Contextual Mixture Tracking

Peng Cui, Li-Feng Sun, *Member, IEEE*, Fei Wang, and Shi-Qiang Yang, *Senior Member, IEEE*

***Abstract*—Multiple Object Tracking (MOT) poses three challenges to conventional well-studied Single Object Tracking (SOT) algorithms: 1) Multiple targets lead the configuration space to be exponential to the number of targets; 2) Multiple motion conditions due to multiple targets' entering, exiting and intersection make the prediction process degrade in precision; 3) Visual ambiguities among nearby targets make the trackers error prone. In this paper, we address the MOT problem by embedding contextual proposal distributions and contextual observation models into a mixture tracker which is implemented in a Particle Filter framework. The proposal distributions are adaptively selected by motion conditions of targets which are determined by context information, and the multiple features are combined according to their discriminative power between ambiguity prone objects. The induction of contextual proposal distribution and observation model can help to surmount the incapability of conventional mixture tracker in handling object occlusions, meanwhile retain its merits of flexibility and high efficiency. The final experiments show significant improvement in variable number objects tracking scenarios compared with other methods.**

***Index Terms*—Contextual observation model, contextual proposal distribution, motion tracking, particle filter.**

I. Introduction

VISUAL tracking, as the primary part of intelligent visual surveillance, can be categorized into two classes, Single Object Tracking (SOT) and Multiple Object Tracking (MOT), according to the number of targets. In the last decades of research, SOT has been well studied. However, the extension of the targets number from Single to Multiple brings the old system three new challenges.

1) **Multiple Targets**. The configuration space of targets temporal correspondences is exponential to the number of targets. Searching the optimal configuration in such high dimensional space is challenging.
2) **Multiple Motion Conditions**. In MOT, the number of targets may vary, and targets may interlace with each other. Thus, each target probably undergo multiple motion conditions, such as entering, exiting, splitting, merging, and updating, which makes the motion prediction process more complex.
3) **Visual Ambiguity**. In SOT, the visual ambiguity between target and cluttered background is a main cause of tracking errors. In MOT, however, this problem becomes more serious, because the targets often have similar appearances. When these targets are nearby or even interlaced, the tracker is prone to erroneously swap their identities.

To address these problems, many methods have been proposed. Most of them can be categorized into two classes. The first category is termed as *joint trackers* [1], [4], [18], which straightforwardly extends the motion state space by concatenating the states of all targets (so all targets' motions are dependent of each other). This state space will expand exponentially with the number of targets. To search the optimal configuration in such space is computationally too expensive. The second category is termed as *mixture trackers* [23], [6], [2], which assumes that the motions of targets are independent with each other, and thus by marginalization the joint state tracker will become to individually evolved local trackers. Therefore, the state space only increases linearly with the number of targets, which makes the trackers much more efficient. However, the assumption of independency among targets in mixture trackers is too strong in practice, which degrades their performances in occlusion handling, local tracker adding/deleting, and ambiguous targets discriminating [17].

The joint trackers and mixture trackers are two extreme instantiations of the tradeoff between dependency and efficiency. In this paper, we address such tradeoff by introducing contexts into mixture trackers, which is named as *contextual mixture tracker* (CMT). In CMT, the evolvement of each local tracker only depends on its context. The context is constituted by only a fraction of targets and some scene priors.

More specifically, the CMT is implemented in *Sequential Monte Carlo* framework (also known as particle filter), which consists of two steps: 1) predict the next motion state based on the current state according to the proposal distribution; and 2) verify the predictions based on new observed data using the observation model. We embed the contexts into these two steps as follows to form the *contextual proposal distribution* (CPD) and *contextual observation model* (COM):

1) Contextual proposal distribution: CPD is a cascaded sampling function, where the particles for motion conditions are first sampled according to its current state and context, and then the particles for motion states are sampled by a motion condition specific proposal distribution. In this step, only spatial contexts are used, including the distances between the local trackers/scene boundaries, and the states of nearby local trackers. These

Manuscript received April 16, 2008; revised October 15, 2008. First published January 16, 2009. This work was supported by National Natural Science Foundation of China (No. 60573167); the National High-Tech Research and Development Plan of China (No. 2006AA01Z118); and the National Basic Research Program of China (No. 2006CB303103). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jiebo Luo.

P. Cui, L.-F. Sun, and S.-Q. Yang are with the Department of Computer Science, Tsinghua University, Beijing 100084, China (e-mail: cuip05@mails.tsinghua.edu.cn; sunlf@mail.tsinghua.edu.cn; yangshq@mail.tsinghua.edu.cn).

F. Wang is with the Department of Automation, Tsinghua University, Beijing 100084, China.

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

Digital Object Identifier 10.1109/TMM.2008.2009722

contexts can help to explicitly predict the motion conditions, and further improve the motion prediction precision.

2) Contextual observation model: The objective of the observation model is to evaluate the prediction likelihoods given new observation data. It is especially challenging when there are visually ambiguous targets nearby. In order to differentiate the local target from them, we propose to use spatial-temporal contexts to adaptively construct the observation models. First, the spatial context is used to find nearby targets. Then the historical appearances of these targets and the local target (temporal contexts) are used to evaluate the discriminative power of the features by an mutual information based feature ranking likelihood method. More discriminative features are weighted higher in the observation model and vice versa. This guarantees that COM is continuously adapted to different contexts and can effectively discriminate the local target from variable surrounding targets.

The rest of this paper is organized as follows. In the next section, we will briefly review the related works. The framework for mixture tracking is specified in Section III. Sections IV and V are respectively devoted to the two elements of CMT: the CPD and COM, followed by the experiments and conclusion sections.

