VEHICLE TRACKING USING FEATURE MATCHING AND KALMAN FILTERING

Kiran Mantripragada
IBM Research Brazil and University of Sao Paulo Polytechnic School, Department of Mechanical Engineering Sao Paulo, Brazil
kiran@usp.br

Flávio Celso Trigo
Flavius P. R. Martins
University of Sao Paulo Polytechnic School, Department of Mechanical Engineering Sao Paulo, Brazil
flavio.trigo@poli.usp.br, flavius.martins@poli.usp.br

Agenor de Toledo Fleury
Centro Universitário da Fundação de Ensino Inaciano
agfleury@fei.edu.br

Abstract. *Aiming at contributing to the development of a robust computer vision traffic surveillance system, in this work a method for vehicle identification and tracking that applies the Scale Invariant Feature Transform (SIFT) and a Kalman filter is proposed. The SIFT algorithm extracts keypoints of the moving object on a sequence of images and the Kalman Filter provides a priori estimates of vehicle position and velocity which are used to improve the said algorithm. This strategy allows reducing the amount of pixels to be tested for matches within the whole image scenario by dynamically redefining the ROI (Region of Interest). Using algorithms from OpenCV Library to compose the required computer vision tracking method, a prototype system was constructed and submitted to off-line experiments based on a series of grabbed traffic image sequences. From the results, it is possible to assert that the joint use of SIFT and Kalman filtering techniques is able to improve the overall algorithm performance concerning quality of matches between the images of the object and the scene, since it reduces in* 50% *the total number of false positives, one the main limitations of the pure SIFT algorithm.*

Keywords: *Vehicle tracking, SIFT, Kalman filter*

1. INTRODUCTION

Tracking salient objects, being people, vehicles or other objects in outdoor image frames is a good challenge within the Computer Vision research community. Moreover, it is also important to observe, detect and track vehicles within traffic surveillance systems in order to better operate the traffic itself, raise tickets, observe abnormal behavior or if one needs only to collect general information, like average speed, jams, collisions and other issues.

There are several approaches to perform object detection within the computer vision research. Some of those are widely used and applied in a variety of applications. For example, the Background Subtraction (Ballard and Brown, 1982; Forsyth and Ponce, 2002; Nixon and Aguado, 2008) is a very well known technique suitable to detect objects in a fairly static background. Because of its low computational costs and easy implementation, this technique is broadly used to segment salient objects. Mean-Shift, Optical Flow (Ballard and Brown, 1982; Bradski and Kaehler, 2008) and Lucas-Kanade (Bradski and Kaehler, 2008; Comaniciu *et al.*, 2003) are other instances of algorithms used for general object tracking.

It is also possible to recognize and track objects by applying feature extraction and matching algorithms. The SURF (Speeded Up Robust Features) algorithm (Bay *et al.*, 2008) claims to be a very robust algorithm for object detection, even after some image transformations. In fact, since Harris Corner Detector (Harris and Stephen, 1988) and Moravec Operator (Moravec, 1980), the studies on interesting keypoints detection became more developed and used within specific object detection algorithms. The work of Shi and Tomasi (1994) also shows an approach to detect keypoints with features more suitable for object tracking applications.

Under the context of searching for objects in sequences of images, video surveillance and camera monitoring, some new requirements must be attended in order to continuously detect objects along all frames. Another interesting approach is SIFT (Scale Invariant Feature Transform) developed by Lowe (1999, 2004) which presents considerable potential, due to its promised robustness for different scales and under low level of affine transformations (like rotation and warp).

The SIFT algorithm extracts keypoints that should to be matched against a template reference object along a sequence of image frames. The match process in every frame would provide itself the partial tracking process, but due to the distortions and transformations that are inbuilt in the video capture, the pure SIFT strategy reduces the matches rate while increase the number of false matches after a few frames.

Jiang *et al.* (2010) presents a SIFT-based strategy for tracking living cells in a videos captured from a differential interference contrast (DIC) microscopy. In the proposed approach, SIFT points are detected around live cells and a structure locality preservation (SLP) scheme using Laplacian Eigenmap is proposed to track the SIFT feature points along successive frames. By assuming that living cells cannot fly from one side to another in consecutive frames, matches with large displacement is filtered out as false matches. On the other hand, to reduce feature vectors dimensionality, the authors compared the wide-used PCA-SIFT against Laplacian Eigen Values, showing an improvement around 115% to 130% for the six different motion cells tested. The authors also claims that the improvements achieved by Eigen-Laplacian-SIFT approach was suitable for the DIC microscopy imagery due to motion characteristics of the living cells.

Rahman *et al.* (2009) suggest an approach to detect and track walking people also using SIFT. Every person entering in the image sequence is detected by applying the background subtraction technique, which is very suitable for real time applications but demands to collect previously one or more clean images of the background. It is also very sensible to luminance variations, depending on the database of backgrounds. On the other hand, for indoor and well controlled environments, background subtraction approach is widely used. The authors compared their results against manual calculations and showed error rates from 2% to 15% for the estimation of geometrical centers.

The project described in this paper presents an approach to track vehicles over outdoor image scenes, which are very susceptible to noise and distortions, by applying the SIFT algorithm (Lowe, 1999, 2004) associated with the Kalman Filter. The Kalman filter intends to improve both algorithm performance and SIFT matching results, since it is used to estimate the *a priori* position of the vehicle (x and y coordinates).The following sections give a brief overview of SIFT, Kalman Filter and finally the method used to solve and improve the vehicle tracking.

2. METHODS

2.1 SIFT algorithm overview

The SIFT algorithm developed by Lowe (1999, 2004) claims to be a robust technique for feature extraction which allows to perform object recognition in different image scenes. Such image scenes can be transformed in regard to their scale, rotation and warp distortions and the reference object can still be detected by matching the scale invariant features. This algorithm can be better described as a 4-steps script: keypoints detection, keypoints filtering, orientation assignment and finally, compute feature vectors to describe each keypoint.

