A novel adaptive object tracking method based on Expected Likelihood Kernel

Hamd Ait Abdelali, Leila Essannouni, Fedwa Essannouni, Driss Aboutajdine
Faculty of Sciences Rabat
GSCM-LRIT Laboratry Associate Unit to CNRST (URAC 29)
Mohammed V-Agdal University Rabat B.P. 1014
Morocco
hamd.abdelali@gmail.com, essleila@yahoo.fr, efedwa@yahoo.fr, aboutaj@fsr.ac.ma

Abstract: Visual tracking is a new line of broad research. It is required for advanced vision-based applications such as visual surveillance and vision-based human-robot interaction. In this paper, we propose a new method of object detection and tracking algorithm using Adaptive Expected Likelihood Kernel. In this algorithm we combine between the probability product kernels as a similarity measure, and the integral image to increase the speed of the algorithm.

Key–Words: Object tracking, Integral Image, Histogram-based, Expected Likelihood Kernel, Mean Shift.

1 Introduction

The goal of object tracking is to estimate the locations and motion parameters of a target in an image sequence given the initialized position in the first frame. Research in tracking plays a key role in understanding motion and structure of objects. It finds numerous applications including surveillance[12], human-computer interaction[13], traffic pattern analysis[14], recognition [15], medical image processing[16], to name a few. Although object tracking has been studied for several decades, and numerous tracking algorithms have been proposed for different tasks [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11], utilizing different characteristics e.g., shape, texture, or color, etc. These methods perform differently depending on the application and are often compared only subjectively. It remains a very challenging problem. There exists no single tracking method that can be successfully applied to all tasks and situations. Therefore, it is crucial to review recent tracking methods, and evaluate their performances to show how novel algorithms can be designed for handling specific tracking scenarios. Notwithstanding decades of efforts, object tracking remains a challenging problem due to factors such as illumination, pose, scale, deformation, motion blur, noise, and occlusion. To account for appearance change, most recent tracking algorithms focus on robust object representations.

A typical tracking system consists of three components: object representation, dynamic model, and search mechanism. As such, tracking algorithms can be categorized in numerous ways. Object representation is a key component as it directly corresponds to the core challenge of tracking, i.e., how to match object appearance despite all the influencing factors. Moreover, it also determines what objective function can be used for searching the target of interest in frames.

In the field of computer vision, visual tracking still is an important and relevant research area [17]. Different methods like Mean Shift [18, 26], belong to this algorithm class and have become standard algorithms. The main challenges for visual tracking algorithms are real-time capability, accuracy and robustness to changes in the object appearance, due to motion or illumination. Especially the mean shift algorithm, also commonly known as kernel based tracking, fulfils these criteria adequately. Mean Shift was first introduced by Fukunaga [19], for data analysis in the field of pattern recognition, as gradient ascent local mode search algorithm for probability density distributions. Comaniciu evolved the Mean Shift algorithm as a kernel based visual object tracking method in 1999 [20, 21], using color histograms of image regions for target and candidate modelling and the Bhattacharyyan coefficient, a similarity measurement for probability density distributions. This approach is limited to spatial tracking and the object scale adaptation is estimated by a brute force search, with different kernel bandwidths. Later different approaches were introduced to handle the scale estimation problem, like mean shift based blob tracking [22], or multi kernel tracker [23], which divide the target into several blocks, track each block separately and have the advantage of additional spatial information of the separated object blocks. In this challenge problem CAMSHIFT [24, 25] and EM-SHIFT[27], have been introduced since 1998, to estimate the scale and orientation changes adaptively, by calculating moment functions on weight values, which are estimated during the Mean Shift procedure. Ning et al. have transferred this calculation to SOAMST (Scale and Orientation Adaptive Mean Shift tracking) [28], as a pedestrian tracking module.

In this paper we present a novel adaptive object tracking method based on Expected Likelihood Kernel $(AELK)$. In this method we combine between the probability product kernels as a similarity measure, and the integral image[32], to compute the histogram-based of all possible target regions of object tracking in video sequence. The advantages of histogram-based representations are their computational efficiency and effectiveness to handle shape deformation as well as partial occlusion.

