

Sparse Representation Tracking with Auxiliary Adaptive Appearance Models

¹Junqiu Wang, ¹Chao Yang, ¹Junqiu Wang, ¹Zhichao Gan and ²Yasushi Yagi

¹Beijing Changcheng Aviation Measurement and Control Institute,
Aviation Industry Cooperation China Beijing, China

²The Institute of Scientific and Industrial Research, Osaka University, Osaka, Japan

Article history

Received: 04-10-2016

Revised: 18-10-2016

Accepted: 13-12-2016

Corresponding Author:

Junqiu Wang

Beijing Changcheng Aviation
Measurement and Control
Institute, Aviation Industry
Cooperation China, Beijing,
China

Email: jerywangjq@gmail.com

Abstract: We propose an effective tracking algorithm based on sparse representation and auxiliary adaptive appearance modeling. Based on a sparse representation, l_1 minimization can follow targets in challenging situations. Unfortunately, tracking approaches based on l_1 minimization are likely to be inefficient because they measure using dense coefficient distributions. The number of target candidates can be very large when the state space is densely sampled. Each minimization takes long time to find the solution. Traditionally, we must calculate the coefficients for each tracking candidates, which is computationally expensive. In this work, we found that l_1 minimization can be limited to a few regions with the reasonable probability based on adaptive appearance modeling and background probability estimation. Therefore, the computational cost is greatly reduced. We have also found that appearance information is useful for the region selection. We form the basis of appearance modeling using colors and shapes. The results of the experiment show that the proposed tracker has good performance.

Keywords: Tracking, Sparse Representation, Auxiliary Appearance Model, Robust Tracking

Introduction

Object tracking has many important applications such as surveillance, human-computer interface and robotics. A good tracker should possess properties such as high efficiency and accuracy. However, in real tracking tasks, many issues are hurdles for developing good trackers. For instance, viewpoint variations and illumination changes tend to make a good model invalid. Additionally, occlusions and unusual motions are also challenging problems for tracking.

Target representation is essential for developing a good tracker. Target appearance usually is not constant in an image sequence. The appearance changes gradually or suddenly as results of viewpoint or illumination variations. Since viewpoints and illuminations are in continuous spaces, the target appearance can have infinite possibilities. Despite from the large number of appearance variations, it has been found that these variations lie in a low dimensional space Tenenbaum *et al.* (2000; Gu *et al.*, 2008; Mei *et al.*, 2011). This space is well approximated by a set of templates. This observation is useful in tracking based on l_1 minimization.

l_1 minimization pursues sparse representations in a large space Donoho (2006). Matching probability in l_1 tracking is partly measured by the density of the coefficient distribution. A target candidate that can be sparsely described has a higher probability to be correctly recognized than those with dense non-zero small coefficients. A tracker based on l_1 minimization can deal with appearance variations, occlusions. Despite from the robustness of l_1 based trackers, they are not efficient due to the coefficient spanning calculation in a large space.

Based on appearance modeling approaches, object tracking algorithms can be classified into two categories: generative and discriminative trackers. Generative trackers model targets using representations that can reconstruct the targets. Such trackers do not consider the background information of the targets which can be helpful in distinguishing a target from its background. Discriminative trackers are effective in distinguishing targets and their background using the differences between foreground and background. However, discriminative trackers are short of the reconstruction ability of generative trackers. In this work, we aim at

combining the merits of discriminative and generative trackers in an integrated way.

Tracking based on $l1$ minimization belongs to generative trackers. We aim at developing a $l1$ tracker supported by discriminative approaches. The mean-shift algorithm Comaniciu *et al.* (2003) is a representative of discriminative trackers. Collins *et al.* (2005) select good color space combinations to improve discriminative ability. Wang and Yagi (2008) consider multiple shape and color features in an integrated framework. We found that the adaptive appearance modeling Wang and Yagi (2008) is helpful for including discriminative power in our tracker. We characterize target and its background using adaptive appearance modeling. Then, foreground likelihood images are computed based the appearance modeling. The searching space of the $l1$ minimization can be greatly constrained with the support from adaptive appearance modeling. We develop a robust and efficient tracker with adaptive appearance modeling and $l1$ minimization.

