Egocentric Object Tracking: An Odometry-Based Solution

Stefano Alletto(✉), Giuseppe Serra, and Rita Cucchiara

University of Modena and Reggio Emilia, Modena, Italy
{stefano.alletto,giuseppe.serra,rita.cucchiara}@unimore.it
http://imagelab.ing.unimore.it

Abstract. Tracking objects moving around a person is one of the key steps in human visual augmentation: we could estimate their locations when they are out of our field of view, know their position, distance or velocity just to name a few possibilities. This is no easy task: in this paper, we show how current state-of-the-art visual tracking algorithms fail if challenged with a first-person sequence recorded from a wearable camera attached to a moving user. We propose an evaluation that highlights these algorithms' limitations and, accordingly, develop a novel approach based on visual odometry and 3D localization that overcomes many issues typical of egocentric vision. We implement our algorithm on a wearable board and evaluate its robustness, showing in our preliminary experiments an increase in tracking performance of nearly 20% if compared to currently state-of-the-art techniques.

Keywords: Visual tracking · Wearable computing · Egocentric vision

1 Introduction and Related Work

The rapid progresses in the development of systems based on wearable cameras and embedded computing devices have created the conditions to allow computer vision technologies to augment experience in everyday life activities such sport, education, social interactions, cultural heritage visits etc. The new and challenging setting that results from the adoption of an egocentric perspective in the video analysis provides a unique insight into many problems that have already been addressed by the traditional computer vision.

Egocentric vision (or ego-vision) is a recent topic that aims at augmenting human visual capabilities and perception by enhancing our field of view [3], analyzing social interactions [4], localizing objects [6] or extracting salient moments from our daily lives [10] based on what we see.

The adoption of an egocentric perspective creates new challenges for traditional computer vision, in particular when facing the task of tracking moving objects. It is a complex field in which many results have been achieved [11], but there still are open issues. A working tracker should handle scale, illumination changes, background clutter, partial occlusions and keep track of the object of interest overcoming these challenges. A notable solution is the Fragments-based

© Springer International Publishing Switzerland 2015
V. Murino and E. Puppo (Eds.): ICIAP 2015, Part II, LNCS 9280, pp. 687–696, 2015.
DOI: 10.1007/978-3-319-23234-8_63

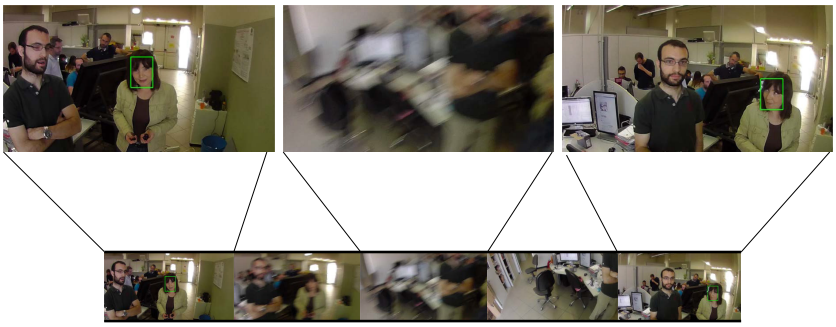

Fig. 1. An example an ego-vision sequence where fast camera motion causes objects and people to exit the camera field of view.

Robust Tracking (FRT) [1] that addresses the problem of partial occlusions representing the object template by multiple image patches. While being very fast and accurate in the case of small changes in object appearance, this method tends to worsen its performances if challenged with severe changes in appearance. To address this issue, tracking approaches that employs discriminative classifiers to identify the target opposed to the background have been proposed. The Hough-Based Tracker (HBT) by Godec *et al.* proposed in [7] is a tracker that aims at non-rigid targets in a discriminative classifier with segmentation of the object itself. The Structured Output Tracking with Kernels (STR) [8] algorithm employs a structured output supervised classifier to acquire training data directly from the image integrating the labeling procedure and its learner. Tracking Learning and Detection (TLD) [9] combines the results of an optical flow tracker and a detector, which can identify errors and learn from them.

Despite being a core component of many algorithms based on video analysis, very few works use visual tracking applied to ego-vision settings. Alletto *et al.* [2] employ HBT with some adjustments to better suit the first person perspective. Fan *et al.* [3] track features clustering motion based on the optical flow of the scene. However, these works employ trackers for a specific task in very constrained settings and lack generality.