To test the performance of $AELK$, we will compare the approach to (SOAMST) and the EM-SHIFT, algorithms are two representative schemes to address the scale and orientation changes of the targets under the Mean shift procedure. The experimental results demonstrated that the new algorithm robust to track the objects in different situations, and can also adapt to changes in shape and scale of the object in the sequences used.

The rest of this paper is organized as follows: Section (2) provides a brief description of probability product kernels. Section (3) introduces the implementation of Adaptive Expected Likelihood Kernel (AELK) algorithm. Section (4) experimental results. Section (5) concludes the paper.

2 Probability Product Kernels

Let p and q be probability distributions on a space χ and ρ be a positive constant. In this work, we are using the probability product kernels ($K_\rho : \chi \times \chi \longrightarrow R$ on the space of normalized discrete distributions over some indexs set Ω) as the similarity measures for comparing two discrete distributions $p_1, p_2, p_3, ..., p_N \varepsilon \chi$ and $q_1, q_2, q_3, ..., q_N \varepsilon \chi$.

The probability product kernel between distributions $\{p\}_{1..N} \epsilon \chi$ and $\{q\}_{1..N} \varepsilon \chi$ is defined as:

$$K_\rho(p,q) = \sum_{k=1}^{N} p(k)^\rho q(k)^\rho \tag{1}$$

It is easy to show that such a similarity measure is a valid kernel , since for any $p_1, p_2, p_3, ..., p_N \epsilon \chi$, the Gram matrix K consisting of elements $K_{ij} = K_\rho(p_i, q_j)$ is positive semi-definite:

$$\sum_i \sum_j \alpha_i \alpha_j K_\rho(p_i, q_j) = \sum_k (\sum_i \alpha_i p_i(k)^\rho)^2 \geq 0 \tag{2}$$

for $\alpha_1, \alpha_2, \alpha_3, ..., \alpha_N \varepsilon IR$.

Different ρ values are corresponded to different types of probability product kernels. For $\rho = 1$, we have:

$$K_1(p,q) = \sum_k p(k)q(k) = IE_p\left[q(k)\right] = IE_q\left[p(k)\right] \tag{3}$$

We call this the expected likelihood kernel[29, 30, 31]. It is worth noting that when dealing with distributions over discrete spaces χ, probability product kernels has a simple geometrical interpretation.

3 Implementation of Adaptive Expected Likelihood Kernel Algorithm

3.1 Tracking principles

First, we will initialize the position of the search template T in the first frame of the video sequence then we will compute the histogram of T. We denote the histogram of T as h_T, and the number of pixels inside T as $\mid T \mid$, which is also equal to the sum over bins, $\mid T \mid = \sum_k h_T(k)$. Let p be the normalized version of h_T given by $p = \frac{h_T}{|T|}$, so we can consider p as a discrete distribution, with $\sum_k p(k) = 1$. Let the histogram obtained at a candidate template R in the first image in the sequence be h_R and its normalized version be q.

Second, we will give a frame as a query, a feature histogram-based algorithm aims to find the location of the object, by evaluating a similarity measure for comparing the feature histogram of the template to that of each possible in the frame of video sequence.

3.2 Expected Likelihood Kernel

The Expected Likelihood Kernel is defined by $K(p,q) = \sum_k p(k)q(k)$. We may compute h_T and p in advance with the given template. For the k-bin of h_T , its value is obtained by counting the pixels that are mapped to the index k :

$$h_T(k) = \sum_{x \epsilon T} \delta\left[b(x) - k\right] \tag{4}$$

Where $\delta\left[x\right]$ is the Kronecker delta, with $\delta\left[x\right] = 1$ if $x = 0$, and $\delta\left[x\right] = 0$ otherwise. The mapping function $b(x)$ maps a pixel x to its corresponding bin index. The computation of the expected likelihood kernel can be expressed as:

$$\begin{aligned} K(p,q) &= \sum_k p(k)q(k) \\ &= \sum_k p(k)\left(\frac{1}{R}\sum_{x\epsilon R}\delta\left[b(x)-k\right]\right) \\ &= \frac{1}{R}\sum_{x\epsilon R}\sum_k p(k)\delta\left[b(x)-k\right] \\ &= \frac{1}{R}\sum_{x\epsilon R} p(b(x)) \end{aligned} \tag{5}$$