II. Related Work

Several approaches have been proposed to find an efficient and robust solution to multiple object tracking, and *Bayesian Recursive Estimation* is well accepted as the framework [12], [20], [9]. In order to make it tractable while maintaining multiple hypotheses, *Particle Filter* (PF) is exploited to approximate the posterior distribution by sampling [11], [26]. In order to simultaneously track multiple targets, [19], [25] extend the standard PF into Joint PF (JPF) where state vector contains the motion information of all targets. When tracking variable targets, the dimension of the state vector has to change accordingly. To maintain a consistent particle space, Smith *et al.* [19] proposed a reversible-jump MCMC PF based on [8]. These methods can indeed naturally model the interactions among targets by maximizing the joint state posterior distribution. However, the high dimensionality of state space requires too much expensive computations.

Another category of methods tend to factorize the JPF into several independent local trackers. Vermaak *et al.* proposed a mixture tracker (MT) [21], where the multimodal distribution caused by multiple targets is approximated by multiple mixture components. Each component corresponds to a local tracker and evolves independently. Okuma *et al.* [15] extend the standard MT into a boosted MT by combining it with an Adaboost object detector, which can detect and track the new entered and exited objects. However, the detector is trained offline. This could be a problem in MOT when there is no prior knowledge on the classes of objects. In these methods, the independency assumption brings efficiency, while making the local trackers know nothing about the global states, which limits the ability of MT to deal with occlusions.

Therefore, the exploration of sufficient but not redundant dependency among targets becomes the central issue. In this paper, we term this kind of dependency as context. Although not explicitly stated, several methods have adopted context information to handle visual ambiguity and occlusions. One category of them exploit the spatial distance as the measure to find the spatial context of a local tracker, and the evolvement of the local tracker is controlled by the observation of both the local tracker and its context [24], [17], [13]. In [25], a collaborative tracking method is proposed to allow for the collaboration among adjacent local trackers (spatial context) by modeling these targets' joint prior using a Markov Random Network. In [Magnetic-Inertia], Gravitation model is introduced to model the interactive power between nearby local trackers. Nguyen *et al.* [13] regard MOT as a classification problem among nearby targets so that the decision on the label of a target is influenced by other targets. The spatial context can be effectively modeled and embedded into the tracking framework by partial joint estimation in these methods. However, they do not further exploit the context information to improve the local trackers' ability of discriminating the local targets from nearby similar targets. Thus the visually ambiguous targets are still probable to be erroneously tracked when they are nearby or occlusion presents. The other category use the historical appearances of the local targets as its temporal context and adaptively construct and update its observation model in hope of an accurate appearance model robust to occlusions and appearance changes [26], [14]. The generative models, including eigenspace [14] and mixture of Gaussians [26] are most commonly applied. The dynamic appearances of targets can be well incorporated by these models which can be straightforwardly embedded into the tracking framework. However, these generative models cannot guarantee sufficient discriminative power if the visual ambiguous targets move nearby or even occlude.

Compared with previous methods, the approach proposed in this paper has the advantage of 1) embedding spatial-temporal context information into a unified probabilistic mixture tracking framework, which is theoretically solid; 2) explicitly predicting the motion condition of local trackers based on spatial contexts, which significantly improve the precision of motion prediction; and 3) exploiting spatial-temporal contexts to adaptively construct discriminative (rather than generative) observation models, which greatly alleviates the problem of visual ambiguity.

III. Mixture Tracker

We denote the state sequence as $x_{1:t}$, and the observations sequence as $z_{1:t}; t \in \mathbb{N}^*$. To deal with multiple objects within a tractable configuration space, we adopt the mixture model proposed in [21] to factorize the configuration space according to the number of targets:

$$p(x_t \mid z_{1:t}) = \sum_{m=1}^{M} \pi_{m,t} p_m(x_t \mid z_{1:t}) \tag{1}$$

where M is the total number of objects, and the mixture weights satisfy $\sum_{m=1}^{M} \pi_{m,t} = 1$.

In Sequential Monte Carlo framework (also known as Particle Filter), the filtering distribution $p(x_t \mid z_{1:t})$ are approximated by a set of weighted particles $\{x_t^i, \omega_t^i\}$:

$$p(x_t \mid z_{1:t}) = \sum_{m=1}^{M} \pi_{m,t} \sum_{i \in \mathbb{C}_m} \omega_t^i \delta_{x_t^i}(x_t). \tag{2}$$

where $\delta()$ denotes the delta-Dirac mass, c_i indicates the component x_t^i attaches to, and $\mathbb{C}_m$ denotes the set of particles attached to component m.

Given the particles $\bigcup_{i \in \mathbb{C}_m} \{x_{t-1}^i, \omega_{t-1}^i\} \sim p_m(x_{t-1} \mid z_{1:t-1})$ at time instant $t-1$, the particles for time instant t is drawn from $\{\tilde{x}_t^i\} \sim q_{c_i}(x_t \mid x_{t-1}, z_{1:t})$, and weighted by

$$\omega_t^i = \omega_{t-1}^i \frac{p_{c_i}(z_t \mid \tilde{x}_i^t)\, p_{c_i}\left(\tilde{x}_t^i \mid x_{t-1}^i\right)}{q_{c_i}\left(\tilde{x}_t^i \mid x_{t-1}, z_t\right)}. \tag{3}$$

Then these samples are resampled to form a set of unweighted samples to approximate the filtering distribution.