Particle filtering, a general Monte Carlo method, has been widely used in non-linear sequential state estimations Isard and Blake (1998; Toyama and Blake, 2001; Zhou *et al.*, 2004; Hastie *et al.*, 2003; Ross *et al.*, 2007; Wang and Yagi, 2009; Das *et al.*, 2012; Kim and Jeon, 2012). Particle filtering estimates the state of a sequential system that evolves in time series. Generally, the system measurement we get is polluted by noise. In order to correctly estimate the state of the system with noisy measurements, particle filtering uses many samples to represent state distributions. The computational cost of a tracker depends on the number of the samples. We constrain our particle filtering using adaptive appearance modeling Wang and Yagi (2008).

This paper is organized as follows. Section 2 introduces our particle filtering framework. Section 3 describes target modeling using a set of target templates and trivial templates. Trivial templates are used in describing occlusions in template matching. Adaptive appearance modeling is also introduced in section 3. Experimental results on real image sequences are demonstrated in Section 4. Section 5 summarizes this work.

Particle Filtering

Particle filtering estimates the state space based on the current and historical information of observations. The state of our tracking system evolves according to $x_t = f(x_{t-1}, u_{t-1})$, where x_t is a vector representing the state of the system at time t ; x_{t-1} is a vector representing the state of the system at time $t-1$, u_{t-1} is a noise vector, f_t is a non-linear time-dependent function. We get observation y_t from the system $y_t = g(x_t, e_t)$, where e_t is a non-linear transform function characterizing the measurement (Doucet *et al.*, 2001).

We estimate the distribution $p(x_t, |y_{0:t})$ using particle filtering. Here, we consider all the observations from y_0 to y_t , where $y_{0:t}$ are the historical observations. Particle filtering consists of two steps: predicting and updating the states. $p(x_t|y_{0:t})$ is computed at time $t-1$:

$$p(x_t | y_{0:t}) = \int p(x_{t-1} | y_{0:t-1}) p(x_t | x_{t-1}) \quad (1)$$

$p(x_{t-1}, |y_{0:t-1})$ is updated with the new observation y_t based on Bayes' rule:

$$p(x_t | y_{0:t}) \propto p(x_{t-1} | y_{0:t-1}) p(y_t | x_t) \quad (2)$$

Particle filtering represents the probability distribution, $p(x_t|y_{0:t})$, as a weighted particle set $\{s_t^{(k)}, \pi_t^{(k)}\}_{k \in \{1, 2, \dots, J\}}$. The weighted particle set consists of samples $s_t^{(k)}$ and their corresponding weights $\pi_t^{(k)}$. The samples in the weighted particle set are calculated using sequential importance re-sampling (Arulampalam *et al.*, 2002). J samples are drawn from the proposal distribution the samples are updated and propagated by the algorithm recursively:

$$p(x_t | y_{0:t}) \cong \sum_{k=0 \sim J} \pi_{t-1}^{(k)} \delta(x_{t-1} - s_{t-1}^k) \quad (3)$$

Given the distribution $p(x_{t-1}|y_{0:t-1})$, particle filtering calculates the filtered distribution $p(x_t|y_{0:t})$. Embedding the weighted particle set into the above equation, we obtain the approximation of $p(x_t|y_{0:t})$:

$$p(x_t | y_{0:t}) \propto p(y_t | x_t) \sum_k \pi_{t-1}^{(k)} p(x_{t-1} | x_{t-1}^k) \quad (4)$$

Target Modeling

We represent the target using a generative method. Correct candidates can be found by $l1$ minimization. We also characterize target and its background using adaptive appearance modeling (Wang and Yagi (2008)), which is a discriminative modeling approach. The auxiliary adaptive appearance modeling better allows for the acceleration of the $l1$ minimization in particle filtering by shrinking the state space of the target.

Generative Target Modeling

We have a few modeling templates denoted by $m_i \in \mathfrak{R}^d$, where d denotes the size of a template. They are used to create a matrix $M = [m_1, m_2, \dots, m_n]$, where $M \in \mathfrak{R}^{d \times n}$.

Considering all the modeling templates, We search for a good target candidate y^* from a set of candidates y . Ideal reconstruction is defined as $y = M\alpha$, where $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)^T$. Here, α_i indicates the contributions of different templates in modeling the candidates.