2.1.1 Detecting keypoints in a scale-space

The generation of interest points can be done by the space scale filtering method, like described in Witkin (1983) and extended by Lindberg (1994). Such a method gives rise to a Gaussian pyramid (Forsith and Ponce, 2002) where the image layers, hereafter named "octaves", are constructed by successively reducing in 50 % the image scale and applying a Gaussian filter to the resultant scaled image. Lowe (Lowe, 2004) suggests to generate smoothed images for each octave, using Gaussian filters increasing progressively, in the interval $[\sigma, 2\sigma]$, where σ is the standard deviation of the first Gaussian mask considered.

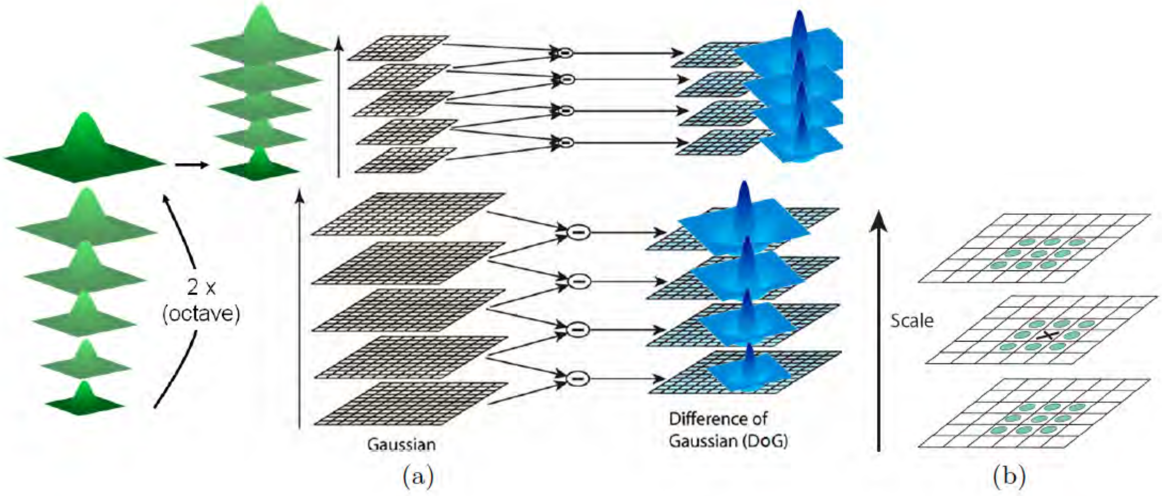

Figure 1: Gaussian Pyramid - smoothing and subsampling; (b) Selection of extrema from their 26 neighbors. Source: van den Linden *et al.* (2009)

As the Difference of Gaussian (DoG) function is a good estimative of the Laplacian of Gaussian (LoG) edge detector (Gonzalez and Woods, 2002; Lowe, 2004), a correspondent DoG pyramid is constructed by subtracting pairs of successive Gaussian pyramid image levels. The resulting images have only the borders of the objects, as shown in figure 2. After

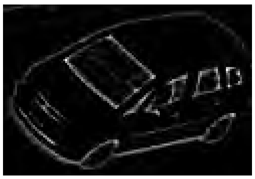

Figure 2: Image generated by a DoG filter

the DoG images are created (2), one must select the local maximum and minimum. These pixels detected are keypoints for further filtering. Hence, searching for local extrema points in all DoG image scales provides a set of possible interest points. Such points are identified by comparing the intensity of a scale-space point (x, y, σ) with that of the points in its $3 \times 3 \times 3$ (3D neighborhood), encompassing the eight neighbor points in the same scale image and the eighteen points in the neighbor scale planes (nine in the upper, nine in the lower). The positions of the extrema points in the scale-space must be determined, preferably, with sub-pixel and sub-scale precisions, as presented in Brown and Lowe (2002), where a quadratic function is fitted to the scale-space of the DoG function by expanding it in a Taylor series up to the quadratic term, *i.e.*,

$$D(X) \simeq D(X_p) + \frac{\partial D(X)}{\partial X}|_{X_p}(X - X_p) + \frac{1}{2}(X - X_p)^T \frac{\partial^2 D(X)}{\partial X^2}|_{X_p}(X - X_p) \tag{1}$$

where $X_p = [x_p \; y_p \; \sigma_p]^T$ is the reference point in the scale-space, $D(X_p)$ is the DoG value in the reference point, $X = [x \; y \; \sigma]^T$ is a point in the scale-space, and $D(X)$ is the DoG estimated value in X.

Therefore, the extrema points of the DoG function can be found with sub-pixel precision, through:

$$\begin{aligned} \frac{\partial D(X)}{\partial X} &= \frac{\partial}{\partial X}(\frac{\partial D(X)}{\partial X}|_{X_p}(X - X_p)) \\ \frac{\partial^2 D(X)}{\partial X^2}|_{X_p}(X - X_p) - \frac{\partial D(X)}{\partial X} &= 0 \\ X_{\text{extrema}} &= X_p + \left[\frac{\partial^2 D(X)}{\partial X^2}|_{X_p}\right]^{-1} \frac{\partial D(X)}{\partial X} \end{aligned} \tag{2}$$

2.1.2 Filtering out non-robust keypoints

Once the keypoints were collected, one must filter out pixels with low level of intensity since such pixels are more susceptible to the noise interference. Lowe (2004) suggests cutting of pixels with less than 0.03 of intensity.

