Paper:

Robust Visual Tracking via Hierarchical Representation

Zihao Ding*, Xuqian Ren*, Chunlei Song*†, and Jianhua Xu*

*School of Automation, Beijing Institute of Technology
No.5 Zhongguancun Nan Dajie, Haidian District, Beijing, 100081, China
E-mail: songchunlei@bit.edu.cn
†Corresponding author

Sparse representation has been applied to visual tracking by solving the target templates' representation coefficient accurately. The robust tracking algorithm needs to construct an appropriate object representation model. However, the existing trackers' representation model is designed without regarding the relationships between templates. In this paper, we propose a novel Hierarchical Visual Tracking(HVT) algorithm, that thoroughly investigated the dictionary's internal structure. The dictionary of templates with a grouping structure is designed in our HVT tracker to study the relationship between target templates. Furthermore, a novel sparse representation model is constructed based on the Hierarchical Lasso model. Quantitative evaluations on challenging benchmark data sets demonstrate that the proposed HVT algorithm performs favorably against several state-of-the-art methods. Experimental results verify the robustness of the proposed HVT algorithm.

Keywords: visual tracking, sparse representation, particle filter

1. Introduction

As a well-known problem in computer vision, visual tracking has critical applications in many fields, including self-driving cars, motion control, and surveillance, to name a few [1]. Although numerous algorithms have been proposed, there are still many visual tracking challenges, such as light change, blur, and the like.

The crux of visual tracking is to extract the appropriate image information and find an accurate representation model. Most existing image feature extraction methods, such as ADN [2] and RISTN [3], have been able to achieve effective results. On the other hand, the sparse representation can adequately represent the characteristics of the target signal [4]. And Lasso model can effectively solve the sparse representation problem, which has been proved in [5].

Many tracking methods based on the Lasso regression model have been proposed, which perform productive tracking results. Mei et al. proposed the l_1 tracker [6], which is the first application of the Lasso regression in the visual tracking algorithm. Bao et al. introduced the accelerated proximal gradient algorithm on the original l_1 tracker [6], and designed the L1APG algorithm [7]. After that, many visual tracking algorithms are generated by applying different sparse representation models [8-12]. In [8], the multi-task tracker considers the joint sparsity of candidate regions, overcoming computational complexity by applying the popular $l_{p,q}$ norm. For improving tracking accuracy, Zhang et al. exploited the underlying low-rank constraints and designed the consistent low-rank sparse tracker [9]. The visual tracking trackers investigated the image's structure also leading to good tracking performance [10,11]. Besides, Zhang et al. preserved the spatial structure among the local patches inside each target candidate region, and proposed the RSST method [12].

All of the above algorithms calculate the target representations by applying sparse linear combinations of dictionary templates. More importantly, the algorithm mentioned above solves the sparse representation from templates independently. This means the sparse representation result in these methods does not consider the underlying relationship between templates. To circumvent this situation, we investigate the templates' structure, and designed the hierarchical visual tracker based on the Hierarchical Lasso model.

Furthermore, many tracking algorithms, combined with reinforcement learning or deep learning, have also achieved excellent results in recent years [13-14].This is mainly because of reinforcement learning can make good use of system optimization [15] and data-driven [16]. However, this type of approach usually does not deal well with the distractions of the tracking environment. The algorithm structure and dictionary updating method proposed in this paper can effectively avoid this problem.

In this paper, we proposed a novel visual tracking algorithm by applying the Hierarchical Lasso model. The contributions of this work are three-fold. (1) Under the premise of considering the correlation between image representation templates, a novel visual tracking algorithm is designed. (2) To represent the candidate regions accurately, we construct a hierarchical dictionary that can reflect the tracking target's structural information. (3) We propose a dictionary updating method. This method can comprehensively evaluate each template's quality in the dictionary and realize the template's real-time update.

This paper's structure is organized as follows. The

second section introduces the relevant knowledge of this method. The third section introduces the algorithm proposed in this paper. In the fourth section, experimental research is given to verify the effectiveness of this work. Section five, at the end of the article, is the conclusion.

2. Related Works

The work related to the HVT algorithm is presented in this section. In section 2.1, we introduced the sparse representation applied in visual tracking. Then, the particle filter, which is the framework of the proposed tracking algorithm, is investigated in section 2.2. Furthermore, the Hierarchical Lasso model is given in section 2.3. Considering the dictionary's template dependencies, the Hierarchical Lasso model achieves better performance than the Lasso model in tracking algorithms.