The reconstruction in $y = M\alpha$ does not consider any noise or occlusion. Unfortunately, observation noise and corruptions are unavoidable in real tracking tasks, which make ideal reconstruction impossible. We have to introduce a noise vector v and trivial template set β for dealing this problem. In each trivial template, only one non-zero element is permitted. Combining many such trivial templates, we can describe noise and occlusion in the reconstruction here, the template set β represents corruptions. A coefficient vector α and a trivial coefficient vector v are put into a matrix $\begin{bmatrix} \alpha \\ v \end{bmatrix}$. A tracking result is approximated by $y = [M, \beta] \begin{bmatrix} \alpha \\ v \end{bmatrix}$.

To reconstruct y , we can calculate the coefficients in α . The values in α should be constrained as nonnegative values because the templates bearing similarity to the tracking target are positively related to the target. The coefficient vector α can be constrained in the reconstruction. However, it is not easy to enforce non-negativity on the trivial coefficient vector v . To solve this problem, trivial coefficient vector v is extended by including the negative trivial templates. Here, target templates M and trivial templates set β are combined to form a matrix $\Lambda = [M, \beta, \beta^{-1}]$, where $-\beta$ defines the negative β . v is also extended to v^+ and v^- , where v^+ is a positive trivial coefficient vector; and v^- is a negative trivial coefficient vector. Coefficients α and trivial noise v^+ and v^- are put into a matrix $\varepsilon = [a^T, v^+, v^-]$. The reconstruction is computed by:

$$y = [M, \beta, -\beta] \begin{bmatrix} a \\ v^+ \\ v^- \end{bmatrix} \quad (5)$$

There are few coefficients with large values for reasonable reconstructions. In the optimization process, we have to keep the number of the large coefficients to be few. This is more difficult than only considering the reconstruction error. To retain a few non-zero coefficients and keep others zero, the problem is transformed into a minimization with a penalized residual sum of absolute values of ε Hastie *et al.* (2003).

We search for good candidate by minimizing an objective with $l1$ penalty Tibshirani (1996; Koh *et al.*, 2007a; 2007b):

$$(\alpha, v) = \arg \min \|\Lambda \varepsilon\|_2^2 + \lambda \|\varepsilon\|_1 \quad (6)$$

Subject to $\varepsilon \succeq 0$

where, $\Lambda = [M, \beta, -\beta]$ consists of target templates M , trivial templates set β and the negative of trivial template

set $-\beta$; λ is a parameter adjusting the complexity level of the solution. There are two kinds of norms in Equation 6: the first $l2$ norm gives penalty on the difference between the template and the candidates; and the second norm considers the coefficients of the sum of absolute values of ε by a $l1$ norm. The coefficients solved in Equation 6 are $l1$ -regularized least squares solutions. The $l1$ -regularized least squares solutions tend to be sparse because the compressibility in the transform domain in solving the problem. The solution become sparser if we set λ to a large value. However, if λ is too large, the solution obtained in Equation 6 is not meaningful. λ also influences the runtime of the minimization process. In this work, we set λ according to the suggestions given in (Koh *et al.*, 2007b). For each minimization, we set λ to $0.1\lambda_{\max}$ the number λ_{\max} gives us an upper bound on the useful range of the regularization parameter λ , which can computed as described in Koh *et al.* (2007b).

Efficient solution to $l1$ minimization has been proposed in Koh *et al.* (2007a; 2007b). Despite from these development, it is still computationally expensive for our tracking tasks. We will introduce adaptive appearance modeling to constrain the cost in section 3.4.

Generative $l1$ minimization

We can find target candidates by directly comparing the pixel difference between a target template and its candidates in images. Unfortunately, such kind of matching is very sensitive to noise. In this work, we search for sparse coefficients for appearance recovery. The residual errors are defined as the difference between the templates and target candidates.

The minimization in Equation 6 can be performed using Forward Stage-wise Linear Regression (FSLR) method Hastie *et al.* (2003). However, this approach is time consuming. Many alternatives have been proposed to speed up the minimization process. We solve the $l1$ minimization problem using the truncated Newton interior-point method Tenenbaum *et al.* (2000). Logarithmic barriers are defined for bound constraints in this method. In the iterative solving process, search directions are computed using preconditioned conjugate gradients Tenenbaum *et al.* (2000).