Amongst the *extrema* points previously retained, those whose intensity is lower than a given threshold are simply eliminated and the remaining ones are submitted to a test to verify if they exhibit the characteristics of vertices. Following Lowe (2004), the eigenvalues of the Hessian matrix $H(x, y)$ are determined for each point (x, y) of the DoG filtered image $I(x, y)$ *i.e.*

$$H(x, y) = \begin{bmatrix} \frac{\partial^2 I}{\partial x^2} & \frac{\partial^2 I}{\partial x \partial y} \\ \frac{\partial^2 I}{\partial x \partial y} & \frac{\partial^2 I}{\partial y^2} \end{bmatrix} \tag{3}$$

and the pixels for which the ratio between the highest and the lowest eigenvalue is greater than 10, as suggested by Lowe (2004), are eliminated.

2.1.3 Assign dominant orientations for the keypoints

A rotational invariant descriptor can be constructed, by calculating the magnitude and orientation of the image gradient level around the position (x, y) of each point of interest according to eqs. 4 and 5.

$$m(x, y) \quad = \sqrt{\lambda^2 + \beta^2} \tag{4}$$

$$\theta(x, y) \quad = \tan^{-1}\left[\frac{\beta)}{\lambda}\right] \tag{5}$$

$$\text{with } \lambda = [G(x + 1, y) - G(x - 1, y)]$$
$$\text{and } \beta = [G(x, y + 1) - G(x, y - 1)]$$

To perform this task it is necessary to construct a histogram of orientations over a circular mask, centered in the interest point, in which a circular radius is given by:

$$r = k.2^{oct} \tag{6}$$

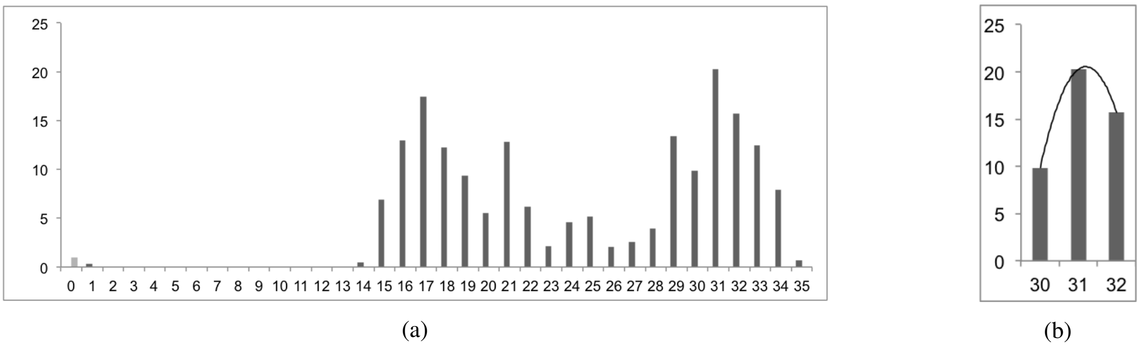

Figure 3: (a) Histogram of orientations. (b) Quadratic polynomial fitted around the histogram maxima.

where k is an empirical parameter and 2^{oct} is the image sub-sampling scale, determined by the octave number of the interest point. The number q of histogram intervals can be arbitrarily set up and the contribution of each orientation is weighted by its magnitude and by a Gaussian mask with standard deviation equal to 1.5 times the standard deviation of the DoG image where the interest point has been detected.

After interpolating the histogram with quadratic polynomial curves around their maximum points (figure 3a), the dominant orientation can be finally determined by searching those local maxima (figure 3b). The remaining keypoints, like shown in figure 4, should be the most robust ones and less susceptible to noise.

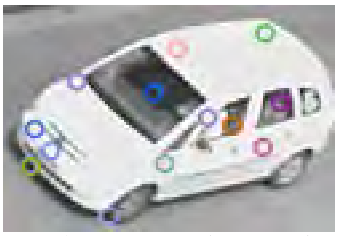

Figure 4: Keypoints detected for the reference object

2.1.4 Describe keypoints with feature vectors

The characteristic vector of each interest point is finally constructed as a set of histograms of orientation gradient vectors collected in its neighbor rectangular regions and weighted by a 2-dimensional Gaussian function, resulting from successively applying n 3×3 mean-filters. In other words, each component of the characteristic vector is a three-dimensional histogram defined at a point (R_x, R_y, θ) where (R_x, R_y) is the coordinate of the neighbor region and θ is the gradient orientation, assuming normalized values in the interval $[0, 1]$. In the example of figure 5, a 4×4 region matrix is generated around a reference point and the local gradients are grouped according to 8 intervals of orientations giving rise to a characteristic vector of 128 components $(= 4 \times 4 \times 8)$. Changes both in the number of regions and in the number of histogram intervals can affect the performance of SIFT algorithm.

2.2 Kalman Estimation

The identification of the object of interest in a straight manner by tracking each image frame is a deterministic procedure; however, grabbed images suffer from inaccuracies due to affine distortion, presence of artifacts such as occlusions and shadows, and sensor (camera) calibration errors which, ultimately, impair the whole identification process. This drawback could be eliminated once the inaccuracies were precisely quantified. Unfortunately, apart the eventual possibility of fine-tuning the sensing equipment, the other above-mentioned errors are difficult to quantify. A possible approach in such instances is to formulate the problem under a *stochastic*, instead of a deterministic point of view. It means that inaccuracies are treated generically as *noise* whose *statistics* are incorporated in the identification process. In this work, the Kalman Filter performs the task, as described in the sequel.

The first step in applying the technique is to tailor the problem in a state-space framework. It is assumed that the kinematics of any moving object in image scenes is constrained to a 2-D environment (Bradski and Kaehler, 2008), in which case the state is a vector $X \in \mathcal{R}^4$ containing both displacements and velocities of the geometric center of the object (see section 3.1for the definition). Once the state is established, it is necessary to define the so called *process* and *observation models*.

