Edge-based Object Tracking

Candidate: Xuebin Qin
Supervisor: Dr. Martin Jagersand

August 10, 2018

Abstract

Visual tracking is an important issue in both computer vision and robotics. Many existing trackers, *e.g.* Lucas-Kanade (KLT), Tracking-Learning-Detection (TLD), Kernelized Correlation Filter tracker (KCF), use regional information to estimate the trajectory and transformation of the to-be-tracked targets. To achieve robust and stable performance, relatively large area of image region with rich textures is required for object tracking. Region-based trackers have some limitations. First, these regions are usually represented by bounding boxes or quadrilaterals which are inaccurate for describing complex targets. Second, region information is not always available. For example, it is difficult to obtain enough region information from these objects: uneven surfaces, hollow objects, thin targets and tiny structures, such as cup rims, cables, rings and so on. Being able to track these objects are essential in applications such as robot manipulation and grasping. Additionally, many targets in our daily life are non-lambertian so that their regional appearance changes drastically with even slight motion or transformation.

Motivated by the above limitations of region-based trackers, the proposed research targets at employing edge and boundary information to facilitate tracking of targets in unstructured environment. First, an effective and efficient edge-based representation method will be proposed to acquire accurate and detailed descriptions of the to-be-tracked targets. Second, a novel tracking framework which utilizes edge and boundary information will be developed. In addition, to evaluate and validate the newly proposed method, a well-annotated tracking dataset which contains video sequences and pixel-accurate ground truth will be created. The success of the proposed work will be beneficial to both computer vision and robotics communities.

Contents

List of Figures

1 Introduction

An image gives a static view of the environment while a video sequence describes the dynamic changes. Visual object tracking is the process of estimating the trajectory and transformation of a moving target (or multiple targets) over time from the given video sequences [61]. As a key technology in computer vision, visual object tracking has a variety of applications, *e.g.* augmented reality [28], visual servoing [17], *etc.*

Although visual tracking achieves significant improvements of robustness with the development of deep learning, learning based trackers [11], [22], [59] are usually low DOF (Degree Of Freedom) and less accurate. Most of them can only estimate image coordinates of object centroid or bounding box, sometimes augmented with scale and rotation. In contrast, high DOF trackers [3], [51], [32] can provide more accurate transformations, *e.g.* homography, affine of planar targets. However, they are less robust in real-world scenario [41]. In addition, most of existing trackers focus on tracking image patches defined by bounding boxes [59] , quadrilaterals [51] or polygons. In real-world robot applications, the tracking targets are often specific parts or structures of objects, which can not be described by those simple shapes. For example, given a mug, its handle or contour has to be tracked in a grasping task, while its rim should be tracked in a pouring task. Tracking of these irregular and complex structures without using fiducial markers is challenging for existing trackers [51], [59] due to harsh conditions including the presence of texture-less target regions, occlusions, non-rigid deformations, non-Lambertian surfaces, changing lighting conditions, cluttered background and drastic contents changing of supportive regions. As we can see in Fig. 1, the texture information inside the red quadrilaterals of the first row is relatively stable and rich while that inside the red contour of the second row changes drastically because of view point changes. It is hard to get enough information from the surrounding areas of the bowl rim to support the rim tracking. Hence, there is a need for an effective and robust real-time approach that can describe and track these kinds of targets accurately.

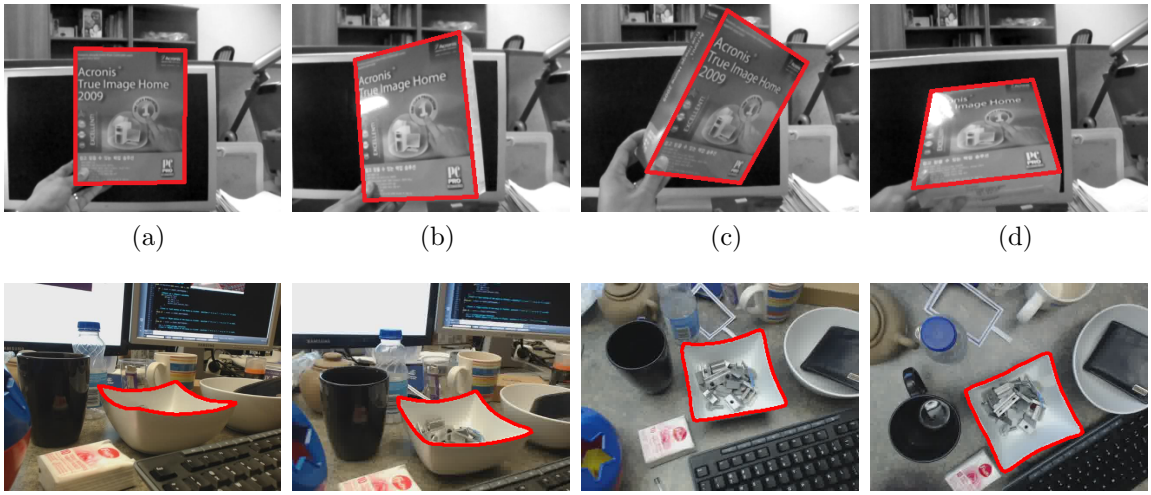


Figure 1. Comparison of region-based tracking [31] and edge-based tracking [42]. The first row shows the region-based homography tracking. Image pixels inside the quadrilaterals are used to support the tracking process. The second row shows the edge-based tracking. Only edge pixels on bowl rims can be taken as tracking cues because their presences are relatively stable with the changing of view angles.

