International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.9, No.9, (2016), pp.113-122
http://dx.doi.org/10.14257/ijsip.2016.9.9.11

An Overview of Compressive Trackers

Zhongyi Hu[1], Lei Xiao[2] and Fei Teng[3]

[1]Intelligent Information Systems Institute, Wenzhou University, Wenzhou 325035, Zhejiang, China
[2]School of Marine Science and Technology, Northwestern Polytechnic University, Xi'an 710072, Shanxi, China
[3]School of Energy and Power Engineering, Wuhan University of Technology, Wuhan 430070, Hubei, China
[1]hujunyi@foxmail.com, [2]xiaolei@wzu.edu.cn, [3]changeerhao_love@126.com

Abstract

Compressive tracking is considerably popular in the visual tracking community in recent years. The very strong theoretic support from compressive sensing motivates many researchers to follow and there are a wide range of compressive trackers with attractive performances. The goal of this paper is to overview some of the most recent state-of-the-art compressive trackers in the literature. First, a variety of compressive trackers are thoroughly introduced and summarized. Second, extensive analyses from different perspectives, including random measurement matrix, compressive features, feature selection strategy and so forth, aim to provide readers a good understanding of the strengths and weaknesses of different trackers. Finally, several possible future trends for compressive trackers are outlined to hopefully bring some insights to interesting researchers.

Keywords: *compressive sensing, compressive trackers, random measurement matrix, compressive features, feature extraction, feature selection*

1. Introduction

Compressive tracking becomes an important branch among the extremely active visual tracking community and there are a wide range of compressive trackers demonstrating superior performances in diverse real-word applications. Zhang *et. al.*, [1] propose the compressive tracking (CT) framework with an appearance model based on features extracted in the compressive domain. This milestone work first employs a sparse non-adaptive measurement matrix to extract the low-dimensional features from a multi-scale image feature space with data-independent basis. Then the compressive features are classified via a naive Bayes classifier with online update. [2] extends CT with a coarse-to-fine search strategy to speed up the time-consuming detection procedure.

In order to improve the CT's performance when the target object suffers illumination variation and cluttered background, extracting more discriminative compressive features is crucial. [3] presents a compressive tracker with its appearance model based on fern features in the compressive domain in order to track in scenes with light and texture changing. Sun *et. al.*, [4] develop a compressive tracker by taking the histogram of gradient (HOG) features instead of the generalized Haar features to achieve favorable performance when the target object undergoes illumination. Teng *et. al.*, [5] propose a compressive tracer that employs two kinds of random measurement matrices to extract two complementary good features to track the target object. Lu *et. al.*, [6] makes the visual representation more abundant through compressing both intensity and speed up robust features (Surf) which have strong power to describe the detailed information like gradient and edge. [7] integrates the sample importance into CT within an online multiple

ISSN: 2005-4254 IJSIP

instance learning (MIL) strategy and employs the co-training criterion into CT tracker to improve the tracking performance. Zeng *et. al.*, [8] propose a compressive tracker whose classifier is trained by positive and negative samples that are weighted according to their similarity with the constant holistic appearance model. Huang *et. al.*, [9] perform frequent affine transformations of training samples to obtain more reliable compressive features.

In order to improve the CT's performance when the target object undergoes extreme scale change, Zhang *et. al.*, [10] construct a set of measurement matrices of different scales offline to perform scale-adaptive compressive tracking. In [11-12], the motion information has been integrated into appearance model by introducing motion estimator, *i.e.*, optical flow to improve the performance of trackers especially when the target is with motion variety. Wu *et. al.*, [13] develop a compressive tracker that integrates an improved appearance model based on normalized rectangle features extracted in the adaptive compressed domain into the bootstrap filter. There are also some other work that naturally integrates their compressive trackers into Kalman filter framework [14] and particle filter framework [15-17].

