Multi-cue Based Tracking

Qi Wang[a,b], Jianwu Fang[a], Yuan Yuan[a,*]

[a] *Center for OPTical IMagery Analysis and Learning (OPTIMAL), State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, Shaanxi, PR China*
[b] *School of Electronic and Control Engineering, Chang'an University, Xi'an 710064, Shaanxi, PR China*

Abstract

Visual tracking is a central topic in computer vision. However, the accurate localization of target object in extreme conditions (such as occlusion, scaling, illumination change, and shape transformation) still remains a challenge. In this paper, we explore utilizing multi-cue information to ensure a robust tracking. Optical flow, color and depth clues are simultaneously incorporated in our framework. The optical flow can get a rough estimation of the target location. Then the part-based structure is adopted to establish the precise position, combining both color and depth statistics. In order to validate the robustness of the proposed method, we take four video sequences of different demanding situations and compare our method with five competitive ones representing state of the arts. Experiments prove the effectiveness of the proposed method.

Keywords: Computer vision, Kinect, optical flow, depth, tracking

1. Introduction

Visual tracking is the central topic in computer vision. The aim of this operation is to identify the examined target in consecutive video frames consistently. To achieve this goal, the target object is usually labeled in the first frame by hand. Then its size and location are automatically determined

*Corresponding author.
Email addresses: crabwq@opt.ac.cn (Qi Wang), fangjianwu@opt.ac.cn (Jianwu Fang), yuany@opt.ac.cn (Yuan Yuan)

in the following frames according to the initially labeled property. Since tracking is widely used in applications such as motion-based recognition, automated surveillance, vehicle navigation, human-computer interaction, and video content analysis, a great deal of efforts have been spent to develop various tracking algorithms [1][2][3][4].

Generally, approaches for tracking can be classified into three categories, *appearance* based, *motion* based and a *combination* of them. 1) For appearance based methods, the object is firstly described by the statistics of a predefined target template. This prior information could be global histograms or local keypoint descriptors. Then the tracker searches for a candidate target that is most similar to the template. Popular tracking strategies include point/region matching (SURF tracking [5][6], SIFT tracking [7], part-based tracking [8], sparse representation [9]), kernel tracking (mean-shift tracker [10], eigentracker [11]), and classifier-based tracking (MIL tracker [12], TLD tracker [13], SVM tracker [14]). 2) For motion based methods, the target movement is estimated in the first place. Then tracking is conducted according to the motion field. Typical example is the optical flow based tracking [15], where a dense velocity field is calculated from adjacent frames. Another example focuses on methodologies tailored for tracking specific objects, mostly humans [16]. In this case, human kinematic motions, such as jogging, running and stretching, are modeled particularly and they cannot be extended to other situations. 3) Though the tracking algorithms are classified into the above two categories, there are still a number of methods that do not correspond to any single prototype, but a combination of them [17][18][19]. These techniques consider the appearance and motion simultaneously and the tracking performance is much more promising.

However, the abundance of emerging tracking algorithms does not mean that this field has achieved perfect success. When problems of occlusion, illumination change and viewpoint variation occur, the accurate tracking in real applications still remains a challenge. This is because the appearance and motion properties at such conditions are different from their corresponding templates, which will cause the difficulty of between-frame association and lead to the drifting problem. Typical examples of these situations are illustrated in Fig. 1. Actually, these exceptional conditions might not be a problem for our human vision system. But the computer is incapable of such tasks. The reason, we think, derives from two aspects. 1) Firstly, the designed algorithms have no comparable learning ability with humans. Though popular machine learning techniques [22] demonstrate certain level of gen-

Figure 1: Typical examples of challenging video sequences from [8][20][21]. There are occlusion, illumination change, scale variation, and rotation in these sequences, which make the tracking task difficult.

eralizing and incremental ability, they are still limited. 2) Secondly, not all useful clues are properly utilized for processing [23][24]. This is similarly indicated from human visual mechanism that people make decisions based on multiple clues, such as color, texture, motion, depth and other prior information. Nevertheless, most algorithms employ not so many clues, which leads to a confused tracking result when the environment changes. For example, when a person walks on the street, the lighting condition may change from one place to another. If only color appearance is considered, the tracker cannot work efficiently. But once the motion or depth continuity is involved, the task becomes easier.

Based on the above considerations, we propose a tracking method based on multiple cues combination (TMC) in this paper. Optical flow, color and depth information are involved simultaneously. Our assumption is that different features can provide complementary supporting information. When a feature fails to track the target, the other features might act as supplemental evidences; or these features may enhance their individual effect together. The general idea of our method is as follows. In the beginning, the target object is manually labeled by a surrounding rectangle. The obtained template is used to determine the promising candidate target in the following frames. Then the optical flow field is calculated based on the two adjacent frames. The obtained displacement for each pixel can provide an estimation of its corresponding position in the next frame. After that, the target candidate is searched in the neighborhood of the estimated location. For every possible location, its appearance statistics is compared with that of the initially labeled template by a part-based model. The depth continuity is also considered in this process to make the result robust to noises from occlusion and illumination.

The main contributions of this work are as follows:

1) Depth maps from Kinect sensor is utilized as a valuable clue for tracking objects. Though depth maps have been applied in existing applications as introduced in Section 2, most of them are based on stereo rig. This makes the speed less efficient because stereo algorithms usually employ an optimization process. After the introduction of Microsoft Kinect sensor, the situation has changed because the depth maps can be obtained in real time. But most related works based on Kinect are focused on the tracking of particular objects, such as hands, face, etc. As for the general tracking problem, few works have been reported. Based on this consideration, we present a tracking method for general objects. No specific prior information is included for the tracking

procedure, which makes the proposed tracker with wider application scope.

2) Part-based model is rephrased in the context of depth information. For tracking problem, part-based method has been recognized for its ability to restrain from occlusion. But this ability still has its limitation when confronting with challenging video sequences. In this work, we propose to employ depth information in the part-based method, together with traditional color statistics. To the best of the authors' knowledge, this is the first time to extent the part-based tracking in this way.

