A dynamic model integrating colour and shape information for objects tracking in conditions of occlusion

L. Marchesotti, S. Piva, C.S. Regazzoni
Department of Biophysical and Electronic Engineering (DIBE) - University of Genoa
Via all’Opera Pia 11A, 16145 Genova (Italy)
e-mail: carlo@dibe.unige.it

Abstract
In this paper, an algorithm for tracking multiple rigid and non-rigid objects in conditions of occlusions is presented. The proposed method is based on a scalable and adaptive model based on joint information of color and shape. Through a GHT (Generalized Hough Transform) based voting method the center of mass of each object can be determined in real time with a good degree of precision. Quantitative and qualitative results are presented to validate efficiency of the method.

1. Introduction
In order to detect, track and analyze objects of interest in a given video sequence, different steps have to be carried out at different abstraction levels. Typically chain of modules can be assembled in order to process video data [1], the first step is however a Dynamic Change Detection performing the difference between the current image and a reference background in order to detect moving regions in the image. Each moving area (called Blob) detected in the scene is bounded by a rectangle to which a numerical label is assigned. The core part of such systems is however represented by tracker algorithms, which have to preserve over time the identity of detected objects. Whereas a situation in which objects are isolated is relatively simple to handle, when multiple blobs overlap creating occlusions is still a not completely solved issue in the state of the art. A common method used to overcome this problem is represented by the Kalman filter [2] that takes into account the position and the velocity of each blob in situation of occlusions. In the presented work, shape information is combined with color information to enhance robustness of the algorithm in conditions of heavy occlusions. In addition, objects’ nature is taken into account in order to insert periodic motion information under the form of a dynamic model.

2. The model
Modeling objects of interest with sufficient generality implies taking into account their different characterizing features. Non rigid objects (e.g.: pedestrians) differs from rigid ones (e.g.: vehicles) not only for dimensions, dynamics and dominant colors but also for their intrinsic self-occluding nature. Therefore, a robust model has to be designed with a sufficient degree of flexibility in order to be representative and effective for all typologies of objects. The model here proposed takes origin from an existing work presented in [6] where shape information under the form of high curvature points (i.e.: corners) was used to estimate the center of mass of each blob in situation of occlusions. In the presented work, shape information is combined with color information to enhance robustness of the algorithm in conditions of heavy occlusions. In addition, objects’ nature is taken into account in order to insert periodic motion information under the form of a dynamic model.

2.1. Shape information extraction
In order to efficiently represent shape information, corners have been chosen for their relatively high stability and persistence [7]. The SUSAN algorithm described in [8] is here used for corners extraction.

Figure 1. (a) Corner extraction. (b) Mask for color evaluation centered on corner.

As it can be seen from fig. 1(a), a quadruplet of parameters \((\omega, dx, dy, p)\) can be extracted and
employed to define a given corner; the first value is the angle direction \( \omega \) of the gradient vector at the corner position with respect to the original image, \( dx \) and \( dy \) are the differences in \( x \) and \( y \) with respect to a reference point and \( p \) is the persistence of each corner. 

The obtained quadruplet is used to model the position and orientation of the corner with respect to the reference point (i.e. the center of mass of the object).

### 2.2. Color information extraction

The color information is extracted and associated to each corner evaluated by the SUSAN algorithm. In particular, as it can be seen from fig. 2(b), given a corner and the change detection sub-image associated to the analyzed object, a \( N \times N \) mask is centered at corner location. In order to reduce as much as possible computational load, only “moving” pixels (e.g.: non zero pixels in change detection image) are taken into account in the estimation of color information. For each chosen corner, a triplet \((c_r, c_g, c_b)\) can be computed, describing the associated colour:

\[
c_p = \frac{1}{N_{nz}} \sum_{i=1}^{N} \sum_{j=1}^{N} v_{ij}^r, \quad c_g = \frac{1}{N_{nz}} \sum_{i=1}^{N} \sum_{j=1}^{N} v_{ij}^g, \quad c_b = \frac{1}{N_{nz}} \sum_{i=1}^{N} \sum_{j=1}^{N} v_{ij}^b
\]

with \( v_{ij}^h \) as the value of the pixel in the \( i\)-th row and \( j\)-th column in the mask for \( h\)-th channel (i.e.: red, green, blue), \( N_{nz} \) as the total number of non zero pixels and \( N \) the dimension of the mask in pixel (typically \( N=5 \)).

