Hypothesis Testing Based Tracking With Spatio-Temporal Joint Interaction Modeling

Hao Sheng, *Member, IEEE*, Yang Zhang, Yubin Wu, Shuai Wang, Weifeng Lyu, Wei Ke, and Zhang Xiong, *Member, IEEE*

***Abstract*— Data association is one of the key research in tracking-by-detection framework. Due to frequent interactions among targets, there are various relationships among trajectories in crowded scenes which leads to problems in data association, such as association ambiguity, association omission, etc. To handle these problems, we propose hypothesis-testing based tracking (HTBT) framework to build potential associations between target by constructing and testing hypotheses. In addition, a spatio-temporal interaction graph (STIG) model is introduced to describe the basic interaction patterns of trajectories and test the potential hypotheses. Based on network flow optimization, we formulate offline tracking as a MAP problem. Experimental results show that our tracking framework improves the robustness of tracklet association when detection failure occurs during tracking. On the public MOT16, MOT17 and MOT20 benchmark, our method achieves competitive results compared with other state-of-the-art methods.**

***Index Terms*— Multi-object tracking, tracking-by-detection, network flow, hypothesis testing, interaction modeling.**

I. Introduction

MULTI-OBJECT tracking (MOT) is an important research in the field of computer vision, aiming at recovering the position of targets in each frame and their complete trajectories as well. Although detection technology has been greatly improved through deep learning methods, frequent target interaction still leads to a large number of trajectory interruptions and false association in crowded scenes. Therefore, tracking targets with complex interactions is still a challenging task.

Tracking-by-detection framework consists of two parts including detection and association. First, targets are located by detectors as accurately as possible from the video. Then, these detections are associated into trajectories of each target. If all targets are detected correctly, it means there is no false detection and generating trajectories is a naive data association problem.

However, even modern detectors ofter fail in crowded scenes and results false detection, missing detection and detection offset. These problems make data association a tough task. As shown in Fig.1, as targets move, there are different types of interactions between the trajectories, such as aggregation, abruption and stabilization. From this view, we introduce a hypothesis testing method with interaction modeling framework to deal with problems in tracklet association.

To improve the accuracy and robustness in tracklet association, we propose hypothesis-testing based tracking (HTBT) method in this paper. First, a non-independent hypothesis is defined to formulate each association condition. Then, the data association is divided into two steps including hypothesis construction and hypothesis testing. Hypothesis construction assumes relationships among each dependent trajectory, while hypothesis testing estimates and tests the assumptions according to the interaction information in the context. In addition, spatio-temporal features of the target interaction are modeled as the basis for hypothesis testing. Finally, HTBT is integrated into the network flow framework for tracking as a MAP problem, including robust tracklet association and enhanced tracklet refinement. Robust tracklet association is assigned to handle association failure when occlusion occurs. Enhanced tracklet refinement re-estimate and update tracklets when the detection is false or missed.

In summary, this paper makes the following contributions:

- Hypothesis-testing based tracking (HTBT) method is proposed to construct and test association assumption, and thus improve the association performance and robustness for tracking.
- Spatio-temporal interaction graph (STIG) is introduced as the basis of hypothesis testing, by modeling the spatio-temporal interaction relationships among tracklets.
- HTBT is integrated into the network flow tracking to improve tracklet association and refinement.

Manuscript received November 13, 2019; revised February 25, 2020 and April 13, 2020; accepted April 15, 2020. Date of publication April 20, 2020; date of current version September 3, 2020. This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFC0806500, in part by the National Natural Science Foundation of China under Grant 61861166002, Grant 61872025, and Grant 61635002, in part by the Science and Technology Development Fund, Macau SAR under Grant 0001/2018/AFJ, in part by the Macao Science and Technology Development Fund under Grant 138/2016/A3, in part by the Fundamental Research Funds for the Central Universities and the Open Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE2019ZX-04, and in part by the support from HAWK-EYE Group. This article was recommended by Associate Editor L. Zheng. *(Corresponding author: Yang Zhang.)*

Hao Sheng, Yang Zhang, and Weifeng Lyu are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China (e-mail: shenghao@buaa.edu.cn; yang.zhang@buaa.edu.cn).

Yubin Wu, Shuai Wang, and Zhang Xiong are with the State Key Laboratory of Software Development Environment, School of Computer Science and Engineering, Beihang University, Beijing 100191, China, and also with the Beihang Hangzhou Institute for Innovation at Yuhang, Beihang University, Hangzhou 311121, China (e-mail: yubin.wu@buaa.edu.cn; shuaiwang@buaa.edu.cn; xiongz@buaa.edu.cn).

Wei Ke is with the School of Applied Sciences, Macao Polytechnic Institute, Macao SAR 999078, China (e-mail: wke@ipm.edu.mo).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2020.2988649

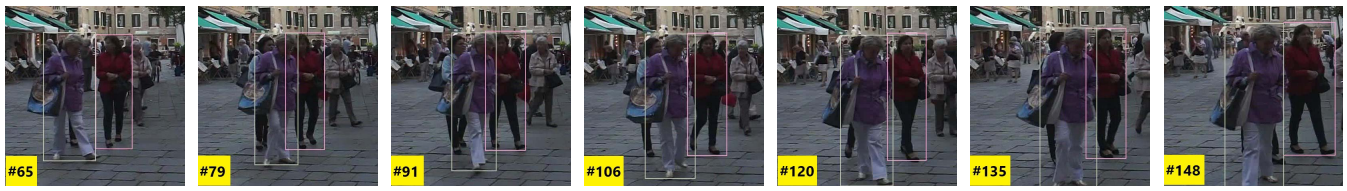


Fig. 1. Target interaction during tracklet association. From frame 65 to frame 148, the interactions between pairwise targets have various features and tendencies, which change with time frame.

- We formulate tracking with hypothesis testing as a MAP problem and solve it by network flow framework.

The rest of the paper is organized as follows. Related work is discussed in Sec.II. In Sec.III, different relationships among tracklets are introduced and the framework of hypothesis-testing based tracking (HTBT) is proposed. In Sec.IV, spatio-temporal interaction graph (STIG) is constructed to formulate the interaction process of tracklets. Network flow tracking with HTBT is solved in Sec.V. Experiments and conclusion are in Sec.VI and VII.

II. Related Work

A. Multi-Object Tracking

MOT has been a popular topic in computer vision for years and most recent tracking methods can be generally categorized into two groups: online and offline. Online methods are widely applied in realtime applications. The state of targets is estimated with only current and past observations by kalman filters [38], particle filters [16] and other deep learning methods. It is hard for online methods to correct trajectories when an early error is made.

In contrast, offline methods take all frames into consideration where the entire or a batch of the sequences is processed. Tracking-by-detection is one of the most popular frameworks in recent research. Detections are gained by detectors in each frame and then linked into trajectories. Therefore, the multi-object tracking can be converted into a data association problem and various methods are proposed.

Conditional random field (CRF) based methods [27], [28] formulate tracking as an energy minimization problem and solve data association by energy minimization. Data association and trajectory estimation are joint in these methods. However, lacking robustness in tracking occluded targets is a common shortcoming of them.

Multiple hypothesis tracking (MHT) [19] is a classic tracking method where association decisions are delayed. To solve the detection failure problem, [9] models the association between detections and scenes and [33] proposes a heterogeneous association graph that fuses high-level detections and low-level image evidence for target association. However, MHT often suffers from the high computation complexity and consumes too much memory.

Zhang *et al.* [42] and Butt and Collins [7] propose a min-cost network flow based data association method for tracking. They solve the optimal problem though linear programming and Lagrangian methods. Since network flow based association methods have the benefit of finding the globally optimal solution efficiently, many following methods [8], [25] [10] are proposed to improve the robustness of association.

In addition, some spatio-temporal model based tracking approaches have been proposed in recent research. Ren *et al.* [29] proposes a spatio-temporal target-to-sensor data association method. Zhang *et al.* [43] proposes an spatio-temporal context learning based method with self-correction under multiple views to track players in soccer videos. To deal with the problems of missed detections, a combined model utilizing the information of spatio-temporal correlation is proposed in [37]. The above approaches still have limitations in terms of interactive target association in crowded scene. In this paper, we focus on this issue and aims to cope with association ambiguities and association failure problems in crowded scene.

B. Visual Tracking With Hypothesis Testing

Hypothesis testing has also been explored to solve tracking problem. In this framework, tracking is formulated as a Maximum A Posteriori (MAP) segmentation problem where each pixel is assigned a binary label indicating whether it belongs to the target or not. Enescu *et al.* [15] proposes an approach to track non-rigid targets based on MAP-MRF framework. An MRF model is used for data association, region smoothness and elliptic shape constraints. Zhang *et al.* [44] presents a method for regional tracking. They use hypothesis testing and statistical methods to judge trajectories and avoid probability distribution problem of the estimated density function.