Finally, the components' mixture weights are updated as

$$\pi_{m,t} \approx \frac{\pi_{m,t-1}\omega_{m,t}}{\sum_{j=1}^{M} \pi_{j,t-1}\omega_{j,t}}, \quad \omega_{m,t} = \sum_{i \in \mathbb{C}_m} \omega_t^i. \tag{4}$$

In this framework, MOT is realized by a number of local trackers, in which there are two crucial elements: the *proposal distribution* $q_m(x_t \mid x_{t-1}, z_{1:t})$ for proposing new particles (i.e., predicting motion states) and the *observation model* for verifying predicted particles (i.e., evaluate the likelihood $p_m(z_t \mid x_t)$ when new observation is obtained). Unlike [21] and [15], the proposal distributions and observation models for different local trackers are not the same. They are influenced by different context information, which will be respectively specified in Sections IV and V. As the following sections only relate to local trackers, we ignore the subscript for local tracker index without confusion.

IV. Contextual Proposal Distribution

Proposal distribution (PD) $q(x_t \mid x_{t-1}, z_t)$ is the central issue in PF framework to predict new particles based upon the current states and the recent observations. In practice, local trackers often share the same transition prior distribution $p(x_t \mid x_{t-1})$ as a trial [21]. Such a general PD would lead to a majority of uninformative particles which consume most computation power while making no contribution. In this section, we extend the proposal distribution in two important aspects: 1) the recent observation is incorporated into the proposal distribution through motion detection; 2) the proposal distribution is transformed into a mixture of motion condition specific proposal distributions.

A. Observation Included PD

Assuming a fixed camera, we firstly establish a pixel-wise background model based on Gaussian Mixture Model. Then the foreground blobs are detected, whose envelop information(i.e., the envelop position and size)are used as the recent observations $\tilde{z}_t$. The proposal state space of $q(x_t \mid x_{t-1}, \tilde{z}_t)$ is then reduced from a Gaussian space into several candidate regions, which significantly saves the computational burden without lessening the

Fig. 1. Observation incorporated particle propose. The left image is the previous frame; the middle one is the foreground mask of current frame; and the right one is current frame. The green and yellow dots are the proposed position samples respectively by the person bounded with the same color.

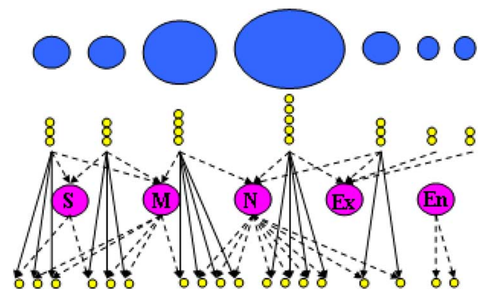

Fig. 2. Motion condition specific particle propose. The four top-down layers correspond respectively to weighted particles at time $t-1$, resampled particles, sampled motion conditions, and proposed particles. The five nodes in motion condition layer represent Split, Merge, Normal update, Exit, and Enter.

cover of uncertainty. In order to tolerate misshapen or discontinuous foreground regions caused by noises, we keep the uncertainties by Gaussian sampling within nearby regions, as shown in Fig. 1. Note that the $\tilde{z}_t$ in proposal distribution is different from z_t in observation model. $\tilde{z}_t$ only considers the envelop information, which excludes the appearance information in the foreground regions in z_t.

B. Motion Condition Specific PD

In MOT, each object may undergo different motion conditions, e.g., entering the scene, occluded by another object, reappearing from the occlusion, updating the appearance and position, and exiting from the scene. In order to propose particles more precisely, we propose a motion condition specific PD, in which the incorporated motion condition set is $\mathbb{H} = \{\mathrm{enter}, \mathrm{split}, \mathrm{merge}, \mathrm{normal}, \mathrm{exit}\}$.

The motion condition specific PD for each component is defined as

$$q(x_t \mid x_{t-1}, \tilde{z}_t) = \sum_{h_t \in \mathbb{H}} q(h_t \mid x_{t-1}, \tilde{z}_t) q_{h_t}(x_t \mid x_{t-1}, \tilde{z}_t). \tag{5}$$

In this PD, the particle proposal process is implemented by firstly sampling the motion condition from $q(h_t \mid x_{t-1}, \tilde{z}_t)$; then, the new particles are drawn from the motion condition specific PD $q_{h_t}(x_t \mid x_{t-1}, \tilde{z}_t)$. Fig. 2 depicts the proposal process.

1) Motion Conditions Sampling: In order to determine the sampling probability of motion conditions for each component, we factor the sampling function into two parts:

$$\begin{aligned} q(h_t \mid x_{t-1}, \tilde{z}_t) &= \frac{p(h_t \mid x_{t-1}) p(\tilde{z}_t \mid x_{t-1}, h_t)}{p(\tilde{z}_t \mid x_{t-1})} \\ &\propto p(h_t \mid x_{t-1}) p(\tilde{z}_t \mid x_{t-1}, h_t) \end{aligned} \tag{6}$$

where $p(h_t \mid x_{t-1})$ is the prediction function, and $p(\tilde{z}_t \mid x_{t-1}, h_t)$ is the likelihood function. As the likelihood function cannot be explicitly calculated, here we use it as a bonus/penalty factor instead.

The prediction function is defined by two kinds of spatial context information: object-object distances (D_O) and object-boundary distances (D_B), which are derived from the states of all components. We denote N_t as the number of objects in the scene at t, and Ne as the number of entrance-exit zone. Then $D_{O,t}$ is an $N_t * N_t$ matrix with the elements $d_{O,t}(i,j)$ representing the distance between object i and object j; $D_{B,t}$ is an $N_t * Ne$ matrix, with elements $d_{B,t}(p,q)$ representing the distance between object p and entrance-exit zone q. Using the two matrices, the motion condition sampling function is defined in the following way.