2.1. Sparse Representation

The basis of sparse representation is to represent each candidate region as a linear combination of dictionary templates. The image of pixel size $m \times n$ can be represented as a matrix of dimension $m \times n$. Moreover, we stack the matrix columns to form a 1D vector to simplify the calculation. Assuming a series of template vectors are generated from the tracking target, any region in image can be structured as $y = Dx + \varepsilon$. $T \in \mathbb{R}^{mn \times n_T}$ is the dictionary generated by n_T target templates,and $x \in \mathbb{R}^{n_T \times 1}$ denotes the sparse representation coefficient vector. This representation method has been proved to be robust against complex tracking environment such as occlusions and blur. In this paper, the pixel information of the image is used to describe the characteristics of the image.

In the tracking problem, how to solve the discriminative representation coefficient becomes the critical factor affecting the algorithm's performance. Therefore, a suitable sparse representation model needs to be found.

2.2. Particle Filter

The particle filter is an approximate Bayesian filter algorithm based on Monte Carlo simulation [17]. As a probabilistic statistical algorithm, particle filter estimates the system's unknown parameters by calculating the sample mean of the particle filter. The particle filter's core idea is to approximate the probability density function of the system random variable by random sampling of a set of discrete points. Particle filter solves the minimum variance of the system state by using the sample mean instead of the essential operation.

The visual tracking problem is equivalent to finding the location of the target in the image. We define $y_{1:t} = \{y_1, y_2, \cdots, y_t\}$ represent the image set. And y_t represents the image at time t, $y_{t,i}$ is the ith candidate area.

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

$$p(x_t|y_{1:t}) = \frac{p(y_t|x_t)\, p(x_t|y_{1:t-1})}{p(y_t|y_{1:t-1})} \quad (2)$$

In particle filter framework, the tracking target can be approximated by sampled particles. Sampled particle set at frame t is defined as $S_t = \left\{x_{t,1}, x_{t,2}, \cdots, x_{t,n_p}\right\}$. The corresponding weight of each particle in the sampling particle set is $W_t = \left\{w_{t,1}, w_{t,2}, \cdots, w_{t,n_p}\right\}$. By selecting the appropriate sequential importance distribution, the particle weight can be updated as $w_t^i \propto w_{t-1}^i p(y_t|x_{t,i})$. In this paper, the sampled particle with the maximum weight is selected as the tracking target, which can be expressed as

$$x_t^* = \arg\max_{x_{t,i}} p\left(x_{t,i}\middle|y_{1:t}\right), \quad i = 1, \cdots, n_p \quad (3)$$

Besides, particle sampling in the next frame is updated according to each particle's weight at the current frame to improve the target tracking accuracy.

2.3. Hierarchical Lasso Model

Assuming a regression problem: given p predictors $x_1, x_2, \cdots, x_p$, the response y can be described as

$$y = \hat{k}_0 + \hat{k}_1 x_1 + \cdots + \hat{k}_p x_p + \varepsilon \quad (4)$$

where $\hat{k} = \left(\hat{k}_0, \hat{k}_1, \cdots, \hat{k}_p\right)$ denotes the predictors' coefficients, and ε is the error term. The Lasso model has been proved that it can be useful in solving the regression problem described in (4). In visual tracking problems, we need to consider the relationships between the atoms represented. The Group Lasso was proposed in [18]. The Group Lasso replaces the sparsity at the single-coefficient level as sparsity at a group level. However, the signals share groups because part of them belong to the same class. To improve the accuracy of presentation, Sprechmann et al. [19] proposed the Hierarchical Lasso model as follows,

$$\min_x \frac{1}{2}\|y - Dx\|_2^2 + \lambda_1\|x\|_1 + \lambda_2 \sum_{g=1}^{G} \left\|x_{[g]}\right\|_2 \quad (5)$$

where λ_1 and λ_2 are manually defined parameters. The solution of Eq. (5) fully considers the structural and intra-group element selectivity. The constraint of Eq.(5) in two-dimensional space is shown in **Fig. 1**.