In this work, we address the problem of tracking a single object of interest from a first person perspective. A typical example is to follow the detected shape of a friend walking with the camera wearer, which is one of the most challenging situations in visual tracking. Due to the novelty of the task if applied to a first-person wearable camera view, this work first discusses the main issues of egocentric tracking such as fast camera motion, see Fig. 1. Then, we propose a 3D localization method based on monocular visual odometry that is used to enforce a tracking framework. By intervening on its detection phase predicting the location where to expect to see the object after it re-enters the field of view following a loss or a total occlusion, we can re-initialize the tracker even when the appearance of the object differs from the learned model in a way the would have otherwise prevented the detection.

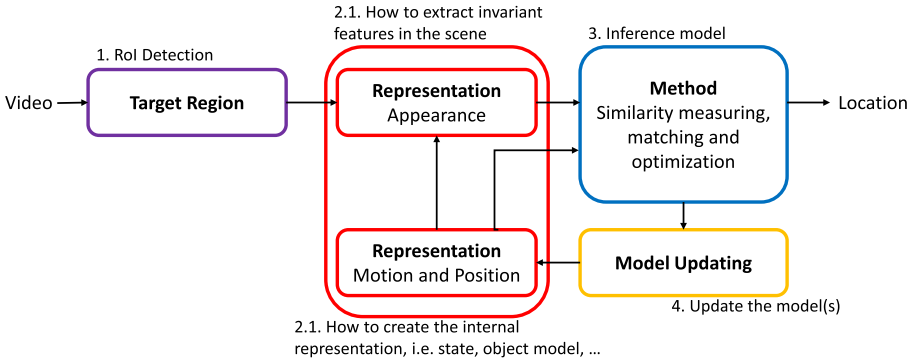

Fig. 2. The typical schema of a tracking algorithm. Each block, represented in different colors, is usually implemented in various ways thus differentiating trackers from one another.

We show that the proposed method outperforms state-of-the-art techniques by nearly 20% and, while being an initial study on the matter of visual augmentation via egocentric object tracking, it provides promising results and encourages further research on the topic. We also implement the described technique on a wearable embedded device coupled with a head mounted camera capable of acquisition and processing.

2 Motivations

In this section, we consider the typical approach to tracking a single moving object and discuss the challenges posed by the setting of first person camera views. Here, tracking-by-detection methods, since they need a specific detector for a specific target (e.g. a people detector) are not taken into account; in fact they cannot be used if the targets are unknown, as in this context.

The typical workflow of a tracking algorithm is shown in Figure 2: after a detection step, several candidate Regions of Interest (RoIs) are automatically selected by the tracker (e.g. around the previous position, with Gaussian scattering, etc) and visual features like appearance, position, and motion are extracted and used both for the frame under evaluation (step 2.1) and for the internal model (step 2.2). Then tracker (step 3), is characterized by an inference method that associates candidate RoIs to model(s), solving an optimization problem or performing a classification. The trackers often differ from each-other in the methods used to update the model (step 4): some of them do not modify the model at all, others keep more models of the target object, updating their short-term and long-term memory with some learning step.

Since the issues that make tracking so difficult are many, there is no tracker that can outperform the others regardless of the setting. In [11], thirteen different problems which can potentially lead a tracker to failure have been considered. In order to analyze them from an egocentric perspective, we could divide them

in three categories: i) Lighting: lighting conditions and variations, the target surface, transparency and its shape in general; ii) Motion: motion smoothness, the motion coherence (between target, camera and background), the camera motion, the camera zoom and the long-term motion (of both target and camera); iii) Scene: the scene clutter, confusion, contrast or occlusions.

All these issues can occur simultaneously making the problem of tracking still unsolved. Considering videos taken from an egocentric perspective, our experiments show that the most crucial characteristic of this setting are the peculiar motion patterns. We can state that in egocentric tracking the main motion issues can be summarized as

- Camera motion: the camera is moving as the head of its owner, thus is unconstrained and often unpredictable. Trackers based on motion estimation alone (e.g. optical flow tracking) are likely to fail due to the significant amount of noise introduced by the ego-motion.
- The long-term motion: since the camera is not fixed, a long-term component of motion must be considered. In fact, the tracked object can change its appearance substantially due to a different point of view of the observer derived from his motion. This results in the need to keep a complex object model capable of recognizing different appearances of the same object, e.g. TLD object model.
- The motion coherence: the complexity of human attention patterns lead to very challenging situations in ego-vision. The motion coherence between the target, the background or the camera is indeed far from granted. Even a still object could bounce in and out of the camera field of view due to ego-motion, or a still background can be all but still, having significant apparent motion. Trackers that rely on robust training such as Struck, utterly fail in this setting due to the impossibility to learn an effective representation of the object vs background motion or appearance.