In order to improve the CT's performance when the target object suffers partial or even full occlusion, part-based appearance models are usually constructed. Zhu *et. al.*, [18] propose a compressive tracking method based on oversaturated sub-region classifiers. [19] takes advantage of the existing online learning appearance model to learning the appearance of each and every part and an affine invariant structural constrains between these parts are online learnt. In [20], the integrated sparse representation combining texture, intensity and local special information from sub-regions is proposed to model the target object.

Furthermore, Luo *et. al.*, [21] utilize sparse Toeplitz projection matrix with random pitches to extract the compressive features. Then, Mean Shift algorithm is used to compute the object candidates' weights and the weighted Bayes classifier is used to determine the reliable object location. In [22], a compressive tracker based on phase congruency is proposed. First, the phase congruency transformation of the image in the search area is calculated. And then the extracted features from the transformed image are used in the classifier to determine the location of the target object. It is well worth noting that there are also some compressive trackers [23-27] that are designed for some specific application.

2. Compressive Tracking Framework

In this section, we will give a brief introduction of CT framework. In [1], a very sparse measurement matrix is constructed to extract the compressive features for the appearance model based on compressive sensing theory:

$$\mathbf{v} = \boldsymbol{R}\mathbf{x} \tag{1}$$

Where $\mathbf{x} \in \mathbb{R}^m$ corresponds to a high-dimensional image space by convolving image patch $\mathbf{z} \in \mathbb{R}^{w*h}$ with a set of rectangle filters defined as:

$$h_{p,q}(x,y) = \begin{cases} 1 & x_i \le x \le x_i + p \;\; y_i \le y \le y_i + q \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

Where (x_i, y_i) is the upper left coordinate of the rectangle filter, $p \in [1, w]$ and $q \in [1, h]$ are the width and height of the rectangle filter respectively. $\mathbf{v} \in \mathbb{R}^n$ corresponds to a low-dimensional space and $\boldsymbol{R} \in \mathbb{R}^{n*m}$ represents the random measurement matrix with entries defined as:

$$r_{i,j} = \sqrt{s} * \begin{cases} 1 \quad \text{with probability } \dfrac{1}{2s} \\ 0 \quad \text{with probability } 1-\dfrac{1}{s} \\ -1 \quad \text{with probability } \dfrac{1}{2s} \end{cases} \tag{3}$$

Practically, the compressive features are a linear combination of generalized Haar-like features:

$$v_i = \sum_{k=1}^{NR} r_{i,k} Rect_{i,k} \tag{4}$$

Where Rect is the randomly generated rectangles and *NR* is the number of rectangles, $r_{i,k}$ is randomly produced from 1 to -1. Then by assuming all elements in $\mathbf{v}$ are independently distributed, the confidence $H(\mathbf{v})$ can be achieved via naive Bayes classifier:

$$H(\boldsymbol{v}) = \sum_{i=1}^{n} \log\left(\frac{\mathrm{p}(\mathrm{v}_i \mid Y=1)}{\mathrm{p}(\mathrm{v}_i \mid Y=0)} \right) \tag{5}$$

$Y \in \{0,1\}$ represents the binary sample label. The conditional distributions for both the molecular and denominator in Equation 5 are assumed to be Gaussian distribution where:

$$\begin{aligned} p(v_i \mid Y=1) &\sim N\left(\mu_i^1, \sigma_i^1\right) \\ p(v_i \mid Y=0) &\sim N\left(\mu_i^0, \sigma_i^0\right) \end{aligned} \tag{6}$$

The classifier parameters are updated by:

$$\begin{aligned} \mu_i^1 &\leftarrow \lambda \mu_i^1 + (1-\lambda)\mu^1 \\ \sigma_i^1 &\leftarrow \sqrt{\left(\lambda\left(\sigma_i^1\right)^2 + (1-\lambda)\right)(\sigma^1)^2 + \lambda(1-\lambda)\left(\mu_i^1 - \mu^1\right)^2} \end{aligned} \tag{7}$$

$\lambda > 0$ is the learning parameter, $\sigma^1 = \sqrt{\sum_{k=0|Y=1}^{n-1} \left(\mathrm{v}_i(k) - \mu^1\right)^2}$ and $\mu^1 = \frac{1}{n}\sum_{k=0|Y=1}^{n-1} \mathrm{v}_i(k)$. Finally, the maximal response classified by H represents the tracking location in the current frame.