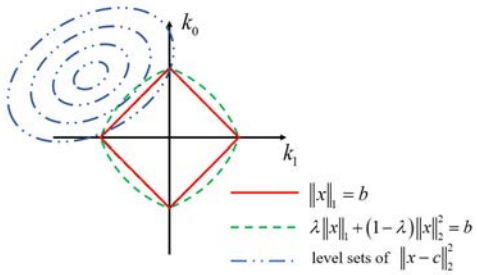

Fig. 1. Hierarchical Lasso model in two-dimensional space

To conclude, the Hierarchical Lasso model has the

grouping structure, while the sparsity of the solutions is guaranteed as well. The above properties make the Hierarchical Lasso model suitable for target tracking problems.

3. Visual Tracking Algorithm

In this section, we proposed a novel hierarchical visual tracking(HVT) algorithm. In section 3.1, we present the tracking target's sparse representation model and its corresponding solution. After that, the dictionary update algorithm is constructed in section 3.2. Moreover, other algorithm details, including similarity function structure and tracking algorithm framework, are introduced in section 3.3.

3.1. Hierarchical Tracking Target Representation

In this section, we designed a novel sparse represent model based on the Hierarchical Lasso model. Given the image matrix y_t at frame t, $y_{t,i}$ denotes the ith candidate region generated by particle sample. And $Y_t = \left[y_{t,1}, y_{t,2}, \cdots, y_{t,n_p}\right]$ represents the candidate matrix at frame t. The dictionary at frame t is defined as $D_t = \left[\begin{array}{cc} T_t & I \end{array}\right]$. In order to solve the representation coefficient of each candidate region under the dictionary, the following model is given

$$\begin{aligned}\hat{X}_t = \arg\min_{X_t} \tfrac{1}{2}\|Y_t - D_t X_t\|_F^2 + \lambda_1 \sum_{i=1}^{n_p} \left\|x_{t,i}\right\|_1 \\ +\lambda_2 \sum_{i=1}^{n_p}\sum_{g=g_1}^{g_n} \left\|x_{t,i[g]}\right\|_2\end{aligned} \quad . \quad (6)$$

where $\hat{X}_t = \left[\hat{x}_{t,1}, \hat{x}_{t,2}, \cdots, \hat{x}_{t,n_p}\right]$ represents the coefficient matrix, $\hat{x}_{t,i}$ denotes the representation coefficient of the ith candidate region. Furthermore, the representation coefficients are divided into several groups for hierarchical model, denoted as $g = \{g_1, g_2, \cdots, g_n\}$. In addition, λ_1 and λ_2 in (6) are preset system parameters. As the algorithms such as L1APG [7] and LRT [9], the parameters λ_1 and λ_2 are model structure and data dependent. The coefficient expression structure solved by Eq.(6) is shown in **Fig. 2**.

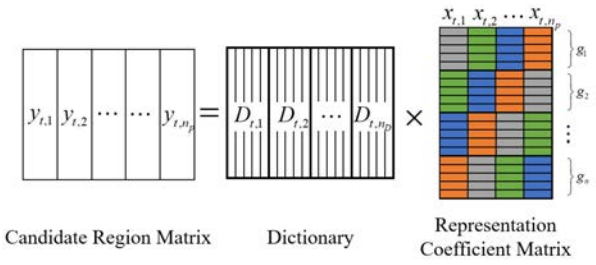

Fig. 2. The structure of Eq.(6)'s solution

In order to ensure the real-time performance of the tracking algorithm, it's necessary to find an appropriate method to solve Eq.(6).

For the purpose of simplifying the representation, we define

$$f(X_t) = \frac{1}{2}\|Y_t - D_t X_t\|_F^2 \quad . \quad (7)$$

$$\varphi(X_t) = \frac{\lambda_1}{\lambda}\sum_{i=1}^{n_p}\left\|x_{t,i}\right\|_1 + \frac{\lambda_2}{\lambda}\sum_{i=1}^{n_p}\sum_{g=g_1}^{g_n}\left\|x_{t,i[g]}\right\|_2 \quad . \quad (8)$$

By simplifying Eq.(7) and Eq.(8), Eq.(6) can be rewritten as the following form

$$\hat{X}_t = \arg\min_{X_t} f(X_t) + \lambda\varphi(X_t) \quad . \quad (9)$$

The optimization problem formed as Eq.(9) can be solved by Sparse Reconstruction by Separable Approximation algorithm(SpaRSA) [20].