Hypothesis testing is often implemented iteratively in tracking. Amit *et al.* [1], [2] implements an iterative hypothesis testing strategy to exploit the appearance features, even targets are only intermittent available. They connect detections across frames based on their position and appearance. However, since their iteration is under the assumption that the target appearance is defined by the key-node appearance estimate, inaccurate appearance estimation and error hypothesis occur in crowded scenes.

Probability hypothesis testing is also used for tracking targets with specific moving. Demirbas [13] proposes a maneuvering target tracking method with hypothesis testing. Target motion is described by nonlinear models in a spherical coordinate system. Hypothesis testing is used to estimate the states of the nonlinear model, which prevents false state estimation due to the model linearization errors. Li *et al.* [22] proposes an

initiation algorithm for dim and small moving target based on spatio-temporal hypothesis testing. Different spatio-temporal features are utilized in hypothesis testing to recognize target and initialize trajectories.

However, the above mentioned hypothesis testing based methods can not handle the data association problem well in crowded scenes because their assumption is based on the low-level image information. High-level potential information is ignored such as target interaction. In this paper, in order to handle tracking failure and ambiguity in crowded scenes, we focus on implementing hypothesis testing for tracking with spatio-temporal interaction modeling.

III. Hypothesis-Testing Based Tracking

Data association can be converted into a MAP problem where individual trajectories are described by a Markov chain. Then, hypothesis-testing based tracking (HTBT) is proposed to describe the state transition likelihood with dynamic trajectory relationship according to successive pairwise detections with appearance and background features. In this section, non-independent hypothesis is introduced to represent the tuple relationship and tracking is formulated as a MAP problem to describe various relationships among tracklets.

A. Non-Independent Hypothesis for Tracking

For a given video sequence, let D denotes a set of detections, where each element $d = (x, y, w, h, f, a) \in V$ consists of location (x, y), scale (w, h), frame t and appearance features a. The appearance features for each detection are extracted after the FC layer in [34] and downsample it as a 256-dimensional vector. Thus a trajectory can be expressed as a set of selected detections as $t = \{d_1, d_2, \ldots, d_k\}$ and each detection can only belong to one trajectory.

The motion of all targets can be regarded as a set of trajectories $T = \{t_1, t_2, \ldots, t_n\}$. Therefore, tracking is formulated as finding an optimal T^* that has the maximum posterior probability $P(T|t)$. Assuming that each trajectory is independent of each other and follows Markov chain, the MAP inference can be expressed as follows:

$$\begin{aligned} T^* &= \arg\max_T P(T|t) \\ &= \arg\max_T \prod_i P(d_i|T) P(T) \\ &= \arg\max_T \prod_i P(d_i|T) \prod_j P(t_j) \end{aligned} \tag{1}$$

where $P(d_i|T)$ is the probability of d_i to describe a real target and $P(t_j)$ represents the probability of t_j to be a correct trajectory. For a trajectory $t = \{d_1, d_2, \ldots, d_k\}$, its probability can be expressed as:

$$P(t) = P(d_1) P(d_k) \prod_{n=2}^{k} P(d_n|d_{n-1}) \tag{2}$$

where $P(d_n|d_{n-1})$ is the probability of d_n and d_{n-1} are two consecutive detections in the trajectory.

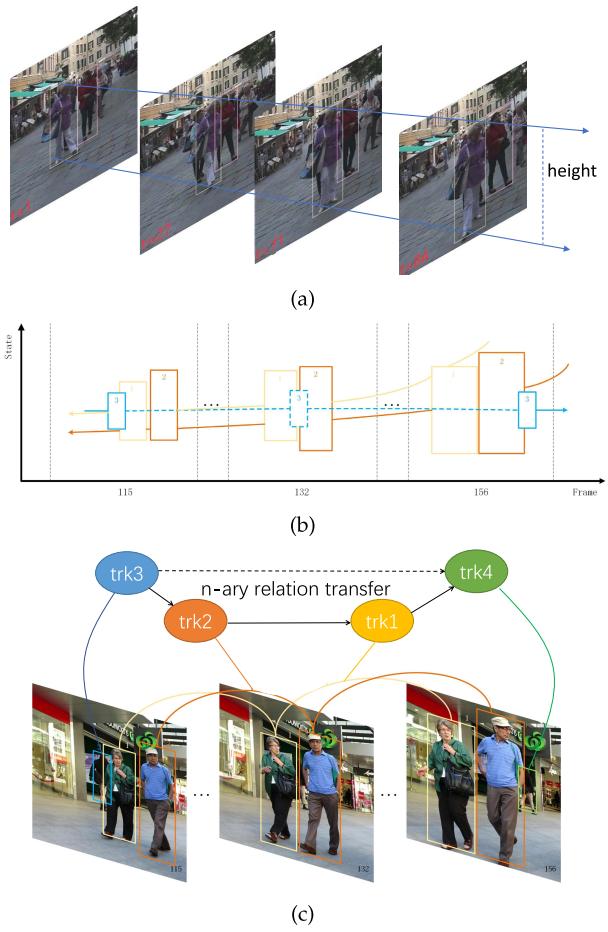


Fig. 2. Illustration of the classification of association relationships of tracklets. (a) shows an example of unary relation, (b) is an example of binary relation and (c) indicates a N-ary relation.

B. Hypothesis Construction

1) Unary Hypothesis: Most people have little difference in height, so for a fixed camera, the heights of targets at a certain position in the scene should be similar. We formulate the distribution of the height of targets as $M(x, y)$. Therefore, we can predict the possible height of the target at (x, y). Given a detection $d_i = (x_i, y_i, w_i, h_i)$, its unary hypothesis is denoted as H_1. d_i is used for tracking only if H_1 is accepted which is defined as follows:

$$\mu_{lower} M(x_i, y_i) \leqslant h_i \leqslant \mu_{upper} M(x_i, y_i) \tag{3}$$

where μ_{lower} and μ_{upper} are the lower and upper limit parameters. They make the height distribution prediction have an offset range, thus being able to describe pedestrians of different heights while enduring detection offset to a certain extent. We set $\mu_{lower} = 0.5$ and $\mu_{upper} = 1.5$ in this paper that can cover the possible height of most people and bear a certain degree of fitting error.

2) Binary Hypothesis: We define three different forms of binary hypothesis H_2 including inclusion, exclusion and coexistence.

a) Inclusion: When two tracklets t_i and t_j are likely to have an inclusion relationship, they do not intersect in neither

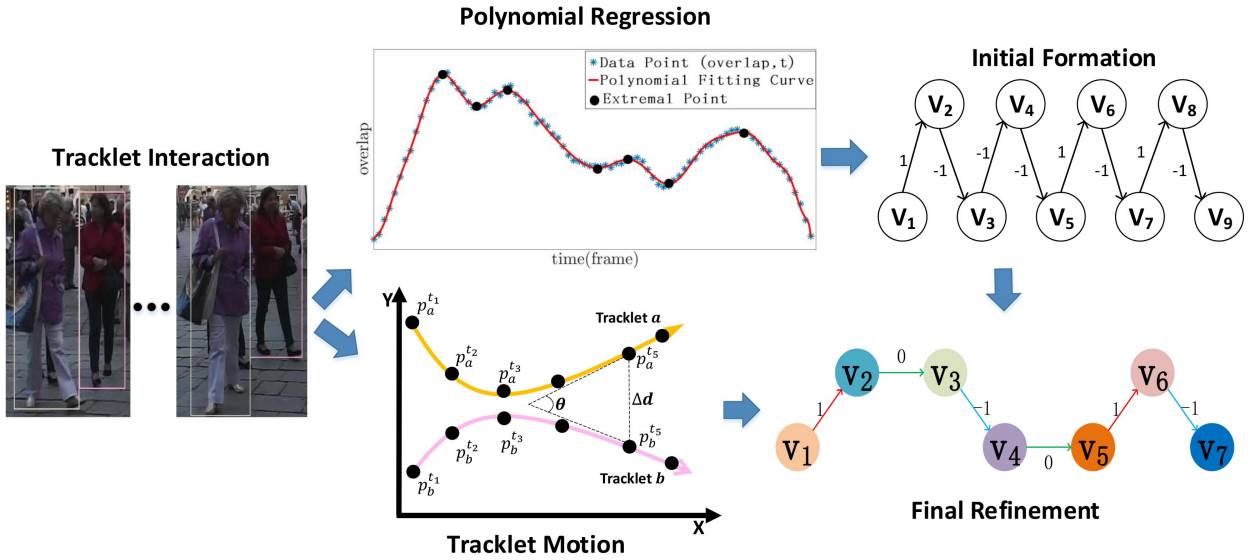


Fig. 3. Illustration of the STIG construction for tracklet interaction. An initial graph is formed by polynomial regression. Then it is refined according to the tracklet motion.