3. Compressive Trackers

In this section, we will focus on introducing and summarizing some most recent state-of-the-art compressive trackers from different perspectives, including random measurement matrix, compressive features, feature selection strategy and so forth, they are sequentially discussed in Section 2.1 to Section 2.3 respectively.

3.1. Random Measurement Matrix and Compressive Features

Random projection and random measurement matrix are two most important elements involved with compressive trackers. Ideally, we expect measurement matrix $\boldsymbol{R}$ satisfies the Johnson-linden Strauss (JL) lemma [28] so that $\mathbf{x}$ can be reconstructed with minimum error from $\mathbf{v}$ with high probability if $\mathbf{x}$ is sparse signal such as audio or image. A variant condition of JL lemma in compressive tracking is restricted isometry property (RIP) [29] that approximately preserves the distances between any pairs of sparse signals when projecting those high-dimensional signals onto low-dimensional space.

A typical measurement matrix that satisfies RIP is the Gaussian matrix $\boldsymbol{R}^G \in \mathbb{R}^{n*m}$ where $r_{ij} \sim N(0,1)$, as used in some compressive sensing-based work [30-32]. However, Gaussian measure matrix is very dense, the memory and computational loads are very expensive when m is large. To ease this problem, [1-3] and [13] employ the very sparse random measurement matrix $\boldsymbol{R}^S$ defined in Equation 3. When $s=3$, $\boldsymbol{R}^S$ satisfies the JL lemma, and two thirds of the computation can be avoided. In [33], it is proved that for $s=o(m)$, the random projections are almost as accurate as the random Gaussian measurement matrix.

Due to the fact that compressive trackers only need to extract almost all the information of the original image patch and not need to reconstruct the original image with minimum error from the low-dimensional features, there are some random measurement matrices that obtain favorable results but do not satisfy JL lemma. [21] constructs a sparse Toeplitz measurement matrix $\boldsymbol{R}^T$ with random pitches:

$$\boldsymbol{R}^T = \begin{bmatrix} r_1 & r_2 & r_{\dots} & r_N \\ r_{N+1} & r_1 & r_{\dots} & r_{N-1} \\ r_{\dots} & r_{\dots} & r_{\dots} & r_{\dots} \\ r_{N+M-1} & r_{N+M-2} & r_{\dots} & r_{N-M+1} \end{bmatrix} \tag{8}$$

In $\boldsymbol{R}^T$, $\boldsymbol{R}^T_{i+1,j+1} = \boldsymbol{R}^T_{i,j}$ and r_i ($i \in k$, k is the selected $\frac{N+M-1}{\Delta}$ indexes from 1 to $N+M-1$ and Δ is the random pitch) is independent and identically distributed Gaussian. [21] also proves that Toeplitz measurement matrix can obtain lower construction error than Gaussian measurement matrix, and the overall computational complexity is about $1/\Delta$ of the Gaussian measurement matrix. [20] utilizes a sparse measurement matrix $\boldsymbol{R}^P$ defined as:

$$r_{i,j} = \sqrt{s} * \begin{cases} 1 & \text{with probability } \frac{0.22}{s} \\ 0 & \text{with probability } 1-\frac{1}{s} \\ -1 & \text{with probability } \frac{0.78}{s} \end{cases} \tag{9}$$

Compared to $\boldsymbol{R}^S$ which satisfies JL lemma, only the signs of some elements are changed (from +1 to -1, or from -1 to +1). The cost for construction does not increase, but yields superior performance.

