Abstract

Usually tracking objects by means of a moving camera means solving the problem of segmenting its motion from the background and maintain the gaze on it. Usually key points, i.e., features like corners, are matched throughout consecutive frames to register frames and detecting objects or to compute a fixation point on the object of interest. None of the methods proposed so far investigates how, more then just matching features between frames, the tuning of their quality would increase the robustness of the methods. Hence, the novelty proposed in this paper concerns the automatic tuning of the focus parameter to increase the tracking capabilities and strength the robustness of the entire system.

1. Introduction

In this paper we propose a new method for tracking moving objects with a moving camera by extracting, selecting and tracking features on it. Moreover, in addition to these common tasks we present a novel technique to tune the intrinsic parameters of the camera in order to control the quality of the features involved in the tracking process. It is indeed true that usual feature trackers [5] do not adopt techniques to locally verify the quality of the features. They just run outliers rejection rules to classify tracked features as well tracked or as outliers (i.e., badly tracked features). In [6], Tommasini et al. use a statistical rule to identify and to reject the outliers but requiring a large number of feature to track. Recently, Sugaya and Kanatani proposed a technique to remove outliers to extract moving objects from a moving camera sequence [4]. The performance of such algorithm, based on subspace separation, does not allow a real-time outliers detection. Indeed, just considering the outliers detection task the system performance shows an upper bound of 4 frame/s. Lately, Guo et al. [1] proposed a linear combination representation for outliers detection. The proposed scheme works on 4 frames by estimating 4 parameters for each of them still avoiding a real-time execution. Then, all these outliers techniques require a consistent number of good features or to operate on multiple consecutive frames. This means that we cannot base our feature tracker on local computations, but we are required to globally consider all the feature extracted and tracked. Instead, with the proposed technique we are able to improve the performance of the feature tracker by enhancing the quality of the objects’ features he tracks. Moreover, by improving the features belonging to an object of interest yields to a better quality of the acquisition of such an object. This can result in an improvement of the robustness of the high level algorithms (i.e., object recognition, face detector, event analysis, etc.).

The architecture of the proposed method can be seen in Fig. 1. An active tracking technique has been adopted to maintain the fixation on the object of interest by tracking its features and rejecting those belonging to the background. Let \( f_1, \ldots, f_n \) be the features classified as belonging to the target. Such features are continuously investigated by a quality criteria based on a semi-quality image function. In particular, let \( SQ \) (Sharpness Quality) be such a semi-quality function [3] used to assess the focusing grade. It is defined as a linear combination of quality operators as follows:

\[
SQ = a_2 \cdot TN + b_2 \cdot FL + c_2 (1 - E) + d_2 (1 - \frac{DM}{255}) + e_2 (1 - V)
\]

where \( a_2, \ldots, e_2 \) are weights used for the normalization and \( E \) is the entropy, \( TN \) is the tenengrad, \( FL \) is the flatness, \( DM \) is the maximum local difference and \( V \) is the variance.

A quality selection function considers the \( SQ \) values of the tracked features and selects two of them as representative of the target quality and therefore more useful to tune the focus parameter. In our solution, the features that respectively have the maximum and the minimum sharpness values are selected as pivots for the focusing strategy. In particular their tenengrad values are exploited by a neural network to decide the amount of focus tuning.

In the following sections the techniques to track the features on the target and to tune the focus parameter on them
them will be described better. Finally, the experimental results will show how the proposed solution achieves a better tracking of the features by reducing features’ residuals, hence improving the object quality.

2. Active Tracking

Let us suppose that an high level module selects an object to track. To maintain the fixation on such an object the work of Tordoff and Murray [7] has been considered. They solve the problem of tracking a fixation point, mostly the centre of mass of an object, by adopting an affine projection. In particular, when the distance between the object and the camera is much greater than the depth of the object it is possible to consider the following affine projection:

\[ x = MX + t \]  

where \( x \) is the fixation point, \( M \) is the affine transform, \( X \) is the real position of the point and \( t \) is a translation vector. The main issue in adopting this technique concerns the determination of the points to use during the computation of the affine transform. To solve this problem we have introduced a feature clustering technique able to classify a tracked feature as belonging to different moving objects or to the background. Then, to distinguish between different clusters we need first to compute the affine transform of the entire image [2]. In particular, let \( \tilde{A} \) be the computed affine transform and \( \tilde{f}_i^{t-1} - \tilde{f}_i^t \) the position of a generic feature respectively at time instant \( t - 1 \) and \( t \). At this point, the effective displacement of the feature \( i \) is \( d_i = f_i^t - f_i^{t-1} \), while the transformed, compliant to the affine transform, is as follows:

\[ \tilde{d}_i = f_i^t - \tilde{f}_i^{t-1} = f_i^t - \tilde{A} f_i^{t-1} \]  

where \( \tilde{f}_i^{t-1} \) represents the position of the feature \( i \) registered by means of the affine transform \( \tilde{A} \).