Matching Probability in Generative Reconstruction

We compute matching probability of a target candidate based on the reconstruction results:

$$p(y_i | x_i) = \frac{\exp \tau(M\alpha - y_i)}{Z} \quad (7)$$

where, α is calculated in the $l1$ minimization, t is a negative coefficient restricting the exponential function; Z is a normalization factor.

Adaptive Appearance Modeling in a Discriminative Way

We represent target candidates using a set of particles. We compute the reconstruction errors for every target candidates if no other information is taken into consideration. Since the number of the particle set can be large, the computational cost related to candidate matching using l_1 minimization is rather high. We would like to alleviate this problem by constraining particle filtering based on foreground likelihood images produced by appearance model back-projection.

Wang and Yagi (2008) proposed a tracking algorithm by selecting reliable features from color and shape-texture cues. The selection is performed according to feature descriptive ability. The target model is updated based on the similarity between the first and most recent models. Thanks to the multi-feature selection mechanism, the target can be characterized effectively. We adopt the feature selection approach in Wang and Yagi (2008) to incorporate appearance guidance in our particle filtering framework.

We compute weighted histogram for multiple cues. Then, log-likelihood ratio images are generated based on these histograms, which is described in Equation 8. We measure the descriptive ability of a certain cue according to its variance ratio. The discriminative features are selected for foreground likelihood computation. Foreground and background histograms are denoted by ξ_F and ξ_B , respectively. Foreground likelihood ratios can be generated by back-projecting the histograms onto the input image Wang and Yagi (2008):

$$\eta^{(bin)} = \max(-1, \min(1, \log \frac{\max(\xi_F^{(bin)}, \sigma_\eta)}{\max(\xi_B^{(bin)}, \sigma_\eta)})) \quad (8)$$

where, σ_η is a very small number.

Log likelihood images are helpful for reducing the searching space of particle filtering. We show a foreground likelihood image in Fig. 1. The distribution of target candidates is closely related the likelihood ratios in a certain region. The samples outside the foreground likelihood distributions are not helpful in particle filtering. We do not need to consider samples in such regions.

We calculate the sum of the likelihood ratios for each candidate region $S_\eta = \sum_{i \in R} \eta_i$. We sort all the likelihood ratio sums in all the candidate regions. The sum calculation and sorting are very efficient.

Candidate matching can be performed by measuring l_2 norm of the difference between the templates and candidates. l_2 norm computation is also more efficient than l_1 minimization. The computation cost of particle filtering can be reduced by applying l_2 norm computation. l_1 minimization is bounded by l_2 norm computation Mei *et al.* (2011):

$$\|Ma - y_i\|_2^2 \geq \|M\hat{a} - y_i\|_1 \quad (6)$$

where, $\hat{a} = \|Ma - y\|_2^2$.

Although l_2 norm is helpful for restricting l_1 minimization cost, it is still less efficient than the foreground likelihood checking strategy. Computational cost analysis will be given in Section 4.2.



Fig. 1. (a) Target tracking result using the VRMS method (Collins *et al.* (2005)). (b) Foreground likelihood image calculated using the method in Wang and Yagi (2008)

Experimental Results

We implemented the proposed tracking method and compare it with a few good trackers. We compare the proposed tracker with a sparse minimization based tracker, the structural local sparse appearance model-based tracker Jia *et al.* (2012) (Struct). This tracker bears certain similarity with ours in the aspect of the sparse representation. We also compare with the visual tracking decomposition tracker Kwon and Lee (2010) (VDT). This tracker is good at dealing with occlusions. Other baseline trackers used in the experiments include the Variance Ratio Mean-Shift tracker (VRMS) Collins *et al.* (2005), which is an extension of the mean-shift algorithm Comaniciu *et al.* (2003); the Incremental Visual Tracking algorithm (IVT), which compute an incremental Principle Component Analysis for target modeling; and the fragment-based tracking method Adam *et al.* (2006) (Frag), which utilizes a voting technique in the localization process.

We have tested the trackers on many image sequences. We evaluate the trackers on challenging sequences. We can find typical difficulties for visual tracking in these sequences: Cluttered background, heavy occlusions, shape deformations and background motion. We also include 1 sequence which seems to be simple. Actually, complicated trackers can fail on simple sequences, which is out of our expectation. We have tested the trackers for many sequences. Here we show 6 typical sequences to demonstrate the performance of our algorithm.

