MVA2015 IAPR International Conference on Machine Vision Applications, May 18-22, 2015, Tokyo, JAPAN

Manufactured Object Sub-Segmentation based on Reflection Motion Estimation

Qinglin Lu, Olivier Laligant, Eric Fauvet
University of Burgundy, Le2i UMR 6306 CNRS
12,Rue de la Fonderie,71200,France
qinglin.lu@u-bourgogne.fr

Anastasia Zakharova
INSA Rouen LMI EA3226
Avenue de l'Universite,76800,France
anastasia.zakharova@insa-rouen.fr

Abstract

In computer vision, reflection is a long-standing problem, it covers image textures, makes original color difficult to recognize, complicates the understanding of the scene. Most of the time, it is considered as "noise". Many methods are proposed in order to reduce or delete the reflection effects in the image, but generally, the performances are not quite satisfactory. While instead of working on "de-noising", we propose a method to take advantage of moving reflections that can be used for different computer vision applications. For instance, the segmentation of reflective manufactured objects is presented in this paper. We focus on tracking reflection components and segmenting continuous surfaces of the object by taking motion information into account. In this paper, a comparison with several conventional segmentation methods is proposed to illustrate how our approach significantly improves the segmentation results.

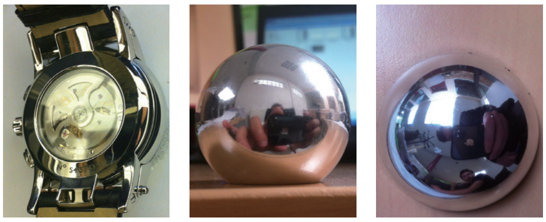
1 Introduction

The images of reflective objects always contain unwished reflections. Those reflections cover object's original information as texture, color, and contour. They also bring extra information that corresponds to the environment into the image of the object. Referring to figure 1, we can see that in (a), reference number of the watch is covered by reflection which makes the components of the watch difficult to distinguish. In (b), the object is showing the reflection of environment on the surface that is not exist on real object. In (c), reflections make scene complicate, even difficult to recognize the object shape. Within the difficulty that reflections bring us, this paper addresses to segment manufacture reflective object into non-continuous surfaces, which we call *object sub-segmentation*.

Various works have been done in dealing with reflections in the image. Most of the works consider reflection as noise, they try to reduce or remove it. Also, a few works attempted to use information contained in the reflection to extract object features.

In the group of removing reflections, Guo and Chao [3] used multiple images acquired from different points of view to find the correspondent reflection and then removed the reflection on the object. D'zmura and Lennie [4] separated reflection components from gray images by a polarizing filter. Tan and Ikeuchi [6, 7] separate reflection components using a single image by comparing iteratively the intensity logarithmic difference between an input image and normalized specular-free image.

In the other group of taking reflection information as an advantage, Savarese and Perona [8, 9] proposed an



(a) (b) (c)

Figure 1: reflective objects. (a) reference number on the object invisible, (b) textures of environment on the object, (c) hard to distinguish the shape of the object.

analysis of the relationship between a calibrated scene composed of lines through a point, and the geometry of a curved mirror surface on which the scene is reflected. This analysis is used to measure object surface profile. DelPozo and Savarese [10] use static specular flows features to detect specular surfaces on natural image. Barrois and Wohler [11] present a method which incorporates different channels of information, one of which is a polarization angle of light reflected from the object surface that provides information on the rotation of an object relative to the camera.

Our proposed method is taking the advantage of reflection. Comparing to all methods in this group, we do not use any prior known information as calibrated camera, textured environment or position of light source. The positions of camera and object are fixed, the light source is moving in this environment. While the reflection of the source is moving on the object surfaces, we extract the direction, the position and the velocity of the reflection in each frame as reflection motion features, then we use these features to match and track each reflection component in the whole sequences. We assume that the reflection is moving smoothly along a continuous surface and irregularly while passing one surface to another. Tracking is stopped while motion features are irregularly comparing to that in previous frames. It guarantees that the trajectory of reflection motion stay on a continuous surface. Then we employ flood fill method for every position in one trajectory to segment one continuous surface on the object. It iteratively covers every trajectory to fill all the surfaces with different labeling colors. Finally, the object is sub-segmented by highlighting different continuous surfaces. Comparing to other segmentation methods, none of them is investigating into reflective object sub-segmentation. While in fact, for a reflective object, sub-segmention is representing the object surface structure and it brings a huge advantage for object recognition.