Therefore, the computation of the expected likelihood kernel can be done by taking the sum of values $p(b(x))$ within candidate template R. As a result, we are able to use the following algorithm to evaluate the expected likelihood kernel over the whole image. The output of the following algorithm is a support map that reflects the similarity between the template and each frame of sequence.

3.3 Adaptive Expected Likelihood Kernel

Adaptive Expected Likelihood Kernel ($AELK$) is an extension of ELK algorithm, it uses the values of the color histogram of the current region defined by support map and covariance matrix V.

We will denote a data set of N independent samples by $\chi = \{x_1, ..., x_N\}$. Let us assume that the probability density function Gaussian $p(x) = \aleph(x_i, map, V)$, is a good generative model for our data.

Let x_i denote the pixel locations of template and i are all the pixels that belong to the template, and map_0 is the initial location of the center of the template in the image. The covariance matrix can be used to approximate the shape of the object:

$$V_0 = \sum_i (x_i - map_0)(x_i - map_0)^t \tag{6}$$

Let $O = [o_1, ..., o_k]^t$ the color histogram model of the object have k-bins and let $b(x_i) : R^2 \longrightarrow 1, ..., K$ be the function that assigns a color value of the pixel at location x_i to its bin. The value of the k-bins is calculated by:

$$O_k = \sum_{i=1}^{N_{v_0}} \aleph(x_i, map, V_0)\delta[b(x_i) - k] \tag{7}$$

where δ is the Kronecker delta function. We use the Gaussian kernel N to rely more on the pixels in the middle of the object and to assign smaller weights to the less reliable pixels at the borders of the objects. We use only the N_{v_0} pixels from a finite neighborhood of the kernel and the pixels further.

The color histogram that describes the appearance of the region is $r_k(map, V)$ and the value of the k-bins is calculated by:

$$r_k(map, V) = \sum_{i=1}^{N_v} \aleph(x_i, map, V)\delta[b(x_i) - k] \tag{8}$$

The weights of the value of k-bins in object are calculated by:

$$\omega_i = \sum_{k=1}^{K} \sqrt{\frac{O_k}{r_k(map, V)}}\delta[b(x_i) - k] \tag{9}$$

The new distribution of color histogram p are calculated by:

$$p_i = \frac{\omega_i \aleph(x_i, map, V)}{\sum_{i=1}^{N} \aleph(x_i, map, V)} \tag{10}$$

The covariance matrix update can be used to approximate the new shape of the object are calculated by:

$$V = \beta \sum_{i=1}^{N} p_i(x_i, map)(x_i, map)^t \tag{11}$$

We should use $\beta = 1.5$. The correct value for the β depends on the noise that is present in the image sequence [27].

3.4 System description

The implementation of the proposed algorithm is done using MATLAB. The basic block diagram for the proposed algorithm is shown below:

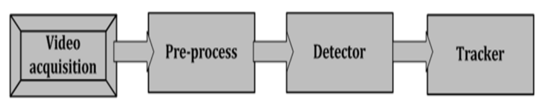


Fig 1: Basic block diagram for proposed algorithm.

The basic block diagram consists of four blocks named as Video acquisition, Pre-process, Detector and Tracker. The functions of these blocks are as follows:

Video acquisition: Video acquisition means to obtain the video frames using the Image Processing Toolbox MATLAB.

Pre-process: In Pre-process, we start video acquisition, converting video into images processing, extracting color information of images.