Normal: In low or moderate complexity tracking, objects undergo Normal condition most often. At each time instant, we first determine the sampling probability of Normal condition:

$$p_m(h_t = \text{Norm} \mid x_{t-1}) = \min\left\{\frac{\min(d_{\cdot,t-1}(m,.))}{d_{\text{thr}}}, 1-\varepsilon\right\} \quad (7)$$

where d_{thr} is the threshold distance, above which the probability of other conditions is near ε.

After the determination of Normal's probability, the other conditions share the rest probability accordingly:

$$\begin{aligned} p_m(h_t = l \mid x_{t-1}) &= (1 - p_m(h_t = \text{Norm} \mid x_{t-1})) \\ &\times \frac{\tilde{p}_m(h_t = l \mid x_{t-1})}{\sum_l \tilde{p}_m(h_t = l \mid x_{t-1})}, l \in \mathbb{H} - \{\text{Norm}\}. \quad (8) \end{aligned}$$

Enter&Exit: The probability of Exit decreases monotonically with respect to the object-boundary distance:

$$\tilde{p}_m(h_t = \text{Ext} \mid x_{t-1}) = \exp(-\min(d_{B,t-1}(m,.))). \quad (9)$$

If there is a blob near x_{t-1} at t which is indicated by $\tilde{z}_t$, then the probability will be penalized by $p(\tilde{z}_t \mid h_t, x_{t-1})$.

In Enter cases, $x_{t-1} \in \phi, p(h_t \mid x_{t-1})$ is ineffective. To remedy this, we use $p(\tilde{z}_t \mid h_t, x_{t-1})$ as bonus if there is a blob near the boundary of the scene.

Merge&Split: The probabilities of Merge and Split monotonically decrease with respect to the object-object distances:

$$\begin{aligned} \tilde{p}_m(h_t = \text{Spl} \mid x_{t-1}) &= \tilde{p}_m(h_t = \text{Mer} \mid x_{t-1}) \\ &= \exp(-\min(d_{O,t-1}(m,.))). \quad (10) \end{aligned}$$

2) Motion States Sampling: Given the sampled motion condition, the particles are sampled as follows:

Normal: A component undergoes Normal condition when the other special cases are excluded. In this case, by considering the recent observation $\tilde{z}_t$ derived from motion detection, the particles are sampled according to the method in Section III-A.

Enter: If the motion condition is sampled as Enter at t, which means $x_{t-1} \in \phi$, then the new particles are sampled from a Gaussian model, whose parameters are estimated from $\tilde{z}_t$. If the Enter condition is verified in the Update process, a new component is added and the new particles are retained to track the new object. However, in some cases, the incorporation of Enter results in ambiguities between new entered objects and objects under other conditions are near the entrance/exit zone. To remedy this, we propose a dispelling rule in which the new particles are invalidated when other motion condition hypothesis is verified in the Update process.

Exit: If the motion condition is sampled as Exit at t, which means $x_t \in \phi$, the particles supporting this hypothesis are invalidated. By including the Enter and Exit conditions, the approach is capable of tracking variable number objects.

Merge: When one object is predicted to be merged with another object, the two objects merge into one blob, which make the particle propose process ambiguous. In this case, we use an AR2 process to predict the object's state, and let their appearances unupdated.

Split: When one object splits up from the merged blob, the recent observation $\tilde{z}_t$ is available and effective, so that the particle propose process is similar with Normal condition.

Until now, local trackers can propose particles according to their contexts. With the aid of these contexts, more particles (i.e., computation power) are assigned to more probable motion states, which greatly improves the predicting precision. The priority of CPD is demonstrated and analyzed in Section VII.

V. Contextual Observation Model

Given the predicted states (represented by newly proposed particles), observation models for targets are needed to validate these predictions. When nearby targets have similar appearances, sufficient discriminative power of the local trackers are desired. The main challenge of observation model construction lies in its adaptability to different contexts. For example, a model of object A that can easily discriminate A from object B may have problems to discriminate A from object C, or it succeeds under sunlight but may fail in shadow. In this section, we propose to adaptively adjust the observation models by online feature re-ranking according to different contexts, so that they can discriminate the local targets from nearby targets even when they have similar appearances.

A. Representation

We use a feature pool $\mathbb{F}$ for observation model construction, in which each feature f_k constitutes an observer whose output is $p(z_t^k \mid x_t)$. We define Field of Context (FOC) by D_O, where i is in FOC if $\exists j,\ s.t.\ d_{O,t}(i,j) < d_{\text{thr}}$. As the observers have different discriminability in different FOCs, we assign to each observer a weight $\rho_{k,t}$ which is higher if the feature can better distinguish the object m from nearby objects. Using above notations, the observation model $p(z_t \mid x_t)$ is transformed into a mixture model:

$$p(z_t \mid x_t) = \sum_{k=1}^{|\mathbb{F}|} \rho_{k,t} p\left(z_t^k \mid x_t\right) \quad (11)$$

where $|\cdot|$ indicates the cardinality of the set. Each component $p(z_t^k \mid x_t)$ can be simply modeled as a Gaussian model. In the following sections, we will investigate the approach for calculating feature weights.

Fig. 3. Feature discriminability evaluation. The left is the illustrating frame; the middle is the samples of A,B,C, and D; these samples are projected into the two dimensional feature space, with the horizontal axis representing the intensity, and the vertical axis representing the width of bounding box.