The rest of the paper is organized as follow. In section 2, we preset the reflection motion features extraction and reflections matching and tracking. In section 3, segmentation is adapted based on the reflection motion trajectory. Results of our approach and a comparison with other segmentation methods are shown in section 4. Conclusion and future works are discussed in section 5.

2 Reflection Motion Estimation

We would like to transfer movement of reflections into useful information that can help us understand how object is composed of different surfaces. According to that, we need to match reflections and track them in different frames, which is a multi-target tracking. The goals of section 2.1, 2.2 and 2.3 is to extract reflection motion features, and use them into reflection matching and tracking.

2.1 Reflection motion features extraction

As the object and the camera are fixed, in the video, movements could only be produced by reflections. We use the motion history image [12, 13] (*MHI*) to extract reflections. The *MHI* $H_\tau(x,y,t)$ can be computed from an update function $\Psi_\tau(x,y,t)$:

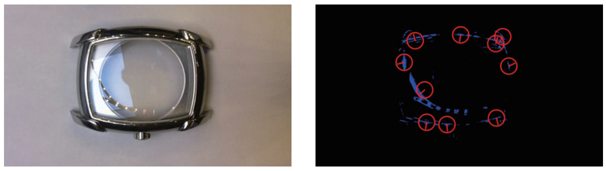
$$H_\tau(x,y,t)=\begin{cases}\tau & if\Psi_\tau(x,y,t)=1\\ max(0,H_\tau(x,y,t-1)-\delta) & if\Psi_\tau(x,y,t)=0\end{cases} \quad (1)$$

Here (x,y,t) in the function are denoted as position and time, $\Psi_\tau(x,y,t)$ denotes whether a motion is exist at position (x,y) in frame t. The duration τ decides the temporal extent of the movement, and δ is the decay parameter. This computation leads to a static scalar valued image where the more recently moving pixels are brighter. Then a direction vector can be efficiently calculated by convolution with separable Sobel filters in the X and Y directions yielding the spatial derivatives. Note that, these direction vectors will point orthogonally to the moving object boundaries. It gives us a normal optical flow representation. Then a downward stepping floodfill is adapted to label motion regions according to the current *MHI*. This computation collects neighbor pixels which has the same motion as a connected component. From the frame at time t, we extract the n moving reflection components as $C_i^t = \{\textit{8-connected pixels of the same motion}\}$ where $i \in [1:n]$. Connected components represent detected moving reflections in the current frame.

From each of the C_i^t, a motion feature vector $f(C_i^t)$ is extracted where $f(C_i^t) = \{d_i^t, p_i^t\}$. Here d_i^t is the direction feature and p_i^t is the position feature. The features extraction is illustrated as follow: d_i^t is obtained by taking the average direction of all the pixels in C_i^t while p_i^t is the center of a bounding box that contains C_i^t. The motion features are used in the next section to match and track each reflection in the image sequences.

2.2 Reflection components matching

Motion features $f(C_i^t)$ are extracted independently from each frame, a matching is adapted to obtain temporal information. The matching of a reference component feature $f(C_i^t)$ and a candidate component feature $f(C_j^{t+\Delta t})$ need to satisfy two following constraints:



(a) (b)

Figure 2: (a) Original frame; (b) motion history image of current frame. Blue pixels represent moving reflections. Red clocks represent moving directions of correspondent reflections.