Detector: In this block we combine between the probability product kernels, and the integral image to compute similarity measure, and the histograms of all possible target regions of object tracking in data sequence.

Tracker: Tracking of object is done on the basis of the region properties of the object such as, Area, Centroid, etc. The objective of object tracking is achieved.

3.5 Proposed algorithm

Adaptive Expected Likelihood Kernel (AELK)

1. Choose the initial location of the search template, and location center template (map_0) and shape defined by V_0 using (6).
2. Compute the normalized histogram p for the template.
3. Quantize the image features of each pixel x in the frame to obtain bin index $b(x)$.
4. Create an auxiliary image as large as the frame. Assign the value $p(b(x))$ to the pixel at the corresponding position in the auxiliary image.
5. Build the integral image of the auxiliary image.
6. Create a support map consisting of the kernel values as the output.
7. Compute the values of the color histogram of the current region defined by map and V from the current frame using (8).
8. Calculate weights ω_i using (9).
9. Calculate the new distribution of color histogram p using (10).
10. Calculate new variance estimate V using (14). Go to step 3.

A basic flowchart diagram for the proposed algorithm is shown below:

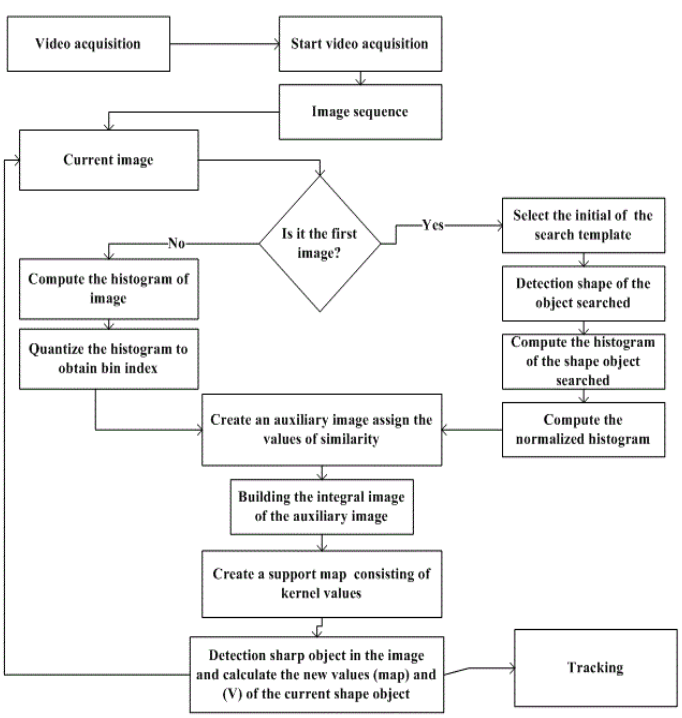


Fig 2: Flowchart diagram for proposed algorithm

4 Experimental Results

This section evaluates the proposed Adaptive Expected Likelihood Kernel (AKEL), to test AKEL algorithm, we used different sequences, each has its own characteristics but the use of a single object in movement is a commonality between these different sequences. We set up experiments to validate our contribution.

In this work, we selected RGB color space as the feature space and it was quantised into $16 \times 16 \times 16$ bins for a fair comparison between different algorithms. One synthetic video sequence and three real video sequences are used in the experiments.

The importance of the RGB color model is that it relates very closely to the way that the human eye perceives color. RGB is a basic color model for computer graphics because color displays use red, green, and blue to create the desired color. Therefore, the choice of the RGB color space simplifies the architecture and design of the system. Besides, a system that is designed using the RGB color space can take advantage of a large number of existing software routines, because this color space has been around for a number of years.

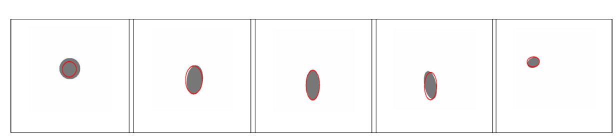
Sequence 1 : This sequence present a simple object ellipse, with different scale and orientation is contained 77 images of (240×352) pixels.