In the first sequence, a bird is tracked in the images with shape and appearance variations. The bird in this sequence changes its pose in a large degree. In addition, it is occluded by other objects in some frames. The Struct tracker give very good performance in this sequence. Our results are a little better than the Struct tracker anyway in this sequence. The Frag tracker's performance is also reasonable although it is a relatively simple tracker.

We track a vehicle in the second sequence (Fig. 3a) the vehicle has a distinctive appearance. The difficulty of this sequence is that the scale of the vehicle always changes. It is small at the beginning of the tracking and becomes large after a few frames and then becomes small again. The Struct tracker become unstable from frame 420 because of the viewpoint variations. It fails at frame 907 due to scale variations. The VRMS tracker does not give good estimation on the scale. The IVT tracker is better in this aspect. However, both of VRMS and IVT lost tracking after a few frames. Although the IVT tracker found the vehicle again by chance in frame 600, it lost tracking again soon. The proposed tracker follows the position and scale of the vehicle well.

A face is tracked in the third sequence (Fig. 3b). The face is occluded by a toy during the tracking. Since the VRMS tracker is not good in dealing with occlusion, the

positions of the face provided by the VRMS deviate from the true positions. The IVT tracker is better in handling occlusions. Both the IVT and our trackers give good estimations under heavy occlusions. The VDT tracker successfully deals with the heavy occlusions in many frames. It fails at the end of the sequence due to the fast motion of the face.

We track another face in the fourth sequence. The face moves forward and backward during the tracking. The *l1*, Struct, VTD, IVT trackers can handle the pose and scale variations before frame 400. The MIL tracker fails due to another face's distraction. The VRMS tracker cannot follow the face after a few frames. Although the IVT tracker follows the face in some frames, its results are not as precise as those given by the proposed tracker.

We track a basketball player in the fifth sequence. The player has large pose variations. In addition, there are sudden illumination changes in the sequence. The VDT tracker and our tracker give good performance in this sequence.

We track a woman in the sixth sequence. The woman is occluded by several cars in the sequence. The VDT and Struct trackers failed during the tracking due to the heavy occlusions.

Qualitative Comparison

We measure the tracking accuracy of the trackers. We compute the error based on the L2 distance between the centers of tracking result bounding boxes and the centers of ground truth bounding boxes. The evaluation results are shown in Table 1. The proposed tracker gives low errors in a few sequences. However, it is not always the most accurate tracker. For example, the tracking results of GirlFace sequence are worse than those of the Struct tracker. In other sequences, our tracker gives the best tracking accuracy. The VRMS usually gives the worst results compared with other trackers. However, it is much more efficient than other trackers.

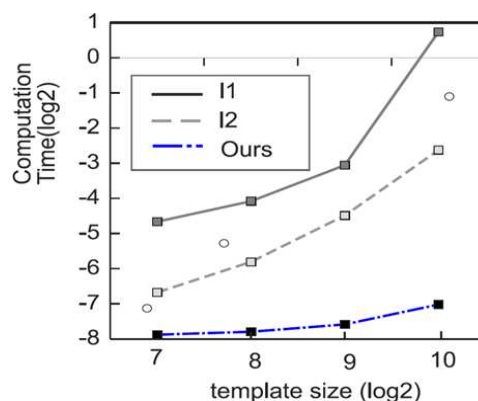


Fig. 2. Computational cost comparison of different approaches. The vertical scale is the log10 time of the running time of each algorithm



Fig. 3. Tracking results qualitative comparison on difference sequences

Table 1. The average tracking errors of the trackers

Tracker	IVT	Frag	VRMS	Struct	VDT	Ours
Aerial	36	9	9	11	8	4.2
BasketBall	3	12	26	17	11	7
GirlFace	16	23	16	6	11	7
Woman	123	107	136	9	192	6
Bird	212	18	128	15	131	12
ManFace	32	11	25	7	10	6

Table 2. Tracking time comparison (frames/second)

Tracker	IVT	Frag	VRMS	Struct	VDT	Ours
Aerial	11.8	2.4	45.6	3.5	0.3	12.8
BasketBall	10.7	2.5	26.1	3.6	0.3	12.3
GirlFace	9.8	2.7	42.3	5.0	1.9	11.7
Woman	10.9	2.8	38.6	3.6	0.7	12.5
Bird	8.9	2.6	26.1	3.7	0.7	10.6
ManFace	8.7	2.4	24.3	4.2	0.3	10.6

Computational Complexity Analysis

We test the likelihood ratios for each target candidates. The computational cost of our approach only depends on the size of the candidate regions d and the number of samples J in particle filtering. The cost is linear to both of the factors, which is $O(dJ)$. This computation is extremely fast in practice.