$$err_p(i,j) = \sqrt{(p_i^t.x - p_j^{t+\Delta t}.x)^2 + (p_i^t.y - p_j^{t+\Delta t}.y)^2} < \delta, \quad (2)$$

$$err_d(i,j) = (d_i^t - d_j^{t+\Delta t})^2 < \alpha. \quad (3)$$

Here equation 2 is the condition of component position and $err_p(i,j)$ is the position difference between C_i^t and $C_j^{t+\Delta t}$. Equation 3 is the condition of moving direction and $err_d(i,j)$ is the difference of direction angle between C_i^t and $C_j^{t+\Delta t}$. Two threshold parameters α and δ are proposed to filter the impossible matching. The matching algorithm is illustrated as follow:

Algorithm 1 Reflection components matching

Input: $f(C_i^t) = \{d_i^t, p_i^t\}$, $f(C_j^{t+\Delta t}) = \{d_j^{t+\Delta t}, p_j^{t+\Delta t}\}$.
Output: $f(C_i^{t+\Delta t})$.

do matching $f(C_i^t)$ and $f(C_j^{t+\Delta t})$ with equation 2,3
if matching is true **then**

1. compute $v_i^{t+\Delta t}$ with equation 4
2. update $f(C_i^t)$ to $f(C_i^{t+\Delta t})$
3. return $f(C_i^{t+\Delta t})$

else

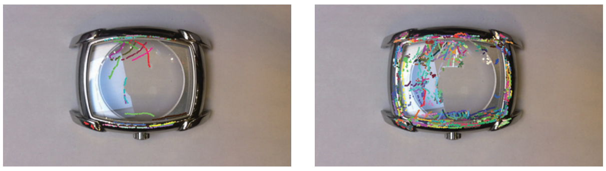
1. $f(C_i^{t+\Delta t}) = f(C_i^t)$
2. return $f(C_i^{t+\Delta t})$

From a pair of matched features, a velocity feature $v_i^{t+\Delta t}$ is computed as follow:

$$v_i^{t+\Delta t} = \frac{\sqrt{(p_i^t.x - p_j^{t+\Delta t}.x)^2 + (p_i^t.y - p_j^{t+\Delta t}.y)^2}}{\Delta t}. \quad (4)$$

Then the reference component feature is updated as $f(C_i^{t+\Delta t}) = \{d_i^{t+\Delta t}, p_i^{t+\Delta t}, v_i^{t+\Delta t}\}$.

If there is no candidate component features can be matched to reference component feature, $f(C_i^{t+\Delta t})$ is updated by the previous reference feature $f(C_i^t)$. In the other case, if there are more than one candidate component features can be matched to $f(C_i^t)$, an



(a) (b)

Figure 3: Reflection motion trajectories. (a) longest ten trajectories, (b) all the trajectories

$Argmin\{err_d(i,j)\}$ will be computed to find the best match.

2.3 Reflection components tracking

The reflection component tracking is composed by iterative matching computation. For tracking a reference component feature $f(C_i^t) = \{d_i^t, p_i^t, v_i^t\}$ at time $t+\Delta t$, a circle window is presented to search the candidate component features. The center of window is the prediction position of $f(C_i^t)$, denoted as $p_i^{t+\Delta t}(pred)$. The computation of $p_i^{t+\Delta t}(pred)$ is presented as:

$$p_i^{t+\Delta t}(pred) = \left\{v_i^t \times \Delta t \times \cos d_i^t + p_i^t.x,\, v_i^t \times \Delta t \times \sin d_i^t + p_i^t.y\right\}. \tag{5}$$

The radius of the window is the threshold parameter δ in the position constraint of matching in equation 2. All the reflection components in the window are considered as candidate components. Then we match the features of these candidate components with $f(C_i^t)$, the minimum $err_d(i,j)$ of all the matches will find the tracking result $f(C_i^{t+\Delta t})$ in current frame. Then a new velocity feature $v_i^{t+\Delta t}$ is calculated by equation 4 and $f(C_i^t)$ will be updated with $f(C_i^{t+\Delta t})$ and $v_i^{t+\Delta t}$.

If no candidate is found by the window in current frame, this frame will be considered as $missingframe$ and the tracking of $f(C_i^t)$ will be continue in next frames. When the number of continuous missing frames is more than 10, the tracking of $f(C_i^t)$ will be stopped and turn to track $f(C_{i+1}^t)$.