3) Several video sequences are recorded and labeled as the benchmark ground truth for depth-based tracking. Though there are data sets publicly available for tracking research [8][20][21], they only have the traditional RGB channels. On the contrary, the constructed data set in this work have both RGB and depth information for each frame.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the proposed multi-cue tracking model. Section 4 conducts extensive experiments to evaluate the presented method, based on the four video sequences taken by ourselves. In the end, conclusion is made in Section 5.

2. Related Work

Since this work is mainly focused on the incorporation of depth information in the tracking process, we restrict the literature review on the depth related scope. This type of methods can be categorized into two situations. The first one explores a mapping from the 3D world to the 2D image plane, instead of directly calculating depth maps in the tracking process. The second one explicitly recovers the depth clue from image sequences or multi-view geometry.

Many examples abound in the first situation. Michel *et al.* [25] presented a monocular model-based 3D tracking approach. They first estimate the camera projection matrix through a calibration procedure. Then the world coordinates of 3D object are connected with its projective camera coordinates and image coordinates. By matching the predefined object model with extracted edges and nodes from current frame, the pose of the tracked object relative to the camera is recursively updated. The work of hand tracking by Stenger *et al.* [26] also followed this prototype, which used an Unscented Kalman Filter (UKF) to minimize the geometric error between the predefined profiles and edges extracted from the images. Li *et al.* [27] schemed

tracking by combining the tracking results from different viewpoints. A discrete relaxation algorithm is employed in this process to reduce the intrinsic combinatorial complexity and unreliable prior information from independent 2D-tracking is pruned by the decision tree. Tu *et al.* [28] proposed an online sequential pose estimation technique for tracking human arms. They utilize the structure-from-motion to provide the 3D arm posture hypotheses of multiple importance sampling by particle filter. Li *et al.* [29] learned a mixture of factor analyzers during off-line training, which can be considered as a local dimensionality reducer that approximates the pose manifold. Then for the online tracking of human motion, the clusters of factor analyzers are utilized in a multiple hypothesis tracking algorithm.

As for the second situation, Tyagi *et al.* [30] extended the 2D mean-shift kernel tracking to 3D by fusing appearance features from all available camera views. They generate 3D point clouds in the scene and re-initialize the tracker itself when necessary. This forms an automatic tracking framework. Ess *et al.* [31] addressed the multiperson tracking problem by a stereo rig mounted on a mobile platform. The interplay between the camera position, stereo depth, object detection and tracking is represented by a graphical model. Göktürk and Tomasi [32] described a head-tracking algorithm based on a sequence of 3D depth images generated from a time-of-flight depth sensor. The tracking process combines recognition and depth sensing. Frati and Prattichizzo [33] combined wearable haptic devices with Kinect depth sensor to develop a hand tracker. Oikonomidis *et al.* [34] treated the tracking of hand articulations as an optimization problem. They seek a result that minimizes the discrepancy between the hypothesized hand model and the actual observation from Kinect sensor. Hu *et al.* [35] tracked the pose of walker user's lower limb with Kinect fixed on the bottom of a walker. A probabilistic approach of particle filtering is used to estimate the the most possible pose.

3. Multi-cue tracking model

In this Section, the proposed tracking model is introduced in detail. We first describe the Kinect sensor employed for generating depth maps and video sequences. Then the optical flow based estimation is derived to get the promising location of the target in the next frame. After that, the part-based appearance matching is presented to establish the accurate target position.

Figure 2: (a) Kinect sensor. (b) The three components of Kinect sensor. This figure is cited from [36].

3.1. Kinect Sensor

Kinect was released by Microsoft on November 2010 to serve as a motion sensing input device. It aims to change the way people playing games and experiencing entertainment. But the consequent *Kinect Effect* is not limited to the gaming industry. Its impact also extends to researchers in computer science and electronic engineering, by changing the way of doing research [36]. The success of Kinect is mainly due to its ability to produce depth information in real time.

The Kinect sensor has three components, the color camera, the infrared (IR) projector, and the IR camera. Its structure is illustrated in Fig. 2. The color camera outputs common RGB images. The IR projector and camera

are used to produce depth maps based on the structured light mechanism. To be specific, the IR projector projects a predefined pattern of dots with varying intensity. The variation of these features relative to the known pattern provides a clue for reconstructing depth [35]. The obtained depth map is represented as gray scale image. The darker a pixel, the closer it is to the camera. Besides, black pixels indicate their depth values are unknown. Typical example images captured by Kinect are shown in Fig. 3.

3.2. Calculating Optical Flow

In our processing, the search for a candidate target in the next frame is bounded in a neighborhood area, which is estimated by the optical flow. The employed optical flow method is based on a variational formulation [37][38]. Before deriving the model, three assumptions should be introduced firstly.

- *The pixel intensities in adjacent frames do not change.* This assumption is intuitively understandable because the two neighboring frames are closely connected. Suppose (u, v) is the displacement (defined as optical flow) for pixel (x, y) of image I, from frame t to $t+1$. Then we have

$$I(x+u, y+v, t+1) = I(x, y, t). \tag{1}$$

  A simple deduction will yield

$$I_x u + I_y v + I_t = 0. \tag{2}$$

- *The pixel gradients in adjacent frames do not change.* This assumption is the direct result of the intensity constancy assumption. Besides, it makes the obtained optical flow more robust in real applications. Because, more or less, the intensity variation cannot be avoided actually. But the relative brightness does not change much. This lead to

$$\bigtriangledown I(x+u, y+v, t+1) = \bigtriangledown I(x, y, t), \tag{3}$$

  where $\bigtriangledown = (\partial_x, \partial_y)$ is the spatial gradient.

- *Smoothness constraint.* This assumption supposes the optical flow varies continuously in the spacial image. Abrupt change is not encouraged and is punished. Most computer vision problems have this constraint too, considering the characteristics of neighborhood pixels are similar.