Different measurement matrices extract different compressive features, and further construct different appearance models. More specifically, by Equation 4, the compressive features are a linear combination of generalized Haar-like features. For measurement matrices $\boldsymbol{R}^G$ and $\boldsymbol{R}^T$, the compressive features are directly weighted sum of some trivial templates. For each row r_i in $\boldsymbol{R}^S$, the possibility p_i that elements in r_i are or 1 or all -1 can be calculated as:

$$\begin{aligned} p_i &= \sum_{j=2}^{4} p(N=j) \prod_{k=1}^{N} p(p_{i,k}=1) + \sum_{j=2}^{4} p(N=j) \prod_{k=1}^{N} p(p_{i,k}=-1) \\ &= \left[\frac{1}{3}*\left(\frac{1}{2}\right)^2 + \frac{1}{3}*\left(\frac{1}{2}\right)^3 + \frac{1}{3}*\left(\frac{1}{2}\right)^4\right]*2 \approx 0.3 \end{aligned} \tag{10}$$

Since $\mathbf{v}_i$ is the weighed sum of N rectangle filters, so if all elements in r_i are 1 or -1, $\mathbf{v}_i$ reflects the homogeneity between the N rectangle filters, with possibility of 0.3. On the contrary, if elements in r_i have both 1 and -1, then $\mathbf{v}_i$ reflects the heterogeneity between the N rectangle filters, with possibility of 0.7. Therefore $\mathbf{v}_i$ highlights the difference among the randomly generated patches, which is in a way similar to texture features [5]. However, for each row r_i in $\boldsymbol{R}^p$, it can easily be computed by Equation 10 that $p_i \approx 0.5$ now, meaning that both the texture and intensity features are given equal probability when performing random projections.

Moreover, there are some further efforts to extract the more discriminative compressive features. [27] and [5] first employ $\boldsymbol{R}^S$ (defined as Equation 3) to extract the compressive features which emphasize more on texture, and then employ another measurement matrix $\boldsymbol{R}^{S'}$ (defined as Equation 11) to extract the compressive features which emphasize more on intensity.

$$r_{2i} = \begin{cases} |r_{1i}| & r_{1i} \text{ are all 1 or } -1 \\ (-1)^k \, |r_{1i}| & \text{otherwise} \end{cases} \tag{11}$$

Finally, these two compressive features are concatenated. Furthermore, [6] presents a two-stage measurement matrices (defined as Equation 12) to extract the compressive Surf features which emphasize more on gradient and edge features of the target object.

$$\mathrm{r}_{i,j} = \sqrt{s} * \begin{cases} \mathbf{r}^{1*4} & \text{with probability } \frac{1}{2s} \\ \mathbf{0}^{1*4} & \text{with probability } 1\text{-}\frac{1}{s} \\ -\mathbf{r}^{1*4} & \text{with probability } \frac{1}{2s} \end{cases} \qquad \mathrm{r}_k = \begin{cases} 1 & \text{with probablility } 1/4 \\ 0 & \text{with probability } 3/4 \\ & k = 0,1,2,3 \end{cases} \tag{12}$$

3.2. Feature Selection Strategy

Having obtained the low-dimensional compressive features for the target object via efficient feature extraction, several feature selection techniques are commonly used in recent compressive trackers in the literature. Overall, feature selection strategy is employed to favor the more discriminative features in the feature pool, thus constructing a more effective and stable appearance model. [20] uses online MIL feature selection strategy to sequentially chosen K most discriminative features h_k from the feature pool by keeping minimizing the loss function, which is defined by:

$$L^n = -\sum_i (y_i \log(p_i^n) + (1-y_i)\log(1-p_i^n)) \tag{13}$$

Where p_i^n is expressed by Noisy-OR model:

$$p_i^n = 1 - \prod_j (1 - p_{ij}^n) \tag{14}$$

And $p_{ij}^n = \sigma\left(H_{ij} + h_n\left(x_{ij}\right)\right)$ denotes the sigmoid function. [23] presents a weighted MIL feature selection scheme by defining the positive and negative bag probability as follows:

$$\begin{aligned} p(y=1 \mid X^+) &= \sum_{j=1}^{N} w_{1j}\, p(y=1 \mid x_{1j}) \\ p(y=1 \mid X^-) &= w \sum_{i=1}^{L} (1 - p(y=1 \mid x_{0i})) \end{aligned} \tag{15}$$