Sequence 2 : This sequence is extracted from a table tennis, and is contained 51 images of (288×352)pixels.

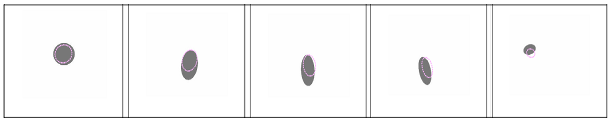
Sequence 3 : This sequence shows a person entering and moving palm in a room, and is contained 50 images of (480×640) pixels.

4.1 Results obtained about sequences of algorithm AELK, EM-Shift and SOAMST

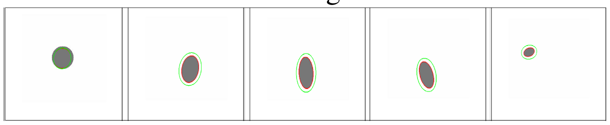
To verify the efficiency of the proposed Adaptive Expected Likelihood Kernel algorithm, we compared our algorithm with two existing algorithms SOAMST [28] and EM-Shift [27], the experimental results show that, the proposed AELK method achieves good estimation accuracy of the scale and orientation of object in the sequences. We listed the estimated width, height and orientation of the synthetic ellipse sequence, player sequence, and palm sequence in Fig 3, Fig 4 and Fig 5 respectively.



AELK algorithm

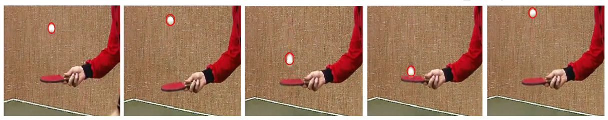


EM-shift algorithm

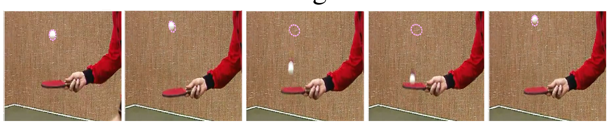


SOAMST algorithm

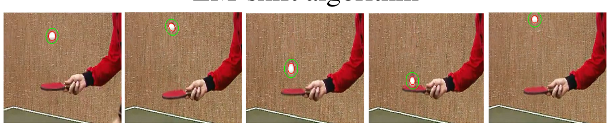
Fig 3: Tracking results of the synthetic ellipse sequence by different tracking algorithms. The frames 1, 15, 20, 23 and 39 are displayed.



AELK algorithm



EM-shift algorithm

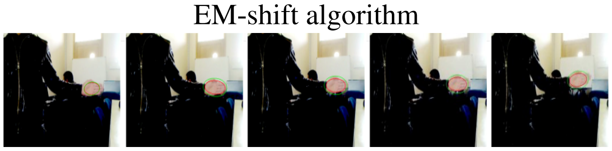


SOAMST algorithm

Fig 4: Tracking results of the player sequence by different tracking algorithms. The frames 1, 7, 16, 18, and 30 are displayed.

AELK algorithm

EM-shift algorithm



SOAMST algorithm

Fig 5: Tracking results of the palm sequence by different tracking algorithms. The frames 1, 5, 9, 14, and 20 are displayed.

This section evaluates the proposed AELK algorithm for scale and orientation changes in comparison with the SOAMST and EM-Shift algorithms. The SOAMST and the EM-shift algorithms are two representative schemes to address the scale and orientation changes of the targets under the mean shift framework. As the weight image estimated by CAMSHIFT is not reliable, it is prone to errors in estimating the scale and orientation of the object.

We first use a ellipse sequence to verify the efficiency of the proposed AELK algorithm. The experimental results show that the proposed algorithm could reliably track in a timely manner the ellipse with scale and orientation changes. Meanwhile, the experimental results by EM-shift and SOAMST is not good because does not estimate the target orientation change and has bad tracking results in a timely manner. The results as shown in Fig 3.

