Multi-target Tracking based on Level Set Segmentation and Contextual Information

Liu Meng[1] and Qingxuan Jia[2]

[1] Automation School of BUPT, Beijing University of Posts and Telecommunications, Beijing (China)
[2] Automation School of BUPT, Beijing University of Posts and Telecommunications, Beijing (China)
1bigbiglemon@163.com,2qingxuan@bupt.edu.cn

Abstract

In the unconstrained environment for video tracking is essential for many applications, such as video surveillance, man-machine interaction. In fact, moving object in the sequences generally has the context information of others or the different moments of its own state. Our research focus on the complex scenes, tracking multiple articulated targets, obtaining the features of the target, getting the precise target segmentation and improving the accuracy and reliability of tracking. We propose using top-down segmentation to feedback object detection, also contains the shape information. And the local appearance information is embedded into the framework of the level set. Then we propose a method to solve the interference of similar appearance target and multi-target tracking, by using context information to create two auxiliary items: Misleading items and support items. Both of them are using continuous random ferns. We experimentally evaluate our proposed approach on challenging sequences and video in real-world demonstrate its good performance in practice.

Keyword*：target segmentation, multi-target tracking, level set, context information, random ferns*

1. Introduction

In order to track multiple moving targets in the scene, the common method consists of two steps: first with detection method for detecting the foreground objects of the current frame, and then by target feature between successive frame matching to achieve tracking. Tracking method which based on object detection--target feature analysis depends on the accuracy of foreground segmentation for target motion features and appearance and shape features, so good segmentation algorithm is particularly important. Level set [1, 2] has a better computational efficiency in many segmentation and tracking tasks [3, 4], it has become more and more popular, it's advantage is the flexibility to adapt to the contour topology changes , but the amount of computation is too large. Some methods proposed level set to track the deform target [4-6]. In particular, Bibby and Reid [4], who demonstrating the robustness of the tracking, including on the busy streets[7] multi-target tracking[8]. And context information has been applied actively in object detection [9], object classification [10, 11], object recognition [12]. It has been employed recently in several tracking methods [13, 14].

So the main research is in the complex scenes, tracking multiple articulated targets, obtaining the features of the target, getting the precise target segmentation and improving the accuracy and reliability of tracking.

First in the segmentation phase, we propose two improvements: (1) We propose using a top-down segmentation to feedback target detection, in order to meet the current needs of the image, we also contains the shape information of a specific category. In addition, our approach without rigid constraints for the target shape, but in the form of a probability map from each pixel to the target /background, the results can be directly integrated into the target model. (2) We propose that the local appearance information is embedded into the framework of the level set. The advantage of this method is to provide the most suitable segmentation contour for tracking.

Secondly in the tracking phase, we propose a method to solve the interference of similar appearance target and target tracking about target leaving the field of view, by using context information to create two auxiliary items: Misleading items and support items. Both of them are using continuous random ferns. Misleading items collect the ones with the similar appearance of the target, continued appear with the target and have a high confidence score. The tracking system must keep track of these misleading, to avoid drift. On the other hand, the support items are co-occurrence and motion correlation in a short time with local key points around the target. Video in real-world prove that the use of context information improved the tracking results.

2. Level Set Segmentation and Tracking

We use a probabilistic level-set framework to perform a segmentation of the target object and track it through the following frames. The tracked object is represented by its contour C (represented with level sets Φ) and its position p in the image. It consists of pixels at coordinates x with color y. Foreground and background regions M are distinguished by appearance models consisting of color histograms and we additionally incorporate a class specific shape model h. Thus, given an initialization for x, y, h, and M, the task is to infer shape Φ and position p. The joint distribution for one pixel given by the model is

$$P(x_i, y_i, h_i, \Phi, p, M) = P(x_i|\Phi, p, M)P(y_i|M)P(h_i|M)P(M)P(\Phi)P(p) \qquad (1)$$