By this way, Eq.(6) can be broken down into the following subproblems as

$$\begin{aligned}x^{(k+1)} \in \arg\min_{\hat{x}}\left(\hat{x} - x^{(k)}\right)\nabla f\left(x^{(k)}\right) \\ +\tfrac{\mu_k}{2}\left\|\hat{x} - x^{(k)}\right\|_2^2 + \gamma\varphi(\hat{x})\end{aligned} \quad . \quad (10)$$

where the parameter $\mu^{(k)}$ is updated as $\mu^{(k)} = \mu^{(0)}\cdot(\eta)^k$. As the discussion in [21], $\mu^{(0)}$ and η need to be chosen properly for the algorithm to converge. Eq.(10) is equivalent to

$$\min_{\hat{X}} \frac{1}{2}\left\|\hat{X} - Q\right\|_F^2 + \frac{\gamma}{\mu^{(k)}}\varphi\left(\hat{X}\right) \quad . \quad (11)$$

where $Q^{(k)} = X_t^{(k)} - \frac{1}{\mu^{(k)}}\nabla f\left(X_t^{(k)}\right)$.

$$\hat{X}_{[g]}^{(k+1)} = \min_{\hat{X}_t}\frac{1}{2}\left\|\hat{X}_t - Q_{[g]}^{(k)}\right\|_F^2 + \frac{\gamma}{\mu^{(k)}}\varphi\left(\hat{X}_t\right) \quad . \quad (12)$$

where $Q_{[g]}$ represents the corresponding item for group g. By substituting Eq.(8) into Eq.(12), Eq.(12) can be rewritten as

$$\begin{aligned}\hat{X}_{t,[g]}^{(k+1)} = \min_{\hat{X}_t}\tfrac{1}{2}\left\|\hat{X}_{t,[g]} - Q_{[g]}^{(k)}\right\|_F^2 \\ +\tfrac{\lambda_2}{\mu^{(k)}}\sum_{i=1}^{n_p}\sum_{g=1}^{G}\left\|x_{t,i[g]}\right\|_2 + \tfrac{\lambda_1}{\mu^{(k)}}\sum_{i=1}^{n_p}\left\|x_{t,i}\right\|_1\end{aligned} \quad . \quad (13)$$

The above optimization problem can be solve as [9]

$$x_{[g]}^{(t+1)} = \begin{cases}\frac{\max\{0, \|\vartheta\|_2 - \tilde{\lambda}_2\}}{\|\vartheta\|_2}\cdot\vartheta, & \|\vartheta\|_2 > 0 \\ 0, & \|\vartheta\|_2 > 0\end{cases} \quad . \quad (14)$$

where $\tilde{\lambda}_1 = \frac{\lambda_1}{\mu^{(k)}}$ and $\tilde{\lambda}_2 = \frac{\lambda_2}{\mu^{(k)}}$ are dynamic parameters. τ_i is the simplified operators in optimization problems, defined as $\tau_i = \hat{x}_{[g_i]}$. And $\vartheta_i = \mathrm{sgn}(\tau_i)\cdot\max\left(0, |\tau_i| - \tilde{\lambda}_1\right)$ is the scalar thresholding of τ_i. The complete tracking model optimization algorithm is summarized in Algorithm 1.

In practice, the solution of Eq.(6) can adequately reflect the associations between dictionary templates. The **Fig. 3** shows the coefficient matrix's numerical visualization. Note that smaller values in the matrix are darker in color. Compared with the $l_{p,q}$ norm sparse representation model's solution [8], Eq.(6)'s solution is more discriminative, as shown in **Fig. 3**. The **Fig. 3** also proved that

Algorithm 1 Tracking model optimization algorithm

Input: candidate region matrix Y_t, dictionary D_t, group set g, constants $\mu^{(0)} > 0$, $\eta = 2$, $[\mu_{\min}, \mu_{\max}]$.

Output: The solution of Eq.(6), $\hat{x}$;

$k = 0$, $x^{(k)} = 0.001 \times 1^T$, where 1^T denotes the unit column vector.
While not convergence
  $q^{(k)} = x_t^{(k)} + \frac{1}{\mu^{(k)}} \nabla f\left(x_t^{(k)}\right)$;
  $\mu^{(k)} = [\mu_{\min}, \mu_{\max}]$;
  While not convergence
    For $i = 1 : n$;
      Calculate Eq.(14);
    End;
  $\mu^{(k+1)} = \eta \cdot \mu^{(k)}$;
  End;
  $k = k + 1$;
End;
return $\hat{x}$;

Eq.(6) could more adequately excavate the solution's internal structure information.