Where X^+ and X^- indicate positive and negative bag, N and L are the number of positive instances in positive bag and negative instances in negative bag, w_{1j} is a monotone decreasing function $w_{1j} = (1/\mathrm{c})\exp\left\{-\left\| l_t(x_{1j}) - l_{t-1}^* \right\|\right\}$, where c is a normalization constant, l_t and l_{t-1}^* denote object location respectively. Then greedily select K features from the feature pool Ω by maximizing the log likelihood function of the bags:

$$f_k = \mathop{\arg\max}_{f \in \{f_1, f_2, \dots f_M\}} \mathrm{L}(H_{k-1} + h) \tag{16}$$

Where $\mathrm{L}(H) = \sum_i (y_i \log p_i + (1-y_i)\log(1-p_i))$ is the log likelihood function of the bags. In the next step, [23] utilize the first-order Taylor expansion to approximately maximize the log likelihood function of the bags as follows:

$$\mathrm{L}(H_{k-1} + h) = \mathrm{L}(H_{k-1}) + < h, \nabla \mathrm{L}(H) > |_{H=H_{k-1}} \tag{17}$$

Where $< h, \nabla \mathrm{L}(H) > = (1/(N+L)) \sum_{J=1}^{N+L} h(x_{ij}) \nabla \mathrm{L}(H)(x_{ij})$ is an inner product and $h(x_{ij}) \nabla \mathrm{L}(H)(x_{ij}) = \frac{\partial \mathrm{L}(H + \delta 1_{x_{ij}})}{\partial \delta} |_{\delta=0} = y_i \frac{w_{1j}^* \sigma(H(x_{ij}))(1-\sigma(H(x_{ij})))}{\sum_{m=1}^{N} w_{1m}^* \sigma(H(x_{im}))} - (1-y_{i)}) \frac{\sigma(H(x_{ij}))(1-\sigma(H(x_{ij})))}{\sum_{m=1}^{L} \sigma(H(x_{im}))}$ is the functional derivative in an inner product space. [5] assign different weights to different features according to their discriminability. Specifically, similar to Equation 5, they model the naive Bayes classifier as:

$$\mathrm{H}(\mathbf{v}) = \sum_{i=1}^{n} w_{i1} * log\left(\frac{p(v_{i1} \mid y=1)}{p(v_{i1} \mid y=0)}\right) w_{i2} * log(\frac{p(v_{i2} \mid y=1)}{p(v_{i2} \mid y=0)} \tag{18}$$

w_{i1} and w_{i2} can be calculated as $w_i = \frac{TP_i + FN_i}{TP_i + FP_i + TN_i + FN_i}$, where TP_i means the number of positive examples that flagged as positive, FP_i means the number of negative examples that flagged as positive, TN_i means the number of negative examples that flagged as negative, FN_i means the number of positive examples that flagged as negative respectively. It can be clearly seen that this strategy treats the features discriminatively according to their cumulative performances. That is, the weights of the more discriminative features are increased and the less discriminative counterparts are decreased. This strategy is easy to implement and strikes a good trade-off between accuracy and efficiency.