During tracking reflection components in frames, positions of all the tracking result are saved as the motion trajectory of reference component. One trajectory is considered as one label for a continuous surface on the object. As the reflection could go through one surface in different directions, we save trajectories separately for each direction. In this case, it guarantees that one trajectory labels only one surface. While on the other hand, some trajectories label a same surface. As shown in figure 3, one color presents one trajectory of moving reflection. Image (a) contains 10 longest trajectories, image (b) contains all the trajectories. In the next section, an iterative surfaces segmentation computation will be employed on the object based on these trajectories which solves perfectly the problem of multi-labelled surfaces.

3 Continuous Surfaces Segmentation

A continuous surface is defined as a surface which has homogeneous curvature profile along the whole surface. As some trajectories are labeling the same surface, an iterative flood fill function is applied to merge segmentation results of different trajectories on the same surface. The seeds are positions in the trajectory. As positions are interspersed among one surface with a continuous trajectory, flood fills all these seeds with a same labeling color votes for one continuous surface. Algorithm of the segmentation process is illustrated as follow:

Algorithm 2 Segmentation process

1. Segmentation
   (a) Update labeling color to the color of current trajectory
   (b) Flood fill all positions with labeling color
   (c) Remove current trajectory from list
2. Morphology components regrouping
   (a) Update labeling color
   (b) Regroup and fill all the components passed by current trajectory with labeling color
   (c) Remove current trajectory from list
3. Final processing
   (a) Fill holes which are surrounded by segmented components with the surrounding color

In the segmentation process, the parts of one surface which has already been labeled by seeds in previous trajectories could be merged into other parts of this surface by the labeling of seeds in latter trajectories. The final processing step fills the holes with the surrounding color to obtain a smooth segmentation result.

4 Results and Comparison

As our approach is supposed to be an industry application. The experiments were conducted in using 5Mpx camera integrated on mobile phone. Resolution of the acquired videos is $720p$. A LED grow light for objects experimented indoor and two light projectors for the outdoor experiment on the car are used to produce reflections on the object.

As the considered objects are reflective and/or transparent, the images contain many high-variability regions. Two of the comparison segmentation methods are graph based method [1]. One of them is based on k nearest neighbors and the other one is based on adjacent. The graph-based methods are chosen since they have the ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions. The third comparison method is EM segmentation [2]. It is a pixel clustering method in a joint feature space. It segments the image with the information from different aspects (color-texture-position). Many objects have been experimented, three of them are shown in the figure 4. Due to the similarity of the two graph-based results and the lack of space, only KNN graph-based results are illustrated in figure 4. From the results, we can see that graph-based methods work reasonably in segmenting the object, but

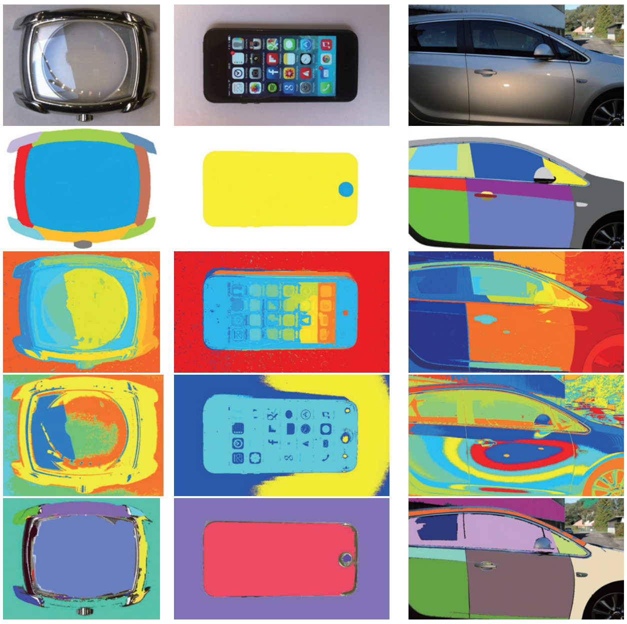


Figure 4: First row: original images. Second row: ground-truth segmentation. Third row: KNN graph-based segmentation. Forth row: EM segmentation. Last row: Segmentation by our proposed method.

