

Improved Mean Shift Tracking Algorithm by Multi Scale Motion Information

Sallama Resen , Hala Bahjat

¹(PhD. Student Sallama Athab Northampton university,)
²(Technology university, head Information System branch)

Abstract: - Tracking object algorithms in moving platform environment are a challenging mission because they treat variety of speeds for moving object problems. Common method is Mean Shift Algorithm (MSA) due to its simple and efficient procedure. However, the lack of dynamic window size scale in its target model makes it unsuitable for tracking objects in real applications system. Dual static cameras placed in front and rear moving platform gathered information surrounding moving platform. Since the scale of the targets diverse irregularly there are needs to adjusted bandwidth scale when the appearances of tracking object changes. In this paper an automatic window-size updating method based mean shift with motion information obtained from Optical Flow (OF). Objects move with variety of speeds and directions surrounding moving platform .The proposed module speed and direction of moving object join together as factor updating scale according to how much moving object near or far from camera. Proposed framework initialized by MSA followed by Multi Scale Motion Measurement (MSMM). Experimental results show that scale update could select the proper window size of tracking region depending on speed of moving object. Evaluate performance of MSMM compare with traditional MSA by mismatch ratio and time complexity.

Keywords: - mean shift algorithm, moving platform, mismatch ratio, optical flow

I. INTRODUCTION

Object tracking is major subject in computer vision applications such as traffic road management, surveillance and auto driver [5-24] . Tracking moving object is one of the most challenging topics in dynamic sight due to the present problems such as occlusions, clutters with background. Environment changes in the objects' shape, size or diversity moving objects speed. A large number of algorithms achieve efficient visual object tracking algorithm, such as mean-shift, kalman filter and particle filters [1-4]. common method is (MS) Mean shift algorithm due to its simple and efficient procedure with suitable accuracies [25]. Mean shift is estimator of nonparametric density function by computes the nearest mode iteratively [26]. original mean shift tracking algorithm limitation is constant target model while object appearance change during tracking. In practical experience appearance of target will be changed and if modeling of target be constant, certainly tracker will loss object. These limitations caused by inaccuracy of tracker's bandwidth or constant target model during tracking. Developed methods try to solve this problem [42-49].

Window size of object tracking would affected by the accuracy of target matching and position, which consequently determines the whole tracking precision. Background confounded with object tracker when window size is too large. However, too small window size make not fully obtained feature of the tracked object causing target lost. In this paper, adaptive scale introduced into object tracking systems to deal with the problem of window size selection and updating. The displacement obtain from optical flow reflected the relation between object appearance and scale. The scale changes of the moving target in video can be regarded as the distance changes between sequence frames in real scene. Therefore motion information obtained from OF used to determine the size changes of the tracking window. The main contribution of the paper is that we have suggested a new definition for updating window size scale related moving object speed and then integrated the proposed method into the mean-shift frameworks. Mismatch Ratio used for comparison between MSMM and MSA. The rest of this paper is as follows: In Section II discusses tracking algorithm update scale development, in section III classical Mean-Shift Algorithm introduced. In section IV. Optical Flow algorithm described, in Section V, proposed algorithm MSMM, experimental tracking results will be showed in Section VI. Lastly, we give a conclusion in Section VII.

II. LITERATURE REVIEW

Mean-Shift Algorithm ((MSA) [4] is used in the Computer Vision field for clustering, tracking and probability density estimation. Tracking achieve in MSA by shift data to minimum distance between target and candidate Probability Density Functions (PDFs).

Traditional MSA in [55] assumed that the object scale is constant during tracking, in experience it is unusual to detect objects whose scale and orientation constantly unchanged. If the object becomes smaller, the kernel window includes background clutter which often leads to tracking failure. In [3] proposed to change the window size by a constant factor (10%) and to run the algorithm multiple times. The scale maximizing the similarity to the target model was chosen. The approach does not cope well with increases of object size since the smaller windows usually have higher similarity and therefore the scale is often underestimated but when scale exceeds window targets and become larger then it cannot work well because window converges to the local maximum in a smaller searching window [4]. In [5] proposed method is based on the theorem that the changes of the target scale and target position within the kernel will not impact the tracking accuracy of the Mean Shift tracking algorithm. However, this method cannot be employed when the scale change of a target becomes smaller. Pu et al. [12] using Mean-Shift to track object estimate translation in forward and backward direction. Produced methods are based on invariant moments to address scale update problem. Yang et al. [14] introduced a new similarity measure which leads to scale estimation by comparison of second moments of target model and target candidate. In [6], image moments are used in [1, 10] also.