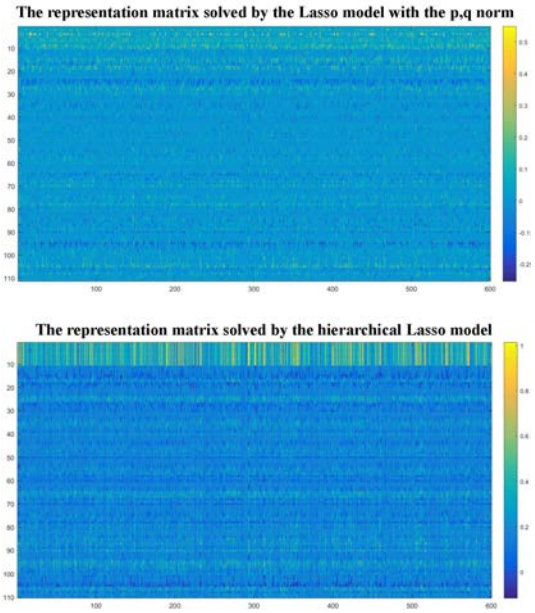

Fig. 3. The visualization of matrices solved by different representation models.

3.2. Dictionary Update

The dictionary selection and updating has a direct influence on the tracking effect. In this paper, the dictionary D_t contains target templates T_t and trivial templates I, where $T_t = [T_{t,1}, T_{t,2}, \cdots, T_{t,n_T}]$.The trivial templates can be treated as a identity matrix determined according to the size of the candidate regions. The dictionary D_t is initialized by random sampling around the initial target like many existing algorithms [6-8]. Furthermore, all templates in the dictionary are divided into groups of n_T, where n_T is the target templates' number. Different from existing tracking methods, the dictionary update algorithm proposed in this paper is based on a comprehensive evaluation of each template's quality.

For each frame t, if the template's corresponding representation coefficient is less than the preset threshold, it begins to detect the similarity between the tracking target and each template. The similarity between the target template and the tracking object is calculated by the following equation

$$\iota_{t,i} = \frac{\vec{\mathrm{T}}_{t,i} \cdot \vec{\mathrm{y}}_t^*}{\left\| \vec{\mathrm{T}}_{t,i} \cdot \vec{\mathrm{y}}_t^* \right\|} \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (15)$$

where $\vec{\mathrm{T}}_{t,i}$ and $\vec{\mathrm{y}}_t^*$ is the vector form correspond to $T_{t,i}$ and y_t^*. The template corresponding to the highest $\sum_{iter=t-3}^{t} \iota_{iter,i}$, as a poor quality template, is replaced directly by the existing trace target y_t^i. As illustrated in experiments, the dictionary update method mentioned above can achieve a robust tracking result.

3.3. Hierarchical Visual Tracking Algorithm

According to previous sections, each candidate region's representation coefficient can be solved obtained to the dictionary. Furthermore, on that basis, it is critical to propose an appropriate similarity function to calculate the reconstruction accuracy of each candidate region.

In order to effectively calculate the reconstruction error of each candidate target, we choose the gaussian kernel function as follows

$$S\left(y_{t,i}, x_{t,i}\right) = \frac{1}{\Gamma} \exp\left\{-\alpha \left\| y_{t,i} - T_t x_{t,i} \right\|_2^2\right\} \quad . \quad . \quad . \quad . \quad (16)$$

where α represents the gaussian kernel parameter, and Γ is the similarity coefficient which controls the the value of Eq.(16). As the particle filter framework discussed in section 2.2, the particle's reconstruction error is proportional to the particle's weight. Therefore, the candidate region with the minimum reconstruction error is selected as the new tracking target, as

$$x_t^* = \arg\max_{s_{t,i}} S\left(y_{t,i}, x_{t,i}\right) \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (17)$$

At the same time, resampling is carried out according to each candidate region's weight in this sampling. And the resampled particles are denoted as candidate regions for the next frame. The whole process of Hierarchical Visual Tracking Algorithm is summarized in algorithm 2.