Another feature selection technique is to measure the Hellinger distances between a feature's distributions of positive and negative samples, and then analyze the feature's ability of discriminating the object from the background [6]. Suppose $f_1(\mathrm{x})$ and $f_0(\mathrm{x})$ are the probability density function (PDF) for samples being positive P^1 and negative P^0, the Hellinger distance is defined as:

$$h^2(P^1, P^0) = \frac{1}{2}\int \left(\sqrt{f_1(\mathrm{x})} - \sqrt{f_0(\mathrm{x})}\right)^2 dx \tag{19}$$

It can be easily observed that $h \in [0,1]$. Generally speaking, the higher value of h indicates higher distance of two distributions, yielding better separation of the feature discriminability.

3.3. Particle Filter Framework

Recently, there are some compressive trackers [11-17] developed within particle filter framework. Then the main goal for these compressive trackers is to design a dynamic and observation model for the tracking system. In the prediction stage, [13] use 2-order auto-regressive model to propagate the samples in the previous frame:

$$x_k(i) = 2x_{k-1}(i) - x_{k-2}(i) + w_k \tag{20}$$

Where $x_k(i)$ represents the state vector of the i th particles at time step k and w_k is a three-dimension, zero-mean, white-noise sequence independent of past and current states with three different standard deviations for each dimension. Thus this dynamic model considers the previous two stages of each sample to fuse the velocity information of the target. In the update stage, observation model defined as Equation 21 is used to estimate the weights for each sample:

$$\mathrm{p}\left(z_k | x_k^*(\mathrm{i})\right) \propto \exp\left(\mathrm{H}\left(v^i\right)\right) = \prod_{j=1}^{m} \log\left(\frac{\mathrm{p}(v_j^{(i)} \mid \mathrm{Y}=1)}{\mathrm{p}(v_j^{(i)} \mid \mathrm{Y}=0)}\right) \tag{21}$$

Where $\mathrm{H}\left(v^i\right)$ represents the classifier response as computed through Equation 5 of the i th particles. To further avoid particle degeneracy, the prior particle set is re-sampled according to the weights of the particles obtained by the observation model, where for any particle index j, $\Pr\left\{x_k(\mathrm{j}) = x_k^*(\mathrm{i})\right\} = \frac{\mathrm{p}\left(z_k | x_k^*(\mathrm{i})\right)}{\sum_{j=1}^{N} \mathrm{p}\left(z_k | x_k^*(\mathrm{i})\right)}$. The re-sampled particle set retains particles whose classifier responses are relatively high and reallocate the distribution of the particles. Motivated by [13-15] employs an extended observation model which further takes marginal color feature into account:

$$\mathrm{p}\left(z_k | x_k^*(\mathrm{i})\right) = \exp\left(\mathrm{H}\left(v^i\right)\right) * \mathrm{E}(v^i) \tag{22}$$

Where E is a function to compute the color similarity between the templates and the candidate image patch.

4. Future Trends

Overall, compressive trackers have obtained many attractive performances in many practical applications. However, there are margins that still push interesting researchers to explore. Some existing compressive trackers track the scale by integrating their trackers into particle filter framework, most of the results tend to be not stable enough mainly because it is not easy to design an observation model which could consider both the location and scale. Another interesting topic is how to exploit the spatial-temporal relationships between the object of interest and its local context. These relationships are extremely useful especially when we intend to model the statistical correlation between the low-level features, such as image intensity and position, from the target and its surrounding regions. The last but not the least, the objective of tracking is to locate the object of interest from frame to frame and the objective of classification is to predict the possibility of the instance label as positive or negative. Therefore, these two objectives are not consistent during tracking, which may lead to inaccurate estimation by maximizing the classifier response, which further makes the tracking task more challenging.

Acknowledgment

We would like to thank the editor and reviewers for their careful read and thoughtful comments on our previous manuscript. This work is supported by Zhejiang Provincial Natural Science Foundation of China (No. LZ15F030002, No. LY16F020022), the National Science Foundation of China (51279152) and the Seed Foundation of Wuhan University of Technology (No. 145211005, No. 155211005).