Conditioning on x_i, y_i, h_i and marginalizing over M yields

$$P(\Phi, p|x_i, y_i, h_i) = \frac{1}{P(x_i)}\sum_k \left\{P(x_i|\Phi, p, M_k)\frac{P(y_i|M_k)P(M_k)}{\sum_l P(y_i|M_l)P(M_l)}P(M_k|h_i)\right\} P(\Phi)P(p) \qquad (2)$$

Where the M_k denote the different regions. We simplifying the expression as:

$$P(\Phi, p|x, y, h) = \prod_{i=1}^{N} P(x_i|\Phi, p, y_i, h_i)\ P(\Phi)P(p) \qquad (3)$$

The term P(Φ) to specify some desired internal properties of the contour: a geometric prior (eliminating the periodic re-initializations [18, 3]) and a prior for the length of the contour [10],then rewarding a smoother contour.

$$P(\Phi)\prod_{i=1}^{N}\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(|\nabla\Phi|-1)^2}{2\sigma^2}\right)\exp(-\lambda|\nabla H_\epsilon(\Phi)|) \qquad (4)$$

Where σ and λ are the weights of the priors. Maximizing the posterior is:

$$\varepsilon(\Phi) = -\log\big(P(\Phi, p|x, y, h)\big) \propto \sum_{i=1}^{N}\Big\{\log\big(P(x_i|\Phi, p, y_i, h_i)\big) - \frac{(|\nabla\Phi|-1)^2}{2\sigma^2} - \lambda|\nabla H_\epsilon(\Phi)|\Big\} + N\log\Big(\frac{1}{\sigma\sqrt{2\pi}}\Big) + \log(P(p)) \qquad (5)$$

In order to optimize the desired values Φ and p, we first optimize the shape and keep the position constant, then optimize the position while keeping the shape constant.

For segmentation, using the Euler-Lagrange equation which minimizes E(Φ):

$$\frac{\frac{\partial}{\partial\Phi_k}P(x|\Phi,p,y,h)}{P(x|\Phi,p,y,h)} + \frac{1}{\sigma^2}\Big[\nabla^2(\Phi_k) - \mathrm{div}\Big(\frac{\nabla\Phi_k}{|\nabla\Phi_k|}\Big)\Big] + \frac{\partial\Phi_k}{\partial t} = -\frac{\partial\varepsilon(\Phi)}{\partial\Phi_k} = \lambda\delta_\epsilon(\Phi_k)\mathrm{div}\Big(\frac{\nabla\Phi_k}{|\nabla\Phi_k|}\Big) \qquad (6)$$

Having obtained the target object's shape, we track it through the following frames by performing a rigid registration. The new position of Φ is described with a warp p, which can be any transformation that forms a group. For this, we introduce the warp W (x_i, Δp) with parameters p. P (p) is dropped here and is handled with drift correction:

The contour of the foreground object is described by the zero level set of a level set embedding function Φ_c.Starting from some initialization; the contour is evolved to maximize its probability given the image, the learned appearance models, and the shape model. The appearance models are rebuilt in each of the n_1 iterations. In the following frame, the new position of the shape is registered and afterwards the contour is adapted by performing n_2 segmentation iterations. In this case, the appearance models are not rebuilt, but only slightly adapted for greater robustness.

3. Detection-Based Top-Down Segmentation

In order to achieve robust tracking performance, we want to make use of the information that we track objects of a certain category (*e.g.*, pedestrians). This is also the motivation behind work on category-specific shape priors. However, such priors do not take into account image-specific information and have difficulties modeling the dynamic shape of strongly articulated objects. Instead, we propose to use top-down segmentation information fed back from object detection.

For this, we build upon class-specific Hough Forest detectors, as they are suitable for processing densely sampled image patches. The votes corresponding to a local maximum in Hough space. It can be backprojected to the image in order to infer top-down segmentation information.