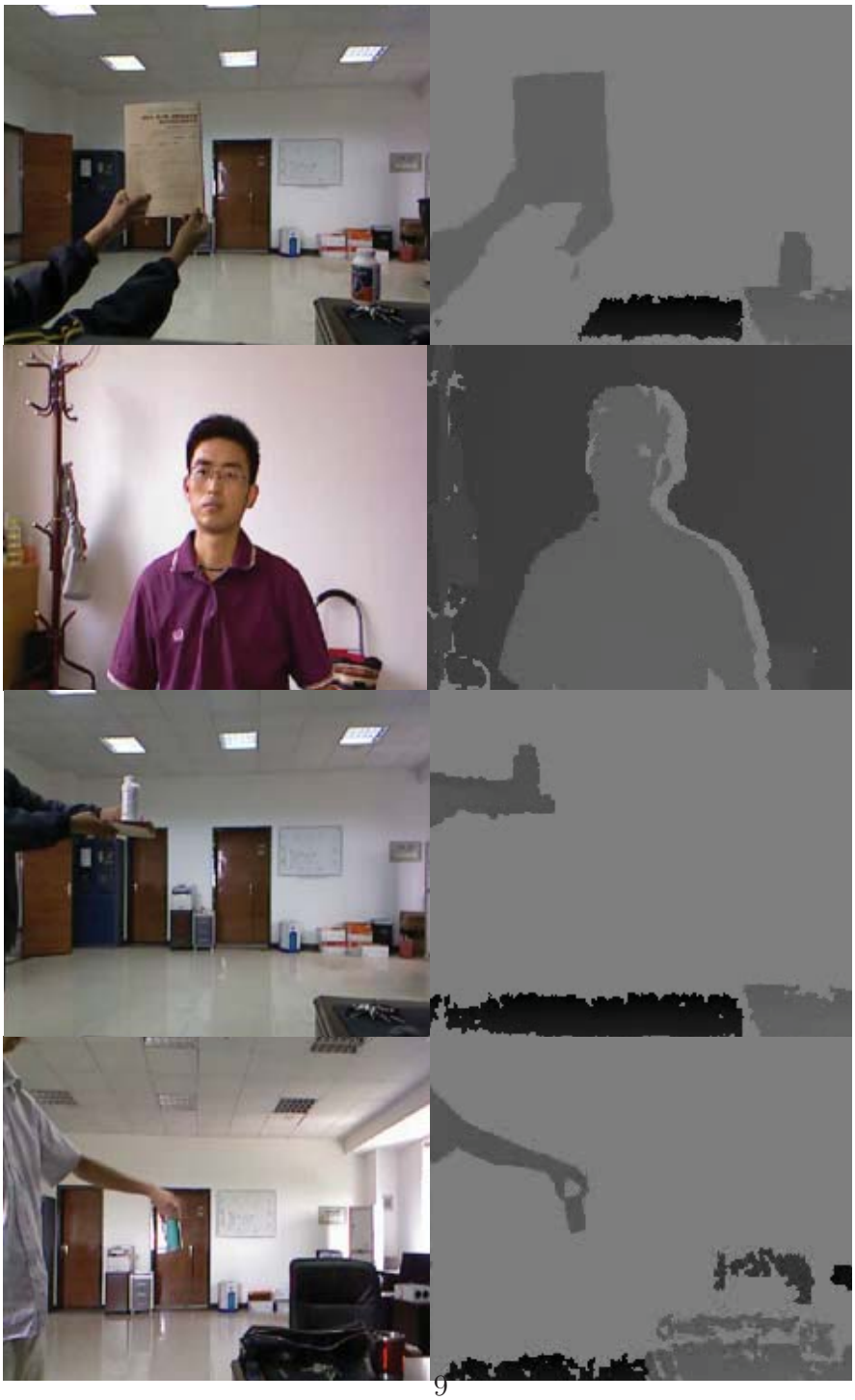

Figure 3: Example images captured by Kinect sensor. They are *Book*, *Face*, *Inno* and *TeaCan*, respectively. Left: color image; Right: depth map.

Based on the above descriptions, an energy function can be derived to embody these assumptions. The energy function is composed of two terms, data term and smoothness term. The data term is defined as

$$E_{data}(u,v) = \int_{\Omega} (|I(x+u, y+v, t+1) - I(x,y,t)|^2 + \lambda| \bigtriangledown I(x+u, y+v, t+1) - \bigtriangledown I(x,y,t)|^2) dxdy, \tag{4}$$

where Ω is the image plane, and λ is the weighting parameter between the intensity and gradient constancy assumptions. The smoothness term is defined as

$$E_{smooth}(u,v) = \int_{\Omega} (| \bigtriangledown_3 u|^2 + | \bigtriangledown_3 v|^2) dxdy, \tag{5}$$

where $\bigtriangledown_3$ is defined as $(\partial_x, \partial_y, \partial_t)$. Finally, the whole energy function is a combination of the data term and smoothness term

$$E(u,v) = E_{data} + \mu E_{smooth}, \tag{6}$$

with μ being the regularization parameter adjusting the balance between E_{data} and E_{smooth}. The optimal u and v that minimize the energy function is the desired optical flow. The solution of this optimization problem is not a trivial task. It can be obtained based on the Euler-Lagrange equation. Details about this topic can be found in [37].

3.3. Estimating New Location

With the acquired optical flow values for every pixel in the image, the new location of the target in the next frame can be estimated. Suppose the center of object O in current frame t is located at (x_t, y_t). Then its position in next frame $t+1$ is calculated as $(x_{t+1}, y_{t+1}) = (x_t + dx, y_t + dy)$, where dx and dy are the displacements estimated by averaging the optical flow within O. To be specific,

$$\begin{aligned} dx &= \sum_{i \in O} OF_x(i)/PixNumInO, \\ dy &= \sum_{i \in O} OF_y(i)/PixNumInO, \end{aligned} \tag{7}$$

where $OF_x(i)$ and $OF_y(i)$ are the optical flow of pixel i in the horizontal and vertical directions respectively, and $PixNumInO$ is the number of pixels within the target object. This procedure is demonstrated in Fig. 4.