The minimization using l_2 norm is simpler than using l_1 minimization. However, it is more expensive than our foreground likelihood test. Although the complexity of l_2 norm computation is also $O(dJ)$, it takes more flops in practice. The l_2 computation is more involved than our foreground likelihood ratio test.

The l_1 minimization problem can be solved by a truncated Newton Interior-point method Koh *et al.* (2007a; 2007b). In the iterative computation, the direction searching is performed by using a preconditioned conjugate gradients method. The computational cost of this method depends on the iteration number required by the truncated Newton interior-point method. It takes a few hundred iterations to reach the solution. The complexity of the preconditioned conjugate gradients method is $O(d^2+dJ)$, which is much more expensive than our foreground likelihood test.

The number of target templates usually is less than the dimensionality of the templates. Therefore, the size of the templates has a large effect on the computational cost of l_1 minimization. In Fig. 2, we compare the computational cost of foreground likelihood checking, l_2 norm computation and l_1 minimization. We use 600 samples in our particle filtering process. The computational costs of different template sizes are shown for all the three approaches. Since the difference between the methods is rather large, we demonstrate the results in log domain. Obviously, all computational costs increase with larger template size. Foreground likelihood checking is the most efficient method compared with the other two methods and the l_2 norm computation takes less time than the l_1 minimization.

We also perform quantitative efficiency comparison of the trackers. Tracking time is indicated by frames per second. We run all the trackers on a computer with Windows 8 system, i5 CPU and 4G RAM. We mark the tracker with the best tracking efficiency with bold fonts and mark the second best tracker using italic fonts. It is clear that the VRMS tracker is far more efficient than all the other trackers. The efficiency of this appearance based mean-shift tracker indicates the importance of overall appearance modeling. Our tracker is the second best in all the sequences. The computational cost of our tracker is much more expensive than that of the VRMS. However, it provides much better performance than

the VRMS. It is more efficient than other trackers (except the VRMS).

Conclusion

We estimate target likelihood image based on adaptive appearance modeling. We represent target motion using less particles with the support from target likelihood image. The particle filtering in this framework is computationally easier compared with matching all the target candidates using expensive minimization process. The proposed method can deal with target appearance and scale variations. It is also good at handling occlusions.

Funding Information

The authors gratefully acknowledge the Ministry of Science and Technology, China, for providing support under 2013YQ350747.

Author's Contributions

Junqiu Wang: He designed the research, implemented the system, did the experiments. He also prepared the initial draft of the manuscript.

Chao Yang: He co-designed the research. He made considerable contributions to the experiments.

Zhichao Gan: He made considerable contributions to designing the experimental plan.

Yasushi Yagi: He made considerable contributions to designing the research plan and experimental procedure.

Ethics

This paper is original and includes unpublished materials. The corresponding author confirms that all other authors have read and approved the manuscript.

References

- Adam, A., E. Rivlin and I. Shimshoni, 2006. Robust fragments-based tracking using the integral histogram. Proceedings of the Computer Vision and Pattern Recognition (CVPR' 06), IEEE Xplore Press, USA, pp: 798-805.
DOI: 10.1109/CVPR.2006.256
- Arulampalam, S. and S. Maskell, N. Gordon and T. Clapp, 2002. A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking. IEEE Trans. Signal Process., 50: 174-188.
DOI: 10.1109/78.978374
- Collins, R.T., Y. Liu and M. Leordeanu, 2005. On-line selection of discriminative tracking features. IEEE Trans. Patt. Anal. Mach. Intell., 27: 1631-1643.
DOI: 10.1109/TPAMI.2005.205