B. Feature Discriminability Evaluation

According to our experiences, the degree to which the objects can be discriminated depends greatly upon the adopted feature set. As shown in Fig. 3, the feature that can discriminate A and B may not work for C and D. In our method, we assign variable weights to observers according to its discriminability in FOC.

In [3] and [22], the online feature selection method for SOT has been proposed, where the features are used to discriminate the target and background. However, the problem in MOT is different from [3], [22] in the following three aspects.

1) Our problem can be treated as a multiclass classification problem, instead of a two-class classification as [3], [22]. So the evaluation criterion must be able to deal with multiclass cases.
2) The number of classes is variable, not fixed as in [3], [22], which poses higher requirements on the scalability of the evaluation criterion.
3) The importance of training samples are not equal. The more recent samples have higher confidence levels in evaluating the features' discriminabilities. For example, after the objects enter into the shadow, the appearance samples under sunlight are not that convincible.

Based on those considerations, we use an information theoretic method to adaptively evaluate features' discriminability online. Given the feature f_i and the class label $y \in \mathbb{Y}$, the dependency of f_i and y can be measured by mutual information [5]:

$$\begin{aligned}\mathcal{I}(f_i, y) &= \mathcal{H}(y) - \mathcal{H}(y \mid f_i) = \mathcal{H}(y) \\ &\quad - \int_{f_i} p(f_i)\mathcal{H}(y \mid f_i) df_i \\ &= \int_y p(y)[-\log p(y)]dy \\ &\quad - \int_{f_i} p(f_i) \int_y p(y \mid f_i)[-\log p(y \mid f_i)]dydf_i \\ &= \int_{f_i}\int_y p(f_i, y) \log \frac{p(f_i, y)}{p(f_i)p(y)} df_i dy. \end{aligned} \tag{12}$$

As the probability densities are unknown, we estimate the mutual information using empirical discrete probability densities:

$$\hat{\mathcal{I}}(F_i, Y) = \sum_{f_i}\sum_{y} P(F_i = f_i, Y = y) \times \log \frac{P(F_i = f_i, Y = y)}{P(F_i = f_i)P(Y = y)} \tag{13}$$

where $P(\cdot)$ is obtained by frequency counting.

A higher mutual information indicates a stronger correlation between the feature values and class labels, which means that the feature is more discriminative with respect to these object classes. This method is efficient for online computation and easy to implement. It satisfies all the three requirements mentioned above.

Multiclass: The intrinsic structure of this method determines its nonrestraint on the number of classes. It is straightforward to apply this method in any number of classes.

Scalability: This method has excellent scalability due to the approximation by histograms. When a new class is added because of a new object entering the scope, we only need to add a histogram bucket on the class dimensions of all histograms and begin to accumulate samples on the new bucket, with all other buckets unchanged.

Weighted Samples: In the histogram representation, the samples' weights can be conveniently incorporated by changing the sample accumulation strategy, which is detailed in the next section.

We use the mutual information to calculate each feature's score, and the scores after normalization are taken as the observers' weights:

$$\rho_{k,t} = \frac{\hat{\mathcal{I}}(F_k, y)}{\sum_{k=1}^{|\mathbb{F}|}\hat{\mathcal{I}}(F_k, y)}. \tag{14}$$

C. Histogram Approximation

We specify in this section the approach of approximating the features' discriminabilities based on histogram estimation. Assuming there are U objects in the FOC, which generate U classes with u as the indices, we denote $\mathbb{S}_t$ as the appearance sample set at time t with its elements $s_{t'}^{(u)}$ representing the sample generated at time t' for class u. The 2-D histogram for feature k derived from those samples is G_k with buckets $G_k(v, u)$, where u is the class index, and v is the feature value or value interval index. Given a sample $s_{t'}^{(u)}$, the value of feature k is denoted by $f_k(s_{t'}^{(u)})$.

In order to make the model adapt to appearance changes caused by motion and environmental factors, the samples' weights are updated by a decay factor so that the recent samples are assigned with comparatively higher weights:

$$\eta_t\left(s_{t'}^{(u)}\right) = \eta_{t-1}\left(s_{t'}^{(u)}\right)e^{-\alpha}\delta(t > t') + \delta(t = t') \quad (15)$$

where α controls the decay speed.

The histogram for each feature is updated accordingly:

$$\begin{aligned} G_k^t(v,u) &= \sum_{t' \le t} \eta_t\left(s_{t'}^{(u)}\right)\delta\left(f_k\left(s_{t'}^{(u)}\right) = v\right) \\ &= \sum_{t' < t} \eta_{t-1}\left(s_{t'}^{(u)}\right)\delta\left(f_k\left(s_{t'}^{(u)}\right) = v\right)e^{-\alpha} \\ &\quad + \delta\left(f_k\left(s_t^{(u)}\right) = v\right) \\ &= G_k^{t-1}(v,u)e^{-\alpha} + \delta\left(f_k\left(s_t^{(u)}\right) = v\right) \end{aligned} \quad (16)$$

where $\delta()$ is the delta-Dirac mass.

From feature histogram to feature discriminability is a process of frequency counting. In order to better approximate the feature value distribution and further achieve a more precise discriminability, we use a pyramid kernel for discriminability estimation.

We map the feature values onto multiresolution histograms $G_{k,2^r}$, where r represents the quantization level with $r = 0$ corresponding to the finest level. We denote the mutual information estimated by $G_{k,2^r}$ as $\hat{\mathcal{I}}_r(F_k, Y)$. Then the mutual information estimated from multiresolution histograms is:

$$\hat{\mathcal{I}}(F_k, Y) = \sum_r \frac{1}{2^r}\hat{\mathcal{I}}_r(F_k, Y). \quad (17)$$

and the weight of the feature is calculated by (14).