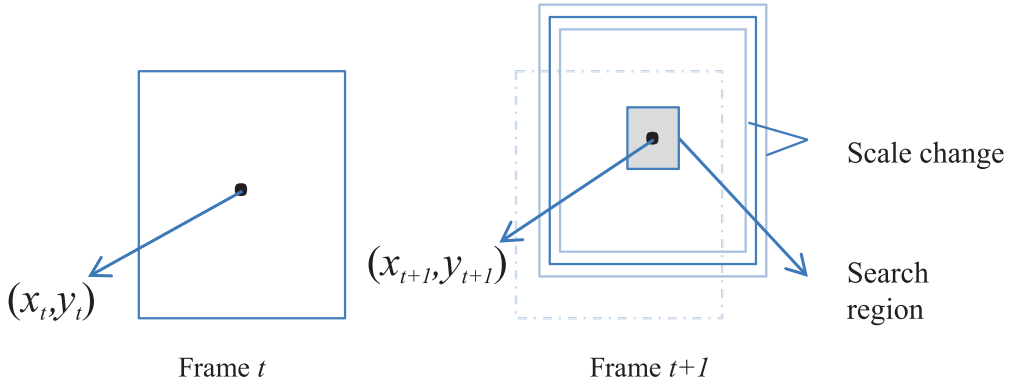

Figure 4: Illustration of the candidate target search. Firstly, the target center (x_t, y_t) in frame t moves to (x_{t+1}, y_{t+1}) in frame $t+1$ according to the optical flow estimation. Then the searching for a more accurate localization is conducted in the gray neighborhood around (x_{t+1}, y_{t+1}). At the same time, scale change of target object is allowed by enlarging or shrinking the rectangle.

The following search then starts from (x_{t+1}, y_{t+1}). The center of O is allowed to shift within a neighborhood region (the gray search region in Fig. 4). To further incorporate the scale changes during tracking, the bounding box of candidate target is allowed with an adaptive enlarging or shrinking. But to ensure a reasonable appearance change between adjacent frames, the scale change is suppressed by a maximum of 10% in the horizontal and vertical directions.

3.4. Part-based Appearance Matching

The optical flow can get a rough estimation of the target location and reduce the searching space. As for the accurate localization, an appearance based matching is required. The region with the minimum distance from the template target is the desired one. In this work, we adopt the part-based model as illustrated in Fig. 5. We do not directly divide the target region into a combination of parts. Instead, we first partition it in different ways (horizontally and vertically) and then treat the acquired four patches (Top, Down, Left, Right) with equal importance. The statistics used for matching is color histogram, depth mean and variance, and pixel number within the target object.

1) For calculating color histograms, the RGB channels are equally divided into 8 intervals, which leads to a total of 8^3 bins. Since the possible target

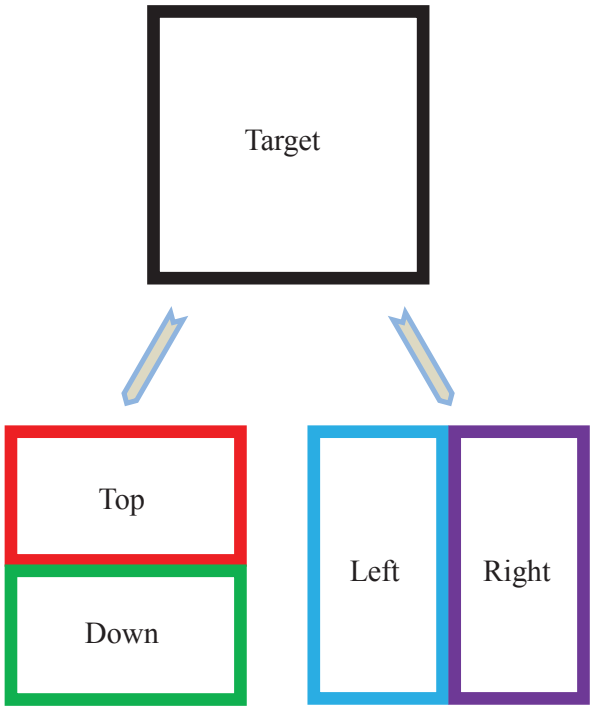

Figure 5: Part-based structure in this work.

region is not a small number, getting their histograms is not a cheap task. For this reason, we employ the *integral histogram* technique [39] which has been adopted by other researchers in tracking to enhance the processing speed. The distance between the candidate target and the template target is then defined as

$$d_c = \sqrt{1 - \rho(P, T)}, \tag{8}$$

where $\rho(P, T)$ is the Bhattacharyya coefficient between the two distributions (histograms) of candidate P and target T. This expression of color distance is proved to be effective by kernel based trackers [10]. It follows as

$$\rho(P, T) = \sum_{n=1}^{N} \sqrt{P_n T_n}, \tag{9}$$

where n is the bin index and $N = 8^3$ is the total bin number. Since the target object is divided into four parts, this distance is indeed calculated separately. In the end, the minimum one is selected as the color distance D_c

$$D_c = min\{d_c^U, d_c^D, d_c^L, d_c^R\}. \tag{10}$$

2) For the depth clue, it is more robust to illumination change than the color clue. We use the depth mean and variance in the target region to measure the difference between the candidate and template. Besides, since the object location does not change much for adjacent frames, we can also suppose a smoothness constraint. That means the depth statistics of next frame is similar with current frame. Mathematically, it follows

$$d_p = \frac{mean(P) - mean(T)}{mean(T)} + \frac{var(P) - var(T)}{var(T)}, \tag{11}$$

where $mean(\cdot)$ represents the mean and $var(\cdot)$ the variance. Similarly, the minimum depth distance is chosen as the representation

$$D_p = min\{d_p^U, d_p^D, d_p^L, d_p^R\}. \tag{12}$$

3) The third parameter for defining appearance is the pixel number within the object region. The difference between pixel numbers of the two target objects in adjacent frames is their distance

$$D_n = \frac{num(P) - num(T)}{num(T)}, \tag{13}$$

where $num(\cdot)$ is the number of pixels.