This step derives a local figure-ground label for the patch $X = \{x_k\}$ conditioned on the patch appearance and the detected object location. When a Hough-space maximum is selected and its constituent votes are backprojected. The resulting patch segmentation label can then be weighted by the weight of the corresponding vote ω_{v_j}. The figure-ground probability of a pixel x is obtained by averaging over all patches X_i

$$P(M_f|h) = \frac{1}{z}\sum_{X_i(x)}\frac{1}{|X_i|}\sum_{v_j \in votes(X_i)}\omega_{v_j}Seg(v_j) \qquad (7)$$

$$P(M_b|h) = \frac{1}{z}\sum_{X_i(x)}\frac{1}{|X_i|}\sum_{v_j \in votes(X_i)} \omega_{v_j}(1 - Seg(v_j))$$

$$z = \sum_{X_i(x)} \sum_{v_j \in votes(X_i)} \omega_{v_j}$$

Figure 1 visualizes the figure and ground probability maps for an example.

The resulting procedure provides an object-specific figure and ground probability. Moreover we use the top down segmentation as initialization for the level set segmentation. The foreground region is given by all pixels x with

$$\frac{\theta P(M_f|h)}{\theta P(M_f|h) + P(M_b|h)} \geq 0.5 \tag{8}$$

We initialize Φ_c with a signed distance function of the obtained contour. The factor θ can be used to shrink or enlarge the obtained contour.

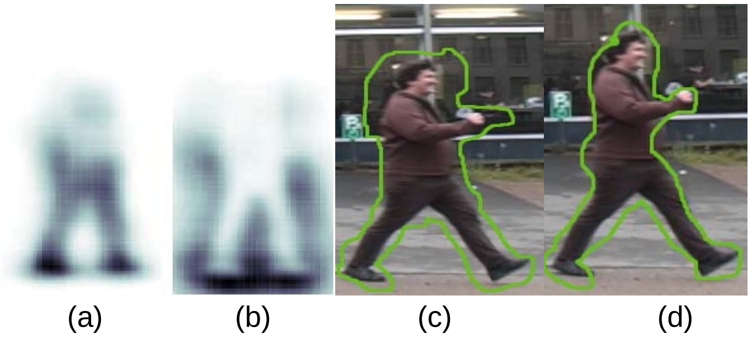


Figure 1. The Figure and Ground Probability Maps (a)$P(M_f|h)$,(b)$P(M_b|h)$,(c) contour segmentation based on the above information, (d) segmentation results from the Hough detector

The combined model can be summarized as follows: Hough forest detector is used to initialize the emerging person. Local appearance model using four color histogram and using detector divided by n1 = 100 iterations to obtain the probability of the image background. Tracking the subsequent frames, the new target position is determined by a simple Kalman filter, using the new test as observations. Shapes split by performing n2 = 100 iterations to adapt to the new image and the background probabilities. If there is no one detected in one frame, P (M | h) is not exist. At this point, we think that a generation model not h, in fact that the formula missed P (M | h). If there is no target after the detecting we will give up tracking.

As everyone is tracking independently, so their level set function does not affect each other. This method does not require additional information (for example, the ground plane or depth map), but it only handle a simple occlusion. For robust tracking targets, when severe occlusion and similar targets appear in the same area, or the target left the visual field, we decided to combine the above method with contextual information to aid tracking. The next section, we will detail the use of context information which add two auxiliary items to effectively achieve the target tracking.

4. Context Information Based on the Object Robust Tracking

Here, we propose to exploit the context information by expressing it in two different terms: 1) Misleading items are regions that have similar appearance as the target. 2) Support items are local key-points around the object having motion correlation with our target in a short time span. Misleading items share the same type as the target. Support items occur in regions belonging to the same object as the target, but are not included in the initial bounding box. The target and Misleading items are detected using shared sequential randomized ferns [15].

4.1. Context Tracker

This section describes how the context tracker exploits context information while tracking, and takes advantage of them to avoid drift.

We use the P-N Tracker [18] as our basic target tracker with several extensions. First, we extend the randomized ferns to accept multiple objects. Second, we use new 6bitBP [19] to boost up the speed of the detector. Third, we don’t use the initial patch as the object model.