Compared with bounding boxes and quadrilaterals, edges and boundaries are more flexible and accurate in describing those irregular and complex structures. They are also more stable than image regions under various illumination changes. Currently, edges are frequently used in 3D pose estimation of low textured targets [9], [13]. These edge templates are obtained from pre-built three-dimensional (3D) models, which are rarely available in unstructured, natural environments [42]. Compared with 3D wireframe models, two-dimensional (2D) edge templates are more accessible. Registration-based methods [23], [25] are usually utilized to track these kind of targets by estimating their affine or homography transformations between consecutive frames. These registration-based template tracking methods are fast. However, they are not able to handle non-rigid transformations and are sensitive to outlier edge pixels. Variational methods, such as level sets [62], [4], [53] and active contour [46], have already been used to track the contours of those non-rigid targets. To use variational methods, there has to be high contrast between the target and the background. Saliency-based methods [50] can also be used to track non-rigid objects. However, these methods are more likely to trap in local minima in cluttered environments [41] and are sensitive to occlusions. Additionally, most of them require that the to-be-tracked targets have to be represented by closed boundaries.

Due to its advantages in flexibility and accuracy, edge-based object tracking is complementary to region-based tracking. However, it has not been fully studied as described above and there is much interest in using it for different applications. For example, it can provide accurate object silhouette for 3D reconstruction by shape from silhouette and for robot hand grasping in visual servoing.

Motivated by the limitations of the existing methods and the demand for real-world applications, I propose to develop new edge-based approaches for target-oriented tracking. Particularly, the following problems will be considered: (1) how to define and initialize the to-be-tracked targets accurately and efficiently? (2) how to track edge-based targets with arbitrary shapes and transformations while overcoming the

impacts caused by low textures, local minima, occlusions and aperture problems? (3) how to create relationships and constraints among multiple targets to achieve more robust and accurate tracking performance? (4) how to design and create a new dataset for edge-based targets evaluation and validation? The success of the proposed research will benefit both academia and industrial communities.

2 Background and Related Work

Edge-based object tracking methods can be categorized into three main classes:

(1) *registration-based edge template tracking*, which estimates the geometric transformations, such as affine and homography, of planar rigid templates from one frame to the next one [23], [58], [25], [34],

(2) *segmentation-based methods for contour tracking*, which determines boundaries on each frame by segmenting images into foreground and background using techniques like graph-cuts [20], level sets [62], [46], [4], [53],

(3) *grouping-based closed boundary tracking*, which searches for a special cycle of low level primitives (e.g. edge fragments, line segments) forming the closed boundaries [50], [42], [43].

2.1 Registration-based Edge Template Tracking

Registration-based trackers can estimate both low and high degree-of-freedom (DOF) transformations like affine and homography $\mathbf{H}$ (or $\mathbf{W}(\mathbf{x};\mathbf{p})$) of planar rigid targets by minimizing the alignment errors between the templates $\mathbf{T}(\mathbf{x})$ and the incoming frames $\mathbf{I}(\mathbf{x})$. This kind of methods have been widely used in image alignment and tracking. They can be divided into two categories: *feature-based methods* and *direct methods*. Given an image template $\mathbf{T}(\mathbf{x})$ and a target frame $\mathbf{I}(\mathbf{x})$,

feature-based methods estimate their geometric transformation $\mathbf{H}$ by minimizing the geometric distances of their detected and matched feature points $\{\mathbf{x}^{\mathbf{c}}_{\mathbf{tmp}}, \mathbf{x}^{\mathbf{c}}_{\mathbf{trg}}\}$. Many keypoints (corners or blobs) detectors and descriptors [19] can be utilized here. *Direct methods* operate on pixel-level intensities directly. They estimate the transformation $\mathbf{H}$ by minimizing the pixels' appearance differences $\mathcal{C}(f(\mathbf{T}(\mathbf{W}(\mathbf{x};\, \mathbf{p}))), f(\mathbf{I}))$ including sum of square difference (SSD) [21], normalized cross correlation (NCC) [49] and mutual information (MI) [10].

In edge-based object tracking, the templates are binary edge maps representing sets of edge pixels, which are designed beforehand (CAD models) or initialized in the first frame, instead of gray scale image pixels. Similar to intensity-based image registration [19], [51], there are also two categories of methods for edge template tracking: *feature-based methods* and *direct methods*.

2.1.1 Feature-based methods

Given two detected edge maps from the reference and current frame, it is hard to extract traditional key points by general feature detectors [19]. Hence, in edge template tracking, "keypoint" usually indicates the local feature pattern, which is a set of closely located edge pixels in local region other than detected corner or blob defined in intensity space.

Zhu *et al.* [63] developed an edge-based method for human face tracking. They first manually selected local regions including left eye region, right eye region, nose region and mouth region as local feature points. The edge template was then tracked by estimating its quadratic transformations with respect to each incoming frame based on these feature points and their correspondences in the current frame. Hofhauser *et al.* [23] proposed to model a given edge template by multiple local features and estimate its perspective transformation with respect to the incoming frame by solving a Direct Linear Transformation (DLT) [2] problem between those features and their correspondences. Instead of manual selection [63], these features were extracted by edge pixels

clustering via k-means [35]. Only features with large gradient magnitude in multiple directions (more than one), which means they are able to handle the aperture problem, were retained. The DLT equation was then formulated using the displacements of those features. The displacement of each feature was obtained by estimating its local transformation with respect to the target frame based on the gradient direction field. Hofhauser *et al.* [24] further extended this feature-based method for non-rigid object tracking by adding constraints on displacements estimation of neighboring local feature points. Besides, Liu *et al.* [34] introduced shape context matching of edge pixels to estimate the translation, rotation and scale change of extracted object contour. Although their method operates on edge pixels without explicit feature points extraction, it actually takes all the filtered edge pixels as feature points and computes the transformations based on the edge pixels' correspondences.

Note that this kind of methods require relatively rich edge or texture information to extraction enough feature points for transformation estimation. In many cases, this requirement is not satisfied.