spatial nor temporal domains. Inclusion hypothesis H_{in} is defined as:

$$H_{in} = \{< t_i, t_j > | T(t_i) \cap T(t_j) = \emptyset, \; S(t_i) \cap S(t_j) = \emptyset\} \tag{4}$$

b) Exclusion: When two tracklets t_i and t_j are likely to have an exclusion relationship, they have intersection only in temporal domain. Exclusion hypothesis H_{ex} is defined as:

$$H_{ex} = \{< t_i, t_j > | T(t_i) \cap T(t_j) \neq \emptyset, \; S(t_i) \cap S(t_j) = \emptyset\} \tag{5}$$

c) Coexistence: When two tracklets t_i and t_j are likely to have a coexistence relationship, they intersect in both spatial and temporal domains. Coexistence hypothesis H_{co} is defined as:

$$H_{co} = \{< t_i, t_j > | T(t_i) \cap T(t_j) \neq \emptyset, \; S(t_i) \cap S(t_j) \neq \emptyset\} \tag{6}$$

3) N-ary Hypothesis: The N-ary relationship R_n among tracklet tuple $< t_{i,1}, \ldots, t_{i,n} >$ can be derived from the unary relationship R_1 and the binary relationship R_2. Given the relationship $R = \{R_1, R_2\}$ on tracklets, we can build the transitive closure $\tau(R)$, which is defined as:

$$\tau(R) = \bigcup_{i=0}^{n} R^i \tag{7}$$

where R^0 is the identity relation and $R^{i+1} = R^i \cdot R$. Then N-ary relationship R_n can be derived from the transitive closure:

$$R_n = \{< t_{i,1}, \ldots, t_{i,n} > | \forall p, q \in [1, n], < t_p, t_q > \in \tau(R)\} \tag{8}$$

Therefore, N-ary hypothesis H_n can be reduced to a combination of unary hypotheses and binary hypotheses.

IV. Spatio-Temporal Interaction Modeling

As introduced in Sec.III-B.2, spatio-temporal interaction information is used for binary hypothesis construction. The spatial and temporal intersection of tracklets is a general description which needs more specific definition for tracking. In this section, a weighted directed graph model named as spatio-temporal interaction graph (STIG) is proposed to formulate the interaction between tracklets in detail.

A. Definition

For a given tracklet t_i, if it consists of a set of detections $\{d_{i,1}, d_{i,2}, \ldots, d_{i,k}\}$ coming from frames $\{f_{i,1}, f_{i,2}, \ldots, f_{i,k}\}$ respectively. The temporal intersection of pairwise tracklets t_i and t_j can be expressed as:

$$\begin{aligned} I_t(t_i, t_j) &= T(t_i) \cap T(t_j) \\ &= \{f_{i,1}, f_{i,2}, \ldots, f_{i,k}\} \cap \{f_{j,1}, f_{j,2}, \ldots, f_{j,k}\} \end{aligned} \tag{9}$$

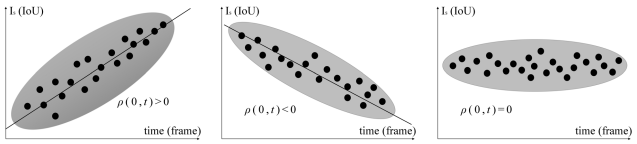
Then, the spatial intersection of pairwise tracklets t_i and t_j at frame k can be expressed as:

$$\begin{aligned} I_s^k(t_i, t_j) &= S(t_i^k) \cap S(t_j^k) \\ &= \frac{d_{i,k} \cap d_{j,k}}{d_{i,k} \cup d_{j,k}} \end{aligned} \tag{10}$$

where $k \in I_t(t_i, t_j)$. I_s^k describes the intersection-over-union (IoU) of two tracklets at certain frame. Obviously, the value of I_s^k ranges 0 to 1. The relation between intersection of time can be formulated by the Pearson correlation coefficient as follows:

$$\rho(I_s, f) = \frac{cov(I_s, f)}{\sigma_{I_s} \sigma_f}, \quad f \in I_t \tag{11}$$

Therefore, three basic patterns of pairwise tracklets interaction are defined according to $\rho(I_s, f)$.



(a) (b) (c)

Fig. 4. Three types of basic interaction patterns. (a) is the state of aggregation in which the overlap is positively related to time. (b) is the state of abruption in which the overlap is negatively related to time. (c) is the state of stabilization in which the overlap is around a constant.

1) Aggregation: It is an interaction pattern in which the spatial intersection I_s is positively related to the time (frame) as illustrated in 4(a), expressed as:

$$\rho(I_s, f) > 0 \tag{12}$$

Aggregation describes an upward intersection trend during tracklets interaction.

2) Abruption: It is another interaction pattern in which the spatial intersection I_s is negatively related to the time (frame) as illustrated in 4(b), expressed as:

$$\rho(I_s, f) < 0 \tag{13}$$

Aggregation describes a downtrend of tracklets intersection.

3) Stabilization: It is an interaction pattern in which the spatial intersection I_s is nearly maintained at a constant as illustrated in 4(c), expressed as:

$$\rho(I_s, f) = 0 \tag{14}$$

Tracklets of stabilization move close with a similar motion while the intersection is in a stable range.

B. STIG Initialization

According to the different value of ρ in Sec.IV-A, the entire period of tracklet interaction can be divided into multiple basic interaction patterns. However, the interaction between tracklets is constantly changing and cannot be described by a single interaction pattern. In this section, a weighted directed graph named as STIG is proposed to describe the changing interaction of pairwise tracklets. Therefore, the interaction modeling problem can be formulated by constructing graph $G = (V, E; W)$. Each edge $e_{i,j} =< v_i, v_j >$ links tracklets to represent a basic interaction pattern where v_i and v_j represent the start and end frames of this interaction. The weight $w_{i,j} \in W$ of the corresponding edge have three possible values of 1, -1 and 0, representing different interaction patterns. As illustrated in Fig.3, the construction of STIG consists of two steps including initialization and refinement.

STIG represents an optimal division of the entire interaction which describes the general trend of tracklets interaction. Specifically, the relation between intersection and time can be modeled by regression analysis to fit the general trend of interaction. For a given interaction period during frames $f = \{f_1, f_2, \ldots, f_n\}$, the entire interaction can be modeled by a polynomial function. Let F^r_{poly} be the polynomial fitting function of order r which is defined as:

$$F^r_{poly}(f_k) = \sum_{i=0}^{r} \omega_i f_k^i \tag{15}$$

For a specific order r, we can use the least square method to find the fitting function F^r_{poly}. However, in the real world, the intersection of pairwise tracklets does not change frequently and greatly in a short time. Based on this consideration, we solve finding the optimal fitting F^r_{poly} as an optimization problem formulated as follows:

$$\arg\min \; E_{rms}(r) = \sqrt{\frac{\sum_{k=f_1}^{f_n} (F^r_{poly}(k) - I_s^k)^2}{\|f\|_0}} + \lambda \sum_{i=0}^{r} \omega_i^2 \tag{16}$$

$$s.t. \; |F^r_{poly}{}'(f_k)| < \theta_{th}, \quad \forall f_k \in f \tag{17}$$

$$|f_p - f_q| > f_{th}, \quad \forall f_p, f_q,$$

$$F^r_{poly}{}''(f_p) = 0, F^r_{poly}{}''(f_q) = 0 \tag{18}$$

$$r = 0, 1, 2, \ldots, n \tag{19}$$

where λ is set to 0.01 for the regular penalty term to penalize over-fitting. Thus the optimal order of the fitting function can be found by minimize the RMS error $E_{rms}(r)$. In addition, θ_{th} and f_{th} are also used to avoid over-fitting. In Eq.17, the first derivative of F^r_{poly} is limited by θ_{th} to prevent the interaction pattern from changing frequently. In Eq.18, the limitation f_{th} on the second derivative ensures the trend changing smoothly. θ_{th} is set to 0.5 and f_{th} is set to 10 frames.

It is difficult to find the optimal fitting F^r_{poly} due to the high computational complexity when the tracklet is too long. So we introduce an efficient algorithm in Alg.1 to find a suboptimal solution instead. We traverse r from small to large in group of 3, and take the minimum $F^{r_o}_{poly}$ from the first group with feasible solution as the suboptimal solution. Therefore, $F^{r_o}_{poly}$

Algorithm 1 Finding the Suboptimal Solution of Polynomial Fitting

Input: Interaction period $f = \{f_1, \ldots, f_k, \ldots, f_n\}$
Output: Polynomial F^r_{poly}
Parameters: λ, θ_{th}, f_{th}, E_{th}

```
for each f_k ∈ f do
    calculate I_s^k by Eq.10
end for

for every five ordered elements {i_1, ..., i_3} ∈ [0, ||f||_0] do
    calculate {E_rms(i_1), ..., E_rms(i_3)} by Eq.16
    if E_rms(i_t) = min(E_rms(i)) satisfies Eq.17 and Eq.18 then
        r = i_t
        F^r_poly = f^{i_t}_poly
        break
    end if
end for
```

is a feasible polynomial with relatively low order that fits the trend and avoids over-fitting problem at the same time.