However, we improve this model by constructing it in binary search tree using k-means. The computational complexity to evaluate a sample is O(logn) instead of O(n) when using Brute-force. We choose the PN-Tracker because it uses scanning window to search for all of possible candidates in the whole image which helps to explore the context at the same time.

Misleading items are regions which have similar appearance as our target. In our tracker, a testing sample confidence score is computed using Normalized Cross-Correlation (NCC) between it and the closest image patch in the object model. The region having the highest confidence is considered as the current target if its score is larger than a threshold $\theta = 80\%$. The remaining regions trigger new Misleading items trackers. These trackers are formulated similarly to our basic tracker.

Assuming that we have the valid target at frame t, the Support items are extracted around the location of that target with a radius R. After that, a sliding window of $k = 5$ frames is used to store and match the previous Support items with the current ones. Each match makes the frequency of that supporter increase by 1.

In practice, there are several candidates similar to our target with very high confidence score. In fact, the right candidate may not even obtain the highest score, especially when the appearance is changing. Without context, the tracker obviously switches to the one with the highest score. Also, in unconstrained environments, our target may leave the FOV, or be completely occluded by other objects. The tracker will simply switch to another region satisfying the threshold θ. Here, our tracker automatically exploits all the Misleading items and pays attention to them by tracking them simultaneously. Also, our tracker discovers a set of Support items to robustly identify the target among other similar regions.

4.2. Detection of Misleading Items

Misleading items are regions which have appearance similar appearance to the target and consistently co-occur with it. Usually, Misleading items are other moving objects sharing the same object category as our target. To prevent our tracker from drifting to these regions, we propose to detect and initiate a simple tracker for each of them so that we can minimize confusion during tracking.

Due to the randomized ferns classifier is used in recognition and tracking, we employ it to detect possible Misleading items in every frame. Randomized ferns were originally proposed by Ozuysal *et al.*, [15] to increase the speed of randomized forest [16]. In our method, each of them corresponds to a set of Binary Pattern features. Each leaf in a fern records the number of added positive and negative samples during training. The posterior probability or that input testing sample in feature vector x_i to be labeled as an object (y = 1) by a fern j is computed as $\mathrm{Pr_j}(y = 1|x_i) = p/(p + n)$, where p and n are the number of positive and negative samples recorded by that leaf. The posterior probability is set to 0 if there is no record in that leaf. The final probability is calculated by averaging the posterior probabilities given by all ferns:

$$\mathrm{Pr_j}(y = 1|x_i) = \textstyle\sum_1^T \mathrm{Pr_j}\,(y = 1|x_i) \qquad （9）$$

T is the number of ferns. To improve the running time, these randomized ferns are shared between our object detector and distracter detector. Each tracker controls the posterior probability by adding its positive and negative samples to the ferns according to the P-constraints and N-constraints. We avoid adding hard negative samples to avoid over-fitting. Also, during tracking, when the appearance of a distracter is different from our target, we discard it. Indeed, it helps to emphasize that our focus is on tracking a single target, not on multiple target tracking.

Therefore, a sample is considered a distracter candidate if it passes the random ferns with a probability$\mathrm{Pr}(y = 1|x_i) > 0.5$, and is not the target. We maintain an M frames sliding window and count the frequency $\mathrm{fd_k}$ of a candidate k based on its appearance consistency spatial consistency related to the target. Then a candidate is classified as a distracter as follows

$$\mathrm{P_d}(y_d = 1|x_i) = \begin{cases} 1 & \text{if } \mathrm{fd_k} > 0.5 \\ \text{and} & d(x_i, M) > 0.8 \\ 0 & \text{otherwise} \end{cases} \qquad (10)$$

Where $\mathrm{P_d}(y_d = 1|x_i)$is the probability for a candidate i in a feature vector x_i having label y_d, while d (x_i, M) is the confidence of this candidate evaluated by the template-based model of the target. The first condition allows detecting Misleading items which repeatedly co-occur with our target, while the second one helps to exploit Misleading items having very similar appearance to our target.