For these reasons, this work focuses on analyzing some of the most promising trackers currently available [11], highlighting their limitations when challenged with ego-vision sequences. Furthermore, instead of focusing on one of the issues and developing a new tracker to handle that particular situation, we develop a module based on visual odometry that can enhance the tracking performance of existing algorithms, a more general solution to a problem which is still to be solved.

3 Proposed Method

In order to overcome the issues of egocentric tracking, we develop a method that integrates 3D target localization into the detection component of the tracking algorithm. Based on experimental results (see the following section) we extend the recent tracker TLD [9] with a module that supports detection with 3D information. However, our approach can be adapted to other visual tracking

techniques such as Struck [8], by introducing in its Structured SVM inference procedure a set of weights learned using 3D target localization.

TLD framework features three main components: a *Tracker* which estimates the object's motion based on a Median-Flow algorithm. This component of the framework is likely to fail if the object exits the camera field of view and it is not able to resume the tracking by itself. A *Detector* intervenes treating each frame independently and performs the detection localizing the appearances of the object which have been observed and learned in the past, recovering tracking after the *Tracker* fails. The *Learning* component observes the performance of both *Tracker* and *Detector*, estimates their error and adds training samples to its object model.

A typical ego-vision characteristic is that the camera wearer can have very fast head motion, e.g. when he is looking around for something. Another example is the object of interest being a person walking with the subject wearing the camera, resulting in him looking at the path they are walking as often as to his companion. With these characteristics in mind, it is clear how important the detection phase of the tracking process is, since the object of interest can be outside of the field of view for a significant part of the sequence.

In particular, the TLD detector is based on a sequence of classifications. Patches are densely sampled in the image at different scales obtaining a large set of candidates which is iteratively reduced by following rejection steps. First all patches with low gray-scale values variance are rejected to rapidly eliminate a large set of non-object candidates. Then patches that passed the first step are classified by an ensemble of classifiers based on pixel comparison trained offline.

The final step is a NN classifier that compares the patches with the learned object model M. This model is composed by a set of positive p^+ and negative p^- patches that respectively encode object and background parts. A patch p is recognized as the object of interest if its *relative similarity* with the model is

Fig. 3. Example sequence of the proposed approach.

greater than a threshold $S^r(p, M) > \theta_{NN}$. The *relative similarity* is defined as $S^r = \frac{S^+}{S^+ + S^-}$ where

$$S^+(p, M) = max_{p_i^+ \in M} S(p, p_i^+) \quad and \quad S^-(p, M) = max_{p_i^- \in M} S(p, p_i^-) \tag{1}$$

are the similarity with the positive and negative nearest neighbors.

However, detection based on the appearance encoded in the learned model M can fail if the object changes too fast or the change takes place out of the camera view. To deal with this issue we extend the detection component by adding 3D motion estimation of the head and the object to model its behavior when it is not visible.

To compute the head motion we use "Semi-Direct Visual Odometry" (SVO) algorithm [5] that can be run in real-time on an on-board embedded computer, since it eliminates the need of costly feature extraction for motion estimation operating directly on pixel intensities. SVO estimates the rigid body transformation between two consecutive camera poses $G_{k,k-1}$ minimizing the negative log-likelihood of the intensity residual:

$$G_{k,k-1} = \arg\min_G \int \int_R \rho[\delta I(G, \mathbf{u})] d\mathbf{u}. \tag{2}$$

where δI is the photometric difference between pixels observing the same 3D point and $\rho := \frac{1}{2} \|\cdot\|^2$.

When the object is visible and tracked, given the bounding-box at the frame k and the head motion estimation $G_{k+\Delta t,k}$ provided by the SVO algorithm, we can estimate its 3D motion model and use it to predict the image coordinates where the target should appear after a loss. In particular, let $c_k = (x_k, y_k)$ be the center of the bounding-box at the frame k, we can predict the image point coordinates where the center should be located when it becomes visible again after an interval Δt:

$$\hat{c}_{k+\Delta t} = P(G_{k+\Delta t,k} \cdot (P^{-1}(c_k, d) + \Delta C(t))). \tag{3}$$

where P is the projection model that maps 3D points to the image coordinates, d is an approximation of the depth based on the scale of the detection at the frame k and ΔC is the 3D target motion that we define, assuming a linear velocity, as:

$$\Delta C_{\Delta t} = C_k + V_k \Delta t \tag{4}$$

where V_k is the velocity vector of the center of the bounding-box at the frame k. While the assumption of linear velocity may appear limiting, the setting of ego-vision often requires the tracking of objects that are somehow related to the person wearing the camera, e.g. people walking beside him, and thus the assumption is often satisfied.