It is a fair assumption to consider the object to be tracked as a rigid body. This way, its configuration in a constrained two-dimensional space can be described by three independent generalized coordinates, namely, the Cartesian coordinates

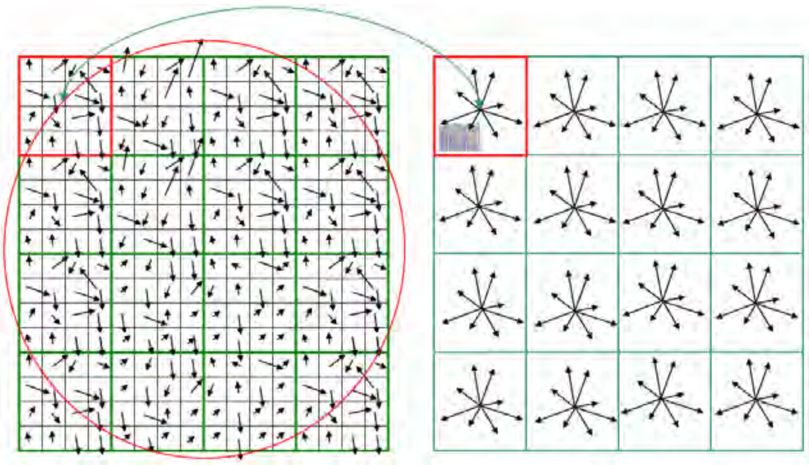

Figure 5: (left): Gradients around a reference point; (right): Histogram of regional gradient orientations of a figure caption (created by the author, based on Lowe (2004))

of its geometrical center, and its attitude about an axis orthogonal to the plane of the movement. However, for the purpose of the present work, only the coordinates of the geometrical center will be employed in describing the kinematics of the body. Moreover, since the time-step between successively grabbed images is small (the order of $1/30$ seconds), the velocity of the geometrical center will be considered constant. Under those hypotheses, it is possible to write the discrete-time state-space model as

$$X_k = FX_{k-1} + w_K \tag{7a}$$

$$Z_k = HX_{k-1} + v_k \tag{7b}$$

where 7a and 7b respectively represent the *process model* and the *observation model*, whereas

$F \in \mathcal{R}^{4x4}$ is the discrete-time state transition matrix;

$w_k \in \mathcal{R}^4 \sim \mathcal{N}(0, Q)$ is the process noise ;

$v_k \in \mathcal{R}^4 \sim \mathcal{N}(0, Q)$ is the measurement error;

$Z_k \in \mathcal{R}^2$ is the states collected from real measurements;

$H_k \in \mathcal{R}^{4x2}$ is the observation model matrix;

Matrices and vectors in eqs. 7a and 7b were defined as follows:

$$X_k = \begin{bmatrix} x \\ y \\ V_x \\ V_y \end{bmatrix} \quad F_k = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = 0, p(w_k)\ N(0, Q) \qquad H_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, p(v_k)\ N(0, R)$$

The state variables for the positions x and y are given in pixels, while the velocities are given in pixels/frame. Finally, it is assumed that process and measurement noise are white and independent of each other and uncorrelated, a consequence of the Gaussian hypothesis. Within this framework, both modeling and measurement disturbances can be accounted for by the estimation algorithm.

Kalman filters are Markovian observers that seek to provide the best estimates of the state in a stochastic least-squares sense, in two stages. On the *propagation* stage, based on noise corrupted observations of the coordinates of the geometrical center of the object (the keypoints matched by the SIFT algorithm) at an instant $k-1$ and on the assumed time-evolution model of the state, the Kalman filter computes optimal estimates of the state $\widehat{x}_k^-$ and of its error covariance matrix, P_k^-, before the arrival of new measurement data. When new observations are available at instant k, new estimates of the state, $\widehat{x}_k^+$, and of the error covariance matrix, P_k^+, are calculated considering a correction term, K_k, the Kalman gain, during the so called *update* stage. Thus, the complete framework of the Kalman predictor-corrector procedure can be summarized by the following equations, according to the terminology from Gelb (1974):

$$\widehat{x}_k^- = A\widehat{x}_{k-1}^- + B\widehat{u}_{k-1}^- \tag{8}$$

$$P_k^- = AP_{k-1}A^T + Q \tag{9}$$

Then, collect the current measurements and compute the update (or correction) equations:

$$K_k = P_k^- H^T(HP_k^- H^T + R)^{-1} \tag{10}$$

$$\widehat{x}_k = \widehat{x}_k^- + K_k(Z_k - H\widehat{x}_k^-) \tag{11}$$

$$P_k = (I - K_k H)P_k^- \tag{12}$$

The ensuing values were admitted as initial conditions:

$$X_0 = \begin{bmatrix} 100 \\ 100 \\ -3 \\ -3 \end{bmatrix}, Q = \begin{bmatrix} 10^{-1} & 0 & 0 & 0 \\ 0 & 10^{-1} & 0 & 0 \\ 0 & 0 & 10^{-1} & 0 \\ 0 & 0 & 0 & 10^{-1} \end{bmatrix}, R = \begin{bmatrix} 10^5 & 0 & 0 & 0 \\ 0 & 10^5 & 0 & 0 \\ 0 & 0 & 10^5 & 0 \\ 0 & 0 & 0 & 10^5 \end{bmatrix}$$

It should be noticed that the values $x = y = 100$, $V_x = V_y = -3$ as initial conditions are only arbitrary values, with the particular characteristic of positioning the initial coordinates close to right side of image scene, according to the street lanes and direction of movement and considering it a "one way" street. These empiric values affect only the number of frames needed for the Kalman estimations to converge. Therefore, they should be selected according to applications.

3. RESULTS AND DISCUSSION

A set of 80 frames from an upper view surveillance camera was selected (refer to the figures 7 and 9 for examples of the image set and camera placement). The reference object (a vehicle) was extracted manually within one of these frames in a way that the object would be well represented across the images sequence. The proposed technique is not restricted to one specific viewpoint, but the initial conditions for Kalman equations would affect the number of frames to converge (section 2.2).