2.1.2 Direct methods

In contrast to feature-based methods, direct methods try to estimate motions and transformations of given templates by iteratively aligning them with the current frame based on information of all pixels other than that of extracted feature points [18]. In intensity-based high DOF tracking, the direct methods mainly refer to the Lucas-Kanade method [3] and its variants. The Lucas-Kanade method aligns a template image $T(\mathbf{x})$, where $\mathbf{x} = (x, y)^T$ denotes the column vector containing the pixel coordinates, to an incoming frame $I(\mathbf{x})$ by minimizing the sum of square intensity differences $\mathcal{C}$, Equ. (2), with respect to the parameters $\Delta\mathbf{p}$ of the warp function $\mathbf{W}(\mathbf{x};\mathbf{p})$, which is usually a homography:

$$\mathbf{W}(\mathbf{x};\mathbf{p}) = \frac{1}{1 + p_7 x + p_8 y} \begin{pmatrix} (1 + p_1)x + p_3 y + p_5 \\ p_2 x + (1 + p_4)y + p_6 \end{pmatrix}. \tag{1}$$

$$\mathcal{C} = \sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x};\ \mathbf{p})) - T(\mathbf{x})]^2. \tag{2}$$

$$\mathcal{C}_{te} = \sum_{\mathbf{x}} [I(\mathbf{W}(\mathbf{x};\ \mathbf{p})) + \nabla I \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \Delta \mathbf{p} - T(\mathbf{x})]^2 \tag{3}$$

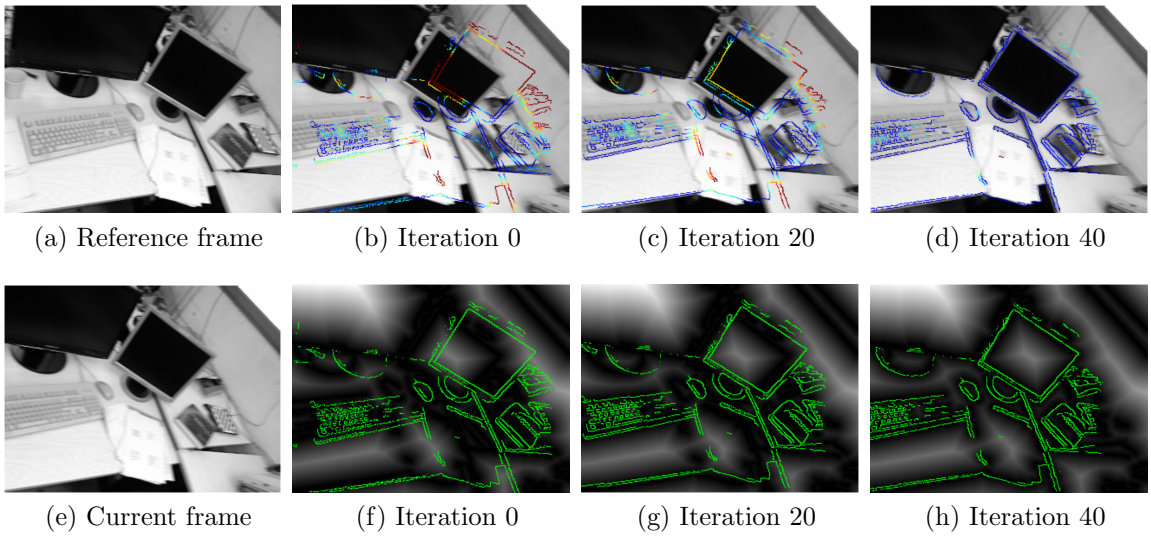
The cost $\mathcal{C}$, Equ. (2), is linearized by the first order Taylor expansion, Equ. (3), where ∇I indicates the gradient of image $I(\mathbf{W}(\mathbf{x}; \mathbf{p}))$. The Lucas-Kanade algorithm iteratively solves for increments to the parameters $\Delta \mathbf{p}$ by optimizing $\arg\min_{\mathbf{\Delta p}} \mathcal{C}_{te}$ and updating $\mathbf{p}$ with $\mathbf{p} \leftarrow \mathbf{p} + \mathbf{\Delta p}$.

One adaption of the intensity-based Lucas-Kanade method [3] for edge template tracking is replacing the differences of intensities in Equ. (2) with that of the distance transform maps as:

$$\mathcal{C}_{et} = \sum_{\mathbf{x}} ||D_I(\mathbf{W}(\mathbf{x}; \mathbf{p})) - D_T(\mathbf{x})||_n \tag{4}$$

where D_I and D_T are the distance transform maps [5], in which the values of the edge pixels and the non-edge pixels are set to zeros and the distances to their nearest edge pixels respectively, generated from the edge map of the incoming frame. $\mathbf{x}$ represents the coordinates of the template pixels. $||\cdot||_n$ denotes the n-th norm. Holzer *et al.* [25] proposed a new method for 3D pose estimation of low textured planar objects based on contour registration. Given a low textured planar template and an input image, this method extracted multiple closed contours from both the template and the input image. The correspondences between the contours extracted from the template and the input image were then created by a trained classifier. To refine the pose estimation, every pair of corresponding contours was registered based on the differences of their distance transform maps as defined in Equ. (4). This method is sensitive to both false positive and false negative edge pixels. Because the distance maps are generated based on the detected closed contours, which is not always guaranteed to be correct in many scenarios.