### 2.3. Assemble a static model

In order to render the extracted information in a coherent parametric model \( T \), color and shape information are joined in an observation vector:

\[
A(t)=[a(i(t)) : i=1, ..., N(t)]
\]

with:

\[
a(i(t))=(\omega, dx, dy, c_r, c_g, c_b, p_i)
\]

\( N(t) \) total number of corners extracted at time \( t \)

Given \( T^*(t-1) \) representing the object model learned at time \( t-1 \) and \( A(t) \), a new model \( T^*(t) \) can be computed by integrating the previous model \( T^*(t-1) \) with current observations \( A(t) \). In order to extract \( T^*(t) \), a function \( T^*(t) = F(T^*(t-1), A(t)) \) is computed. This function is quite complex and it is structured onto two steps:

1) First an integration step is performed by searching for the best matching between all the corners in \( T^*(t-1) \) with corners included in \( A(t) \). Matching is performed by investigating the existence of corners in \( A(t) \) laying in a neighborhood of corners contained in \( T^*(t-1) \). A match between a corner in \( T^*(t-1) \) and a corner in \( A(t) \) is found if the following conditions are satisfied:

\[
|\omega_{i(t-1)} - \omega_{j(t)}| < \text{angle\_th}
\]

\[
\sqrt{(dx_{i(t-1)} - dx_{j(t)})^2 + (dy_{i(t-1)} - dy_{j(t)})^2} < \text{dist\_th}
\]

where we introduced the Bhattacharyya coefficient for color distance [9].

Let us indicate \( A_i(t) \) as the list of corners that satisfies the previous conditions. For deciding the best matching between the corners in \( T^*(t-1) \) and the corners in \( A_i(t) \) the minimum distance criterion is selected. The matching corner is added to the table \( T^*(t) \) by using the current values at frame \( t \) \((\omega_i, dx_i, dy_i, c_{r_i}, c_{g_i}, c_{b_i})\) and the persistence \((p_i(t-1))\) of the corresponding corner in the model at frame \( t-1 \) incremented by one: \( p_i(t) = p_i(t-1) + 1 \).

If a corner in \( T^*(t-1) \) does not match with any corner in \( A(t) \), i.e. \( A_i(t) = \emptyset \) (no corner of \( A(t) \) satisfies the equations), then its persistence is decremented by one, i.e. \( p_i(t-1) = p_i(t-1) - 1 \). If the value of the obtained persistence \((p_i(t-1))\) is higher than a threshold \( \text{low\_pers\_th} \), then the corner is added to the model \( T^*(t) \). Otherwise, the corner is discarded.

2) Corners in \( A_i(t) \) detected at time \( t \), that do not match any entry of table \( T^*(t-1) \) are added to \( T^*(t) \) with a persistence equal to 1. [6]

Figure 2. (a) Model for a single object represented through the \( R \)-Table containing shape and color information (b) Estimation of corners’ center of mass

Results of the integration between observations \( A(t) \) and old model are stored under the form of a look-up table (\( R \)-table), a \( 7 \times n \) matrix outlined in fig. 2(a).

### 2.4. The dynamic model

The model previously outlined performs well in the representation of rigid objects such as vehicles; however performances are less satisfactory when blobs are pedestrian; periodic oscillations of arms and legs make shape model for these object non stationary. In order to take this into account, blobs classified as humans are modeled using a dynamic version of the model. The first step to achieve a time-varying model is to estimate periodicity of the movements of the object in analysis. To perform this, a simple algorithm (fig.3) based on change detection images is applied:
1) The change detection image for the object is stored as reference image when the entire object has entered the scene ($t_{\text{start}}$) and a counter $k$ is set to 0.
2) At frame $t_{\text{start}} + k$ the new change detection image is superimposed to the reference image by aligning centers of mass.
3) The overlapping region between the two frames is estimated and stored:
   \[ I(k) = \frac{\text{overlapping \_ pixels}}{\text{All \_ pixels}} \] (6)
4) The algorithm stops when two consecutive local maxima are found in $k_{\text{max1}}$ and $k_{\text{max2}}$ and the oscillation period is set to $P = k_{\text{max2}} - k_{\text{max1}}$.