The second video is a player sequence shown in Fig 4. The experimental results show that the proposed AELK algorithm estimates more accurately the scale changes than SOAMST and EM-shift algorithms.

The last experiment is on a more complex sequence of palm. The object exhibits with partial occlusion. The results by AELK, EM-shift and SOAMST algorithms as can be seen in Fig 5, both EM-shift and SOAMST algorithms can track the target over the whole sequence. However, the AELK scheme works much better in estimating the scale and orientation of the target, especially when occlusion occurs.

4.2 Estimation of the execution time

An evaluation of the execution time of the algorithms (AELK, SOAMST and EM-Shift), was carried out for different sequences. This measurement was made in a completely software Matlab, knowing that our platform for the experiments is described as follows:

Processor: Intel (R) Core (TM) 2 Duo CPU T6570

Frequency: 2.10 GHz

Memory: 2GB

System (OS): Operation System 32-bit

The results of the measurement of detection time are presented by the following figures:

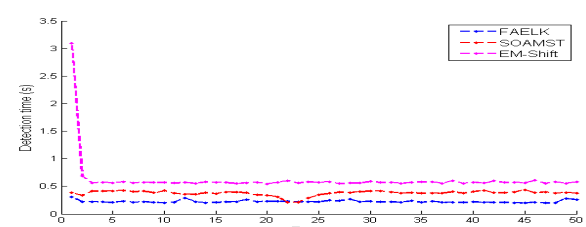


Fig 4: Measuring of detection time for palm sequence by AELK, SOAMST and EM-Shift.

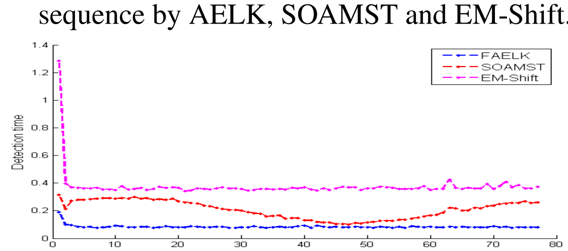


Fig 5: Measuring of detection time for ellipse sequence by AELK, SOAMST and EM-Shift.

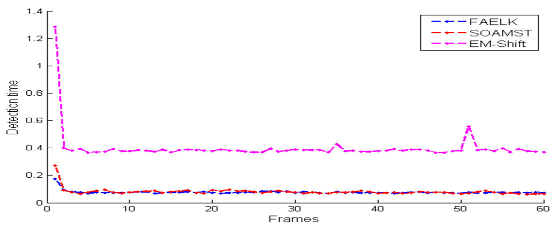


Fig 6: Measuring of detection time for player sequence by AELK, SOAMST and EM-Shift.

The graphs above (Fig 4,Fig 5 and Fig 6) represents the detection time of object in the frames of sequences videos (palm, ellipse and player) respectively by applying the algorithms AELK, SOAMST and EM-SHIFT. According to this group, we notice that the algorithm EM-SHIFT having a high detection time. For example, in 2th frame the detection time with EM-SHIFT reaches more that 0.4s in all sequences of videos, on the other hand the detection time of the algorithm SOAMST is less than 0.4s in all sequences of videos, which is better by report EM-SHIFT. However, by applying our algorithm AELK we were able to detect object in the frames with a timely manner, in comparison to others algorithms EM-SHIFT and SOAMST. Consequently, according to this comparison we can deduct that the application and experimentation show that our AELK always best to achieve the detection object in a timely manner.

We note that, the tracking time of SOAMST and EM-SHFT is much larger than the AELK algorithm, table 1 lists the average time by different methods.

Table 1: The average time by different methods on the sequences.

Methods / sequences	AELK	SOAMST	EM-SHFT
Ellipse (77 frames)	0.08 s	0.20 s	0.37 s
Player (60 frames)	0.07 s	0.08 s	0.4 s
Palm (50 frames)	0.22 s	0.37 s	0.62 s

We observe that in the three experimented videos (Ellipse sequence, Player sequence, and Palm sequence), the average time of the EM-Shift algorithm is always much larger that the other applied algorithms such as SOAMST and AELK. Nevertheless, we notice that our algorithm AELK has an average time of execution better.