Let \( TFS_{obj} \) be the set of features extracted from a window (i.e. fovea) centred on the object of interest. Then, the clusters computation is performed by the following rule:

\[ C_{obj}(\tilde{d}) = \left\{ f_i \in TFS_{obj} \left| \| \tilde{d}_i - \tilde{d} \|_2 \leq r_{tol} \right. \right\} \]  

where \( C_{obj}(\tilde{d}) \) is the cluster having all the features \( i \) whose displacement \( \tilde{d}_i \) is such that the norm between it and the vector \( \tilde{d} \) is lower than a defined threshold \( r_{tol} \). Such a threshold has been introduced to sidestep the radial effect visible during zoom operation. Experimentally, we have seen that such a value is linearly dependent to the zoom speed (i.e. magnification grade between two consecutive frames) and that under real time computation the range \([0, 2]\) demonstrated to be a good range. Once the computation of all the clusters is done, we can easily find the background cluster (i.e. those having a null displacement) and therefore all the features that have been erroneously extracted from the background in the previous feature extraction step. Such features are considered outliers and rejected. After having deleted all the features either belonging to the background or to a cluster with cardinality lower than three, we can apply the technique proposed by Tordoff-Murray over each cluster to determine the fixation point of each moving object inside the image.

Let \( g_t \) be the previous fixation point of the object \( i \). We need to solve the following equation:

\[ g_{d_k'} = N g_t + r \]  

where \( g_{d_k'} \) is the new position of the fixation point. To solve such an equation the cluster \( C_{obj}(\tilde{d}_k) \) having the features previously used to compute \( g_t \) is used to compute \( N \). Finally, the features belonging to the selected target are therefore used to tune the acquisition quality.

3. Neural Focusing

To tune the focus parameter with respect to the quality of the target’s features we have developed a strategy based on a neural hierarchy that has resulted to be really efficient and reliable. In particular, we have identified two typical cases on the basis of the SQ-value of the representative features: a) the object of interest is really defocused or b) it is not optimally focused. In the first case we have developed a four steps strategy:...
• Compute a first focus position inside the entire range. *Entry Point*

• Given the entry point move the focus in a position where the tenengrad function starts to have a bell-shaped shape

• Compute the slope of the function surface between two focus positions whose tenengrad values are higher than the threshold value.

• Use the slope computed at previous step to estimate the optimal focus position *OF*.

In particular, if the best feature (i.e. the one with the best SQ value) is really out-of-focus (i.e. the tenengrad value is lower than a threshold) an *entry point* neural network is applied on the current focus value, its related tenengrad measure and the zoom value. The developed feed forward network is full connected and it is composed by three input nodes, three hidden layers each composed by three nodes and a output node. The returned value identifies a first focus position $F_1$ within the entire focus range.

When the camera reaches the position $F_1^2$ a second feed forward network is applied to the pattern $\{F_1, TN_1, Zoom\}$ given by the current focus position, the tenengrad value of the worst feature and the zoom level. Topologically, such a network is full connected and composed by three input nodes, three hidden layer of three nodes each and a output node. Such a node identifies a step for the computation of the second focus position $F_2$. Hence, the camera is requested to tune the optics to reach the focus position $F_1^2 + F_2 = F_2^2$ where a new tenengrad measure $TN_2$ is computed for the worst feature. At this point a further feed forward network takes as input the pattern $\{F_1, TN_1, F_2^2, TN_2, Zoom\}$ given by the two computed focus positions with corresponding tenengrad measures and the zoom position. Such a feed forward network is fully connected and composed by five input nodes, two hidden layers of five nodes each and two output nodes related to two focus steps. The two focus steps $F_3$ and $F_4$ computed by this NN are used to tune the focus parameter first toward the position $F_2^2 + F_3 = F_3^2$ where $TN_3$ is computed then toward the position $F_3^2 + F_4 = F_4^2$ where $TN_4$ is computed. Finally, the system is able to estimate the optimum focus position by running a final neural network. This NN takes the pattern $(F_3^2, TN_3, F_4^2, TN_4, Zoom)$ given by two focus positions, the corresponding tenengrad measures and the actual zoom position and gives the estimated position of the optimum focus $OF$.