References

[1] K. Zhang, L. Zhang and M. H. Yang, “Real-time compressive tracking”, 12th European Conference on Computer Vision (ECCV), **(2012)**, pp. 864-877.
[2] K. Zhang, L. Zhang and M. H. Yang, “Fast Compressive Tracking”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 10, **(2014)**, pp. 2002-2015.
[3] F. Teng and Q. Liu, “Multi-scale ship tracking via random projections”, Signal Image and Video Processing, vol. 8, no. 6, **(2014)**, pp. 1069-1076.
[4] X. Y. Sun and F. L. Chang, “Object tracking based on sparse representation of gradient feature”, Guangxue Jingmi Gongcheng/Optics and Precision Engineering, vol. 21, no. 12, **(2013)**, pp. 3191-3197.
[5] F. Teng and Q. Liu, “Robust multi-scale ship tracking via multiple compressed features fusion”, Signal Processing: Image Communication, vol. 31, **(2015)**, pp. 76-85.
[6] W. Lu, Z. Y. Xiang, H. B. Yu and J. L. Liu, “Object compressive tracking based on adaptive multi-feature appearance model”, Zhejiang Daxue Xuebao (Gongxue Ban)/Journal of Zhejiang University (Engineering Science), vol. 48, no. 12, **(2014)**, pp. 2132-2138.
[7] J. Zhu, Y. Ma, Q. Qin, C. Zheng and Y. Hu, “Adaptive weighted real-time compressive tracking”, IET Computer Vision, vol. 8, no. 6, **(2014)**, pp. 740-752.
[8] F. X. Zeng, Z. T. Huang and Y. F. Ji, “Embedding holistic appearance information in part-based adaptive appearance model for robust visual tracking”, Electronics Letters, vol. 49, no. 19, **(2013)**, pp. 1219-1221.
[9] H. T. Huang, G. Yuan, J. Zhang and Y. F. Cha, “Application of affine transform in compressed sensing tracking”, Journal of Xidian University, vol. 44, no. 2, **(2015)**, pp. 162-166.
[10] L. Zhang, J. Han, B. Li and L. Wang, “The scale adaptive feature compressed tracking”, Guofang Keji Daxue Xuebao/Journal of National University of Defense Technology, vol. 35, no. 5, **(2013)**, pp. 146-151.
[11] X. Chen and J. Wu, “Scalable compressive tracking based on motion”, IEEE International Conference on Robotics and Biomimetics (ROBIO), **(2013)**, pp. 504-509.
[12] J. Wu, D. Chen and R. Yi, “Real-time compressive tracking with motion estimation”, IEEE International Conference on Robotics and Biomimetics (ROBIO), **(2013)**, pp. 2374-2379.
[13] Y. X. Wu, J. Ni and J. P. Sun, “Real-time multi-scale tracking based on compressive sensing”, The Visual Computer, vol. 25, no. 5, **(2014)**, pp. 1-11.
[14] H. Li, P. Bai and H. Song, “Car tracking algorithm based on Kalman filter and compressive tracking”, 7th International Congress on Image and Signal Processing (CISP), **(2014)**, pp. 27-31.
[15] J. P. Sun and N. Jia, “Human target matching and tracking method in coal mine video”, Journal of China University of Mining and Technology, vol. 44, no. 3, **(2015)**, pp. 540-548.
[16] L. Wu, X. Wu, W. Zhang and Y. Zhang, “Particle filtering based on compressive sense for target tracking”, 7th International Congress on Image and Signal Processing (CISP), **(2014)**, pp. 492-497.
[17] X. Yao and Y. Zhou, “Real-time compressive tracking with a particle filter framework”, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 8836, **(2014)**, pp. 242-249.
[18] Q. Zhu, J. Yan and D. Deng, “Compressive tracking via oversaturated sub-region classifiers”, IET Computer Vision, vol. 7, no. 6, **(2013)**, pp. 448-455.
[19] W. Xiang and Y. Zhou, “Part-Based Tracking with Appearance Learning and Structural Constrains”, Neural Information Processing (NIP), **(2014)**, pp. 594-601.
[20] J. Yan, X. Chen, D. Deng and Q. Zhu, “Visual object tracking via online sparse instance learning”, Journal of Visual Communication and Image Representation, vol. 26, **(2015)**, pp. 231-246.
[21] L. Huilan, Z. Baokang and K. Fansheng, “Object tracking algorithm by combining the predicted target position with compressive tracking”, Journal of Image and Graphics, vol. 19, no. 6, **(2014)**, pp. 875-885.
[22] L. Zhang, Y. J. Wang and S. W. He, “Real-time compressive tracking method based on phase congruency”, Guangzi Xuebao/Acta Photonica Sinica, vol. 43, no. 8, **(2014)**, 0810003-1-0810003-8.
[23] W. Wang, Y. Xu, Y. Wang, B. Zhang and Z. Cao, “Effective weighted compressive tracking”, 7th International Conference on Image and Graphics (ICIG), **(2013)**, pp. 353-357.
[24] S. Wuzhen, N. Jifeng and Y. Yongfeng, “Feature selection and target model updating in compressive tracking”, Journal of Image and Graphics, vol. 19, no. 6, **(2014)**, pp. 932-939.
[25] Y. Zhang, J. Li and Z. Qie, “Improved compressive tracker via local context learning”, 33rd Chinese Control Conference (CCC), **(2014)**, pp. 4691-4695.
[26] H. Xu and F. Yu, “Improved compressive tracking in surveillance scenes”, 7th International Conference on Image and Graphics, (ICIG), **(2013)**, pp. 869-873.
[27] Q. P. Zhu, J. Yan, H. Zhang, C. E. Fan and D. X. Deng, “Real-time tracking using multiple features based on compressive sensing”, Guangxue Jingmi Gongcheng/Optics and Precision Engineering, vol. 21, no. 2, **(2013)**, pp. 437-444.
[28] D. Achlioptas, “Database-friendly random projections: Johnson-Lindenstrauss with binary coins”, Journal of computer and System Sciences, vol. 66, no. 4, **(2003)**, pp. 671-687.
[29] E. J. Candes and T. Tao, “Decoding by linear programming”, IEEE Transactions on Information Theory, vol. 51, no. 12, **(2005)**, pp. 4203-4215.