After all these terms are defined, the final distance function D is a linear combination of the three distances

$$D = \lambda_1 D_c + \lambda_2 D_p + \lambda_3 D_n, \tag{14}$$

where $\lambda_1, \lambda_2, \lambda_3$ are the weighting parameters.

4. Experiments

4.1. Data Set

In this Section, we conduct intensive experiments to evaluate the performance of the proposed method. We first introduce the data set employed here. Though there are many publicly available data sets for evaluating tracking algorithms, they are not suitable for our method. This is because existing data sets contain video sequences of only RGB channels, without depth information. For this reason, we employ the Kinect sensor to take four videos for our purpose. They are *Book*, *Face*, *Inno* and *TeaCan* as illustrated in Fig. 3. These video sequences represent different challenging cases, such as occlusion, rotation, illumination change, shape variation of flexible object, and small target. We manually marked the ground truth every five frames and the obtained results are used for evaluation.

4.2. Comparative Algorithms

In order to gauge the absolute performance, we also compare our method with five competitive ones, representing state of the arts. They are Frag-Track [8], OAB [40], SemiBoost [41], MIL-Track [12], and ASLSAM [42]. Frag-Track is a canonical work for part-based tracking. It demonstrates great robustness to occlusion because of its part structure. OAB is the pioneer of boosting technique applied in tracking and SemiBoost is its development aiming at alleviating drifting problem. MIL-Track uses the multiple instance learning to train the classifier to discriminate target and background. ASLSAM is a sparse based tracking method exploring both partial information and spatial information. The codes for implementing these algorithms are downloaded from the authors' respective homepages.

4.3. Evaluation Metrics

Besides presenting subjective evaluation, objective measurements are also necessary to provide a quantitative comparison. For the objective metrics, two indexes are mostly adopted in the tracking community. The first one is the Center Location Error (CLE), which measures the center distraction of the target rectangle from the ground truth center. The smaller the error, the better the method is. The second one is the precision, which summarizes the percentage of frames within a certain level of distance from the ground truth. Given an accepted threshold bias, the bigger percentage indicates a more robust performance.

4.4. Implementation Details

Some detailed points in our experiments are clarified in this Section. The first problem we should note is that the color image and depth image do not have the same content. There is a displacement of the scene. Take Fig. 3 for example. A close look of the fourth row *TeaCan* image pair will find that the two images do not match exactly. In the color image, half of the human body appears in the image. But in the depth map, only the arm is present. In our experiments, we find this displacement is correlated with the distance between the object and camera sensor. As for the four captured video sequences, their horizontal and vertical shifts are respectively *Book* (10,15), *Face* (5,12), *Inno* (6,13) and *TeaCan* (5,10) in pixels. In fact, there are other reported techniques for Kinect calibration [36]. We didn't utilize them because our simple adjustment suffices for the tracking task and the calibration is not the focus of this work.

Another question should be addressed is the choice of parameters. The weighting parameter μ in Eq. 6 is set as 1 according to [43]. The $\lambda_1, \lambda_2, \lambda_3$ are all set to 1 for video sequences of *Face*, *Inno* and *TeaCan*. As for *Book*, the parameters are set as $\lambda_1 = \lambda_2 = 1, \lambda_3 = 0.2$, because the target object book in this sequence is changing its shape and the pixel number within it varies from frame to frame. Therefore, we put less weight on the influence of pixel number on the final distance function.

Besides, in our implementation, the enlarging and shrinking operations for the target rectangle are allowed in two directions simultaneously or differently. That means the shape change in the horizontal and vertical directions may occur at the same time or only at one direction.