4.3. Selection of Support Items

We aim to build an efficient Support items set which helps to quickly verify the location of the target. They also have a strong correlation in motion with our target. The Support items are also detected from the local region around each candidate. After that, these supporter detection responses are matched with the ones from previous frames to find the co-occurrence between them and our target. Moreover, unlike the Support items proposed in [14] which are expensive to detect and match in the whole frame, our Support items are efficiently detected and matched around the locations of the very few candidates having high probability to be the target in each frame.

To detect Support items, we use the Fast Hessian Detector and employ SURF descriptor as in [17] to describe the region around them. We store all of these Support items in a sliding window of k frames (k = 5). There are two types of Support items: active and passive. The active Support items are the ones co-occurring with our target in high frequency $f_s > 0.5$ within the sliding window, while passive ones are the rest. Finally, the supporting score is computed as follows

$$S_i = \frac{n_{am}}{n_{ta}} \qquad (11)$$

Where n_{am} and n_{ta} are the numbers of active matched Support items and total active Support items in the model. A supporter model is considered strong if $S_i > 0.5$ and $n_{ta} > 5$. Then all of the matched results are used to update the supporter model.

5. Experiments

5.1. Experiment Settings

The algorithm uses C + +, Intel Open Source Computer Vision Library (OpenCV) visual processing, 1 Intel Core (TM) i5-2450M 2.5GHz 4G computer

The experiments to test the two types of video clips the video clips: the TUD tunnel data set, fixed camera on people walking in a cloudy day, which is observed from the side.

5.2. Segmentation Performance

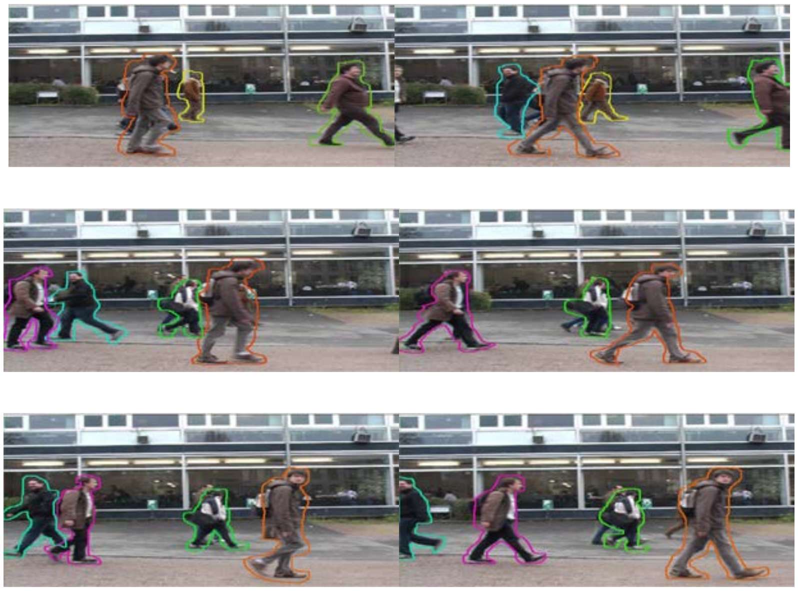
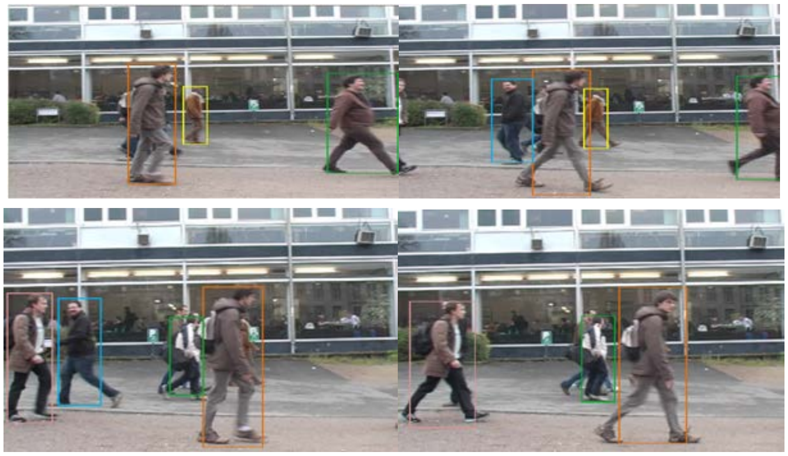