Second order moment consuming time calculation, it is not appropriate for real-time application. Another improvement direction present in [8–10]. Affine projection is also introduced to solve the problem of scale adaptation However, simply introducing affine Transformation in these methods cannot get a satisfied scale adaptation. [16] Exploit affine structure to recover object relative scale from feature point correspondences between consecutive frames the interest point matching in these methods is not efficient to obtain correct affine structures. [8] search for the object boundary by correlating image with four templates. Positions of the boundaries directly determine the scale of the object. Estimating scale from feature correspondences takes times and has difficulties dealing with a non-rigid or a deformable object.

Update the tracking bandwidth required when the size of the tracked objects changes due to the camera or object motion. When an object move slowly becomes larger cause poor localization since not all pixels belonging to the object are included in the search window .

We present adaptive scale estimation mechanism which employ motion information of moving object as a scale factor, built simply on the MSA procedure for target modeling and OF for motion information. Present scale adaptation by a Multi-Scale Information Measurement (MSIIM) object speed between three sequence frames was scale by the MSIIM to adjustment bandwidth which consider as the scale estimation.

III. MEAN-SHIFT TRACKING: AN OVERVIEW

In this section we introduce the Mean Shift Tracking Algorithm described in [26].The Mean Shift tracker works by searching in each frame for the location of an image region whose color histogram is closest to the reference color histogram of the target [27]. Features and representation of target model, target candidates and geometric relations between target models and candidates region are the main subject of the MSA [25].

Object modeling using color probability density, q object model probability of color u , $u = 1 \dots m$ is

$$q_u = C \sum_{i=1}^{n_x} k(\|x_i\|)^2 \delta(b(x_i) - u) \quad (1)$$

C is the normalization constant

$$C = (\sum_{i=1}^{n_x} k(\|x_i\|)^2)^{-1} \quad (2)$$

σ is the Kronecker delta function

$$\delta(a) = \begin{cases} 1 & \text{if } a = 0 \\ 0 & \text{otherwise} \end{cases}$$

The probability p of color u in the target candidate is

$$p_u = c_h \sum_{i=1}^{n_h} k(\|\frac{y-y_i}{h}\|)^2 \delta(b(y_i) - u) \quad (3)$$

Use mean shift to estimate probability and target location. Track target candidate in video by matching probability of target model with candidate neighbor. Measure similarity between probability object model q and target p at new location y use Bhattacharyya coefficient ρ .

Let z denote estimated new target location near y , and color probability does not change significantly.

$$\rho(p(z), q) = \frac{1}{2} \sum_{u=1}^m \sqrt{p_u(y)q_u} + \frac{C_h}{2} \sum_{i=1}^{n_h} w_i (\|\frac{z-y_i}{h}\|)^2 \quad (4)$$

Where weight w_i is $w_i = \sum_{u=1}^m \delta(b(y_i) - u) \sqrt{\frac{q_u}{p_u(y_i)}}$ $\delta[b(x_i) - u]$ (5)

z new location of candidate target eq.4

$$z = \frac{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{y - y_i}{h}\right\|\right)^2 y_i}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{y - y_i}{h}\right\|\right)^2} \quad (6)$$

Mean Shift Tracking Algorithm step:

- (1) Calculate target model by eq.(1)
- (2) Initialize y_0 location of candidate target in previous frame.
- (3) Calculate candidate model by eq. (3) in current frame.
a window of y_i pixels is related to h size of window
- (4) Compute weights w_i according to eq.5.
- (5) Apply mean shift: Compute new location z of target candidate eq.6
- (5) Compute $\{p_u(z)\}$, $u = 1, \dots, m$, and $\rho(p(z), q)$ eq.5.
- (6) While $\rho(p(z), q) < \rho(p(y), q)$, do $z \leftarrow 1$ This step used to validate the target's new location.
To adapt to change of scale, can modify window size h and let algorithm converge to h that maximizes $\rho(p(y, q))$.
- (7) If $z_k - y_k$ is small enough, stop.
Else,
set $y \leftarrow z$ and go to Step 1.