VI. Implementation

In this section, we specify the flow chart of the proposed algorithm. We use $x_t = \{P_x, P_y, B_x, B_y\}$ as the state, where P_x and P_y represent the position of an object, and B_x and B_y represent the width and height of the minimum bounding box. Thus, the state of an object actually specifies a rectangle image region that the object reside in. The feature pool for observation model includes color features and shape features. We firstly divide rectangle image regions into top and bottom regions, and respectively calculate the normalized color histograms for the two regions. We use each bucket in the histogram as a color feature. If denoting the quantization step by λ, and adopting the color space of RGB, then we can compute the number of color features as $2 \times 3 \times (255)/(\lambda)$. As to the shape feature, we use the width-height ratio of the bounding box to describe the shape information. These features' discriminabilities are online evaluated, and those scored lower than the threshold are filtered. Note that more features, like texture, can be straightforwardly incorporated into the observation model, although we find the color and shape features are sufficient in our experiments.

The algorithm is initialized by background subtraction, where each blob is assigned with a local tracker with its parameters defined by the centroid and bounding box of the blob. In each local tracker, 40 particles are used for approximation. The context information can also be derived from the blobs. Given the initialized particles and context information, the algorithm estimates the targets' motions in a recursive way which is specified in Algorithm 1.

Algorithm 1 The algorithm flowchart of CMT

Given the particle set of CMT at $t - 1\{x_{t-1}^i, \omega_{t-1}^i, c_i, \pi_{c_i,t-1}\}_{i=1}^N$, weights of features $\{\rho_{j,t-1}^m\}_{j,m}$ (j and m are the indices of features and local targets), and the context informations $\{D_{O,t-1}^m, D_{B,t-1}^m, \mathbb{S}_{t-1}^m\}_m$ the algorithm proceeds as follows at time t:

■ **Local Tracking**: For each local tracker m, its feature weights are $\{\rho_{j,t-1}\}_j$, the context is $\{D_{O,t}, D_{B,t}, \mathbb{S}_{t-1}\}$, and the assigned particle set is represented as $\{x_{t-1}^i, \omega_{t-1}^i\}_{i\in\mathbb{C}_m}$ whose size is N_m. These particles evolve as follows:

(1) **Motion Prediction**:

- Background subtraction on frame t to derive $\tilde{z}_t$
- Sample motion conditions $h_t \sim q(h_t \,|\, x_{t-1}^i, \tilde{z}_t)$
- Given the sampled motion conditions, particles are sampled $x_t^i \sim q_{h_t}(x_t \,|\, x_{t-1}^i, \tilde{(z)}_t)$.

(2) **Verification**: For $i = 1 \dots N_m$,

- For $j = 1 \dots |\mathbb{F}|$, calculate $p(z_t^j \,|\, x_t^i)$;
- Proposed particles are weighted by
$$\omega_t^i = p\left(z_t \,|\, x_t^i\right) = \sum_{j=1}^{|\mathbb{F}|} \rho_{j,t-1} p\left(z_t^j \,|\, x_t^i\right).$$

(3) **Local Decision**: Cluster the particles $\{x_t^i, \omega_t^i\}$, and let C be the cluster with maximum weight. Then x_t is decided to be
$$\hat{x}_t = \frac{\sum_{i\in C}\omega_t^i x_t^i}{\sum_{i\in C}\omega_t^i};$$

■ **Global Decision**:

(1) **Component Weight Update** Update the component weight from $\pi_{m,t-1}$ to $\pi_{m,t}$ by (4).

(2) **Local Decision Verification** If $\pi_{m,t}$ is lower than a predefined threshold, then the local decision of local tracker m at t is canceled and the local tracker is deleted.

■ **Context Update**: Update the context to $\{D_{O,t}^m, D_{B,t}^m, \mathbb{S}_t^m\}_m$ by the decisioned local tracker states.

■ **Feature Reevaluation**: For each local tracker m, for each feature $j = 1 \dots |\mathbb{F}|$,

(1) **Histogram Construction** Construct the histogram G_j^t using $\mathbb{S}_{m,t}$.

(2) Discriminability Evaluation} Evaluate the discriminability of F_j: $\hat{\mathcal{I}}(F_j)$.

(3) **Feature Weighting** Calculate the normalized feature weights $\rho_{j,t-1}$.

TABLE I
CHARACTERISTICS OF EXPERIMENT DATA SETS. RES: RESOLUTION; FN: TOTAL NUMBER OF FRAMES; VON: WHETHER THE NUMBER OF OBJECT IS VARIABLE; Occ: WHETHER OCCLUSIONS HAPPEN; MOC: WHETHER MULTIPLE OBJECT CATEGORIES EXIST

	Res	FN	VON	Occ	MOC
$PETS$	352*288	5530	$\surd$	$\surd$	$\times$
$Vehicles$	352*288	6400	$\surd$	$\surd$	$\times$
$Pedestrians$	352*288	1350	$\surd$	$\surd$	$\times$
$Hybrid$	352*288	975	$\surd$	$\surd$	$\surd$

Fig. 4. Experiment results on PETS.

TABLE II
TRACKING ERROR RATES ON PETS

	OE	DE	SE
MTb	10.2%	11.5%	8.1%
$MTb + CPD$	4.7%	3.8%	4.0%
$MTb + CPD + COM$	3.3%	3.8%	2.6%

Fig. 5. Experiment results on Vehicle.