Thus, we can deduct that the new algorithm AELK always gives perfected results at the average time of execution, for example in Ellipse sequences video, we obtained an average time 0.08(s) by applying our algorithm AELK, on the other hand we obtained 0.20(s) and 0.37(s) by applying respectively SOAMST and EM-SHIFT.

5 Conclusion

Adaptive Object Tracking based on Expected Likelihood Kernel is a basis for a number of important applications such as real-time surveillance and visual tracking. However, it is computationally expensive and resource hungry. For performance evaluation, we have implemented and evaluated our algorithms as well as a well accepted approach for Object tracking that is based on Histogram-based. In the experiments performed both in indoor and outdoor environments, our approaches considerably reduce the detection delay and memory usage. As our algorithms are more efficient in terms of delay and memory consumption. The efficiency and reliability of the Adaptive Expected Likelihood Kernel has been tested. These results demonstrate that our approach is able to track objects in all types and characteristics of the video sequences used.

We demonstrated that it new algorithm can robustly track the objects in different situations, and can also adapt to changes in shape and scale of the object. Adaptive Expected Likelihood Kernel works in real-time, that can be useful for many other vision problems. This is a topic of our further research.

References:

[1] Schugk, D., Kummert, A., and Nunn, C.,(2014), 'Adaptation of the Mean Shift Tracking Algorithm to Monochrome Vision Systems for Pedestrian Tracking Based on HoG-Features,' SAE Technical Paper 2014-01-0170, doi:10.4271/2014-01-0170.

[2] Shengfeng He; Qingxiong Yang; Lau, R.W.H.;(2013), 'Jiang Wang; Ming-Hsuan Yang, Visual Tracking via Locality Sensitive Histograms,' Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on , vol., no., pp.2427,2434, 23-28.

[3] A. Chan and N. Vasconcelos,(2012) 'Counting people with low-level features and Bayesian regression,' IEEE Trans. Image Process., vol. 21, no. 4, pp.21602177.

[4] B. Babenko; M.-H. Yang; and S.Belongie,(2011),' Robust object tracking with online multiple instance learning,' PAMI, 33(8):1619 1632.

[5] Y.-L. Hou and G. K. H. Pang,(2011),'People counting and human detection in a challenging situation,' IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 41, no. 1, pp. 2433 .

[6] L.Cehovin, M. Kristan, and A. Leonardis,(2011),'An adaptive coupled-layer visual model for robust visual tracking,' In ICCV.

[7] J. Kwon and K. M. Lee.(2010),' Visual tracking decomposition'. In CVPR,pages 12691276.

[8] Yuan Chen; Shengsheng Yu; Weiping Sun; Hongxing Li,(2008), Objects Detecting Based on Adaptive Background Models and Multiple Cues in Computing, Communication, Control,and Management, CCCM 08. ISECS International Colloquium on , vol.1, no., pp.285-289.

[9] Jianguo Tao; Changhong Yu,(2008), Real- time detection and tracking of moving object in Intelligent Information Technology Application, IITA 08. Second International Symposium on, vol.2, no., pp.860-863.

[10] Guanglun Li; Yanling Wang; Weiqun Shu,(2008), Real-Time Moving Object Detection for Video Monitoring Systems inIntelligent Information Technology Application, IITA 08. Second International Symposium on , vol.3, no., pp.163-166.

[11] Arnab Roy; Sanket Shinde; Kyoung-Don Kang,(2010), An Approach for Efficient Real Time Moving Object Detection, in Hamid R. Arabnia and Ashu M. G. Solo, ed., ESA ,CSREA Press, , pp. 157-162 .

[12] I. Haritaoglu, D. Harwood, and L. Davis,(1998) 'W4s: A real-time system for detecting and tracking people,' in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition , pp. 962968.