F_{score}	Watch	Phone	Car
KNN graph [1]	0.7409	0.5059	0.7322
ajdacent graph [1]	0.7464	0.4754	0.6558
EM [2]	0.4702	0.7921	0.5403
our method	**0.8355**	**0.9124**	**0.8592**

Table 1: Best F_{score} of the objects.

about the sub-segmentation of the object surfaces, it does not work meaningfully. EM segmentation preserves very well the contour of the objects but also the contour of the reflection that yields the poor sub-segmentation performance. Conspicuously, the results obtained by our proposed method are very accurate. Our method is able to sub-segment correctly the object surfaces through all the experiments. In consequences of a high sub-segmentation performance, the object surface structure is well presented.

4.1 Quantitative evaluation

To evaluate the performance of our approach against the segmentation methods in literature (KNN and adjacent graph-based, EM), we employ the f_{score} (FS) as an criterion for each object. FS is a the harmonic mean of precision and recall which globally evaluates the segmentation performance.

In table 1, for the experiment of object 'Phone', pixel values vary a lot but not rendezvous in small regions because that the surface structure is less complicated. Thus, the final processing of our method can fill the holes and yield the value of FS to 0.9124. For the other two experiments, we also obtain the FS which are more than 0.83. We would like to emphasize that, in dealing with reflective and transparent objects, our method outperforms significantly (at least 9% higher) the state-of-the-art methods. However, the proposed method takes advantage of temporal information while the others use static data.

5 Conclusion

We have presented a segmentation method based on reflection motion features to deal with reflective and transparent objects. It has very little constraints which means it can be widely used in the industry for object recognition. The significant results showed that this feature can be used as a robust signature for labeling continuous surfaces on reflective and transparent objects. Instead of removing reflections, taking its advantage is pioneering a new direction. Within the movement of reflections on the object surface, the shapes of reflections are changing. This additional information which can be added into reflection motion features is the subject of our future investigation.

6 Acknowledgment

This work is supported by the company *ALITHEON* and french national research and technology association (*ANRT*).

References

[1] PF. FELZENSZAWALB, *et al.*: "Efficient Graph-Based Image Segmentation," *IJCV*, vol.59, no.2, pp.167-181, 2004.

[2] C. CARSON, *et al.*: "Blobworld: image segmentation using expectation maximization and its application to image querying," *PAMI*, vol.24, no.8, pp.1026-1038, 2002.

[3] X. GUO and X. CHAO: "Robust Separation of Reflection from Multiple Images," *CVPR*, 2014.

[4] M. D'ZMURA and P. LENNIE: "Mechanism of color constancy," *JOSA*, vol.3, no.10, pp.1162-1172, 1986.

[5] Y. LI and MS. BROWN: "Exploiting Reflection Change for Automatic Reflection Removal," *ICCV*, 2003.

[6] R.T. TAN and K. IKEUCHI: "Separating Reflection Components of Textured Surfaces using a Single Image," *PAMI*, vol.25, no.2, pp.178-193, 2005.

[7] R.T. TAN and K. IKEUCHI: "Reflection components decomposition of textured surfaces using linear basis functions," *CVPR*, 2005.

[8] S. SAVARESE and P. PERONA: "Local Analysis for 3D Reconstruction of specular surfaces," *CVPR*, 2001.

[9] S. SAVARESE and P. PERONA: "Local Analysis for 3D Reconstruction of specular surfacesPart ii," *ECCV*,2002.

[10] A. DELPOZO and S. SAVARESE: "Detecting Specular Surfaces on Natural Images," *CVPR*, 2007.

[11] B. BARROIS and C. WOHLER: "3D Pose Estimation based on Multiple Monocular Cues," *BenCOS*, pp.1-7, 2007.

[12] J.DAVIS: "Hierarchical Motion History Images for Recognizing Human Motion," *IEEE workshop DREV*, 2001.

[13] MAR. AHAD *et al.*: "Motion History image: Its variants and applications," *MVA*, 2012.