Another adaption is combining the Lucas-Kanade method [3] with



(a) Reference frame (b) Iteration 0 (c) Iteration 20 (d) Iteration 40

(e) Current frame (f) Iteration 0 (g) Iteration 20 (h) Iteration 40

Figure 2. Edge-based frame alignment for visual odometry [30]. Given the reference frame (a) and the current frame (e), direct edge alignment method extracts edges from both frames. It then generate distance map from the reference edge map, as the gray background terrain shown in (f)-(h). The current edge map, green pixels in (f)-(h), is then aligned with the reference distance map iteratively.

the Chamfer Matching (CM) method [6] as:

$$\mathcal{C}_{et} = \sum_{\mathbf{x}} ||D_T(\mathbf{W}((\mathbf{x};\mathbf{p})))||_n \tag{5}$$

where D_T is the distance map of the edge template and $\mathbf{x}$ indicates the edge pixels extracted from the current frame. It tries to minimize the alignment error between the distance map of the template and the edge pixels of the current frame.

Wu *et al.* [58] employed the combination of a condensation particle filter and an improved chamfer matching for edge-based two DOF object tracking. This method reduced the tracking problem into searching for the 2D locations with minimal alignment cost, which is similar to Equ. (5). Note that this method is still low DoF tracking. The

idea of edge alignment for tracking has also been introduced to visual odometry. Kuse and Shen [30] developed a 6-DOF camera motion estimation method based a direct edge alignment approach, as shown in Fig. 2. They used the Distance Transform in their energy formulation and obtained robust performance against the harsh conditions, such as small convergence basin, noise, changing illumination and fast motion. Wang *et al.* [56] proposed an RGB-D visual odometry method that minimizes both the photometric error and edge alignment error. The combination of these two errors leads to high tracking accuracy and more robust performance on texture-less scenes. In visual odometry methods, the whole frame is utilized to estimate the camera pose. The ratio of misclassified edge pixels is usually small so that their impacts on camera pose estimation are very limited. However, only part of the frame information (regions around the to-be-tracked target) is used in object tracking and the false positive ratio and false negative ratio of the edge pixels classification are usually large. Hence, more techniques for reducing the impacts caused by the misclassified edge pixels are needed.

Based on this edge alignment pipeline, I proposed a novel real-time method for tracking planar edge template [45]. This method estimates the homography of a given edge template by minimizing the alignment error: $\mathcal{C}_{et} = \sum_{\mathbf{x}} ||F_T(\mathbf{W}((\mathbf{x};\mathbf{p})))||$, where $F_T = \sqrt[4]{D_T}$ and D_T is the distance transform map of the edge template, $\mathbf{x}$ represents the coordinates of the detected edge pixels on the current frame, $\mathbf{p}$ denotes the homography parameters. Combined with well designed workflow and edge pixels filtering techniques, this tracker achieves competitive performance in terms of robustness, accuracy and time complexity.

2.2 Segmentation-based Methods for Contour Tracking

In contrast to most existing trackers that use bounding boxes or quadrilaterals to specify the to-be-tracked targets, segmentation-based track-

ing methods produce accurate contours as tracking outputs. Therefore, this kind of methods are often used for non-rigid object contour tracking.

Many object contour tracking methods adapted level-set method [40] as their segmentation baseline. In image segmentation [8] and object tracking [37], the level-set method represents a closed 2D curve Γ by the intersection of a higher dimensional auxiliary function ϕ and a plane, as shown in Fig. 3. The curve Γ is usually described by the zero-level set of ϕ as: $\Gamma = \{(x, y)|\phi(x, y) = 0\}$. Mansouri [37] started using level-set method for region tracking which requires no explicit estimations of motion fields or parameters. Yilmaz *et al.* [62] proposed to track the complete object region by evolving its contour from frame to frame. To reduce the segmentation time and avoid local minima, the energy function of the contour evolving was evaluated in the contour vicinity defined by a band. To track deformable objects with large motions, Rathi *et al.* [46] formulated the particle filter algorithm in the level set framework. The particle filter was used to produce the global and coarse motion of the to-be-tracked contour by estimating a rigid transformation (Affine). The local and fine motion of the target contour was obtained by the level set method. Sun *et al.* [53] proposed a novel supervised level set model for non-rigid object tracking. Instead of using a static or generative model to describe the target, they constructed a discriminative model to differentiate the foreground and background pixels.

Besides, Geodec *et al.* [20] developed a non-rigid object tracking method, HoughTrack, based on generalized Hough-transform. This method couples the voting based detection and back-projection with a rough segmentation obtained by GrabCut [48]. Stefan and Christophe [14] proposed a faster version of HoughTrack, PixelTrack, by using pixel-based descriptors. However, PixelTrack only produces bounding boxes other than contours of targets.

Although these segmentation methods can be used for contour tracking, their prerequisite is the relatively stable regional information enclosed by the target contour, which is not available in many scenarios.

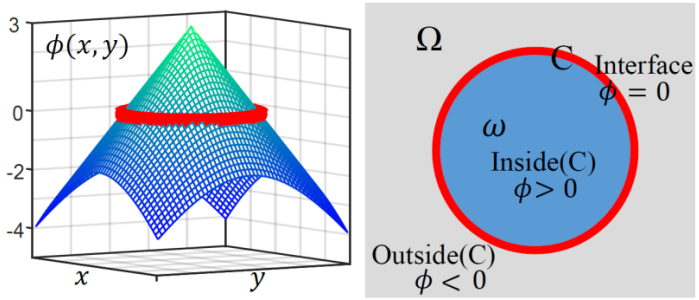
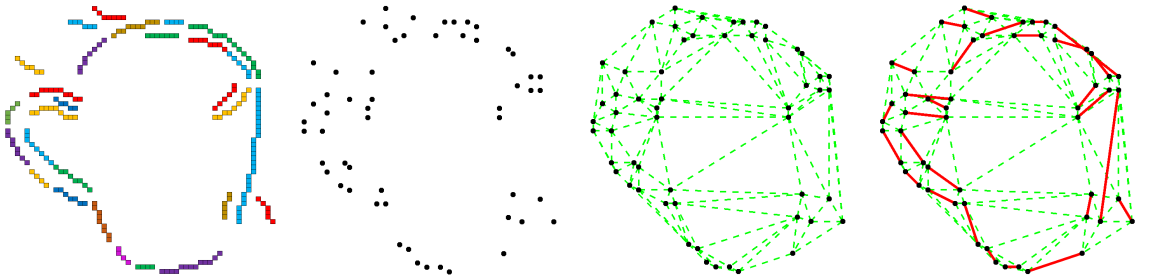


Figure 3. A example of level set segmentation [26]. Left figure visualizes the terrain of a level set function $\phi(x, y)$ and its zero level set, which is represented by a red closed curve. Right figure shows the corresponding segmentation boundary C of the zero level set in 2D space.