Based on this estimation the we extend the *relative similarity* including the displacement between the estimated center of the bounding-box $\hat{c}_{k+\Delta t}$ and the

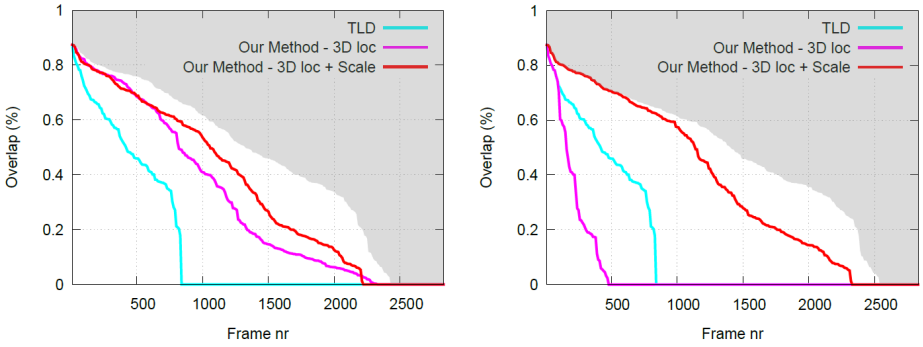

Fig. 4. Comparison between TLD and the two variations of our approach. On the left: the results of our method using the SLAM localization. On the right: SLAM data perturbed with gaussian noise. The displayed results are obtained concatenating and sorting the frames of the different video sequences.

center of the candidate patch p in the image coordinates. The new similarity function is defined as:

$$S^r(p, M, \hat{c}_{k+\Delta t}) = (1 + e^{-(\frac{d_x^2 + d_y^2}{2\sigma^2})}) \cdot S^r(p, M) \tag{5}$$

where d_x and d_y are the displacement components in x and y, and σ is the variance of two-dimensional Gaussian function center in $\hat{c}_{k+\Delta t}$ (based on preliminary experiments we fix $\sigma = 20$). This new similarity is used to identify whether the patch is recognized as the object of interest by comparing it to the threshold θ_{NN}.

We observed that patches where the relation $S^r(p, M) < \theta_{NN} < S^r(p, M, \hat{c}_{k+\Delta t})$ is satisfied are likely to contain a detail of the target. While these patches are sufficient to restart the *Tracker*, they are not suitable to provide an accurate localization of the object of interest due to scale errors. To address this issue we adjust the scale considering the size of the patch and the dimension of the bounding-box at the frame k. This allows us to resume tracking with a more robust initialization and follow the target more properly.

Figure 3 presents an example of our method applied to ego-vision sequence. The green bounding box is the chosen detection, the blue ones represent the image patches obtained from the NN classifier. The cyan patch in the last frame satisfies $S^r(p, M, \hat{c}_{k+\Delta t}) > \theta_{NN}$ and is used as input in the scale adjustment step to compute green detection. The two-dimensional Gaussian function, that predicts the center of the object of interest c_k, is represented in shades of red and yellow.

4 Experimental Results

We described the differences and challenges posed by egocentric perspective compared to the traditional tracking setting.

We now evaluate the following trackers on first-person sequences to show their performance: STR [8], HBT [7], TLD [9], FRT [1]. We also employ a baseline NCC to show the performance of a simple tracking by detection approach compared to more complex methods. All these trackers achieve good results on standard benchmarks and datasets [11], but substantially different performances are to be expected when considering the egocentric perspective of first person videos. Indeed, these trackers are not designed to cope with the abrupt losses of the target due to head and camera motion, or changes in scale that are a consequence of movement. To validate this statement we collected a set of five ego-vision sequences that contain people interactions in both indoor and outdoor environment. Videos are recorded and processed using a wearable Odroid-XU board, that embeds the ARM Exynos 5 SoC, and a glass-mounted Matrix Vision BlueFox global shutter camera. We add a 3000 mAh battery pack to make it portable.

Figure 5 shows the results of this evaluation of the aforementioned trackers on one of the ego-vision sequences that contains changes in illumination and fast camera motion induced by head motion and walking.