- Comaniciu, D., V. Ramesh and P. Meer, 2003. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25: 564-577.
DOI: 10.1109/TPAMI.2003.1195991
- Das, S., A.A. Kale and N. Vaswani, 2012. Particle filter with a mode tracker for visual tracking across illumination changes. *IEEE Trans. Image Process.*, 21: 2340-2346. DOI: 10.1109/TIP.2011.2174370
- Donoho, D., 2006. Compressed sensing. *IEEE Trans. Inform. Theory*, 52: 1289-1306.
DOI: 10.1109/TIT.2006.871582
- Doucet, N., Freitas and N. Gordon, 2001. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York. ISBN-13: 978-0387951461
- Gu, J., S. Nayar, E. Grinspun, P. Belhumeur and R. Ramamoorthi, 2008. Compressive structured light for recovering inhomogeneous participating media. *IEEE Trans. Patt. Anal. Mach. Intell.*, 35: 1.
DOI: 10.1109/TPAMI.2012.130
- Hastie, T., R. Tibshirani and J.H. Friedman, 2003. *The Elements of Statistical Learning*. 2nd Edn., Springer Verlag, ISBN-13: 978-0387848570
- Isard, M. and A. Blake, 1998. Condensation-conditional density propagation for visual tracking. *Int. J. Comput. Vis.*, 29: 5-28.
- Jia, X., H. Lu and M.H. Yang, 2012. Visual tracking via adaptive structural local sparse appearance model. *Proceedings of the Computer Vision and Pattern Recognition (CVPR' 12)*, Providence, RI, pp: 1822-1829.
DOI: 10.1109/CVPR.2012.6247880
- Kim, D.Y. and M. Jeon, 2012. Spatio-temporal auxiliary particle filtering with l_1 -norm-based appearance model learning for robust visual tracking. *IEEE Trans. Image Process.*, 22: 511-522.
DOI: 10.1109/TIP.2012.2218824
- Koh, K., S.J. Kim and S. Boyd, 2007a. An efficient method for large-scale l_1 -regularized convex loss minimization. *Proceedings of the IEEE Information Theory and Applications Workshop, (TAW)*, pp: 223-230.
- Koh, K., S.J. Kim and S. Boyd, 2007b. An interior-point method for large-scale l_1 -regularized logistic regression. *J. Mach. Learn. Res.*, 8: 1519-1555.
- Kwon, J. and K.M. Lee, 2010. Visual tracking decomposition. *Proceedings of the Computer Vision and Pattern Recognition*, Jun. 16-21. IEEE Xplore Press, San Francisco, CA, pp: 1-8.
DOI: 10.1109/CVPR.2010.5539821
- Mei, X., L. Ling, Y. Wu and E.P. Blasch, 2011. Minimum error bounded efficient l_1 tracker with occlusion detection. *IEEE Trans. Image Process.*, 22: 2661-2675.
DOI: 10.1109/TIP.2013.2255301
- Tenenbaum, J.B., V.D. Silva and J.C. Langford, 2000. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290: 2319-2323.
DOI: 10.1126/science.290.5500.2319
- Ross, D., J. Lim, R.S. Lin and M.H. Yang, 2007. Incremental learning for robust visual tracking. *Int. J. Comput. Vis.*, 77: 125-141.
DOI: 10.1007/s11263-007-0075-7
- Tibshirani, R., 1996. Regression shrinkage and selection via the lasso. *J. Royal Stat. Society*, 58: 269-288.
DOI: 10.1111/j.1467-9868.2011.00771.x
- Toyama, K. and A. Blake, 2001. Probabilistic tracking with exemplars in a metric space. *Int. J. Comput. Vis.*, 48: 9-19. DOI: 10.1023/A:1014899027014
- Wang, J. and Y. Yagi, 2008. Integrating color and shape-texture features for adaptive real-time object tracking. *IEEE Trans. Image Process.*, 17: 235-240.
DOI: 10.1109/TIP.2007.914150
- Wang, J. and Y. Yagi, 2009. Adaptive mean-shift tracking with auxiliary particles. *IEEE Trans. Syst. Man Cybernet.*, 39: 1578-1589.
DOI: 10.1109/TSMCB.2009.2021482
- Zhou, S.K., R. Chellappa and B. Moghaddam, 2004. Visual tracking and recognition using appearance-adaptive models in particle filters. *IEEE Trans. Image Process.*, 11: 1491-1506.
DOI: 10.1109/TIP.2004.836152