4. Experiments

In the experiments, we present experimental results on evaluation of the proposed tracking algorithm against several existing sparse representation tracking methods. The selected datasets and comparison methods are introduced in section 4.1. And the analysis of the experimental results is given in Section 4.2.

Algorithm 2 Hierarchical Visual Tracking algorithm

Input: Current image y_t at frame t, Current candidate region set Y_t, Dictionary D_t

Output: Tracking target x_t^* at frame t

1: Build the target representation model based on Eq.(6);
2: Solve the representation coefficients of each candidate region by Algorithm 1;
3: Calculate the each candidate region's reconstruction error by Eq.(16);
4: Select the tracking target by Eq.(17), and resample according to each candidate target's reconstruction error;
5: Update the dictionary for the next frame;
6: **return** x_t^*, D_{t+1}

4.1. Datasets and Comparison Methods

We select a set of 8 challenging videos with object's ground truth location to verify the effect of the tracking method mentioned in this paper. The video set including car1, car2, coupon, faceocc2, fish, toy, twinnings, and vase. These datasets are available online[1] and contain complex scenes with challenging visual tracking factors, including illumination variation, scale variation, occlusion, deformation, motion blur, and fast motion.

We compared the proposed HVT algorithm with L1APG [7], MTT [8], LRT [9] and SST [10]. The center location error and the overlapping rate are chosen as the evaluation parameters for measuring each algorithm's quality. The center location error is the euclidean distance between the tracking result's center and the ground truth. And the overlapping rate is calculated as

$$score = \frac{S_T \cap S_{GT}}{S_T \cup S_{GT}} \quad . . . . . (18)$$

where S_T represents the target region marked by tracking algorithms in the image, and S_{GT} denotes the region labeled by the ground truth. It is a measure of the proportion of overlapping areas between two different regions.

We calculate the average center location error and average overlap score across all frames of each image sequence as existing methods to rank the tracking performance. **Table 1** records the Average Center Location Error on 8 Different Datasets. **Table 2** records the Average Overlap Score on 8 Different Datasets.

4.2. Discussion

In this section, the performance of the proposed algorithm in each data set is analyzed. Unlike [22] and [23], the gray value feature of the image is extracted in the proposed tracker. For all experiments, we set $\lambda_1 = 0.001$, $\lambda_2 = 0.0005$, the number of particles $n_p = 500$. Some representative results in the experiment are shown in **Fig. 4** and **Fig. 5**. The topics discussed include Illumination Variation, Scale Variation and Occlusion, as follows.

Illumination Variation. In these data sets, the illumination of the target region changed significantly. Car1, car2, and faceocc2 contain the illumination Variation. The experimental results show that the algorithm in this paper performs stably in datasets with illumination Variation, especially in vehicle tracking. Even in long-distance tracking(such as car1 and car2), our tracking method has no noticeable drift, which verifies the effectiveness of the proposed method in long-time tracking.

Scale Variation. When the ratio of the bounding boxes of the first frame and the current frame is out of the range, the dataset is defined as contain Scale Variation. The formula of scale variation is $S_{T,t}/S_{T,t-1} > 1$. Car1, car2, toy, twinnings and vase contain the scale variation. The experimental results proved that the proposed algorithm has an excellent tracking effect under scale variation.

Occlusion. Occlusion means the tracking target in datasets is partially or fully occluded. Faceocc2 contains occlusion interference. Depending on the dictionary's effective tracking method, the algorithm presented in this paper is still effective in the case of interference.

Fast Motion and In-Plane Rotation. Fast motion presents the dataset containing the ground truth's motion larger than the default threshold(in pixels), and In-Plane Rotation means the target rotates in the image plane. The dataset toy and twinnings have Fast Motion and In-Plane

Table 1. Average Center Location Error of 5 Different Trackers on 8 Different Datasets

Video	L1APG	MTT(l_{12})	SST	LRT	Ours
Car1	3.2	2.9	2.7	5.1	**2.5**
Car2	2.8	2.6	2.9	3.6	**1.9**
Coupon	61.4	8.2	8.1	9.7	**5.5**
FaceOcc2	15.1	8.1	**6.1**	9.0	8.5
Fish	27.8	10.8	**8.2**	9.5	8.8
Toy	12.7	9.1	8.9	7.9	**7.5**
Twinning	19.3	6.6	6.3	7.2	**5.8**
Vase	14.4	8.3	8.1	**7.5**	7.6