Due to the inherent transformations in image process capture, the object is subject to affine distortion (rotation and scale) which would affect the performance of the SIFT detector. In order to understand how the Kalman filter may help the tracking process, two experiments were conducted: (a) to detect and track the object applying only the SIFT algorithm over all image frames and; (b) to detect and track the object applying the SIFT algorithm helped by a Kalman filtering.

3.1 SIFT-only tracking

As described in the section 2.1 the first step is to compute the object descriptors in form of feature space. Each selected keypoint is described by a feature vector. The same task must be executed for both the scene and object images, *i.e.*

$X = \{X_1, X_2, X_3, ..., X_n\}, n =$ number of keypoints detected for the object image
$Y = \{Y_1, Y_2, Y_3, ..., Y_m\}, m =$ number of keypoints detected for the scene image

Being $X_i, i = 1...n$ the feature vectors of the object keypoint and $Y_j, j = 1...m$ the feature vectors of the scene keypoints,

$$X_i = [x_1, x_2, x_3..., x_{128}]$$
$$Y_j = [y_1, y_2, y_3..., y_{128}]$$

then the Euclidean distance between each pair of feature vector must be computed as follows:

$$D(X, Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_{128} - y_{128})^2} \tag{15}$$

The minimum distance between a pair of feature vectors, each one representing a keypoint, would indicate a possible match:

$$Match(X, Y) = argmin_{X,Y}\{D(X_i, Y_j)\} \qquad i = 1...n, j = 1...m \tag{16}$$

In this project, the SIFT algorithm provided 17 keypoints, meaning 17 matches. According to Lowe (2004), at least 3 matches are required in order to identify an object match. For the purpose of avoiding false positives, the first 5 matches were selected, thus looking for the most "robust" ones. Still, in some frames there remained false positives which had to be discarded. A second constraint was to limit the area of the keypoints to be equal or less the area of keypoints detected in the object image.

$$rectangle(Keypoints_{scene}) \leq rectangle(Keypoints_{object}) \tag{17}$$

It is assumed that the position of the object is represented by its geometrical center corresponding to the best matched keypoint which, in turn, is the pair of object/scene feature vector bearing the smaller distance. This procedure is iteratively applied throughout all frames, and the results are shown in figure 6.

For the SIFT-only approach, there is no ROI defined; as a consequence, the SIFT matching process has to be performed against keypoints collected from the whole image scene. However, as it shown by the frame in 7, an example of the "object lost" outcome during a tracking sequence, despite some good matches, the false ones jeopardize object recognition. Then, it is possible to conclude that the fewer keypoints in the background, the better SIFT-algorithm performance.