2.3 Grouping-based Closed Boundary Tracking

Perceptual grouping algorithms aim at recognizing and organizing subsets of detected independent small geometric primitives, *e.g.* points, edge fragments and line segments, into meaningful intact entities, as shown in Fig. 4. They have been widely used in contours completion [27, 38, 47] and salient closed boundaries extraction [16], [1], [36], [55], [52], [39]. The closure, proximity and good continuation are their most frequently used basic principles draw from Gestalt laws [57]. The closure refers to the tendency of humans' perception to see complete figures or forms even if a picture is incomplete. The proximity indicates the relative Euclidean distances among the detected geometric primitives, which means closer primitives are more likely to be grouped together. The good continuation refers to that humans tend to perceive multiple parts of an interrupted object as an intact entity based on the gradient magnitude and direction information.

Grouping methods have been proposed for salient closed boundary extraction. Wang *et al.* [55] developed a contour grouping method (Ratio Contour, RC) by encoding the proximity as the ratio of the total gap length among the extracted line segments and the perimeter of the target closed boundary $\Gamma = \frac{|B_g|}{\oint_B ds}$, where $|B_g|$ denotes the summation of



(a) Edge fragments (b) Endpoints (c) Gap filling (d) Graph for grouping

Figure 4. General process of perceptual grouping [43]. Given the detected edge fragments of an image, the gaps among those detected edge fragments are firstly filled (c) by endpoints (b) connecting. A graph structure (d) and a grouping cost are then built and defined based on these connections. Finally, the optimal closed boundary is extracted from this graph by graph-based optimization.

the gap filling segments (green dash segments in Fig.4d) length along the closed boundary B, $\oint_B ds$ is the perimeter of boundary B. Stahl and Wang [52] proposed to replace the denominator of the ratio in RC with the area of the region enclosed by the closed boundary of the target $\Gamma = \frac{|B_g|}{\iint_{R(B)} dx\, dy}$, where the denominator $\iint_{R(B)} dx\, dy$ is the area of region $R(B)$ which is enclosed by the boundary B. Salient closed boundary tracking actually can be formulated as a prior shape constrained perceptual grouping problem. Elder *et al.* [15] combined the prior probabilistic knowledge of the target appearance with probabilistic models to improve contour grouping. Schoenemann and Cremers [50] first defined the prior shape constraint using terms, such as penalties for shape stretching and shrinking, the tangent angles and the pixel gradients of curves. They then formulated a pixel-wise elastic shape matching and grouping model based on the defined prior shape constraints to track the target boundary.

In my previous works, I adapted the grouping techniques [52] to track a salient closed boundary B using both detected line segments [42] and edge fragments [43]. Particularly, I utilized the similar grouping

cost as $\Gamma = \frac{|B_g|}{\iint_{R(B)} dx\, dy}$ in [52] and added area and perimeter variation constraints to the grouping process. Besides, I developed a novel graph based search method "Bi-Directional Shortest Path (BDSP)" based on Dijkstra [12] to find the optimal closed boundary. This method achieved real-time speed and state-of-the-art performance in terms of accuracy and robustness on our newly created dataset. Currently, few tracking works are grouping-based. However, this is a promising solution for tracking targets with low textures.

3 Proposed Work

In this section, the objectives of the proposed research are first illustrated. Then, the methodologies, challenges and possible solutions are introduced. Finally, the evaluation methods and milestones are discussed, followed by the expected contributions.

3.1 Objectives

The purpose of the proposed research is to develop an edge-based object tracking framework. The research mainly contains the following two parts:

- target description and benchmark: As mentioned in section 2, most existing trackers describe targets as square bounding boxes or contour with single closed boundary. Hence, in this research my first goal is to propose a new edge-based representation for accurately describing targets with arbitrary shapes and structures. Then, a standard dataset with manually labeled ground truth will be created to serve as a benchmark for my proposed method and related studies. Additionally, to reduce human workload in image and video labeling, a novel tool for interactive object annotation will be developed.

- A novel edge-based framework will be developed to track rigid and non-rigid targets with arbitrary shapes and transformations. In particular, this tracking framework is expected to achieve high accuracy and robustness, even for extreme cases such as low textures, cluttered backgrounds, illumination changes, occlusions and non-lambertian.

3.2 Challenges and Possible Solutions

I almost finished the main work of the firt part[1] [44]. Currently, I am working on the second part, developing a general framework for edge-based object tracking.