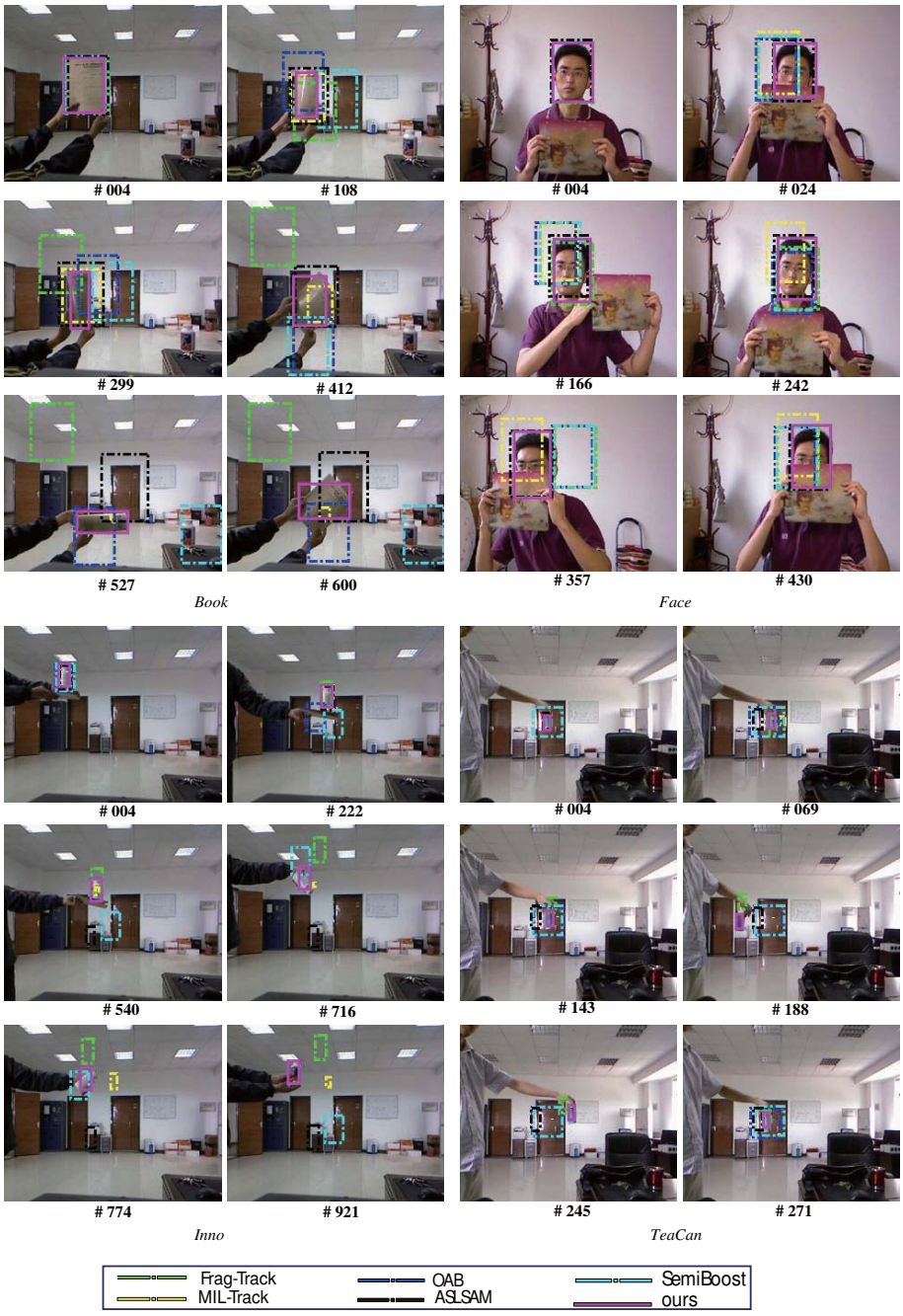

Figure 6: Screenshots of tracking results.

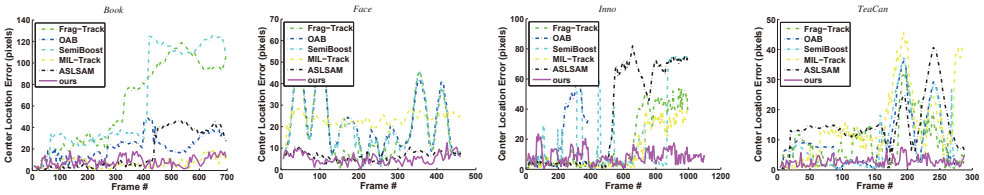

Figure 7: Plots of center location error. Horizontal axis: frame number. Vertical axis: displacements (in pixels) of target center from the ground truth center.

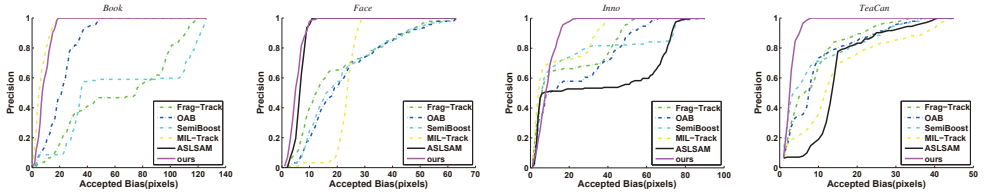

Figure 8: Precision curves. Horizontal axis: threshold bias (in pixels) between the target center and the ground truth center. Vertical axis: percentage of frame numbers with the center location error below the threshold.

4.5. Results

We perform experiments on the four video sequences. The manually labeled template rectangle for each video in the first frame is kept the same for all methods. This can ensure a fair comparison. The captured clip shots are shown in Fig. 6 for subjective evaluation. Objective analysis is displayed in Fig. 7, Fig. 8, and Table 1, where the center location error, precision, and averaged overlap ratio are compared. Detailed discussion is presented below.

Book. There are 700 frames for the *Book* video. In the first 100 frames, the book movements include 2D translation, 3D moving towards and beyond the camera. Then in the remaining frames, the book is subject to rotating, folding and unfolding. In fact, it is a non-rigid object that goes through flexible transformation. Besides, there is also certain level of illumination change because the book cover reflects the light from the ceiling. Therefore, it is a challenging sequence. From Fig. 7, it is clear that our method and MIL-Track have comparable performance of lower errors; both outperform the other four tracking methods. Similar results can be found in Fig. 8. Nevertheless, though the MIL-Track demonstrates a good adaptivity as our method from Fig. 7 and Fig. 8, its actual tracking results are not satisfying

Table 1: Comparison of overlap ratios. The statistics reflect the percentage of frames with an overlap ratio greater than 40%. Higher value indicates a better performance.

Methods	Successful frames (%)			
	Book	*Face*	*Inno*	*TeaCan*
Frag-Track	37.86	66.02	63.56	42.91
OAB	66.43	56.28	19.02	0.0
SemiBoost	9.29	59.52	69.57	0.0
MIL-Track	56.43	27.06	41.54	21.45
ASLSLAM	63.57	99.57	50.05	7.61
ours	**80.71**	**99.57**	**71.57**	**87.20**

because the rectangle box is finally shrank to a small box, which covers only part of the book. This can be seen from frame #527 and #600 in Fig. 6. For the percentage of successful frames with an overlap ratio larger than 40%, our method performs the best among all the competitors. This is clearly shown in Table 1. Considering all these aspects, the proposed method is better than the other five competitive ones on the *Book* video.

Face. For the *Face* sequence, there are 460 frames in together. It is mainly recorded for testing the robustness to occlusion. From the beginning, we use a mouse mat to shade the face from all directions, keeping the head fixed at the initial place. Then the occluded head moves horizontally, together with the mouse mat. Statistics from Table 1 indicate our method is equally well as ASLSAM. But the results in Fig. 7 and Fig. 8 are manifest that the proposed method performs far better than the other five ones. Our method has a more accurate localization no matter how the face is occluded. This can be proved in Fig. 6.