It can be noticed how the challenging aspects of the ego-vision scenario, namely the fast camera motion and the target exiting the camera field of view after very few frames, significantly worsen the performance of state-of-the-art trackers. In particular, HBT and FRT fail due to the lack of the ability to cope with the exit of the target from the frame. STR, while trying to adapt to such a situation, does not perform any loss detection and quickly adapts its model to the background. A simple tracking by detection approach (NCC) can recover tracking if the appearance of the object becomes close to the initial template but it is shown to be unable to provide sufficient results in most cases. Among the evaluated trackers, due to its hybrid framework of tracking and detection, TLD results in being the more robust to the recurring loss of the target but still presents a low overlap measure. In fact its detector, while often being able to resume tracking after a loss, requires the new appearance to have already been

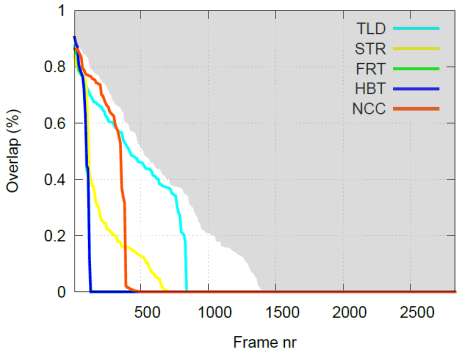

Fig. 5. Tracking results on the ego-vision sequences.

observed and encoded in the model. In egovision it often occurs that the target can change its appearance out of the camera field of view thus compromising its ability to detect the object.

Furthermore, consider the tracking upper bound of Figure 5, which is the performance obtained by the combination of the evaluated trackers by taking at each frame the best result in terms of overlap. This upper bound shows little room for improvement and demonstrates the requirement of a different approach to the task of egocentric visual tracking.

Figure 4 shows the improvement tied to the enforcing of visual tracking with 3D localization. In particular, we present a comparison between the TLD tracker and two variations of our approach: 3D localization estimation with no additional scale adjustment (Our Method - 3D loc) and improved scale estimation considering the size of the patch and the dimension of the bounding-box at the previous frames (Our Method - 3D loc + Scale). Our complete approach can achieve an average overlap of 35.26% while on the same data TLD scores a 15.28%, featuring an increase of 19.98%.

It can be noticed how if the prediction of the 3D position of the center of the object is accurate enough, performing the scale adaptation step is not strictly required since it only slightly improves results. On the other hand, if the localization results are less precise, not taking into account the errors in the scale of the detection severely impact of the performance of our method. As Fig. 4 shows, adding a gaussian noise of $\sigma = 15px$ to the predicted position requires the scale adjustment step to work properly. This is due to the error introduced by the noise excessively perturbing the localization resulting in the impossibility to resume tracking.

5 Conclusions

In this paper we presented a method that uses a semi-direct monocular visual odometry algorithm to infer the head motion of the camera wearer and subsequently compute the 3D location of the target. This allows us to build a target motion model used to predict the image coordinates where to expect it to reappear after a loss. By exploiting this information we can intervene in the detection component of a tracker and effectively leading it to a more robust detection. While this is an initial study on the matter, our preliminary results validate our method by showing a significant improvement of the state-of-the-art performance.

References

1. Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: Proc. of CVPR (2006)
2. Alletto, S., Serra, G., Calderara, S., Solera, F., Cucchiara, R.: From ego to nos-vision: detecting social relationships in first-person views. In: Proc. of CVPR Workshops (2014)

3. Fan, K., Huber, J., Nanayakkara, S., Inami, M.: Spidervision: extending the human field of view for augmented awareness. In: Proc. of ACM Augmented Human (2014)
4. Fathi, A., Hodgins, J., Rehg, J.: Social interactions: a first-person perspective. In: Proc. of CVPR (2012)
5. Forster, C., Pizzoli, M., Scaramuzza, D.: Svo: fast semi-direct monocular visual odometry. In: Proc. of ICRA (2014)
6. Funk, M., Boldt, R., Pfleging, B., Pfeiffer, M., Henze, N., Schmidt, A.: Representing indoor location of objects on wearable computers with head-mounted displays. In: Proc. of ACM Augmented Human (2014)
7. Godec, M., Roth, P.M., Bischof, H.: Hough-based tracking of non-rigid objects. Computer Vision and Image Understanding **117**(10), 1245–1256 (2012)
8. Hare, S., Saffari, A., Torr, P.H.: Struck: structured output tracking with kernels. In: Proc. of ICCV (2011)
9. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **34**(7), 1409–1422 (2012)
10. Lee, Y.J., Ghosh, J., Grauman, K.: Discovering important people and objects for egocentric video summarization. In: Proc. of CVPR, vol. 1, pp. 1346–1353 (2012)
11. Smeulders, A., Chu, D., Cucchiara, R., Calderara, S., Dehghan, A., Shah, M.: Visual tracking: An experimental survey. IEEE Transactions on Pattern Analysis and Machine Intelligence **36**(7), 1442–1468 (2014)