Then we generate an initial STIG $G = \{V, E; W\}$ by $F^{r_0}_{poly}$. The peaks and troughs of $F^{r_0}_{poly}$ are the most potential critical points in the interaction period. We define V as follows:

$$V = \Theta(F^{r_0}_{poly}) \cup f \tag{20}$$

where $\Theta(F^{r_0}_{poly})$ is the set of all extreme points. Two adjacent extreme points are the start and end points of a basic interaction and elements in V are sorted in ascending order through frames and define E as follows:

$$E = \{e_{i,i+1} | v_i \in V, v_{i+1} \in V\} \tag{21}$$

Since $F^{r_0}_{poly}$ is a polynomial function, it is continuous and differentiable. So the trend of IoU through frames can be reflected by its derivative. If the derivative is positive from v_i to v_{i+1}, the basic interaction in this period can be regarded as an aggregation process with growing IoU. Similarly, if the derivative is negative from v_i to v_{i+1}, we consider this basic interaction as an abruption process. Otherwise, the basic interaction between v_i and v_j is classified as a stabilization process. Therefore, the weight of $e_{i,i+1}$ can be defined as:

$$w_{i,i+1} = \begin{cases} 1, & \forall f_k \in [f_{v_i}, f_{v_{i+1}}],\ {F^{r_0}_{poly}}'(f_k) > 0 \\ -1, & \forall f_k \in [f_{v_i}, f_{v_{i+1}}],\ {F^{r_0}_{poly}}'(f_k) < 0 \\ 0, & \forall f_k \in [f_{v_i}, f_{v_{i+1}}],\ {F^{r_0}_{poly}}'(f_k) = 0 \end{cases} \tag{22}$$

where f_{v_i} and $f_{v_{i+1}}$ are the start and end frames of the interaction according to $e_{i,i+1}$.

C. STIG Refinement

The derivative of $F^{r_0}_{poly} = 0$ is a sufficient and unnecessary condition of the interaction to be a stabilization process. It is an ideal situation that the Pearson correlation coefficient $\rho(I_s, f)$ equals 0 when a tracklet interaction belongs to stabilization. In the real world, if two pedestrians are walking side-by-side with similar motion, I_s is always changing through frames. However, it ofter just changes in a certain extent. Thus we further propose a method to refine the initialized STIG which can better describe the trend of tracklet interaction.

For a given interaction during frames $f = \{f_1, f_2, \ldots, f_n\}$, it can be described as an edge $e_{i,i+1}$ with weight $w_{i,i+1}$ in the initialized STIG. Its weight is updated by the following formula:

$$w_{i,i+1} = \begin{cases} 0, & max(I_s) - min(I_s) < I_{th} \\ w_{i,i+1}, & otherwise \end{cases} \tag{23}$$

where I_{th} represents the maximum degree that IoU can shift during a stabilization process. I_{th} is set to 0.5 in this paper.

Since the weights of the STIG are updated, there are adjacent edges such as $e_{i,i+1}, e_{i+1,i+2}, \ldots, e_{j-1,j}$ with weights of 0. These edges are merged into a new edge as $e_{i,j}$ and remove the corresponding vertices except v_i and v_j. Therefore, the refined STIG is constructed to describe the interaction of pairwise tracklets as shown in Fig.3

D. STIG for Hypothesis Testing

In Sec.III-B.2, three different types of binary hypothesis for tracklets are introduced. Since STIG describes the spatio-temporal interaction of pairwise tracklets, it can test hypotheses among tracklets.

For give pairwise tracklets t_i and t_j, we find the polynomial fitting $F^{r_0}_{poly}$ by Alg.1 and construct a STIG graph $G = \{V, E; W\}$ to describe the interaction between them through frames. According to G, the entire interaction process in divided into several basic interaction patterns. If two targets approach each other from a distance, the weight $w_{i,j}$ of the corresponding edge in G is 1, representing that the value of $F^{r_0}_{poly}$ is gradually increases from near-zero. If two targets are moving away from each other, the weight of the corresponding edge is -1 while $F^{r_0}_{poly}$ drops from a high level. If the weight of the edge of the corresponding interaction is 0, it mean the value of $F^{r_0}_{poly}$ does not change greatly and the two targets are probably walking side-by-side.

If t_i and t_j have intersection in temporal domain, exclusion hypothesis H_{ex} and coexistence hypothesis H_{co} are constructed between them. Then, these hypothesis can be tested and decided whether or not to accept according to G.

As defined in Eq.5, tracklets with exclusion hypothesis do not have intersection in spatial domain. According to the meaning of G, if all the weights in G is 0 and the value of $F^{r_0}_{poly}$ is always equal to 0, it means that the pairwise tracklets do not have spatial intersection. Formally, the exclusion hypothesis H_{ex} between t_i and t_j is accepted only if the following statement is true.

$$\forall w_{i,i+1} \in W, \quad w_{i,i+1} = 0 \wedge F^{r_0}_{poly} \equiv 0 \tag{24}$$

According to the definition of H_{co} in Eq.6, pairwise tracklets with coexistence hypothesis H_{co} have intersection in both temporal and spatial domains. Corresponding to G, the value of $F^{r_0}_{poly}$ is always positive. Formally, the coexistence hypothesis H_{co} is accepted only if the following statement is true.

$$min(F^{r_0}_{poly}) > 0 \tag{25}$$

Another binary hypothesis is inclusion hypothesis H_{in}. Different from H_{ex} and H_{co}, pairwise tracklets with inclusion hypothesis H_{in} do not have intersection in temporal domain. A typical example is that a pedestrian is occluded by another one in a certain period, so its tracklets are visible only before and after the occlusion. This common phenomenon can be described by STIG. For a given target with two separated tracklets t_i and t_j, where t_i is ahead of t_j in time, another target that occludes it is defined as tracklet t_k. Inclusion hypothesis H_{in} for t_i and t_j is constructed and tested by STIG. Let $G_{i,k} = \{V_1, E_1; W_1\}$ describe interaction between t_i and t_k, and $G_{j,k} = \{V_2, E_2; W_2\}$ represent t_j and t_k. If the last edge $e_{n-1,n}$ in $G_{i,k}$ shows an aggregation process while the first edge $e_{1,2}$ in $G_{j,k}$ describes abruption, t_k is regarded as an occlusion. Formally, the inclusion hypothesis H_{in} of t_i and t_j is accepted only if the following statement is true.

$$w_{n-1,n} \in W_1, \quad w_{n-1,n} = 1 \wedge w_{1,2} \in W_2,\ w_{1,2} = -1 \tag{26}$$

Therefore, three different types of binary hypotheses for pairwise tracklets can be tested through STIG and the polynomial fitting.

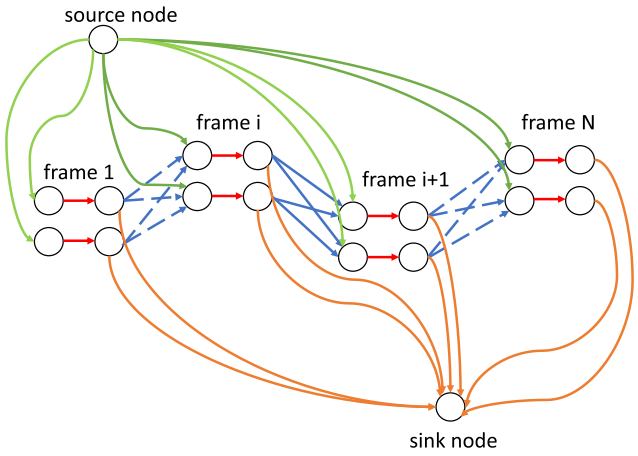


Fig. 5. Illustration of network flow tracking. A trajectory is start with a green line and end with an orange line. Red lines represent detections in different frames while the blue lines link detections in adjacent frames.

V. HTBT Application With STIG

In this section, the traditional formulation of tracking with network flow framework is introduced. Then, we analyze the shortcoming of the traditional network flow in terms of data association and propose a method to handle it with HTBT.

A. Network Flow Tracking

Network flow tracking simultaneously estimates the trajectories based on the tracking-by-detection paradigm. The input of the algorithm mainly includes two aspects. One is the observation information of the target to be tracked in the video, which is typically obtained by the target detection algorithm. The other is the similarity information among detections, which can be obtained via various methods, such as optical flow [35], appearance model [12], motion model [20], etc.

Therefore, tracking multiple targets can be converted into a data association problem and formulated as a maximum a posteriori (MAP) problem. The maximum posterior probability includes the detection cost and the linking cost between detections in adjacent frames. In addition, to ensure the validity of the tracking results, the results should correspond to the actual target trajectory. The network flow can be defined as a graph $G = \{V, E_1, E_2; W\}$, then MAP can be formulated as the following integer programming problem:

$$\arg\max_{x_i} \; E = \sum_{x_i \in V} w_{ij} x_i x_j \tag{27}$$

$$s.t.\ \forall e_{ij} \in E_1, \quad x_i = x_j, \tag{28}$$

$$x_i = 0 \ or \ 1 \tag{29}$$

where E_1 is the set of edges linking pairwise nodes to describe a detection and E_2 represents association between detections in adjacent frames. Each edge e_{ij} in E_1 links a pairwise nodes (x_i, x_j) to represent a detection, so its corresponding weight w_{ij} is defined as the confidence of the detection. Each edge e_{ij} in E_2 links nodes in adjacent frames to describe the association between targets, its weight w_{ij} is defined by the similarity between detections. In addition, the value of each node $x_i = 1$ if the corresponding detection is used in the trajectory. Otherwise, $x_i = 0$ to represent a false detection.