Figure 2. The Results of Level Set Segmentation



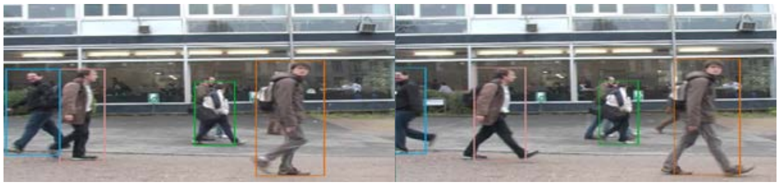


Figure 3. The Tracking Results with the Context Tracker

In Table 1 (left), we compare the segmentation performance of the Hough Forest detector top-down segmentation. It can be seen that all parts contribute to improve the segmentation results. The full model without the localized appearance models or without the Hough Forest top-down segmentation both do not reach the performance of the full model, proving that both are necessary to achieve this improvement.

Table 2(right), shows how the segmentation performance of the raw Hough Forest detector is improved through the integration of the level set tracker with the probabilistic shape models and our localized appearance models for different detector thresholds. The localized appearance models improve performance on top of the integration with the probabilistic shape models

Table 1. Segmentation Performance.BR: Level Set; LS: Level Set with 2 Appearance Models; LAM: Level Set with our Appearance Models; HF: Hough Forest Detector

	recall	Performance
BR	57.5%	83.1%
LS	60%	88.4%
LAM	64.5%	85.5%
HF	65.7%	90.1%
LS+HF	64.5%	92.7%
LAM+HF	68.8%	92.1%

Tracking performance after use context information, in the TUD campus data sets, the background is complex, many people on the move at the same time a few people leave the field of vision and new ones come into the view. It contains a number of challenges, such as the plane of rotation, completely occlusion and the target leaves the field of view. However, due to the use of misleading items and support items, our tracking system is easy to overlook other targets. In order to avoid the randomization, each of tracking run 5 times, observing that what makes tracking system can't return. The results are shown in Figure 3.

Our own video clips from the elevator outside the laboratory, the staff is complex, and lights is dimly. This sequence is very interesting and challenging, because the goals are similar clothing. When there is a change in appearance, the tracking system will jump to another target. Our tracking system is successful track the correct target until the end. Some results are shown in Figure 3. Noting that, in most cases, there is no a powerful context information, the track is still working, the results is better than use level set segmentation tracking system only, and superior to other state-of-the-art methods. Quantitative analysis (Table 3)

Comparison the running time, since the original reference for different method having different type of search range, thereby it greatly affecting the tracking speed. Due to the increase in candidate, so broaden the search range, which resulting the tracking speed is slower. Co-Tracker tracking system using the particle filter, the search range is also affected by the impact of the number of particles. Our method scans the whole image to find the candidate. The running time also depends on the tracking the number of misleading items. According to our observations under normal circumstances, there are few misleading items. Our tracking with the help of the context information has better overall performance than other methods. Although they may have a good performance in a constrained environment, but in a long sequence, and unconstrained environment, it is difficult to always track the target.

Table 2. The Average of Central Location Error（PNT：PNTracker, DNBS：DNBSTracker, COTT：CO-Tracker, MILT：MILTracker）

Sequence	Frame	OURS	PNT	DNBS	COTT	MILT
Animal	72	9	37	19	8	9
Clutter	1528	4	4	6	9	
Scale	1911	2	6		6	11
Speed	560	7	12	7	2	14

6. Conclusion

Tracking multiple targets in a complex environment facing various problems, the study use the improve the level set segmentation method combined with contextual information for robust tracking multiple targets in complex environments, effectively improve the efficiency of the track, for the target block, the deformation of the target and the target leaves the field of view all have a good tracking results. Experiments show that the method proposed in this chapter is practical and effective, effective for the real-time tracking of occlusion and deformation, when the target leave the field of vision, the tracker can flexible transfer the tracking to a new target.