I outline some of the challenges as follows: **(1) How to track with limited image information?** For many low textured targets such as cup rim and cables, it is hard to find stable regional texture that many tracking algorithms require. To achieve robust tracking performance and avoid aperture problems, taking full advantage of these edge information is necessary. **(2) How to define the target motion model?** The proposed tracking framework has to handle both planar and non-planar, rigid and non-rigid targets. Planar and linear transformations such as Affine, Homography, is limited in this framework. **(3) How to handle outlier introduced by cluttered backgrounds?** Compared with tracking using region-based representation, edge-based target tracking are even more likely to be influenced by cluttered backgrounds. Because both inside and outside of the target region may have numbers of outlier edge pixels. **(4) How to handle partial occlusions and recover from complete occlusions?** Partial and complete occlusions are common yet challenge problems in tracking. Novel mechanism in handling partial and complete occlusions plays an important role in improving the robustness of the tracking framework. **(5) How to achieve real-time performance?** Real-world applications often require real-time performance. The tracking speed relies on

[1] `https://webdocs.cs.ualberta.ca/~xuebin/`

multiple factors including the frame size, the number of edge pixels, the algorithm's time complexity, hardware and *etc.*

As mentioned in section 2.1.2 and section 2.3, I have already developed two kinds of edge-based trackers based on registration [45] and grouping [42] [43]. However, they are limited to certain scenarios and not able to overcome all the challenges above. For example, grouping-based trackers [42] [43] work well with low textured targets and non-rigid transformations. But it is easy to fail because of partial occlusions and cluttered backgrounds. Registration-based edge tracker 2.1.2 performances better than grouping-based trackers in handling partial occlusions. However, it is not able to track non-planar targets and non-rigid transformations.

To overcome these challenges, the to-be-developed tracking framework will take following possible solutions into consideration. **(1)** Given an edge template and a new frame, a more robust edge detection method will be developed to extract more robust and holistic edge information from the current frame. Particularly, I will try to combine the conventional local edge detection [7] [54] and learning-based global boundary detection [60] [33] together. Besides, I plan to use direct method other than feature-based method for the tracking, which means every edge pixel of the edge template will be aligned to their corresponding ones on the current frame. These two techniques will guarantee the stable and rich information for tracking. **(2)** Any complicated motions of a target can be approximated by two DOF translations of the target's sub-parts (edge pixels in my research). However, it is difficult to have stable estimations based on too many independent two DOF translations. Therefore, to provide a universal motion model to the planar and non-planar targets, rigid and non-rigid targets, I will define the two DOF motion model (translation) on each edge pixel and impose geometry, topology and graph constraints on their relative motions to estimate the complicated motions. **(3)** To reduce the impacts of outlier edge pixels, I plan to build a shape constrained generative or discriminative model to filter most of those noisy edge pixels. Meanwhile, a new cost function for non-linear registration will be proposed

to overcome small ratio of noisy edge pixels. Specifically, there are mainly two possibilities: (a) design a classifier (could be a neural network [29] or other classifiers) to filter outlier pixels, and then develop another constrained non-linear registration method for tracking, (b) integrate the outlier removal and the non-linear registration into one optimization process. **(4)** To handle occlusions and tracking failure, a new procedure will be developed to recover tracker from tracking failures. First, a criterion for judging the tracking failure will be proposed. Second, a novel detection method will be designed to detect unsuccessfully tracked target from the whole frame. **(5)** To achieve real-time performance, methods of edge pixels sampling will be studied to reduce the computation time while maintaining the tracking robustness and accuracy. Additionally, both CPU-based and GPU-based implementations will be tested. Different implementations will be used in the corresponding scenarios.

3.3 Evaluations and Milestones

To evaluate the performance of my proposed tracking method qualitatively and quantitatively, I plan to conduct experiments on both synthetic datasets and real datasets.

The synthetic datasets will be used to quickly validate my proposed method by simulating the following tracking conditions: targets with different shapes, rigid or no-rigid deformations, cluttered backgrounds, occlusions and *etc.* In particular, given a synthetic target template and the frame size, I plan to generate the video frames by assigning zero (non-edge pixel) or one (edge pixel) to the pixels on the synthetic frames. To evaluate the performance of my proposed method in real-world, real-life video sequences are essential. I plan to extend our previous collected and annotated real-life datasets [2][3] to have more comprehensive evaluations. Currently, these two datasets only have

[2] `https://github.com/NathanUA/SalientClosedBoundaryTrackingDataset`

[3] `https://github.com/NathanUA/Edge_Template_Tracking_Dataset`

single target tracking sequences. Video sequences for multiple targets tracking will also be collected and annotated for the evaluations.

To quantitatively evaluate the accuracy and robustness of my proposed tracking method, I plan to use two commonly used error metrics: alignment error [43] and success rate [51]. The success rate on a sequence is defined as the ratio of the number of frames where the tracking alignment error is less than a threshold of pixels and the total number of frames. Additionally, the speed of our tracking method will be studied with both CPU and GPU implementations. First, the time complexity will be analyzed theoretically. Then, the relationship between the time cost and the number of sampled edge pixels will be studied based on real-world experiments.

The tentative schedule is given as follows:

1. (2018.09 - 2018.10) Design and build the tracking framework and interfaces. The dataset collection will be conducted at the same time.

2. (2018.11 - 2018.12) Develop an edge-based generative or discriminative classifier to recognize foreground and background edge pixels.

3. (2019.01 - 2019.04) Iteratively update and improve motion models and cost functions for pixel-wise accurate registration.

4. (2019.05 - 2019.06) Develop the mechanism for tracking failure recovering, which will be mainly based on novel detection methods.

5. (2019.07 - 2019.08) Conduct the experiments and analyze the results on robustness, accuracy and time complexity.

6. (2019.09 - 2019.12) Thesis drafting and defence.

3.4 Expected Contributions

The contributions of our proposed research are threefold.

(1) In the computer vision community, the proposed project will be complementary to the region-based tracking methods. It can be used to improve the capability of region-based tracking with support to complex yet commonly used objects so that it will further extend the possible applications of visual tracking. It is also possible to be used in semantic segmentation, registration and other related computer vision studies.

(2) In the robotics field, the proposed project will improve the capability of vision guided robot arm and hand manipulation in wider areas and applications by providing highly accurate and robust tracking results. For example, it can be used to help a robot arm to carry out some complicated tasks like rescue or maintenance in post-disaster buildings or constructions where people can not enter. In some assembly lines, the tracker can help improve the efficiency of the robot operating on moving components.