IV. OPTICAL FLOW

One of basic operation in video analyses is Optical Flow used in computer vision application such as segmentation, structure motion and compression figure(1). OF algorithms consider intensity change in 2D for image pixels caused by motion so can calculate displacement between at least two sequence frames.

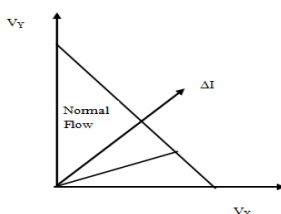


Figure 1 -optical constrict line

The idea of brightness conservation and optical flow were first proposed by Fennema [3].

$$I(x, y, t) = I(x + dx, y + dy, t + dt)$$

Where I is an image sequence, $[dx, dy]$ is the displacement vector for the pixel at coordinate $[x, y]$ and t and Δt are the frames at time t and $t+1$ displacement of the image sequence. Find approximation derived by Taylor expansion.

$$\frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t} = 0$$

Two unknown variable with one equation . Horn and Schunck [7] using differential technique technique has been used to calculate the optical flow.

Horn and Shunck were the first to impose a global smoothness constraint, assuming the flow to be smooth across the image. Their minimization function is:

$$\int \int (I_x u + I_y v + I_t)^2 + \lambda^2 (\|\nabla u\|_2^2 + \|\nabla v\|_2^2) dx dy$$

Where λ reflects the influence of the smoothness term ,can be expressed as a pair of Gauss-Siedel iterative equations.

$$u_{n+1} = u_n - \frac{I_x [I_x u_n + I_y v_n + I_t]}{\alpha^2 + I_x^2 + I_y^2}$$

$$v_{n+1} = v_n - \frac{I_y [I_x u_n + I_y v_n + I_t]}{\alpha^2 + I_x^2 + I_y^2}$$

The estimated motion by Horn & Shunck algorithm is dense and precise, because it is calculated for each pixel in the image. However, the use of the temporal intensity conservation and the spatial motion coherence assumptions make it more adapted for the small displacements than for the long displacements. Motion criteria is reinforced by the connectivity criteria in order to localize each moving object in the scene, and to find the number of objects moving with the same apparent velocity.

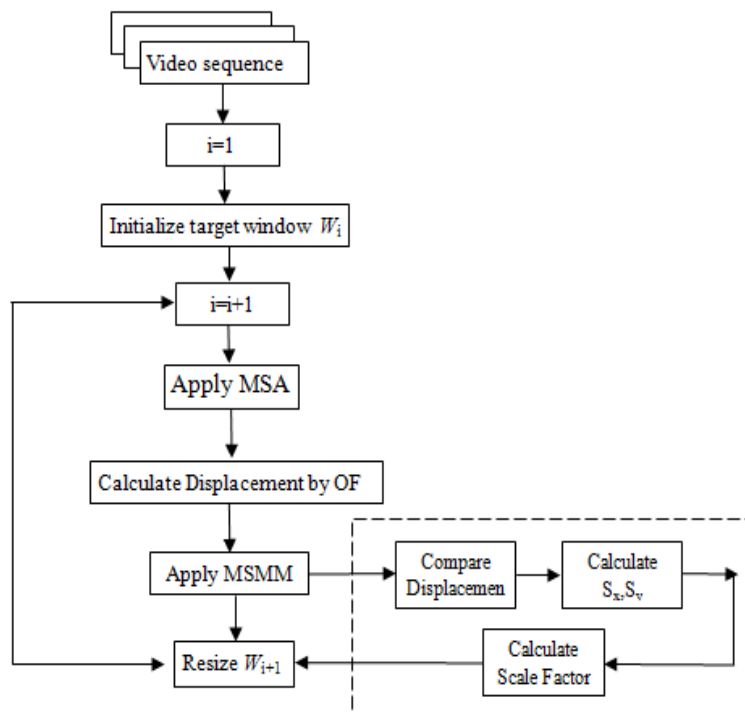
V. MSMM ALGORITHM

Mean shift algorithm assumed that the object scale during the course of tracking is constant. It is not true to detect objects with complex shapes whose scale and orientation constantly change due to the platform and object motions.