TABLE III
TRACKING ERROR RATES ON VEHICLES

	OE	DE	SE
MTb	6.6%	12.5%	2.4%
$MTb + CPD$	3.0%	2.9%	1.5%
$MTb + CPD + COM$	1.3%	2.5%	0%

VII. EXPERIMENTS

A. Data Sets

To validate the proposed method, we conducted experiments on four real world data sets. The first data set, denoted by **PETS**, is a subset of the public PETS2004 data set. All these sequences include multiple targets, and the environmental illumination is unstable and non-uniform, which poses more challenges to the tracker. The second data set, denoted by **Vehicles**, is an aerial footage of vehicles driving through the multiple overhead bridges. This video is used to evaluate the proposed method in the abilities of tracking variable number vehicles, and vehicles occluded by overhead bridges. The third data set, denoted by **Pedestrians**, is captured in front of a hall where multiple pedestrians come into the field of view. As a number of occlusions happen among pedestrians, we use this video to evaluate the method's ability of occlusion handling. The fourth data set, denoted by **Hybrid**, is an aerial footage of a crossing road, where multiple kinds of moving objects, like pedestrians, cars, and bicyclers, enter and exit from the field of view. As our method place no constraint on the tracked object's category, we use this video to demonstrate the capability of our method in tracking multiple categories of targets. The detailed information about the three data sets is listed in Table I. Note that the occlusions occurring in Vehicles and Pedestrians are different. In Vehicles, the targets are often occluded by background objects (background occlusions), while in Pedestrians the targets sometimes occlude each other (inter-objects occlusions).

B. Evaluation Measures

In order to fairly evaluate the proposed method, we use the following three criteria for performance evaluation:

1) Overall Error (OE): This criterion is used to evaluate the overall performance. It is defined as the ratio of the number of false tracked frames to total frame number.
2) Delay Error (DE): This criterion is used to evaluate the variable number targets tracking, which is defined as the ratio of the number of untracked entered targets tracked exited targets to total exited and entered objects.
3) Swap Error (SE): This criterion is used to evaluate the discriminability of the proposed method to tackle visual ambiguity. It is defined as the ratio of the number of false identification swap frames to the total frame number.

C. Results

In order to demonstrate the benefits derived from context information, we implement the standard mixture tracker (denoted as MTb) proposed in [21] for a baseline, where no context information is incorporated, and the features in observation models always have equal weights. Then we add the module of CPD to take into account the spatial context information, which is denoted as $\mathrm{MTb} + \mathrm{CPD}$. Finally, the COM is added, and the resulted tracker is the proposed CMT. We evaluate the three algorithms on four data sets.

1) PETS: The error rates of the three algorithms on PETS are listed in Table II, and some typical results of CMT are shown in Fig. 4. There are mainly two causes of errors. One is that the targets in these sequences are often nearby and have similar appearances, which leads to some swap errors. The other is that the illumination is nonuniform, which make the appearances of targets photometrically vary as they moves. Therefore, the tracker sometimes lose one of its tracks when the target move through the bright region. These errors are especially serious in MTb. With the assist of CPD, the second type of errors are reduced because the spatial context forbid the local tracker to be deleted if it is not near an exit/enter zone. At the same time, as the temporal context in COM lay more emphasis on recent appearances, the observation model can quickly incorporate the photometric changes. Thus, in Fig. II, the target in green window can be correctly tracked. In addition, the online feature evaluation mechanism automatically broaden the differences of nearby targets, so the swap error problem is alleviated.

2) Vehicles: The error rates of the three algorithms on Vehicles are listed in Table III, and some typical results of CMT are shown in Fig. 5.

In the Vehicle video, the background is constituted by complex multiple overhead bridges, and numerous vehicles enter

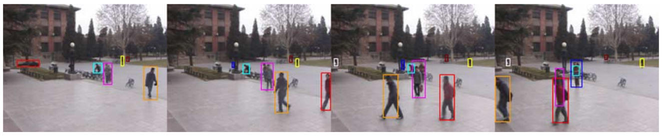

Fig. 6. Experiment results on Pedestrians.

TABLE IV
TRACKING ERROR RATES ON PEDESTRIANS

	OE	DE	SE
MTb	7.3%	8.3%	6.4%
$MTb+CPD$	6.1%	2.8%	5.8%
$MTb+CPD+COM$	2.1%	2.8%	1.8%

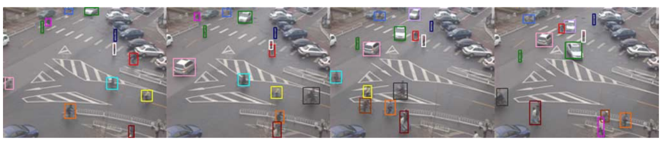

Fig. 7. Experiment results on Hybrid.

TABLE V
TRACKING ERROR RATES ON HYBRID

	OE	DE	SE
MTb	5.7%	10.0%	5.2%
$MTb+CPD$	4.7%	5.0%	4.2%
$MTb+CPD+COM$	1.8%	5.0%	1.3%

and exit from the filed of view at times. As is shown in the top row of Fig. 5, our method can detect the new entering objects in time and assign new particles to track the object. Also, as denoted by the yellow arrow, when the car is occluded by the background, the motion condition of the car is recognized as Merge, as it disappears not in an exit zone. In this case, the appearance of the car is recorded, and its motion is predicted by AR2 model. When the car reappear, the system can still re-track it. In MTb, however, all these cases are erroneously regarded as exit. The majority of the remained error in CMT is caused by background occlusion, especially when multiple similar vehicles are occluded by the background which is also difficult for people to retrack when they reappear.