Therefore, a set of trajectories that satisfies the conditions has an energy according to Eq.30, so the tracking task is converted to find an optimal solution with the maximum energy.

In summary, network flow tracking converts the data association problem into a MAP problem and set flow constraints to ensure that the network flow solution corresponds to the correct tracking results. Only association between targets in adjacent frames are considered while the potential associations across frames are ignored. To exploit more possible associations, we integrate HTBT into the network flow as a new framework. In our framework, more potential associations are built by constructing various kinds of hypotheses between tracklets. Then, these additional hypotheses are tested by STIG model for tracklet association refinement. The whole process is performed iteratively during network flow tracking, as shown in Fig.5.1. In this way, the tracking results are obtained by finding the optimal solution of the network flow.

Our network flow tracking is based on tracklets instead of discrete detections that are used in traditional network flow. We separate the entire video into a series of windows with fixed frames. In this paper, we set the size of the window to 5 frames as same as our baseline method TEM [10]. Then each node x_i in the network flow G represents a tracklet. In addition, e_{ij} in E_1 represents a tracklet and e_{ij} in E_2 links tracklets in adjacent windows.

B. Robust Tracklet Association With HTBT

The main drawback of the traditional network flow method is that it cannot represent the relationship between detections with large gap in spatial or temporal domain. As a result, lots of potential associations are ignored which makes it difficult to describe complex relationship between targets. To handle this defect, we further exploit the association between targets in the network flow by constructing and testing possible hypotheses.

Considering the structure of the network flow framework, E_2 in G is extended with edges linking tracklets in the same window and across windows as hypothesis terms. Therefore, the objective energy function with hypothesis testing term h_{ij} can be defined as follows:

$$\arg\max_{x_i} \; E = \sum_{x_i \in V} w_{ij} h_{ij} x_i x_j \tag{30}$$

$$s.t.\ \forall e_{ij} \in E_1, \quad x_i = x_j \tag{31}$$

$$x_i = 0 \ or \ 1 \tag{32}$$

$$h_{ij} = 0 \ or \ 1 \tag{33}$$

where h_{ij} represents the hypothesis between tracklets t_i and t_j. Hypothesis term h_{ij} is 1 if the corresponding hypothesis is accepted, otherwise it equals 0. Hypothesis h_{ij} for edge in E_1 represents unary hypothesis while h_{ij} for edge in E_2 describes the binary hypothesis between tracklets. Unary hypothesis is tested by Eq.3 and binary hypothesis can be tested by the method in Sec.IV-D.

These additional hypothesis terms can model more relationship between tracklets than traditional network flow

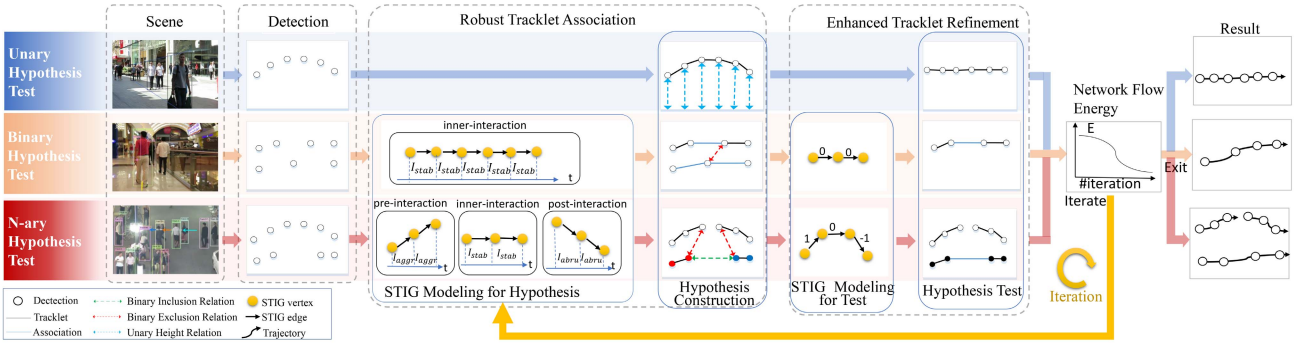


Fig. 6. The framework of hypothesis-testing based tracking (HTBT) with network flow. Each iteration consists of two main steps, including robust tracklet association based on hypothesis construction and enhanced tracklet refinement with hypothesis testing.

framework. Tracklets are linked not only in adjacent windows, but pairwise tracklets across windows are also associated by the hypothesis terms. By integrating with HTBT, the ability of traditional network flow framework in representing more complicated relationship between targets is improved.

C. Enhanced Tracklet Refinement With HTBT

Since the potential association among tracklets are described with hypothesis terms, we further discuss the weight of these additional edges. For a given edge e_{ij}, if it belongs to E_1, its weight w_{ij} represents the confidence of the corresponding tracklet and unary hypothesis, which can be defined as follows:

$$w_{ij} = w_c + w_{H_1} \tag{34}$$

$$w_{H_1} = ave(\frac{|h_{ij} - \tilde{h_{ij}}|}{\tilde{h_{ij}}}) \tag{35}$$

where w_c is the average confidence of detections in the tracklet and w_{h_1} is the average of the relative error between the height of the detection h_{ij} and the predicted height $\tilde{h_{ij}}$ calculated by $M(x, y)$ in Eq.3. w_{H_1} describes the confidence of accepting the corresponding unary hypothesis.

If edge e_{ij} belongs to E_2 linking tracklets t_i and t_j, the corresponding weight w_{ij} represents the similarity between tracklets and the cost of the binary hypothesis. The binary hypothesis is categorized into three types in Sec.III-B, including inclusion, exclusion and coexistence. According to the definition of the binary hypothesis, different binary hypotheses are not compatible with each other, so there is at most one binary hypothesis between a pairwise tracklet. Therefore, different weight for the corresponding binary hypothesis $H_2(i, j)$ can be defined respectively as follows:

$$w_{ij} = w_a + w_{H_2} \tag{36}$$

$$w_a = ave(cos(a_i, a_j)) \tag{37}$$

$$w_{H_2} = \begin{cases} w_{H_2} = e^{-ave(d_{ij})}, & H_2(i, j) \subset H_{in} \\ w_{H_2} = 0, & H_2(i, j) \subset H_{ex} \\ w_{H_2} = -ave(I_s), & H_2(i, j) \subset H_{co} \end{cases} \tag{38}$$

where $cos(a_i, a_j)$ is the cosine distance between the appearance features of t_i and t_j and the similarity is defined as the average cosine distance. $ave(d_{ij})$ is the average distance between detections in t_i and t_j. The weight w_{H_2} for binary hypothesis has different definition according to its type. If $H_2(i, j)$ belongs to inclusion hypothesis, it means the pairwise tracklet is likely to be of the same target. Thus w_{H_2} encourages two tracklets to be linked. In contrast, if $H_2(i, j)$ belongs to coexistence hypothesis, t_i and t_j have high probability to represent different target. w_{H_2} is set as a penalty to discourage them from being linked, where w_{H_2} is a non-positive value.

VI. Experiments

Platform: HTBT tracking in this paper is implemented through MATLAB 2019b and the parallel optimization by the GPU toolbox is used as well. The configuration of our hardware platform consists of i7-9700K, GeForce RTX 2080 and 32GB DDR4 RAM.

Dataset: Our method is tested on both MOT16 [26], MOT17 and MOT20 [14] benchmarks. MOTChallenge benchmarks have been widely used for fair comparison in recent years, which contain video sequences in various unconstrained environments. There are 7 training sequences and 7 test sequences with 11235 frames in the MOT16 benchmark and 21 training sequences and 21 test sequences with 33705 frames in the MOT17 benchmark. MOT20 is the latest benchmark for multi-object tracking which contains 4 training and 4 test sequences with 6811 frames. It is the most difficult challenge for MOT at present with density over 100 per frame. Sec.VI-A presents the results of hypothesis testing for tracklet association on MOT16 training dataset, which are used to evaluate the performance of HTBT. Sec.VI-B evaluates the performance of our proposed STIG model. Sec.VI-C presents the results on the training sequences to evaluate tracking performance of network flow with HTBT. Sec.VI-D shows the comparison results of our method with other state-of-the-art trackers. For fair comparison, we use the public detections provided by the benchmark as the inputs of our methods.