Inno. The *Inno* sequence is a long video containing 1100 frames. The target object is a medicine bottle with different appearances from different viewpoints. One of its sides is completely white with no texts and pictures on the surface. This makes the appearance similar with the background wall. The target movement is also complex. It is taken close to or far beyond the camera, leading to a scale change. It is also moved from all directions in a 2D plane. At the same time, the bottle itself is rotated from all directions, making what it seems varied. Experimental results from Fig. 7 and 8 show that our method does not perform best before frame 400. But this inferior is not so much because the subjective evaluation from Fig. 6 indicates our tracking results are also satisfying. Later in the video

sequence, the target appearance changes rapidly and the background wall interferes with the tracking procedure. In this case, our method can track the target until the last frame but the other five methods all drift away finally. This demonstrates great robustness of the proposed method. The successful frames in Table 1 agree with this conclusion.

TeaCan. The *TeaCan* video sequence is designed to evaluate the tracker's ability to track small object. There are in total 290 frames. The target of tea can occupies a small portion of the image content and it moves in the 3D space from all directions. For this testing video, our method is absolutely superior to the other five ones. The marked rectangle size is no much difference with the actual target size. On the contrary, the other five methods generate tracking rectangles much larger than it should be, or even drifting away. This conclusion is consistent from Fig. 7, Fig. 8 and Table 1.

4.6. Discussion

Based on the above analysis, we can see that the proposed method is effective and robust. It can endure different kinds of transformations and changes, in a certain level. This success is primarily due to the incorporation of depth information. The depth clue can provide a discriminative ability from the background. It is more stable compared with the appearance statistics. Once the color information cannot provide the discriminative clue, the depth information can still act as a supplement. Besides, the optical flow can provide a rough estimation of the target position, which can ensure no drifting problem occurs. All these factors together make the tracker work effectively. On the other hand, the five competitive ones only utilize color information. If there is similar background or illumination change, the trackers will fail most of the time.

To further justify the superiority of combining the color clue with the depth clue, we conduct comparative experiments with each individual information. Then the averaged center location error is calculated to evaluate the performance. Table 2 shows the statistics. We can see clearly that the color and depth along cannot achieve the best performance. Besides, the tracking results are not stable enough. For *Face* and *TeaCan* sequences, the color clue (ourC) performs better. But for *Book* and *Inno* sequences, the depth clue (ourD) is better. Nevertheless, combing them together (ourCD) always has the best performance. This proves that the strategy proposed in this work is reasonably effective.

Table 2: Comparison of color and depth clues for the proposed method. The statistics are averaged center location errors.

Methods	Averaged center location error			
	Book	*Face*	*Inno*	*TeaCan*
Frag-Track	63	19	17	8
OAB	21	22	22	10
SemiBoost	64	22	19	9
MIL-Track	6	23	12	15
ASLSLAM	18	6	34	15
oursC	11	5	39	3
oursD	39	15	11	57
ourCD	**8**	**5**	**8**	**2**

5. Conclusion

In this paper, a multi-cue based tracker is presented. Unlike most other methods that only utilizing color clue, the optical flow, color and depth clues are all incorporated in the tracking process. This provides more supporting information for the determination of tracked objects. To justify its robustness, we took four video sequences representing various challenging situations, such as occlusion, scaling, illumination change, and shape transformation. Each sequence has both RGB channels and depth channel. Experiments compared with five popular trackers representing state of the arts indicate that the proposed method is effective.

In the future, we plan to adaptively change the target template during the tracking procedure, because the target object always changes its shape and appearance. How to model this adaptivity and reduce the accumulated error are the keypoints for a successful tracking method.

Acknowledgment

This work is supported by the National Basic Research Program of China (Youth 973 Program) (Grant No. 2013CB336500), the National Natural Science Foundation of China (Grant No. 61172143, 61379094 and 61105012), the Natural Science Foundation Research Project of Shaanxi Province (Grant No. 2012JM8024), and the Fundamental Research Funds for the Central Universities, Chang'an University (Grant No. 2013G3324005).

References

1. Yang, H., Shao, L., Zheng, F., Wang, L., Song, Z.. Recent advances and trends in visual tracking: A review. *Neurocomputing* 2011; **74**(18):3823–3831.

2. Tracking objects using shape context matching. *Neurocomputing* 2012; **83**:47–55.

3. Wang, Q., Chen, F., Xu, W.. Adaptive multi-cue tracking by online appearance learning. *Neurocomputing* 2011;**74**(6):1035–1045.

4. Yuan, Y., Fang, J., Wang, Q.. Robust superpixel tracking via depth fusion. *IEEE Transactions on Circuits and Systems for Video Technology* 2013;**PP**(99):1–1.

5. He, W., Yamashita, T., Lu, H., Lao, S.. SURF tracking. In: *ICCV*. 2009, p. 1586–1592.

6. Miao, Q., Wang, G., Shi, C., Lin, X., Ruan, Z.. A new framework for on-line object tracking based on surf. *Pattern Recognition Letters* 2011;**32**(13):1564–1571.

7. Zhou, H., Yuan, Y., Shi, C.. Object tracking using SIFT features and mean shift. *Computer Vision and Image Understanding* 2009; **113**(3):345–352.

8. Adam, A., Rivlin, E., Shimshoni, I.. Robust fragments-based tracking using the integral histogram. In: *CVPR*. 2006, p. 798–805.

9. Mei, X., Ling, H.. Robust visual tracking and vehicle classification via sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2011;**33**(11):2259–2272.

10. Comaniciu, D., Meer, P.. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2002;**24**(5):603–619.

11. Black, M.J., Jepson, A.D.. Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. *International Journal of Computer Vision* 1998;**26**(1):63–84.

12. Babenko, B., Yang, M.H., Belongie, S.. Robust object tracking with online multiple instance learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2011;**33**(8):1619–1632.

13. Kalal, Z., Mikolajczyk, K., Matas, J.. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2012; **34**(7):1409–1422.