[30] L. Hanxi, S. Chunhua and S. Qinfeng, "Real-time visual tracking using compressive sensing", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), **(2011)**, pp. 1305-1312.
[31] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry and M. Yi, "Robust face recognition via sparse representation", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 2, **(2009)**, pp. 210-227.
[32] L. Liu and P. W. Fieguth, "Texture classification from random features", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 3, **(2012)**, pp. 574-586.
[33] P. Li, T. J. Hastie and K. W. Church, "Very sparse random projections", 12th ACM international conference on Knowledge discovery and data mining, **(2006)**, pp. 287-296.

Authors

Zhongyi Hu, is an associate professor at Wenzhou University. He received BS degree in automation from the Beijing Institute of Mechanical Industry in 1999, and MS and PhD degrees in control engineering and intelligent traffic engineering from Wuhan University of Technology in 2006 and 2015. His current research interests include intelligent systems, target tracking, image processing and information management. He is a member of IEEE and CCF.

Lei Xiao, is an associate professor at Wenzhou University. He received his BS degree in CS from the Xiangtan University in 1999, and his MS degree in CS from Nanjing Tech University in 2009. He is now a PhD candidate in School of Marine Science and Technology, Northwestern Polytechnical University. His current research interests include intelligent systems, target tracking and image processing. He is a member of CCF.

Fei Teng, received the BS degree in School of Automation, Taiyuan University of Science and Technology in 2011, the MS degree in School of Automation, Wuhan University of Technology in 2013. He is now a PhD candidate in School of Energy and Power Engineering, Wuhan University of Technology. His research interests include machine learning and computer vision, with a focus on visual tracking techniques.