(3) It will also benefit other researches and applications in computer vision and graphics. For example, the detected and tracked accurate geometric primitives can be used in structure from motion or shape from silhouette for improving the accuracy and details of 3D reconstructions. In augmented reality, the robust and accurate tracking results will help to improve the performance of visual odometry. The newly built dataset will enable validation of other tracking methods.

References

[1] T. D. Alter and Ronen Basri. Extracting salient curves from images: An analysis of the saliency network. *International Journal of Computer Vision*, 27(1):51–69, 1998.

[2] Alex M. Andrew. *Multiple View Geometry in Computer Vision*, by richard hartley and andrew zisserman, cambridge university

press, cambridge, 2000, xvi+607 pp., ISBN 0-521-62304-9 (hardback, £60.00). *Robotica*, 19(2):233–236, 2001.

[3] Simon Baker and Iain A. Matthews. Lucas-kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, 2004.

[4] Charles Bibby and Ian D. Reid. Real-time tracking of multiple occluding objects using level sets. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*, pages 1307–1314, 2010.

[5] Gunilla Borgefors. Distance transformations in digital images. *Computer Vision, Graphics, and Image Processing*, 34(3):344–371, 1986.

[6] Gunilla Borgefors. Hierarchical chamfer matching: A parametric edge matching algorithm. *IEEE Transactions on pattern analysis and machine intelligence*, 10(6):849–865, 1988.

[7] John F. Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, 1986.

[8] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

[9] Changhyun Choi and Henrik I. Christensen. 3d textureless object detection and tracking: An edge-based approach. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*, pages 3877–3884, 2012.

[10] Amaury Dame. *A unified direct approach for visual servoing and visual tracking using mutual information*. PhD thesis, Université Rennes 1, 2010.

[11] Martin Danelljan, Fahad Shahbaz Khan, Michael Felsberg, and Joost van de Weijer. Adaptive color attributes for real-time visual tracking. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1090–1097, 2014.

[12] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.

[13] Tom Drummond and Roberto Cipolla. Real-time visual tracking of complex structures. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):932–946, 2002.

[14] Stefan Duffner and Christophe Garcia. Pixeltrack: A fast adaptive algorithm for tracking non-rigid objects. In *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*, pages 2480–2487, 2013.

[15] James H. Elder, Amnon Krupnik, and Leigh A. Johnston. Contour grouping with prior models. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(6):661–674, 2003.

[16] James H. Elder and Steven W. Zucker. Computing contour closure. In *Computer Vision - ECCV'96, 4th European Conference on Computer Vision, Cambridge, UK, April 15-18, 1996, Proceedings, Volume I*, pages 399–412, 1996.

[17] Chaumette F. and Hutchinson S. *Visual Servoing and Visual Tracking*. Springer, Berlin, Heidelberg, 2008.

[18] Christian Forster, Matia Pizzoli, and Davide Scaramuzza. SVO: fast semi-direct monocular visual odometry. In *2014 IEEE International Conference on Robotics and Automation, ICRA 2014, Hong Kong, China, May 31 - June 7, 2014*, pages 15–22, 2014.

[19] Steffen Gauglitz, Tobias Höllerer, and Matthew Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *International Journal of Computer Vision*, 94(3):335–360, 2011.

[20] Martin Godec, Peter M. Roth, and Horst Bischof. Hough-based tracking of non-rigid objects. *Computer Vision and Image Understanding*, 117(10):1245–1256, 2013.

[21] Gregory D Hager and Peter N Belhumeur. Efficient region tracking with parametric models of geometry and illumination. *IEEE transactions on pattern analysis and machine intelligence*, 20(10):1025–1039, 1998.

[22] David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 FPS with deep regression networks. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I*, pages 749–765, 2016.

[23] Andreas Hofhauser, Carsten Steger, and Nassir Navab. Edge-based template matching and tracking for perspectively distorted planar objects. In *Advances in Visual Computing, 4th International Symposium, ISVC 2008, Las Vegas, NV, USA, December 1-3, 2008. Proceedings, Part I*, pages 35–44, 2008.

[24] Andreas Hofhauser, Carsten Steger, and Nassir Navab. Harmonic deformation model for edge based template matching. In *VISAPP 2008: Proceedings of the Third International Conference on Computer Vision Theory and Applications, Funchal, Madeira, Portugal, January 22-25, 2008 - Volume 2*, pages 75–82, 2008.

[25] Stefan Holzer, Stefan Hinterstoisser, Slobodan Ilic, and Nassir Navab. Distance transform templates for object detection and pose estimation. In *2009 IEEE Computer Society Conference on*

Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA, pages 1177–1184, 2009.

[26] Ping Hu, Bing Shuai, Jun Liu, and Gang Wang. Deep level sets for salient object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 540–549, 2017.

[27] Ryan Kennedy, Jean H. Gallier, and Jianbo Shi. Contour cut: Identifying salient contours in images by solving a hermitian eigenvalue problem. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 2065–2072, 2011.

[28] Georg Klein. *Visual tracking for augmented reality*. PhD thesis, In: Siciliano B., Khatib O. (eds) Springer Handbook of Robotics. Springer, Berlin, Heidelberg, 2006.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012.

[30] Manohar Kuse and Shaojie Shen. Robust camera motion estimation using direct edge alignment and sub-gradient method. In *2016 IEEE International Conference on Robotics and Automation, ICRA 2016, Stockholm, Sweden, May 16-21, 2016*, pages 573–579, 2016.

[31] Junghyun Kwon, Hee Seok Lee, Frank C. Park, and Kyoung Mu Lee. A geometric particle filter for template-based visual tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 36(4):625–643, 2014.

[32] Pengpeng Liang, Yifan Wu, and Haibin Ling. Planar object tracking in the wild: A benchmark. *CoRR*, abs/1703.07938, 2017.

