currently a strategy consultant with McKinsey & Company, New York.

Allen Tannenbaum (F'09) received the Ph.D. degree in mathematics from Harvard University, Cambridge, MA, in 1976.

He is a faculty member with the School of Electrical and Computer Engineering and the School of Biomedical Engineering, Georgia Institute of Technology, Atlanta. His research interests are in image processing, computer vision, and control.

(a)

(b)

(c)

Fig. 1.

(a) Original image (left) and a disparity map (right). Segmenting the cone: initial, intermediate and final contours using (b) only image intensity information and (c) a disparity map. Images are zoomed in for better visualization. The test data set is taken from the Middlebury Stereo Vision Page (<http://vision.middlebury.edu/stereo/>) [36].

Fig. 2.

Concept of the weighting range information with a Gaussian kernel $\mathcal{N}(d_p, \sigma_w^2)$. The dashed cube describes the probable region, which is weighted heavily by the given kernel.

Fig. 3. Gaussian weighting functions (left column), the weighted depth maps (middle column), and segmentation results (right column). The Gaussian weighting kernels are constructed with (a) $\sigma_w^2 = 0.1^2$, (b) $\sigma_w^2 = 1^2$ and $\sigma_w^2 = 10^2$. γ and d_p are 255 and 18, respectively.

Fig. 4.

Top row of (a) and (b), left to right: original left image with an initial contour, depth map, and the weighted depth map with (a) $f_w \sim \mathcal{N}(18, 1^2)$ and (b) $f_w \sim \mathcal{N}(57, 2.5^2)$. Bottom row of (a) and (b), left to right: segmentation results using only image intensity information, depth map, and the weighted depth map. Images are zoomed in for better visualization. $\gamma = 255$.

Fig. 5. Shape sequence. Frame order: from left to right, top to bottom. Note that the reacquisition of the tracked square is achieved after (k) frame 180 and that the other shapes except the tracked square are ignored during the course of tracking. (a) Frame 1. (b) Frame 57. (c) Frame 70. (d) Frame 89. (e) Frame 105. (f) Frame 119. (g) Frame 135. (h) Frame 149. (i) Frame 165. (j) Frame 176. (k) Frame 180. (l) Frame 195.

Fig. 6.

(a) Template shape. (b)–(e) Shape energies corresponding to various shapes of the shape sequence. (f)–(i) Shape energies of other shapes. (a) Φ_{TS} . (b) 0.66. (c) 0.62. (d) 0.5. (e) 0.37. (f) 0.52. (g) 0.45. (h) 0.39. (i) 0.21.

Fig. 7.

Graph of the shape energy for each frame of the shape sequence. The dashed-dotted line and solid line denote $\overline{E}_{\text{shape}}$ and E_{shape} , respectively. $\eta_d = 0.45$ and $\eta_r = 0.65$.

Fig. 8.

Graph of the degree of similarity for some similarity measures over some frames of the sequence in Fig. 12; the proposed shape-similarity energy (SE), the Bhattacharyya distance (BH), the diffusion distance (DF), Kullback-Leibler divergence (KL), the normalized mutual information (MI), the spatiogram (SG), and the edge-orientation histogram (EO) were tested. In the degree of similarity, 0 and 1 indicate complete mismatch and perfect similarity, respectively.

	SE	BH	DF	KL	MI	SG	EO
MAX	0.601	0.983	0.833	0.796	0.621	0.997	0.975
MIN	0	0	0	0	0	0	0
AVG	0.257	0.708	0.438	0.268	0.420	0.699	0.576
VAR	0.028	0.171	0.077	0.083	0.060	0.167	0.118

Fig. 9.

Table displaying statistical information in the disappearance section during ≈ 50 frames of the sequence in Fig. 8. MAX, MIN, AVG, and VAR denote a maximum value, a minimum value, an average, and a variance, respectively.

σ_n^2	Shapes with a noise and their energies			
10%	0.637	0.567	0.421	0.321
25%	0.63	0.567	0.425	0.309
50%	0.621	0.565	0.415	0.299
100%	0.615	0.555	0.392	0.268
$d_{\sigma_n^2}$	0.045	0.065	0.108	0.102

Fig. 10.

Quantitative results for the robustness of the proposed shape energy to a noise for various shapes of the sequence in Fig. 5. Shape energies corresponding to various shapes in the diverse noise levels are displayed at the bottom of each image. Gaussian noises with $\sigma_n^2=10\%$ (first row), $\sigma_n^2=25\%$ (second row), $\sigma_n^2=50\%$ (third row), and $\sigma_n^2=100\%$ (fourth row) were added, respectively. Refer to Fig. 6 for the template shape and the shape energy of the Gaussian noise with $\sigma_n^2=1\%$. $d_{\sigma_n^2}$ denotes the difference of shape energies between $\sigma_n^2=1\%$ and $\sigma_n^2=100\%$.

Fig. 11. Overall framework diagram of the proposed algorithms (I_L and I_R denote a left image and a right image, respectively).

Fig. 12.

Truck sequence I. Frame order: from left to right, top to bottom. The upper and bottom left images to each frame are a depth map and the weighted depth map, respectively. Note that disappearance and reappearance handling is achieved between (c) and (e), and between (i) and (k). The condition of illumination is changed after 865th frame. (a) Frame 1. (b) Frame 232. (c) Frame 272. (d) Frame 295. (e) Frame 332. (f) Frame 450. (g) Frame 702. (h) Frame 865. (i) Frame 920.

Fig. 13.

Graph of the shape energy for some frames of the truck sequence I. The dashed–dotted line and solid line denote $\overline{E_{\text{shape}}}$ and E_{shape} , respectively. Note that the detection of disappearance and reappearance of the tracked truck is achieved with $\eta_d = 0.25$ and $\eta_r = 0.55$.

Fig. 14.

Truck sequence II. Frame order: from left to right, top to bottom. (a) Frame 26. (b) Frame 65. (c) Frame 108. (d) Frame 135. (e) Frame 176. (f) Frame 201. (g) Frame 234. (h) Frame 307. (i) Frame 324. (j) Frame 345. (k) Frame 452. (l) Frame 560. (m) Frame 617. (n) Frame 650. (o) Frame 743. (p) Frame 770. (q) Frame 799. (r) Frame 838.

Fig. 15.

Van sequence. Frame order: from left to right, top to bottom. Note that the detection of disappearance is achieved in the last part of the sequence. (a) Frame 1. (b) Frame 3. (c) Frame 16. (d) Frame 35. (e) Frame 40. (f) Frame 45.

Fig. 16.

Graph of the shape energy for each frame of the van sequence. The dashed-dotted line and solid line denote $\overline{E_{\text{shape}}}$ and E_{shape} , respectively. $\eta_d = 0.2$ and $\eta_r = 0.55$.