Dual static cameras placed in front and rear moving platform gathered information which is used to tracking objects .objects move with different speed and direction surround moving platform. In this paper, we propose framework allows update window size scale depending on the speed of the platform and objects movement by integrate displacement measurement with appearance changing.

Speed of moving object related with size a scale, i.e., the closer one to camera sees bigger to observe while far one become smaller .proposed method being able to measure and distinguish the object’s scale related to speed.

Displacement moving object obtained from OF, employs scaling factors are applied to improve the MSA tracking performance by MSMM algorithm Figure (2).



MSMM algorithm Figure (2)

The scaling factors are update based on the speed value which considers scale adaption with respect to appearance change. Tracking algorithm adjusts the tracking bandwidth when the tracked objects move far away or near moving platform.

When object moving speed higher than platform speed become far away from the camera and it seems smaller to observer scale window size decreases until object disappear from the scene. In the same way if the object speed is less than platform speed scale window size become larger .window size unchanged if object and moving platform move at the same speed .

Here we present a Multi-Scale Motion Measurement (MSMM) to achieve the goal. If we set the distance d between moving object and moving platform as a scale factor. High speed moving object increase average difference displacement (D_x, D_y) and object seem smaller in this case window size become smaller according to *value*.

While window size become larger if object moving in slower speed. Unchangeable window size when both of objects, platform movement speed are equal.

MSMM process stage explains as follow:

- 1- Find Optical flow for moving objects in three sequence frames eq.7
- 2- Calculate average difference displacement in X, Y direction from eq.

$$D_x = ((I_{x3} - I_{x2}) + (I_{x2} - I_{x1}))/2 \quad (7)$$

$$D_y = ((I_{y3} - I_{y2}) + (I_{y2} - I_{y1}))/2 \quad (8)$$

- 3- Update window size of the tracking object according to D_x, D_y value.

Here the scale changing is dependent on object displacement that determined by optical flow. on the current tracking window, then multiply a factor of average difference displacement

If $D_x > E_x$ THEN $S_x = S_x * (1 + D_x)$

If $D_y > E_y$ THEN $S_y = S_y * (1 + D_y)$

If $D_x < E_x$ THEN $S_x = S_x * (1 - D_x)$

If $D_y < E_y$ THEN $S_y = S_y * (1 - D_y)$

If $D_x = E_x$ THEN $S_x = S_x$

If $D_y = E_y$ THEN $S_y = S_y$

$$S_{n+1} = S_n * \sqrt{\frac{S_y}{S_x}} \quad (9)$$

α constant, S_x, S_y high, width scale change. In proposed method, accuracy in target modeling and adaptive bandwidth allows to have the target model adaptation with variable appearance of the target. Our proposed method solves the mentioned problem of the original algorithms.

- (1) Choose the initial location of the search window.
- (2) Perform Mean Shift Algorithm on current frame.
- (3) Repeat Steps 2 until convergence condition.
- (4) Average difference displacement (D_x, D_y).by eq.7.eq.8 .
- (5) Update search window size by eq.9
- (6) Get new frame.

VI. EXPERIMENT RESULTS

This section evaluates the proposed MSMM scale update algorithm in comparison with the MSA algorithm. Results examined on real videos with variable platform (using normal car as the moving platform mentioned before in experiments to test adaptive scale related with speed). The target moves far away from moving platform then tracking region scale from large to the small size and object speeds. Front, rear camera placed inside car in the middle of it is longitudinal axis's .dual camera has same properties which are Number of Pixels Approx. 18.9 Megapixels, Dimension (W x H x D) =121.6x86.6x93.3 mm, Focal length 35 mm equivalents, Image Sensor type (7.76mm) Exmore R CMOS sensor, Video Out type HD. speeds of car are 20-60 mile in hour .Video length 25 sec, resolution 320x240 and frame rate is 25fps . video1, video2 acquired by front, and rear camera synchronization with 30 m/h platform speed.Video3, video4 capture from moving platform in motorway 60m/h car speed. Proposed method compared with MSA. Fig.3 (a, b) show result for low speed moving object video1, video2 capture by front rear camera respectively while Fig. 4 (a, b) show result for high speed moving object video3, video4 capture by front, rear camera respectively . Method runs in with 8GB multi processor core TM-i7 with software environment MATLAB.