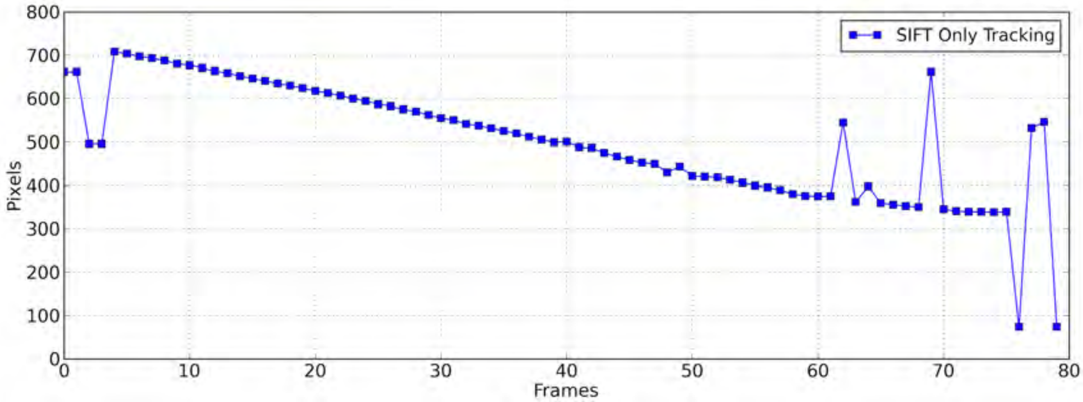

Figure 6: SIFT tracking - magnitude of displacement vector in pixels

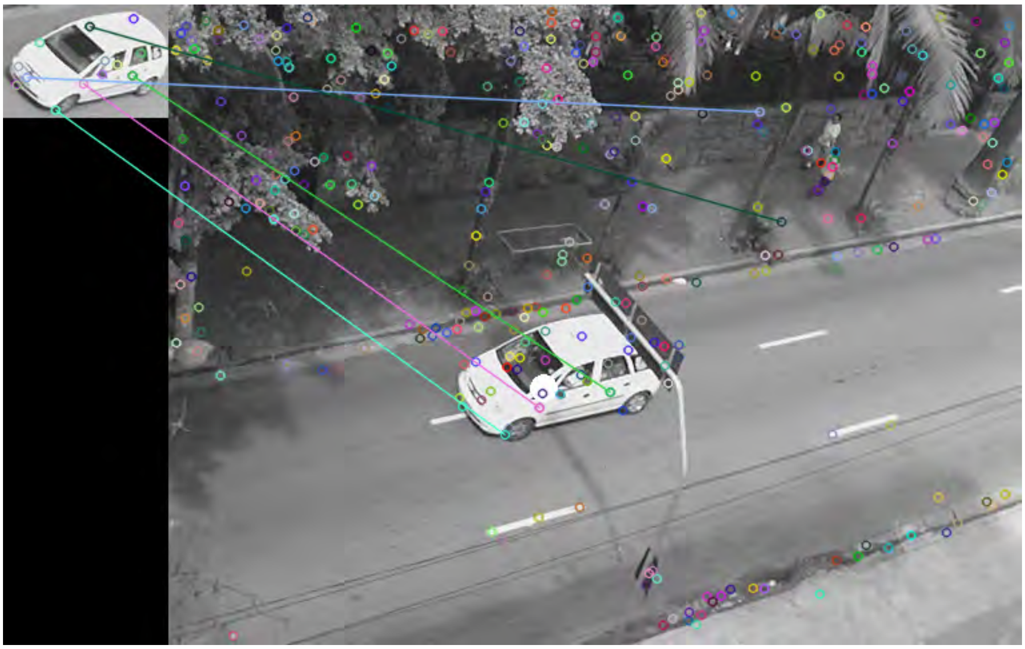

Figure 7: Tracking the vehicle with SIFT algorithm only

For the SIFT-only approach, there is no ROI defined, incurring the SIFT matching process to be performed against all the keypoints collected from the whole image scene. By observing such situation, one can assume that the SIFT would perform better if there is no (or less) keypoints in the background when matching against the vehicle keypoints. The figure 7 shows that despite some good matches, the false matches jeopardize the object recognition. The frame shown in figure 7 is one example of the "object lost" during the tracking sequence.

3.2 SIFT-Kalman Tracking

Under the hypothesis the SIFT tracking can be improved by Kalman estimation, the latter is used to compute *a priori* estimates of the vehicle state (position and speed), that are, then, used to reduce the ROI (Region of Interest) within the image scene, therefore decreasing the computational burden necessary to find the matches. The size of ROI rectangle was empirically set to double of the object size (figure 4), centered in the position estimated by the Kalman Filter. This size should be fine tuned to accommodate the errors in estimation and also to allow abrupt changes in the displacement direction.

The thicker rectangle in figure 9 suggests that the object was successfully found while the thinner rectangle indicates the ROI. One can note that only the keypoints inside the ROI was used to perform the match subprocess.

It can be seen from figure 8 that the number of keypoints reduced dramatically when the Kalman estimates converged to the object position. The criterion to consider that the Kalman estimatives finally converged is based on the distance

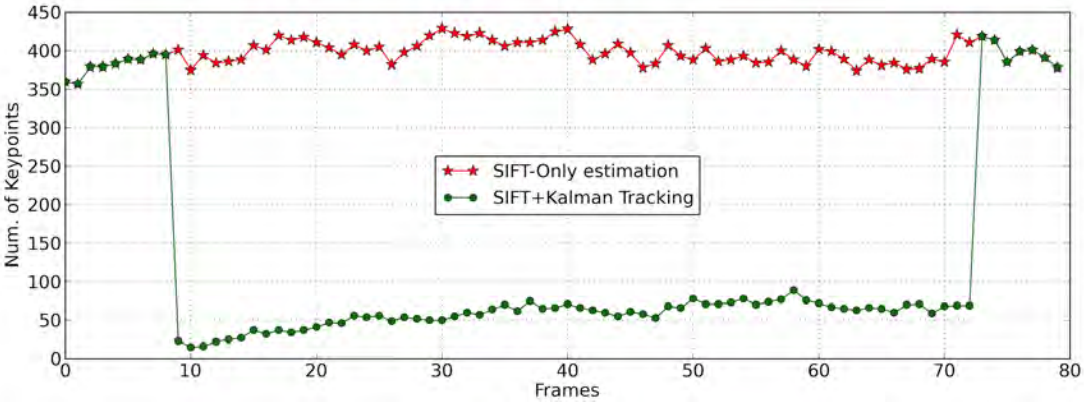

Figure 8: Number of keypoints - whole image and ROI

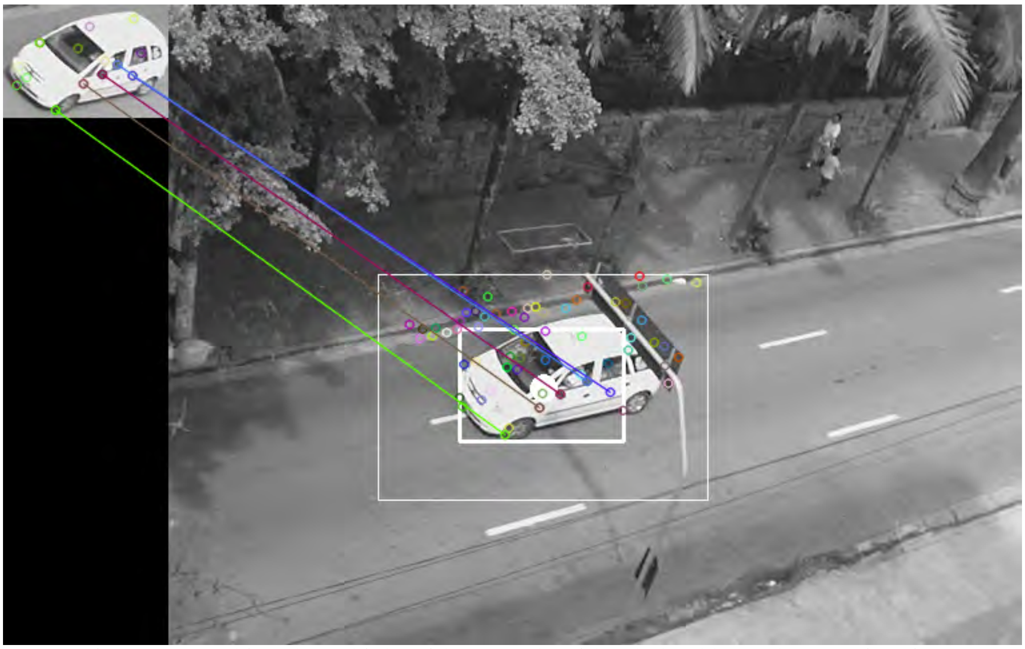

Figure 9: Tracking the vehicle with SIFT algorithm helped by Kalman estimation

between the "Kalman estimated position" and the "detected geometrical center" , as follows:

$$err = \sqrt{dx^2 + dy^2} \tag{18a}$$

$$diag = \sqrt{w_{ROI}^2 + h_{ROI}^2} \tag{18b}$$

where

dx and dy are the distance between estimated Kalman coordinates and the detected geometrical center, for x and y respectively and;

w_{ROI} and h_{ROI} are respectively width and height of the rectangle built for Region of Interest and $diag$ is the length of ROI diagonal.