Table 2. Average Overlap Score of 5 Different Trackers on 8 Different Datasets

Video	L1APG	MTT(l_{12})	SST	LRT	Ours
Car1	0.53	0.86	0.87	0.72	**0.89**
Car2	0.69	0.87	0.83	0.69	**0.89**
Coupon	0.33	0.79	0.81	0.75	**0.83**
FaceOcc2	0.67	0.73	**0.74**	0.71	0.73
Fish	0.42	0.65	0.68	**0.79**	0.62
Toy	0.55	0.67	0.72	0.66	**0.78**
Twinning	0.57	0.70	0.68	**0.79**	0.77
Vase	0.54	0.51	0.57	0.60	**0.74**

1. http://cvlab.hanyang.ac.kr/tracker_benchmark/datasets.html

Rotation. The experimental results show that the proposed algorithm is robust in these datasets.

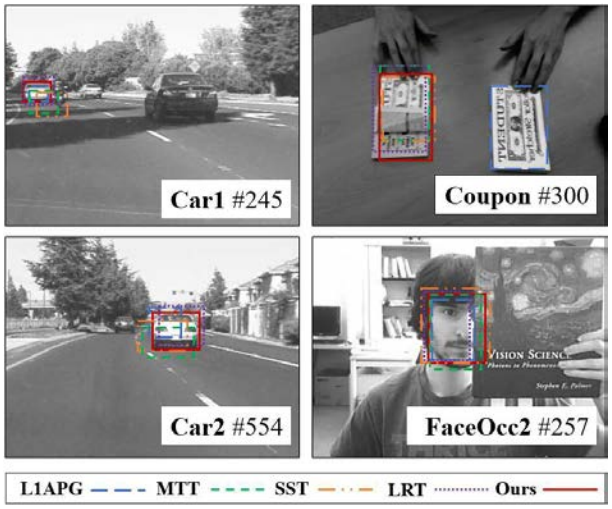

Fig. 4. Tracking results on Car1, Car2, Coupon, FaceOcc2

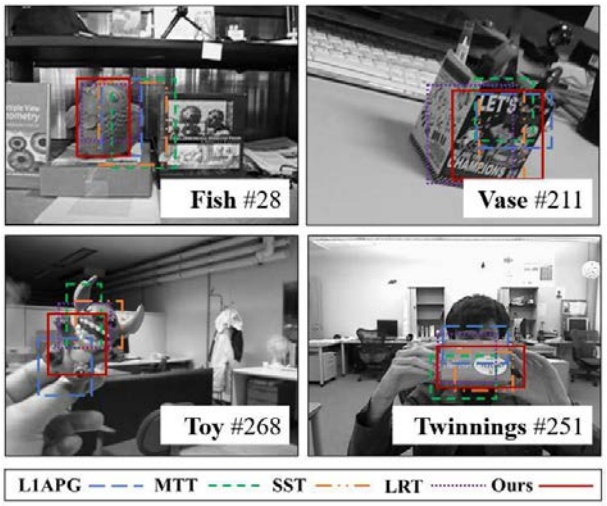

Fig. 5. Tracking results on Fish, Vase, Toy, Twinnings

5. Conclusion

In this paper, we designed a novel tracking target sparse representation model and proposed the Hierarchical Visual Tracking algorithm. The Hierarchical Visual Tracking algorithm makes full use of the information between the templates in the dictionary. To mine the internal links in the dictionary templates, we group the templates in the dictionary. Candidate regions with different features can be distinguished more effectively by the representation coefficients obtained from the hierarchical dictionary. Experimental results show that the proposed HVT algorithm has an excellent tracking result in datasets. The comparison with other algorithms also verifies the robustness of the HVT algorithm.

References:

[1] A. W. M. Smeulders, D. M. Chu et al., "Visual Tracking: An Experimental Survey,".Trans. on Pattern Analysis and Machine Intelligence, Vol.36, No.7, pp. 1442-1468, 2014.

[2] M. D. Zeiler, G. W. Taylor and R. Fergus, "Adaptive Deconvolutional Networks for Mid and High Level Feature Learning," Proc. of the International Conference on Computer Vision, pp. 2018-2025, 2011.