Evaluation Metrics: We use the standard CLEAR MOT metrics [4] to evaluate the tracking performance. MOTA↑ (multiple object tracking accuracy) combines three types of errors: FP↓ (false positives), FN↓ (false negatives), IDs↓ (identity switches) and Hz↑ (computational speed). MOTP↑ (multiple object tracking precision) is the precision of the output trajectories relative to the ground truth. IDF1 [30] is the

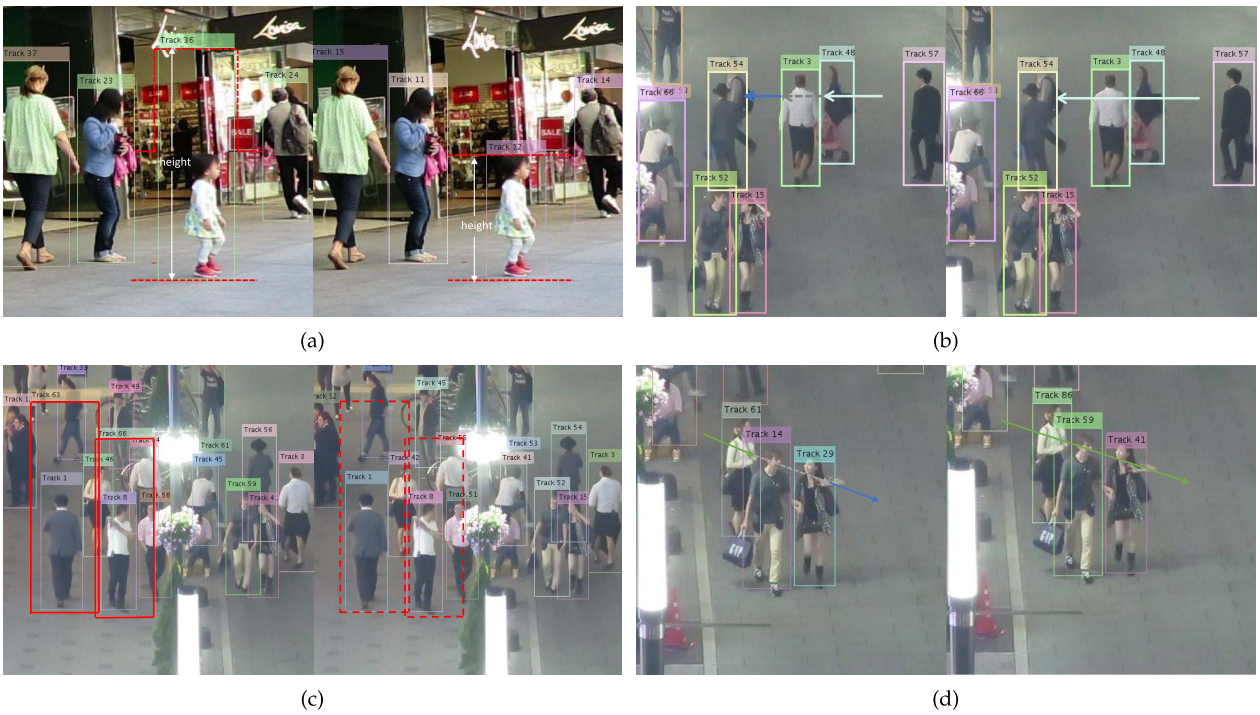


Fig. 7. Results without and with hypothesis testing. Arrows represent the trajectories of targets. (a) On the left, it is the result without unary hypothesis and the result with unary hypothesis testing on the right. (b) On the left, it is the result without binary inclusion hypothesis and the result with binary inclusion hypothesis on the right. (c) On the left, it is the result without the binary exclusion hypothesis and results with binary exclusion hypothesis on the right. (d) On the left, it is the result without the N-ary hypothesis and results with N-ary hypothesis on the right.

ratio of correctly identified detections to the average number of ground truth and computed detections. MT↑ (the number of mostly tracked trajectories), ML↓ (the number of mostly lost trajectories), and FM↓ (track fragmentations) are also reported. MOTA is mainly used to compare trackers. However, MOTA does not properly account for identity switches [24], [41]. Unlike MOTA, IDF1 penalizes switches over the whole trajectory in which fragments are assigned the wrong identity [30], [32]. Therefore, MOTA and IDF1 are reported in our results. The indicator ↑ denotes the higher the better while ↓ denotes the lower the better.

A. Hypothesis Testing for Tracklet Interaction

In this subsection, we analyze the effectiveness of the hypothesis testing in HTBT qualitatively. Since the hypothesis are used to revise the tracklet association in tracking, the number of tracklets with different interaction is counted manually in the baseline method [10] on MOT16 training dataset, including unary hypothesis, binary hypothesis and N-ary hypothesis. In addition, the number of the revised tracklets with HTBT method is listed in Tab.I. The percentage of the revised tracklets can reflect the effectiveness of HTBT in revising tracklet association.

First, we present the results of revising tracklets with unary hypothesis. Unary hypothesis is constructed to find false detections. As demonstrated in Fig.7(a), there are false tracklets with abnormal height in the scene. Through our unary hypothesis construction and testing, we discover false detections according to the height normalization method. More than 50% of the false tracklets with unary relationship are revised in HTBT compared with the baseline. It shows the effectiveness of unary hypothesis testing on revising tracklets.

Then, we present the results of the binary hypothesis testing for tracklets with inclusion relationship. Inclusion hypothesis describes mutual inclusion relationship between interactive tracklets. As shown in the left of Fig.7(b), tracklet (ID: 48) without inclusion hypothesis testing leads to fragmentation and identity switches problems. Through binary hypothesis testing, we construct binary hypothesis between potential pairwise tracklets with inclusion relationship. Therefore, in Tab.I, near 70% of the false tracklets are found and revised in HTBT, which shows the effectiveness of the binary inclusion hypothesis.

The results of binary exclusion hypothesis are also presented in the table. Exclusion hypothesis represents exclusion relationship between tracklets. As shown in Fig.7(c), there are false overlapping trajectories (ID: 63 and 66) without constructing exclusion hypothesis. We are able to avoid this kind of false trajectories with exclusion hypothesis testing. In Tab.I, 63.3% of these false tracklets are revised. The high percentage shows the performance of the exclusion hypothesis testing in HTBT.

N-ary relationship is a kind of complex relationship among targets with various forms. We show a typical N-ary relationship in the left of Fig.7(d). Tracklet (ID: 61) is occluded by two pedestrian (ID: 14 and 29) which leads to trajectory fragmentation and identity switches. Through constructing N-ary hypothesis among targets, we generate the complete trajectory (ID: 86) without fragmentation in the right of Fig.7(d). According to Tab.I, a total of 71 false tracklets with N-ary relationship are revised, near 60% of the false tracklet

TABLE I
RESULTS OF HYPOTHESIS TESTING FOR TRACKLET INTERACTION IN MOT16 TRAINING

Hypothesis	Number of tracklets in baseline[10]	Number of revised tracklets	Percentage
Unary hypothesis	171	102	59.6%
Binary inclusion hypothesis	183	127	69.4%
Binary exclusion hypothesis	158	100	63.3%
N-ary hypothesis	119	71	59.7%

TABLE II
RESULTS ON MOT16 TRAINING DATASET

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓	Hz↑
baseline[10]	40.2	78.2	44.5	76	232	5329	60235	408	480	**23.7**
baseline+H_1	41.9	**78.3**	45.5	76	233	**3237**	60512	383	463	22.5
baseline+H_2	42.1	78.1	49.5	**79**	**230**	5003	**58667**	**308**	**414**	14.8
baseline+H_1+H_2 (HTBT)	**43.1**	78.1	**49.6**	**79**	**230**	3862	58683	309	415	14.6

TABLE III
RESULTS ON MOT17 TRAINING DATASET

Method	MOTA↑	MOTP↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓	Hz↑
baseline[10]	51.5	**85.2**	54.0	341	618	7034	155284	1190	1341	**21.3**
baseline+H_1	52.1	85.0	57.9	357	614	**6391**	153887	930	1206	19.2
baseline+H_2	53.3	84.7	59.4	387	598	8994	147557	916	1190	13.9
baseline+H_1+H_2 (HTBT)	**53.7**	84.6	**60.8**	**396**	**593**	9725	**145395**	**905**	**1175**	13.1

in the baseline method. It proves that N-ary hypothesis testing performs expectably in handling relationship among multiple tracklets.

B. Spatio-Temporal Interaction Modeling

In Sec.IV, we propose a STIG model to formulate various interaction between tracklets which is used for binary hypothesis testing. In this section, we set comparison experiments to test the performance of our spatio-temporal interaction modeling approach. In Tab.II and Tab.III, we show the results of the baseline method, baseline with unary hypothesis, baseline with binary hypothesis and HTBT tracking which integrated with both unary and binary hypothesis. With unary hypothesis testing, our method achieve higher MOTA and IDF1 by reducing the number of the FP and IDs. By integrating binary hypothesis testing into the baseline, we achieve a higher score on MOTA and IDF1. Then, we test HTBT tracking with both unary and binary hypothesis testing and get the best results on both MOT16 and MOT17 training datasets. By compare the tracking performance of baseline with and without HTBT, almost all of the evaluation metrics are better with HTBT, especially on identity switches. It proves that HTBT can substantially improve the overall tracking performance.