Currently, our tracking system still has fault that is the running time is a little long,as level set segmentation use most of time and in the complex environment too many misleading items also may lead the running time longer. So in the future we will focus on improve the running time to handle this issue. Also we hope to use our method in more systems.

Reference

[1] V. Caselles, R. Kimmel and G. Sapiro, "Geodesic Active Contours", IJCV, vol. 22, **(1997)**.

[2] D. Cremers, M. Rousson and R. Deriche, "A Review of Statistical Approaches to Level Set Segmentation Integrating Color, Texture, Motion and Shape", IJCV, vol. 72, **(2007)**.

[3] C. Li, C. Xu and C. Gui, "Level Set Evolution without Re-initialization: A New Variational Formulation", CVPR, USA, **(2005)**.

[4] C. Bibby and I. Reid, "Robust Real-Time Visual Tracking using Pixel-Wise Posteriors", ECCV, France, **(2008)**.

[5] D. Cremers, "Dynamical Statistical Priors for Level Set Based Tracking", PAMI, vol. 28, **(2006)**.

[6] D. Cremers, "Nonlinear Dynamical Shape Priors for Level Set Segmentation", J. Sci. Comput., vol. 35, **(2008)**.

[7] D. Mitzel, E. Horbert and A. Ess, "Multi-Person Tracking with Sparse Detection and Continuous Segmentation", ECCV, Greece, **(2010)**.

[8] C. Bibby and I. Reid, "Real-time Tracking of Multiple Occluding Objects using Level Sets", CVPR, Greece, **(2010)**.

[9] S. K. Divvala, D. Hoiem and J. H. Hays, "An empirical study of context in object detection", CVPR, vol. 10, **(2009)**, pp. 1271-1278.

[10] L. J. Li, R. Socher and L. F. Fei, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework", CVPR, Miami, **(2009)**.

[11] D. Munoz, J. A. Bagnell and N. Vandapel, "Contextual classification with functional max-margin markov network", CVPR, Miami, **(2009)**.

[12] M. Ozuysal, P. Fua and V. Lepetit, "Fast keypoint recognition in ten lines of code", CVPR, Minneapolis, **(2007)**.

[13] D. A. Ross, J. Lim and R.S. Lin, "Incremental learning for robust visual tracking", IJCV, vol. 77, **(2008)**.

[14] H. Grabner, J. Matas and L. V. Gool, "Tracking the invisible: Learning where the object might be", CVPR, Greece, **(2010)**.

[15] M. Ozuysal, M. Calonder and V. Lepetit, "Fast keypoint recognition using random ferns", PAMI, vol. 32, **(2010)**.

[16] L. Breiman, "Random Forests", ML, vol. 45, **(2001)**, pp. 5-32.

[17] H. Bay, A. Ess and T. Tuytelaars, "SURF: Speeded up robust features", CVIU, vol. 110, **(2008)**, pp. 346-359.

[18] Z. Kalal, J. Matas and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints", CVPR, Greece, **(2010)**.

[19] T. B. Dinh, N. Vo and G. Medioni, "High resolution face sequences from a PTZ network camera", FG, Santa Barbara, **(2011)**.

Authors

Liu Meng, Ph.D, She received her M.Sc. in China University of Petroleum Beijing(2010), Now she study in University of Posts and Telecommunications, supervised by professor QingXuan Jia. Her research interest covers machine learning and machine vision.

QingXuan Jia, Ph.D., professor and doctoral supervisor. He received his M.Sc. and Ph.D. in Beijing University of Aeronautics and Astronautics master. Now he is a professor of Beijing University of Posts and Telecommunications, deputy director of national high-tech aerospace industry space robotics engineering research center. Member of United States Institute of Electrical and Electronics Engineers (IEEE), China Mechanical Engineers Senior Member, and served as the international and domestic academic conferences, members and group chairman several times.