Object seems bigger when objects slow movement because it become near from camera. Fig.2 (a) Fig.2 (b) blue rectangle and Fig.3 (a) Fig.3 (b) yellow rectangle .Visa versa in height speed Fig.4 (a) Fig.4 (b) red rectangle and Fig.5 (a) Fig.5 (b) green rectangle. Since window size processed by MSMM changes according to how much near or far from camera this cause different processing time while traditional MSTA may process large window size because of initialized even when object seems small so it is clear processing time in MSMM

less than MSTA . There are three parts in MSMM algorithm cost time computational which are: calculation of the Mean Shift tracking, motion information, and the MSMM computation. Also target tracking size is related to the time cost .The bigger size is cost more time. The time cost is less than 55ms when the number of the pixels in the target is less than 10 X 45 pixels table (1) mismatch ratio for MSMM and MSA.

Table-1- mismatch ratio

Frame no.	MSMM	MSA
1	0.01	0.01
25	0.09	0.16
50	0.19	0.32
75	0.27	0.55
100	0.16	0.76

Proposed method can handle with quick response to variety of video sequence .In experiment result achieved very encouraging results to real time applications about 15 frames per second. Using motion information is more accurate to matching than the statistic feature because it avoid color histogram mismatch problem, which occur in MS. The tracking window will be larger or smaller depend on how much the target far from camera.

The scales of the tracking windows are incorrect when the spatial color Mean-Shift tracking Algorithm is used to track the high speed car. Furthermore, the errors become larger when the target becomes smaller. Our MSMM algorithm can adapt to the scale changing very well. The scale of the moving platform in Figure 3,4 changes frequently. Because the video is taken by the camera which follows the moving platform, the scale of the moving platform changes from the large to small size or vice versa. MSMM performs extremely well and has the lowest number of errors. Tracking algorithms have more errors after 20 frames, and there are more and more errors as the target moves away from camera. Because the true target scale changes suddenly from the large size to the small size, the cumulative errors of the Mean Shift increase.



Figure (3-a) video1 MSMM



Figure (3-b) video2 MSA



Figure (4-a) video2 MSMM



Figure (4-b) video2 MSMM



Figure (5-a) video3 MSMM



Figure (5-b) video3 MSA



Figure (6-a) video4 MSMM



Figure (6-a) video4 MSA

VII. CONCLUSION AND FUTURE WORK

In this paper we proposed MSMM tracking algorithm for moving object in video sequence. The traditional Mean-Shift tracking algorithm is used to obtain the target model in each iteration while resizing tracking region by MSMM. Motion information obtained from OF is used as scaling factor to update window size between adjacent three frames. Depending on object moving characteristics to update scale factor parameters. Proposed technique built on two levels, the first level apply MSA and in each iteration motion information is obtained using OF algorithm. Second level is used to resize scale by MSMM method. Experiments are carried out on four videos sequences acquired by dual camera placed in front of and rear moving platform.

Real videos are tested to validate scale factor update performance, Mismatch Ratio measurement and computation time is analyzed and compared with the traditional MSA. Algorithm evaluated by two measurements applied to MSA and MSMM time complexity and mismatch ratio. The experimental results show that our algorithm can effectively adapt scale with respect to speed and direction of moving object and platform without additional time consuming and it is still suitable for real-time tracking applications. Mismatch ratio show that MSMM more accurate than classical MSA. motion information used to track object with different situation such as appear, stop and disappear object in view as future work, since this approach solves static scale in MSA detect different motion speed for the scene flow, it can be used for motion analysis in existing 2D tracking based methods or to define scene flow descriptors.