[13] M. de La Gorce, N. Paragios, and D. Fleet,(2008) 'Model-based hand tracking with texture, shading and self-occlusions,' in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition.

[14] Z. Sun, G. Bebis, and R. Miller,(2006) 'On-road vehicle detection: A review,' IEEE Transactions on Pattern Analysis and Machine Intelligence 28(5), pp. 694711.

[15] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley,(2008), 'Face tracking and recognition with visual constraints in real-world videos,' in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 18.

[16] X. Zhou, D. Comaniciu, and A. Gupta,(2005), 'An information fusion framework for robust shape tracking,'IEEE Transactions on Pattern Analysis and Machine Intelligence 27(1), pp. 115129.

[17] Yilmaz A., Javed O. and Shah M.,(2006) 'Object tracking:A survey,'ACM Comput. Surv., vol. 38, no. 4, p. 13+, doi:10.1145/1177352.1177355

[18] Comaniciu D., Ramesh V. and Meer P.,(2003) 'Kernel-basedobject tracking,' Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 25, no. 5, pp. 564-577,doi:10.1109/TPAMI.2003.1195991

[19] Fukunaga K. and Hostetler L.,(1975) 'The estimation of the gradient of a density function, with applications in pattern recognition,' Information Theory, IEEE Transactions on, vol.21, no. 1, pp. 32-40, doi:10.1109/TIT.1975.1055330

[20] Comaniciu D. and Meer P.,(1999) 'Mean shift analysis and applications,' in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, Kerkyra, doi:10.1109/ICCV.1999.790416

[21] Comaniciu D., Ramesh V. and Meer P., (2000) 'Real-time tracking of non-rigid objects using mean shift,' in Computer Vision and Pattern Recognition. Proceedings. IEEE Conference on, Hilton Head Island, SC, doi:10.1109/CVPR.2000.854761

[22] Collins R., (2003) 'Mean-shift blob tracking through scale space,' inComputer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, Madison, Wisconsin, doi:10.1109/CVPR.2003.1211475

[23] Yan Y., Huang X., Xu W. and Shen L., (2012) 'Robust Kernel-Based Tracking with Multiple Subtemplates in VisionGuidance System,' Sensors, vol. 12, no. 2, pp. 1990-2004, doi:10.3390/s120201990

[24] Bradski G. R.,(1998) 'Computer vision face tracking for use in a perceptual user interface,' Intel Technology Journal, vol. 2.

[25] Lee L.-K., An S.-Y. and Oh S.-y., (2011) 'Robust visual object tracking with extended CAMShift in complex environments,' in IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society , Melbourne, VIC, doi:10.1109/IECON.2011.6120057

[26] Comaniciu, D.; Ramesh, V.; Meer, P.,(2003),Kernel-Based Object Tracking, IEEE Trans.Pattern Anal. Mach. Intell. 25 (5) , pp. 564-575 .

[27] Zivkovic, Z.; Krose, B., (2004), An EM-Like Algorithm for Color-Histogram-Based Object Tracking, in CVPR (1) , pp. 798-803 .

[28] Ning, J.; Zhang, L.; Zhang, D.; Wu, C.,(2012),Scale and orientation adaptive mean shift tracking, Computer Vision, IET , pp.52-61.

[29] Huang-Wei Chang and Hwann-Tzong Chen,(2010), A square-root sampling approach to fast histogram-based search, in CVPR , IEEE, pp. 3043-3049.

[30] Tony Jebara and Risi Kondor and Andrew Howard and Kristin Bennett and Nicol Cesabianchi,(2004), Probability Product Kernels., Journal of Machine Learning Research 5 , pp.819-844 .

[31] Tony Jebara and Risi Kondor,(2003), Bhattacharyya Expected Likelihood Kernels, in Bernhard Schlkopf and Manfred K. Warmuth,ed., COLT , Springer, , pp. 57-71.

[32] Paul Viola and Michael Jones, (2004), Robust Real-Time Face Detection, International Journal of Computer Vision 57 (2) , pp. 137-154.