[33] Yun Liu, Ming-Ming Cheng, Xiaowei Hu, Kai Wang, and Xiang Bai. Richer convolutional features for edge detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 5872–5881, 2017.

[34] Zhao Liu, Hui Shen, Guiyu Feng, and Dewen Hu. Tracking objects using shape context matching. *Neurocomputing*, 83:47–55, 2012.

[35] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[36] Shyjan Mahamud, Lance R. Williams, Karvel K. Thornber, and Kanglin Xu. Segmentation of multiple salient closed contours from real images. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(4):433–444, 2003.

[37] Abdol-Reza Mansouri. Region tracking via level set pdes without motion computation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(7):947–961, 2002.

[38] Yansheng Ming, Hongdong Li, and Xuming He. Connected contours: A new contour completion model that respects the closure effect. In *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*, pages 829–836, 2012.

[39] Vida Movahedi and James H. Elder. Combining local and global cues for closed contour extraction. In *British Machine Vision Conference, BMVC 2013, Bristol, UK, September 9-13, 2013*, 2013.

[40] Stanley Osher and James A Sethian. Fronts propagating with curvature-dependent speed: algorithms based on hamilton-jacobi formulations. *Journal of computational physics*, 79(1):12–49, 1988.

[41] Muriel Pressigout and Éric Marchand. Real time planar structure tracking for visual servoing: a contour and texture approach. In *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, Edmonton, Alberta, Canada, August 2-6, 2005*, pages 251–256, 2005.

[42] Xuebin Qin, Shida He, Camilo Perez Quintero, Abhineet Singh, Masood Dehghan, and Martin Jägersand. Real-time salient closed boundary tracking via line segments perceptual grouping. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pages 4284–4289, 2017.

[43] Xuebin Qin, Shida He, Zichen Zhang, Masood Dehghan, and Martin Jagersand. Real-time salient closed boundary tracking using perceptual grouping and shape priors. *28th British Machine Vision Conference, BMVC, London, UK, September 4-7*, 2017.

[44] Xuebin Qin, Shida He, Zichen Zhang, Masood Dehghan, and Martin Jagersand. Bylabel: A boundary based semi-automatic image annotation tool. In *Applications of Computer Vision (WACV), 2018 IEEE Winter Conference on*, pages 1804–1813. IEEE, 2018.

[45] Xuebin Qin, Shida He, Zichen Zhang, Masood Dehghan, Jun Jin, and Martin Jagersand. Real-time edge template tracking via homography estimation. In *submitted to 2018IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2018, Madrid, Spain, October 1-4, 2018*.

[46] Yogesh Rathi, Namrata Vaswani, Allen Tannenbaum, and Anthony J. Yezzi. Tracking deforming objects using particle filtering for geometric active contours. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(8):1470–1475, 2007.

[47] Xiaofeng Ren, Charless C. Fowlkes, and Jitendra Malik. Scale-invariant contour completion using conditional random fields. In

10th IEEE International Conference on Computer Vision (ICCV 2005), 17-20 October 2005, Beijing, China, pages 1214–1221, 2005.

[48] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, 23(3):309–314, 2004.

[49] Glauco Garcia Scandaroli, Maxime Meilland, and Rogério Richa. Improving ncc-based direct visual tracking. In *European conference on Computer Vision*, pages 442–455, 2012.

[50] Thomas Schoenemann and Daniel Cremers. A combinatorial solution for model-based image segmentation and real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(7):1153–1164, 2010.

[51] Abhineet Singh and Martin Jägersand. Modular tracking framework: A fast library for high precision tracking. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017, Vancouver, BC, Canada, September 24-28, 2017*, pages 3785–3790, 2017.

[52] Joachim S. Stahl and Song Wang. Edge grouping combining boundary and region information. *IEEE Trans. Image Processing*, 16(10):2590–2606, 2007.

[53] Xin Sun, Hongxun Yao, Shengping Zhang, and Dong Li. Non-rigid object contour tracking via a novel supervised level set model. *IEEE Trans. Image Processing*, 24(11):3386–3399, 2015.

[54] Cihan Topal, Cuneyt Akinlar, and Yakup Genc. Edge drawing: A heuristic approach to robust real-time edge detection. In *20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23-26 August 2010*, pages 2424–2427, 2010.

[55] Song Wang, Toshiro Kubota, Jeffrey Mark Siskind, and Jun Wang. Salient closed boundary extraction with ratio contour. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(4):546–561, 2005.

[56] Xin Wang, Wei Dong, Mingcai Zhou, Renju Li, and Hongbin Zha. Edge enhanced direct visual odometry. In *Proceedings of the British Machine Vision Conference 2016, BMVC 2016, York, UK, September 19-22, 2016*, 2016.

[57] Max Wertheimer. Untersuchungen zur lehre von der gestalt. *Psychological Research*, 1(1):47–58, 1922.

[58] Tao Wu, Xiaoqing Ding, Shengjin Wang, and Kongqiao Wang. Video object tracking using improved chamfer matching and condensation particle filter. In *Image Processing: Machine Vision Applications*, volume 6813, page 681304. International Society for Optics and Photonics, 2008.

[59] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang. Object tracking benchmark. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(9):1834–1848, 2015.

[60] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. *International Journal of Computer Vision*, 125(1-3):3–18, 2017.

[61] Alper Yilmaz, Omar Javed, and Mubarak Shah. Object tracking: A survey. *ACM Comput. Surv.*, 38(4):13, 2006.

[62] Alper Yilmaz, Xin Li, and Mubarak Shah. Contour-based object tracking with occlusion handling in video acquired using mobile cameras. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(11):1531–1536, 2004.

[63] Guopu Zhu, Qingshuang Zeng, and Changhong Wang. Efficient edge-based object tracking. *Pattern Recognition*, 39(11):2223–2226, 2006.