REFERENCES

- [1] Jwu-Sheng Hu, "3D object Tracking Using Mean-Shift and Similarity-Based Aspect-Graph Modeling", The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON) Nov. 5- 8, 2007, Taipei, Taiwan. pp.2383-2388]
- [2] Y.Wang, "Fuzzy Colour Histogram Based Kernel Tracking under Varying Illumination", Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)0-7695-2874- 0/07 \$25.00 © 2007 ,International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)0-7695-2874-0/07 \$25.00 © 2007 IEEE.
- [3] P.Pan, F.Porikli, "A new method for tracking performance evaluation based on a reflective model and perturbation analysis.", ICASSP, pp.3529-3532, 2009. G.Lihua,Z,Yuanjian , "Video Object Tracking Method Based on Snake Model Using Object's Histogram",0-7803-9584-0/06/\$20.00(2006)IEEE.

- [4] V.Kettnaker,R.Zabih,"Bayesian multi-camera surveillance, "in Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Fort Collins, CO, 1999, pp. 253–259.
- [5] R.Collins, "Algorithms for cooperative multisensor surveillance", Proceedings of the IEEE, vol. 89, no. 10, pp. 1456–1477, 2001.
- [6] M.Greifenhagen, D.Comanicu, "Design, analysis and engineering of video monitoring systems: An approach and a case study", Proceedings of the IEEE, vol. 89, no. 10, pp. 1498–1517, 2001.
- [7] L.Li, "An Efficient Sequential Approach to Tracking Multiple Objects through Crowds for Real-Time Intelligent CCTV Systems", IEEE transactions on system, man, and cybernetics—part B: cybernetics, vol. 38, no. 5, October 2008.
- [8] P.Bai, "Person-Tracking with Occlusion Using Appearance Filters", Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems October 9 - 15, 2006, Beijing,China.
- [9] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, "Multi-camera multi-person tracking for Easy Living", in Proc. IEEE Intl. Workshop on Visual Surveillance, Dublin, Ireland, 2000, pp. 3–10 .
- [10] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body", IEEE Trans. Pattern Anal. Machine Intell, vol. 19, pp. 780–785, 1997.
- [11] J. Garcia, J. Valdivia, and X. Vidal, "Information theoretic measure for visual target distinctness", IEEE Trans. Pattern Anal. Machine Intell, vol. 23, no. 4, pp. 362–383, 2001. X. Li, M. Porikli," hidden markov model framework for traffic event detection using video features", IEEE, International Conference on Image Processing, ICIP, 2004.
- [12] H. Veeraraghavan, "Combining Multiple Tracking Modalities for Vehicle Tracking at Traffic Intersections", New Wan, April 2004.
- [13] Y. Wu, F. Lian, T. Chang, "Traffic Monitoring and Vehicle Tracking using Roadside Cameras", IEEE International Conference on Systems, Man, and Cybernetics October 8-11, 2006, Taipei, Taiwan. C.Picus, C.Beleznai, "Kernel-based storm tracking in radar data". G. R. Bradski, "Computer vision face tracking as a component of a perceptual user interface" in Proc.IEEE Workshop on Applications of Computer Vision, Princeton, NJ, October 1998, pp. 214–219.
- [14] P.Vadakkapat,"Multimodal Approach to Human-Face Detection and Tracking", IEEE transaction on industrial electronics,vol. 55,no. 3, March 2008. A. D. Bue, D. Comanicu, V. Ramesh, and C. Regazzoni, "Smart cameras with real-time video object generation", in Proc. IEEE Intl. Conf. on Image Processing, Rochester, NY, volume III, 2002,pp.429– 432.
- [15] C. LIEBE, "Accuracy Performance of Star Trackers—A Tutorial", IEEE transactions on aerospace and electronic system,vol. 38,no. 2 APRIL 2002. Advanced Computer Vision Notes Series 6,' Optical Flow and Motion Analysis',Ying Wu Electrical Engineering & Computer Science Northwestern University Evanston, IL 60208, EECS432.
- [17] S.Kahlouche, O.Djekoune, D.Djebrouni, D.Merliche, 'Informatique Graphique 52 Segmentation by Motion Based on Optical Flow Histogram , Session A2 : Recalage, Calibrage et Segmentation JIG'2007 - 3èmes Journées Internationales ,Centre de Développement des Technologies Avancées CDTA Houch Oukil, BP 17, Baba Hassen – Alger – Algérie.Tel: (213)21-35-10-18, Fax (213)