When $error \leq diagonal$ from equations (18), this $error$ can be neglected and the ROI is applied to the subsequent frame. Experiments have shown that the Kalman estimation in supporting the SIFT object tracking brings some considerable advantages, such as the reduction of false positives, as depicted in figure 10. While the SIFT-only approach gives 120 false matches (keypoint matches), the Kalman-aided tracking algorithm reduced the amount of pixels to process and also the number of errors caused by keypoints located on the background scene.

The Kalman estimation was applied in two different ways. In the first one, the measurement noise and process noise covariance matrices were kept constant as initially set, thus forcing the estimation towards the model dynamics, since there is no guarantee that measurements are reliable. When the SIFT algorithm detects an object match, the above-mentioned covariance matrices are respectively adjusted to values $1.0e^{-3}I_4$ and $1.0e^{-2}I_4$ (I_n is the identity matrix of order n) in order to continue the estimation procedure. The result of such experiment is shown in figure 11.

In the second experiment, the Kalman filter algorithm was run updating the measurement noise error covariance matrix R at every iteration, based on new measurement data from grabbed images. The result is shown in figure 12.

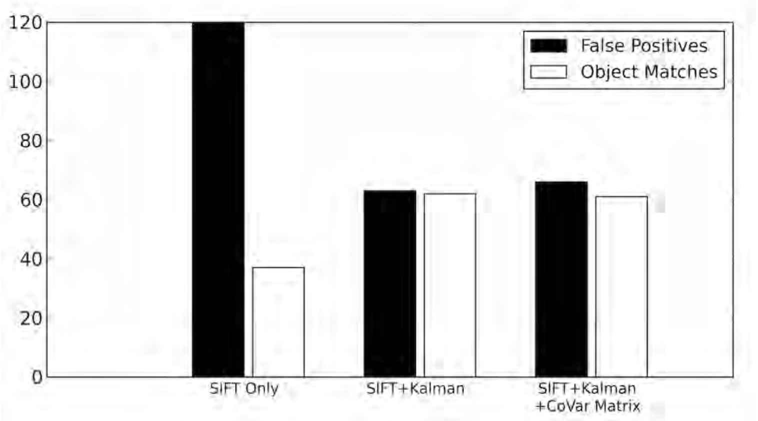

Figure 10: False positives and object matches

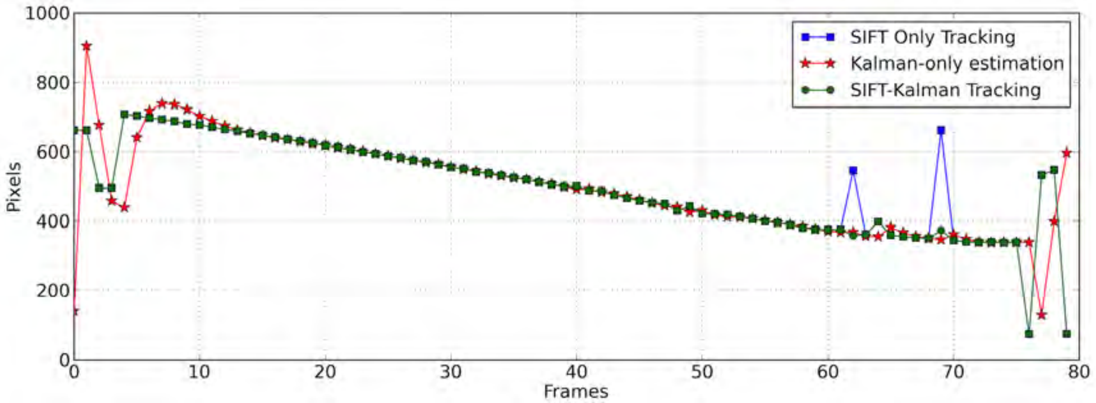

Figure 11: SIFT-Kalman tracking and Kalman-only estimation - fixed error and covariance matrices (magnitude of displacement vector in pixels)

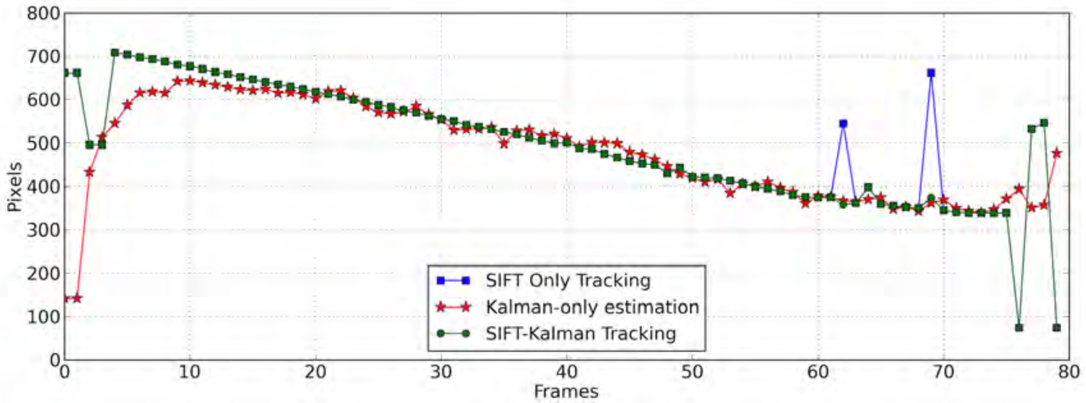

Figure 12: SIFT-Kalman tracking and Kalman-only estimation - updating error covariance matrices (magnitude of displacement vector in pixels)

From figures 11 and 12, one realizes that, no matter the measurement error covariance update, estimates obtained with the aid of the Kalman filter avoids loosing track of the object between frames 60 and 70, which represents another advantage over the SIFT-only method. It should be noticed that the poor results obtained by all methods after around frame 75 is a consequence of the tracked object leaving the scene.