[3] X. Zhu, Z. Li et al., "Residual Invertible Spatio-Temporal Network for Video Super-Resolution," Proc. of the 33th AAAI Conference on Artificial Intelligence, pp. 5981-5988, 2019.

[4] Y. Eldar and G. Kutyniok, "Compressed Sensing: Theory and Applications," The Cambridge University Press, 2012.

[5] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," J. Roy. Statist. Soc., Series B, Vol.73, No.3, pp. 273-282, 2011.

[6] X. Mei and H. Ling, "Robust Visual Tracking using l_1 Minimization," Proc. of the 12th International Conference on Computer Vision. pp. 1436-1443, 2009.

[7] C. Bao, Y. Wu, H. Ling and H. Ji, "Real time robust l_1 tracker using accelerated proximal gradient approach," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1830-1837, 2012.

[8] T. Zhang, B. Ghanem, S. Liu and N. Ahuja, "Robust visual tracking via multi-task sparse learning," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2042-2049, 2012.

[9] T. Zhang, S. Liu, N. Ahuja, M. Yang and B. Ghanem, "Robust Visual Tracking Via Consistent Low-Rank Sparse Learning," International J. of Computer Vision, Vol.111, No.2, pp. 171-190, 2015.

[10] T. Zhang, S. Liu et al., "Structural sparse tracking," Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 150-158, 2015.

[11] Y. Sui and L. Zhang, "Robust Tracking via Locally Structured Representation," International J. of Computer Vision, Vol.119, pp. 110-144, 2016.

[12] T. Zhang, C. Xu and M. Yang, "Robust Structural Sparse Tracking," Trans. on Pattern Analysis and Machine Intelligence, Vol.41, No.2, pp. 473-486, 2019.

[13] B. Zhong, B. Bai, J. Li, Y. Zhang and Y. Fu, "Hierarchical Tracking by Reinforcement Learning-Based Searching and Coarse-to-Fine Verifying," Trans. on Image Processing, Vol.28, No.5, pp. 2331-2341, 2019.

[14] C. Ma, J. Huang, X. Yang and M. Yang, "Robust Visual Tracking via Hierarchical Convolutional Features," Trans. on Pattern Analysis and Machine Intelligence, Vol.41, No.11, pp. 2709-2723, 2019.

[15] Y. Yang, H. Modares, D. C. Wunsch and Y. Yin, "LeaderFollower Output Synchronization of Linear Heterogeneous Systems With Active Leader Using Reinforcement Learning," Trans. on Neural Networks and Learning Systems, Vol. 29, No. 6, pp. 2139-2153, 2018.

[16] Y. Yang, Z. Guo et al., "Data-Driven Robust Control of Discrete-Time Uncertain Linear Systems via Off-Policy Reinforcement Learning," Trans. on Neural Networks and Learning Systems, Vol.30, No.12, pp. 3735-3747, 2019.

[17] A. Doucet, N. de Freitas, and N. Gordon, "Sequential Monte Carlo Methods in Practice," The Springer-Verlag, 2001.

[18] M.Yuan and Y. Lin, "Model selection and estimation in regression with grouped variables," J. Roy. Statist. Soc., Series B, vol.68, pp. 4967, 2006.

[19] P. Sprechmann, I. Ramirez, G. Sapiro and Y. C. Eldar, "C-HiLasso: A Collaborative Hierarchical Sparse Modeling Framework," Trans. on Signal Processing, Vol.59, No.9, pp. 4183-4198, 2011.

[20] J. Wright and R. Nowak, "Sparse reconstruction by separable approximation," Trans. on Signal Processing, Vol.57, No.7, pp. 2479-2493, 2009.

[21] S. J. Wright, R. D. Nowak and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," Trans. on Signal Processing, Vol. 57, No. 7, pp. 2479-2493, 2009.

[22] B. Zhuang, H. Lu, Z. Xiao and D. Wang, "Visual Tracking via Discriminative Sparse Similarity Map," Trans. on Image Processing, Vol.23, No.4, pp. 1872-1881, 2014.

[23] X. Guo, Y. Liu, Q. Zhong, and M. Chai, "Research on Moving Target Tracking Algorithm Based on Lidar and Visual Fusion," J. Adv. Comput. Intell. Intell. Inform., Vol.22, No.5, pp. 593-601, 2018.