Instead, if the object is not really defocused, hence the feature with the worst SQ value has a tenengrad value greater then the threshold $TN$ a different strategy has been adopted. Let $F_3'$ and $F_4'$ be the current and previous focus position, then $TN_3'$ and $TN_4'$ are the tenengrad values of the current worst feature respectively at focus positions $F_3'$ and $F_4'$. Such values are used as pattern for the optimum focus neural network to compute the $OF$ position.

To train the neural networks we have followed a step by step strategy by simulating the tuning process. Then, starting from defocused objects we have automatically trained the first network by searching for the first value on the bell-shape and using a randomly selected value outside the bell. The outputs of the first trained neural network have been used as input of the second neural network as previously described. At each neural training the value to use in the training patterns have been automatically identified by scanning the entire focus range of the camera.

4. Experimental Results

To test the effectiveness of the proposed method and its impact on active vision problems, the experiments have been conducted on images acquired by a SONY DFW-VL 500 with adjustable parameters and a CoHu 3812 with automatic tuning of the intrinsic parameters. A first experiment
Figure 3. Evaluation of the proposed technique on the basis of the tenengrad value computed
on a selected object with respect its position inside the image. Chart (a) plots the tenengrad value
computed on images acquired with the proposed tuning strategy, (b) plots the tenengrad value com-
puted on images tuned by an auto focusing technique and (c) shows a couple of frames obtained by
the automatic tuning strategy.

has been executed by acquiring the same scene, represented
by a moving object in an indoor environment, with both
cameras and by comparing the residuals computed over tar-
get’s features. As shown in Fig. 2 the use of the proposed
tuning strategy has the result to reduce the features’ resid-
uals (i.e. intensity differences between current and previous
features) between consecutive frames. This is really im-
portant since reducing the residuals of the features means
maintaining the goodness of the feature hence improving
the tracking capabilities. During this test phase, the mean
residual value computed over all the well tracked feature
decreased of about 22% yielding to an increment of the fea-
tures life (i.e. number of frames for which they are con-
tinuously tracked) of about 10%.

A second experiment has been conducted on an outdoor en-
vironment by acquiring vehicles moving around a parking
lot with both cameras. After few frames, needed to estimate
the first optimum focus position, the performances of the
proposed strategy show a mean increment of the tenengrad
value near to 15%. If we consider the case in which the ob-
ject is not in the center of the image the performances have
an increment of about 25%. The same strategy has been
followed for 20 sequences of 30s each of footage to give an
overall evaluation of the proposed method compared with
a commercial automatic technique. As can be seen in Fig-
ure 3, when the position of the object of interest is the cen-
ter of the image the tenengrad value computed over images
acquired by a commercial camera decreases (Figure 3(b)).
Instead, the proposed technique, being focused on the ob-
ject, maintains the tenengrad value almost constant regard-
less the position of the object (Figure 3(a)). The overall gain
of the proposed solution is estimated in a fair 6% for objects
in the image center to a remarkable 23% when the object is
almost in a corner of the image.

5. Conclusions

In this paper, we have proposed a novel method to ac-
tively track features on objects by also tuning the focus pa-
rameter to increase the quality of the object. Results have
shown that by focusing the features the tracking capabili-
ties increase as well as the object quality if compared with
standard camera solutions.

References

tation for outlier detection in motiontracking”. In IEEE Int.
Conf. on Comp. Vis. and Pat. Rec., volume 2, pages 274–281,
San Diego, CA, USA, June 20-25 2005.
tering method for object detection with an activecamera. In
IEEE International Conference on Image Processing, pages
abilistic system fo adaptive regulation of imageprocessing pa-
1996.
a moving camera video sequence. In 10th Symposium on
Sensing and Image Information, pages 279–284, Yokohama,
features. Technical Report CMU-CS-91-132, Carnegie Mel-
lon University, Pittsburgh PA, 1991.
good features track better. In IEEE International Conference
on Computer Vision and Pattern Recognition, pages 178–183,