Once oscillation period $P$ is estimated, a new model $\overline{T}$ can be built using static models:
   \[ \overline{T}(t) = [T_0^*(t), T_1^*(t), \ldots, T_{P-1}^*(t)] \] (7)
Instead of a single R-Table, $P$ different R-tables are stored to represent a single object.
In this case, given an input observation vector $A(t)$, an updated model $\overline{T}(t)$ can be computed (exactly as described in section 2.3) by integrating the correct static model $T_i^*(t-1)$ present in $\overline{T}(t-1)$ with current observations $A(t)$ according to (3-5) and with:
   \[ i = \left[ (t-t_{\text{start}}) \bmod P \right] \] (8)

2.5. The Tracking Algorithm
The tracking algorithm used to locate the center of mass for each blob is presented in fig.4. As it can be seen, different alternatives (i.e.: gray modules) are available to estimate the center of objects in relation to their status (i.e.: isolated, occluded) and nature (i.e.: rigid, non rigid):
- Estimation through Bounding Box
- Estimation through Static Model
- Estimation through Dynamic Model
A two step approach is required:
- Model Learning Phase (MLP)
- Model Operative Phase (MOP)

In MLP the model is learned from observations. Period estimation regulates the switching between the initialization step in which the model $T^*_i(t)$ or $\overline{T}(t)$ (i.e.: static or dynamic) has to be filled with new observations and the model update carried out according to rules proposed in [6]. It is assumed that period estimation returns 1 in case of rigid object, therefore after an initialization step $T^*_i(t)$ is continuously updated. In case of non-rigid object a number of cycles equal to the period $P$ is performed and then each static model $T^*_i(t)$ within $\overline{T}(t)$ is independently updated.

In this phase, center of mass estimation is carried out through the intersection of diagonals of Object’s Bounding Box (OBB) (see Fig.2(b)). When an occlusion occurs, center of mass cannot be evaluated anymore with diagonals of OBB because of noisy calculation of the BB; therefore MOP is entered.

3. Results
Several qualitative and quantitative tests have been performed. A comparison example of two screen shots is shown in figure 5: qualitative observations of real time working phases of the classical static model-based tracking vs the new dynamic colour based model tracking algorithm demonstrate a clear improvement in occlusion situations. To evaluate the model improvement, examples of the most interesting quantitative tests are also reported in this section. The
analysis has been led in terms of number of voting corners in a comparison between the new model and the model without non-rigid object handling support and color information.

The number of voting corners parameter for a given blob appears to be in inverted proportion with respect to the number of classification errors affecting the tracking system. For this reason this parameter can be used to quantitatively evaluate the tracking performances. Figure 6 depicts a temporal example of the voting corners number for a chosen video sequence. Furthermore in figure 7 the mean number of voting corners for a single blob is summarized for three different sequences. Quantitative results, as well as qualitative ones, confirm the performance improvement due to the new model employment. Test sequences can be found for reader interest at the following url: http://ginevra.dibe.unige.it/ISIP/sequencesLuca.html.

Last issue to consider commenting the proposed system regards the computational weight of the steps needed to implement our tracking system: as can be easily understood the amount of data processed to manage the correspondences detection frame by frame and limited to the described R-table is small.

This is surely true for the traditional static model, but the addition of the color information and the repeated periodic model structure do not influence the real-time applicability of the technique on today common PCs, provided with 2 GHz frequency processors and 512MB of RAM memory as the ones used for our tests.

4. Conclusions and future work
In this paper an innovative model for tracking rigid and non-rigid objects has been presented. The color and shape information combined with an elegant method for center of mass estimation have enhanced robustness of the algorithm in conditions of heavy occlusions.

Results show that it can be adopted as valid solution for real-time object tracking.

5. References