14. Avidan, S.. Support vector tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2004;**26**(8):1064–1072.

15. Senst, T., Eiselein, V., Sikora, T.. Robust local optical flow for feature tracking. *IEEE Transactions on Circuits and Systems for Video Technology* 2012;**22**(9):1377–1387.

16. Fan, J., Xu, W., Wu, Y., Gong, Y.. Human tracking using convolutional neural networks. *IEEE Transactions on Neural Networks* 2010; **21**(10):1610–1623.

17. Bhandarkar, S.M., Luo, X.. Integrated detection and tracking of multiple faces using particle filtering and optical flow-based elastic matching. *Computer Vision and Image Understanding* 2009;**113**(6):708–725.

18. Shin, J., Kim, S., Kang, S., Lee, S., Paik, J.K., Abidi, B.R., et al. Optical flow-based real-time object tracking using non-prior training active feature model. *Real-Time Imaging* 2005;**11**(3):204–218.

19. Shi, J., Tomasi, C.. Good features to track. In: *CVPR*. 1994, p. 593–600.

20. Ross, D.A., Lim, J., Lin, R.S., Yang, M.H.. Incremental learning for robust visual tracking. *International Journal of Computer Vision* 2008; **77**(1-3):125–141.

21. Birchfield, S.. Elliptical head tracking using intensity gradients and color histograms. In: *CVPR*. 1998, p. 232–237.

22. Wang, Q., Yuan, Y., Yan, P., Li, X.. Saliency detection by multiple-instance learning. *Cybernetics, IEEE Transactions on* 2013;**43**(2):660–672.

23. Wang, Q., Yuan, Y., Yan, P.. Visual saliency by selective contrast. *IEEE Transactions on Circuits and Systems for Video Technology* 2013; **23**(7):1150–1155.

24. Wang, Q., Yan, P., Yuan, Y., Li, X.. Multi-spectral saliency detection. *Pattern Recognition Letters* 2013;**34**(1):34–41.

25. Michel, P., Chestnutt, J.E., Kagami, S., Nishiwaki, K., Kuffner, J.J., Kanade, T.. GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing. In: *IROS*. 2007, p. 463–469.

26. Stenger, B., Mendonça, P.R.S., Cipolla, R.. Model-based 3D tracking of an articulated hand. In: *CVPR*. 2001, p. 310–315.

27. Li, Y., Hilton, A., Illingworth, J.. A relaxation algorithm for real-time multiple view 3d-tracking. *Image Vision Computing* 2002;**20**(12):841–859.

28. Tu, M.H., Huang, C.M., Fu, L.C.. Online 3D tracking of human arms with a single camera. In: *ICRA*. 2012, p. 1378–1383.

29. Li, R., Yang, M.H., Sclaroff, S., Tian, T.P.. Monocular tracking of 3D human motion with a coordinated mixture of factor analyzers. In: *ECCV*. 2006, p. 137–150.

30. Tyagi, A., Keck, M.A., Davis, J.W., Potamianos, G.. Kernel-based 3D tracking. In: *CVPR*. 2007, .

31. Ess, A., Leibe, B., Schindler, K., Gool, L.J.V.. Robust multiperson tracking from a mobile platform. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 2009;**31**(10):1831–1846.

32. Göktürk, S.B., Tomasi, C.. 3D head tracking based on recognition and interpolation using a time-of-flight depth sensor. In: *CVPR*. 2004, p. 211–217.

33. Frati, V., Prattichizzo, D.. Using kinect for hand tracking and rendering in wearable haptics. In: *World Haptics*. 2011, p. 317–321.

34. Oikonomidis, I., Kyriazis, N., Argyros, A.. Efficient model-based 3D tracking of hand articulations using kinect. In: *BMVC*. 2011, p. 101.1–101.11.

35. Hu, R.L., Hartfiel, A., Tung, J., Fakih, A., Hoey, J., Poupart, P.. 3D pose tracking of walker users' lower limb with a structured-light camera on a moving platform. In: *CVPRW*. 2011, p. 29–36.

36. Zhang, Z.. Microsoft kinect sensor and its effect. *IEEE Transactions on MultiMedia* 2012;**19**(2):4–10.

37. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.. High accuracy optical flow estimation based on a theory for warping. In: *ECCV*. 2004, p. 25–36.

38. Bruhn, A., Weickert, J., Schnörr, C.. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *International Journal of Computer Vision* 2005;**61**(3):211–231.

39. Porikli, F.M.. Integral histogram: A fast way to extract histograms in cartesian spaces. In: *CVPR*. 2005, p. 829–836.

40. Grabner, H., Bischof, H.. On-line boosting and vision. In: *CVPR*. 2006, p. 260–267.

41. Grabner, H., Leistner, C., Bischof, H.. Semi-supervised on-line boosting for robust tracking. In: *ECCV*. 2008, p. 234–247.

42. Jia, X., Lu, H., Yang, M.H.. Visual tracking via adaptive structural local sparse appearance model. In: *CVPR*. 2012, p. 1822–1829.

43. Liu, C.. *Beyond Pixels: Exploring New Representations and Applications for Motion Analysis*. Ph.D. thesis; Massachusetts Institute of Technology; 2009.

Qi Wang received the B.E. degree in automation and Ph.D. degree in pattern recognition and intelligent system from the University of Science and Technology of China, Hefei, China, in 2005 and 2010 respectively. He is currently an associate professor with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China. His research interests include computer vision and pattern recognition.

Jianwu Fang received the B.E. degree in automation and M.E. degree in Traffic Information Engineering and Control from the Chang'an University, Xi'an, China, in 2009 and 2012 respectively. He is currently a candidate Ph.D. with the Center for Optical Imagery Analysis and Learning, State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China. His research interests include computer vision and pattern recognition.

Yuan Yuan is a full professor with the Chinese Academy of Sciences (CAS), China. Her major research interests include Visual Information Processing and Image/Video Content Analysis. She has published over a hundred papers, including about 70 in reputable journals, like IEEE transactions and Pattern Recognition, as well as conferences papers in CVPR, BMVC, ICIP, ICASSP, etc.