This comparison experiment indicates that binary hypothesis has better performance on enhancing tracklet association than only using unary hypothesis. In addition, the results show that binary hypothesis works together with unary hypothesis.

C. Framework Verification

Our HTBT framework with STIG modeling is an iterative process with energy minimization, as discussed in Sec.IV. When the number of trajectories is greater than that in the previous iteration, we terminate the iteration and take the previous results as the final trajectories.

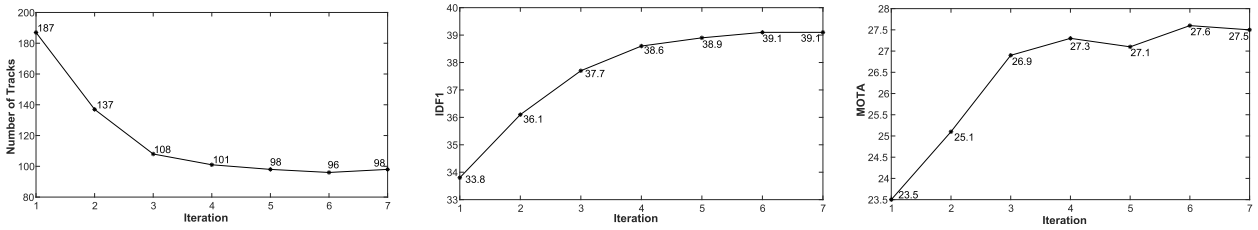
Experimental results have demonstrated that this strategy is effective on most sequences, e.g., MOT17-02-DPM. During the process of iteration, the changes in the numbers of trajectories, ID switches and MOTA on MOT17-02-DPM sequences are presented in Fig.6.2. The number of trajectories decreases with the process of iteration because additional tracklets are linked into longer trajectories in each iteration. As a result, MOTA and IDF1 are improved and remain steady as the number of iterations increases. These metrics change in the same way on the other sequences in the MOT17 dataset.

In addition, we analyze the computational efficiency of our method. As shown in the last column in TabII and Tab.III, the baseline method shows absolute speed advantage compared with HTBT. It can process more than 20 frames per second. However, our method integrates HTBT into the network flow but can still maintain a processing speed of 10 frames per second, which is much higher than lots of other offline tracking methods. Since our implementation of HTBT is based on MATLAB, which suffers from low efficiency on memory management, there is still much room to improve the efficiency.

D. Benchmark Comparison

Finally, we evaluate HTBT tracker on the MOT16 and MOT17 benchmarks. The comparison of our method with other state-of-the-art trackers is presented in Tab.IV and Tab.V.

Tab.IV presents the results on the MOT16 benchmark. Our method gets competitive score on two aggregative metrics including MOTA and IDF1. Our tracker takes the first place on MOTA by 50.3 and the second highest score on IDF1 by



(a) (b) (c)

Fig. 8. The number of trajectories, MOTA and IDF1 during the iterative process on the MOT17-02-DPM sequence. With the gradual stabilization of the number of trajectories, MOTA and IDF1 grow higher and tend to be more stable.

TABLE IV
RESULTS ON MOT CHALLENGE 2016 BENCHMARK (2020.4)

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓
Ours (HTBT16)	**50.3**	55.0	19.2%	39.8%	8341	81843	490	754
HCC [23]	49.3	50.7	17.8%	39.9%	**5333**	86795	**391**	**535**
AFN [31]	49.0	48.2	**19.1%**	**35.7%**	9508	**82506**	899	1383
KCF16 [11]	48.8	47.2	15.8%	38.1%	5875	86567	906	1116
LMP [36]	48.8	51.2	18.2%	40.1%	6654	86245	481	595
TLMHT [32]	48.7	**55.3**	15.7%	44.5%	6632	86504	413	642
eHAF16 [33]	47.2	52.4	18.6%	42.8%	12586	83107	542	787
MHT_DAM [19]	45.8	46.1	16.2%	43.2%	6412	91758	590	781
INTERA_MOT [21]	45.4	47.7	18.1%	38.7%	13407	85547	600	930
EDMT [9]	45.3	47.9	17.0%	39.9%	11122	87890	639	946

TABLE V
RESULTS ON MOT CHALLENGE 2017 BENCHMARK (2020.4)

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓
JBNOT [31]	**52.6**	50.8	19.7%	35.8%	31572	232659	3050	3792
Ours (HTBT)	52.3	54.5	22.5%	36.4%	28743	238268	1959	2973
eHAF17 [33]	51.8	**54.7**	**23.4%**	37.9%	33212	**236772**	1834	**2739**
AFN17 [31]	51.5	46.9	20.6%	35.5%	**22391**	248420	2593	4308
FWT [17]	51.3	47.6	21.4%	**35.2%**	24101	247921	2468	4279
jCC [18]	51.2	54.5	20.9%	37.0%	25937	247822	**1802**	2984
MHT_DAM [19]	50.7	47.2	20.8%	36.9%	22875	252889	2314	2865
TEM [10] (baseline)	49.1	45.4	17.0%	38.3%	**22119**	261797	3439	3881

TABLE VI
RESULTS ON CVPR CHALLENGE 2019 AND MOT20 BENCHMARK (2020.4)

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDs↓	FM↓	Remark
Tracktor++[3]	**51.3**	47.6	313	326	**16,263**	253,680	2,584	4,824	CVPR19
Ours (HTBT)	48.9	**54.6**	**384**	**274**	45,660	**216,803**	**2,187**	**3,067**	MOT20
DD_TAMA19[40]	47.6	48.7	342	297	38,194	252,934	2,437	3,887	CVPR19
V_IOU[6]	46.7	46.0	288	306	33,776	261,964	2,589	4,354	CVPR19
HAM_HI[39]	43.0	43.6	353	**274**	72,018	243,055	4,153	4,801	CVPR19
IOU_19[5]	35.8	25.7	126	389	24,427	319,696	15,676	17,864	CVPR19

55.0. On most other metrics, our method performs comparably to other popular trackers.

On the more recent MOT17 benchmark, our results are presented in Tab.V. We gets the second highest score on both MOTA and IDF1, by 52.3 and 54.5 respectively. Due to constructing various hypothesis between targets, we are able to generate more complete trajectories. It is shown in the results that our method outperforms most of the others in terms of identity switches and fragmentations. TEM [10] is the baseline method which is based on network flow as well. Compared with it, our method shows substantial improvements on almost all metrics. Considering the high computational efficiency of the network flow, we are convinced that our method is practical for applications requiring both quality and efficiency.

In addition, we have conducted a comparison experiment on the latest MOT20 benchmark which is the most difficult dataset for MOT at present. Since the submission of MOT20 has just been opened, there are few methods that can be compared. However, MOT20 is almost the same dataset as CVPR Challenge 2019 (submission closed), expect that public detections are slightly different. Therefore, we evaluate our method on MOT20 and compare it with others on CVPR19 in Tab.VI. Our method takes the second place on MOTA by 48.9 while achieves the best performance on IDF1, MT, ML, FN, IDs and FM. Especially on IDF1 and IDs, HTBT tracker is obviously much better than others which shows that our method generates more complete trajectories.

VII. Conclusion

This paper proposes a hypothesis-testing based tracking (HTBT) method to construct and test hypothesis of tracklets association. It improves the performance and robustness of association for tracking in crowded scenes. According to the different features of interaction between trajectories, spatio-temporal interaction graph (STIG) model is proposed to describe the basic patterns of the interaction. By using STIG as the basis of hypothesis testing in HTBT, various association relationships between tracklets are built. Then, HTBT is integrated into traditional network flow framework to solve tracking as a MAP problem. The experimental results show that our method accurately describes various relationship between trajectories and improves the association between tracklets as well. Experimental results show that HTBT has great improvement compared with traditional network flow method. Our method achieves much better tracking performance and maintains the advantages of network flow method in computational efficiency at the same time. On the public MOT16, MOT17 and MOT20 benchmark, our method achieves competitive results compared with other state-of-the-art trackers.

References

[1] K. K. C. Amit, D. Delannay, L. Jacques, and C. D. Vleeschouwer, "Iterative hypothesis testing for multi-object tracking with noisy/missing appearance features," (Lecture Notes in Computer Science), vol. 2, pp. 412–426, 2012.

[2] A. Kumar K. C., D. Delannay, and C. De Vleeschouwer, "Iterative hypothesis testing for multi-object tracking in presence of features with variable reliability," 2015, *arXiv:1509.00313*. [Online]. Available: http://arxiv.org/abs/1509.00313

[3] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 941–951.

[4] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The clear mot metrics," *Eurasip J. Image Video Process.*, vol. 2008, no. 1, 2008, Art. no. 246309.

[5] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information," in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Aug. 2017, pp. 1–6.

[6] E. Bochinski, T. Senst, and T. Sikora, "Extending IOU based multi-object tracking by visual information," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.

[7] A. A. Butt and R. T. Collins, "Multi-target tracking by lagrangian relaxation to min-cost network flow," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1846—1853.