Concerning figures 11 and 12 it is important to mention that when updates are performed, the convergence time is slightly higher; nevertheless, it does not represent a drawback, since the overshooting that occurs in the first experiment is avoided. In practice, this means that if one is not able to *ad-hoc* fine tune the covariance matrices, the algorithm itself (SIFT-Kalman with R matrix update) is robust enough to provide good estimates of the state from a totally arbitrary initial condition. Moreover, the amount of computer work is decreased from about $O(n^2) = 400^2 = 16e^4$ to about

$O((n/16)^2) = 625$ cycles to obtain the matches.

4. CONCLUSIONS

Despite of the claimed robustness of SIFT algorithm to track objects over outdoor image scenes, it still suffers from degradation which reduces considerably the matching performance, and may cause the object loss. In this work, by using Kalman Filter to estimate the *a priori* states, it was shown that one could considerably improve the performance of SIFT algorithm, since the number of keypoints is reduced 16 times and, as a consequence, the number of false matches is reduced by about 50%. This improvement was achieved because, by knowing previously the most likely position of the object in the subsequent frames, it was possible to redefine a ROI inside the whole pixel matrix of the image scene. Therefore, from the above rationale, it can be asserted that the initial objectives are attained.

5. ACKNOWLEDGEMENTS

First author acknowledges partial support from IBM Research - Brazil.

6. REFERENCES

Ballard, D.H. and Brown, C.M., 1982. *Computer Vision*. Prentice-Hall, Englewood Cliffs, NJ.

Bay, H., Ess, A., Tuytelaars, T. and Gool, L.V., 2008. “Surf: Speeded up robust features”. *Computer Vision and Image Understanding (CVIU)*, Vol. 110, pp. 346–359.

Bradski, G.R. and Kaehler, A., 2008. *Learning OpenCV - computer vision with the OpenCV library: software that sees*. O’Reilly. ISBN 978-0-596-51613-0.

Brown, M. and Lowe, D.G., 2002. “Invariant features from interest points groups”. In *BMVC 2002, Proceedings of the British Machine Vision Conference*. British Machine Vision Association, Cardiff, UK.

Comaniciu, D., Ramesh, V. and Meer, P., 2003. “Kernel-based object tracking”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 5, pp. 564–577. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1195991`.

Forsith, D. and Ponce, J., 2002. *Computer Vision: A Modern Approach*. Prentice Hall.

Forsyth, D.A. and Ponce, J., 2002. *Computer Vision: A Modern Approach*. Prentice Hall Professional Technical Reference. ISBN 0130851981.

Gelb, A., 1974. *Applied optimal estimation*. MIT Press.

Gonzalez, R. and Woods, R.E., 2002. *Digital Image Processing*. Prentice Hall.

Harris, C. and Stephen, M., 1988. “A combined corner and edge detector”. In *AVC 1988, Proceedings of the 47h Alvey Vision Vision Conference*. University of Manchester, Manchester, UK, pp. 147 – 151.

Jiang, R., Crookes, D., Luo, N. and Davidson, M., 2010. “Live-cell tracking using sift features in dic microscopic videos”. *Biomedical Engineering, IEEE Transactions on*, Vol. 57, No. 9, pp. 2219–2228. ISSN 0018-9294. doi:10.1109/TBME.2010.2045376.

Lindberg, T., 1994. “Scale-space theory: a basic tool for analysing structures at differnet scales”. *Journal of Applied Statistics*, Vol. 21, No. 2, pp. 225 – 270.

Lowe, D.G., 1999. “Object recognition from local scale invariant features”. In *ICCV 1999, Proceedings of the 7th International Conference on Computer Visio*. IEEE, Kerkira, Greece.

Lowe, D.G., 2004. “Distinct image features from scale invariant keypoints”. *International Journal of Computer Vision*, Vol. 2, No. 60, pp. 59 – 67.

Moravec, H.P., 1980. *Obstacle avoidance and navigation in the real world by a seeing robot rover*. Ph.D. thesis, Stanford University, Stanford, CA, USA.

Nixon, M. and Aguado, A., 2008. *Feature Extraction & Image Processing*. Academic Press. ISBN 9780123725387. URL `http://books.google.com.br/books?id=jXmJqzQgdY8C`.

Rahman, M., Saha, A. and Khanum, S., 2009. “Multi-object tracking in video sequences based on background subtraction and sift feature matching”. In *Computer Sciences and Convergence Information Technology, 2009. ICCIT ’09. Fourth International Conference on*. pp. 457–462. doi:10.1109/ICCIT.2009.164.

Shi, J. and Tomasi, C., 1994. “Good features to track”. In *1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR’94)*. pp. 593 – 600.

van den Linden, R., Neerbos, J. and Havinga, T., 2009. “Distinctive image features from scale-invariant keypoints”. Technical report, University of Groningen. URL `http://svn.assembla.com/svn/rob_ai/articles/essay_SIFT.pdf`.

Witkin, A.P., 1983. “Scale-space filtering”. In *IJCAI 83 - Proceedings of the 8th International Joint Conference on Artificial Intelligence*. University of Manchester, Karlsruhe, Germany, pp. 1019 – 1022.

7. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.