[8] V. Chari, S. Lacoste-Julien, I. Laptev, and J. Sivic, "On pairwise costs for network flow multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5537–5545.

[9] J. Chen, H. Sheng, Y. Zhang, and Z. Xiong, "Enhancing detection model for multiple hypothesis tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 18–27.

[10] J. Chen, H. Sheng, Y. Zhang, W. Ke, and Z. Xiong, "Community evolution model for network flow based multiple object tracking," in *Proc. IEEE 30th Int. Conf. Tools Artif. Intell. (ICTAI)*, Nov. 2018, pp. 532–539.

[11] P. Chu, H. Fan, C. C. Tan, and H. Ling, "Online multi-object tracking with instance-aware tracker and dynamic model refreshment," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2019, pp. 161–170.

[12] C. Dai, Y. Zheng, and X. Li, "Pedestrian detection and tracking in infrared imagery using shape and appearance," *Comput. Vis. Image Understand.*, vol. 106, no. 2, pp. 288–299, 2007.

[13] K. Demirbas, "Maneuvering target tracking with hypothesis testing," *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-23, no. 6, pp. 757–766, Nov. 1987.

[14] P. Dendorfer *et al.*, "MOT20: A benchmark for multi object tracking in crowded scenes," 2020, *arXiv:2003.09003*. [Online]. Available: http://arxiv.org/abs/2003.09003

[15] V. Enescu, I. Ravyse, and H. Sahli, "Visual tracking by hypothesis testing," in *Advanced Concepts for Intelligent Vision Systems*, J. Blanc-Talon, W. Philips, D. Popescu, and P. Scheunders, Eds. Berlin, Germany: Springer, 2007, pp. 13–24.

[16] Z. Fu, P. Feng, F. Angelini, J. Chambers, and S. M. Naqvi, "Particle PHD filter based multiple human tracking using online group-structured dictionary learning," *IEEE Access*, vol. 6, pp. 14764–14778, 2018.

[17] R. Henschel, L. Leal-Taixe, D. Cremers, and B. Rosenhahn, "Fusion of head and full-body detectors for multi-object tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Los Alamitos, CA, USA, Jun. 2018, pp. 1428–1437.

[18] M. Keuper, S. Tang, B. Andres, T. Brox, and B. Schiele, "Motion segmentation & multiple object tracking by correlation co-clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 1, pp. 140–153, Jan. 2020.

[19] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4696–4704.

[20] L. Kratz and K. Nishino, "Tracking pedestrians using local spatio-temporal motion patterns in extremely crowded scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 987–1002, May 2012.

[21] L. Lan, X. Wang, S. Zhang, D. Tao, W. Gao, and T. S. Huang, "Interacting tracklets for multi-object tracking," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4585–4597, Sep. 2018.

[22] Z. Li, L. Qi, W. Li, G. Jin, and M. Wei, "Track initiation for dim small moving infrared target based on spatial-temporal hypothesis testing," *J. Infr., Millim., THz Waves*, vol. 30, no. 5, pp. 513–525, May 2009.

[23] L. Ma, S. Tang, M. J. Black, and L. V. Gool, "Customized multi-person tracker," in *Proc. Comput. Vis. (ACCV)*. Springer, Dec. 2018, pp. 612–628.

[24] A. Maksai, X. Wang, F. Fleuret, and P. Fua, "Non-Markovian globally consistent multi-object tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2544–2554.

[25] N. McLaughlin, J. M. D. Rincon, and P. Miller, "Enhancing linear programming with motion modeling for multi-target tracking," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 71–77.

[26] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," 2016, *arXiv:1603.00831*. [Online]. Available: https://arxiv.org/abs/1603.00831

[27] A. Milan, K. Schindler, and S. Roth, "Detection- and trajectory-level exclusion in multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3682–3689.

[28] A. Milan, K. Schindler, and S. Roth, "Multi-target tracking by discrete-continuous energy minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2054–2068, Oct. 2016.

[29] G. Ren, I. D. Schizas, and V. Maroulas, "Distributed spatio-temporal multi-target association and tracking," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 4010–4014.

[30] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 17–35.

[31] H. Shen, L. Huang, C. Huang, and W. Xu, "Tracklet association tracker: An end-to-end learning-based association approach for multi-object tracking," 2018, *arXiv:1808.01562*. [Online]. Available: https://arxiv.org/abs/1808.01562

[32] H. Sheng, J. Chen, Y. Zhang, W. Ke, Z. Xiong, and J. Yu, "Iterative multiple hypothesis tracking with tracklet-level association," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 12, pp. 3660–3672, Dec. 2019.

[33] H. Sheng, Y. Zhang, J. Chen, Z. Xiong, and J. Zhang, "Heterogeneous association graph fusion for target association in multiple object tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 11, pp. 3269–3280, Nov. 2019.

[34] H. Sheng, Y. Zheng, W. Ke, D. Yu, X. Cheng, W. Lyu, and Z. Xiong, "Mining hard samples globally and efficiently for person re-identification," *IEEE Internet Things J.*, early access, 2020, doi: 10.1109/JIOT.2020.2980549.

[35] J. S. Supancic, III, and D. Ramanan, "Self-paced learning for long-term tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2379–2386.

[36] S. Tang, M. Andriluka, B. Andres, and B. Schiele, "Multiple people tracking by lifted multicut and person re-identification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3701–3710.

[37] J. Wei, M. Yang, and F. Liu, "Learning spatio-temporal information for multi-object tracking," *IEEE Access*, vol. 5, pp. 3869–3877, 2017.
[38] S.-K. Weng, C.-M. Kuo, and S.-K. Tu, "Video object tracking using adaptive Kalman filter," *J. Vis. Commun. Image Represent.*, vol. 17, no. 6, pp. 1190–1208, Dec. 2006.
[39] Y.-C. Yoon, A. Boragule, Y.-M. Song, K. Yoon, and M. Jeon, "Online multi-object tracking with historical appearance matching and scene adaptive detection filtering," in *Proc. 15th IEEE Int. Conf. Adv. Video Signal Based Surveill. (AVSS)*, Nov. 2018, pp. 1–6.
[40] Y.-C. Yoon, D. Y. Kim, K. Yoon, Y.-M. Song, and M. Jeon, "Online multiple pedestrian tracking using deep temporal appearance matching association," 2019, *arXiv:1907.00831*. [Online]. Available: http://arxiv.org/abs/1907.00831
[41] S.-I. Yu, D. Meng, W. Zuo, and A. Hauptmann, "The solution path algorithm for identity-aware multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3871–3879.
[42] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2008, pp. 1–8.
[43] P. Zhang, L. Zheng, Y. Jiang, L. Mao, Z. Li, and B. Sheng, "Tracking soccer players using spatio-temporal context learning under multiple views," *Multimedia Tools Appl.*, vol. 77, no. 15, pp. 18935–18955, Aug. 2018.
[44] Y. Zhang, L. Guo, and L. Q. Gao, "New method for object tracking based on similarity measure and hypothesis testing," *Control Decis.*, vol. 26, no. 12, pp. 1900–1903, and 1908, 2011.

Hao Sheng (Member, IEEE) received the B.S. and Ph.D. degrees from the School of Computer Science and Engineering, Beihang University, China, in 2003 and 2009, respectively. He is currently an Associate Professor at the School of Computer Science and Engineering, Beihang University. He is working on computer vision, pattern recognition, and machine learning.

Yang Zhang received the B.S. degree from the School of Computer Science and Engineering, Beihang University, China, in 2014, where he is currently pursuing the Ph.D. degree. His research interest is computer vision, and he is particularly interested in multiple object tracking.

Yubin Wu received the B.S. degree from the School of Computer Science and Engineering, Beihang University, China, in 2016, where he is currently pursuing the Ph.D. degree. His research interest is computer vision, and he is particularly interested in multiple object tracking.

Shuai Wang received the B.S. degree from the School of Computer Science and Engineering, Beihang University, China, in 2019, where he is currently pursuing the Ph.D. degree. His research interest is computer vision, and he is particularly interested in multiple object tracking.

Weifeng Lyu is currently a Professor and the Dean of the School of Computer Science and Engineering and the Vice Director of the State Key Laboratory of Software Development Environment. He is also the Leader of the special expert group of "Key Technology and Demonstration of Internet of Things and Smart City" of Ministry of Science and Technology, China. His research interests include intelligent transportation and data analysis.

Wei Ke received the Ph.D. degree from the School of Computer Science and Engineering, Beihang University. He is currently an Associate Professor of the Computing Program at the Macao Polytechnic Institute. His research interests include programming languages, image processing, computer graphics, and tool support for object-oriented and component-based engineering and systems. His recent research focuses on the design and implementation of open platforms for applications of computer graphics and pattern recognition, including programming tools and environments.

Zhang Xiong (Member, IEEE) received the B.S. degree from Harbin Engineering University in 1982, and the M.S. degree from Beihang University in 1985. He is currently a Professor and a Ph.D. Supervisor at the School of Computer Science and Engineering, Beihang University, China. He is working on computer vision, information security, and data vitalization.