3) Pedestrians: The error rates of the three algorithms on Vehicles are listed in Table IV, and some typical results of CMT are shown in Fig. 6.

In the Pedestrian video, the density of pedestrians is moderate, and there exit a number of complex occlusions. Different from the Vehicle video, these occlusions are mostly inter-objects occlusions. This kind of occlusions is more challenging, especially when the occluder and occludee have similar appearances. The discriminability of the observation model can be well evaluated in these scenarios. Our method demonstrate its excellent ability to handle the inter-object occlusions owing to the contextual observation model, as shown in the Fig. 6. The average tracking error rate of our method in this video is 2.1%. Most of these errors happen when the interlaced objects cannot be well discriminated by any feature in the feature pool.

4) Hybrid: The error rates of the three algorithms on Vehicles are listed in Table V, and some typical results of CMT are shown in Fig. 7.

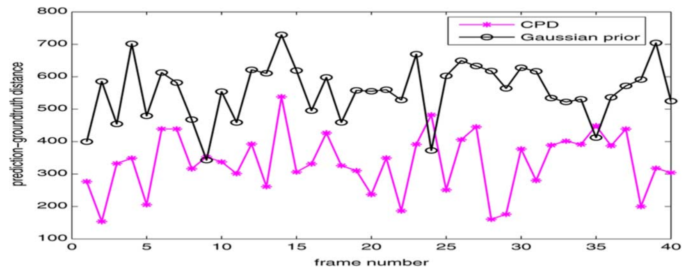

Fig. 8. Comparison on motion prediction precision between CPD and Gaussian prior.

In the Hybrid video, most of the moving objects are pedestrians, bicyclers and vehicles. As we don't place any constraints on the object category in our method, we can track all these objects at the same time, as shown in Fig. 7. The overall error rate of CMT is 1.8%. Most of these errors are caused by occlusion-entering, that is multiple targets together enter the scene with occlusions, so that the tracker can not assign right number of local trackers to the blob. The solution to this problem is a general pedestrian detector, which is very computationally expensive.

In all the four data sets, the proposed CMT show significantly improvements on MTb. Here, we empirically analyze the reasons. On one hand, as mentioned that the priority of CPD is that the context information helps to improve its motion prediction precision. In order to explicitly demonstrate this, we use a subsequence of PETS2004 to evaluate the prediction precision. In PETS2004, the centroids of targets in each frame are manually annotated. We use the position information of each proposed particle (P_x^i, P_y^i) to calculate the distance from the groundtruth position to the predicted position and sum over the distances of all particles to represent the prediction precision of the proposal distribution. We compare CPD with commonly used Gaussian prior [19], [1] on the sequence. The result is shown in Fig. 8. It can be seen that in most cases, the prediction-groundtruth distance of CPD is smaller than that of Gaussian prior. The improvement on motion prediction precision is not trivial, because it saves computation power, meanwhile reduce the tracking errors result from erroneous predictions.

On the other hand, the COM has much more discriminative power than commonly used equally weighted observation models (EWOM) [6], [2]. We demonstrate this using the annotated PETS2004 sequence either. The image patches of two near and similar targets (A and B) are collected. We online train the observation model of one target (A) with the other (B) as its context. Given a frame, we use the likelihood of A's appearance to minus the likelihood of B's appearance, and the likelihood difference is used as the measure of discriminative power. The result is shown in Fig. 9. We can see that in most cases, the COM has large differences than EWOM. At the beginning frames, the two model perform all square. With the samples accumulating, the COM raise up the weights of discriminative features, and the difference between the two visual ambiguous targets is gradually broadened. However, the likelihood differences in EWOM maintain at a low level, and around the frame 8, the difference is minus, which represents that a swap error occurs.

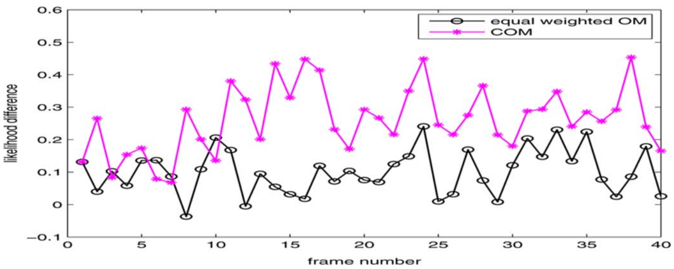

Fig. 9. Comparison on likelihood difference between COM and equally weighted observation model.

VIII. Conclusion

In this paper, we have proposed an contextual mixture tracker for variable number objects tracking. In this tracker, each object is tracked by a local tracker, which greatly factorize the configuration space. At the same time, the local trackers implicitly interact through the contextual proposal distribution and contextual observation model, where spatial and temporal context information plays an important role. We have testified our method in four real-life data sets, and the experimental results show its significant superiority in handling variable number objects, and discriminating nearby or occluded objects compared with previous mixture trackers.

References

[1] D. Angelova and L. Mihaylova, "Extended object tracking using Monte Carlo methods," *IEEE Trans. Signal Process.*, vol. 56, pp. 825–832, 2008.

[2] Y. Z. Cai, N. Freitas, and J. J. Little, "Robust Visual Tracking for Multiple Targets," in *ECCV*, 2006.

[3] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 27, pp. 1631–1643, Nov. 2005.

[4] A. Declercq and J. H. Piater, "On-line simultaneous learning and tracking of visual feature graphs," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2007, pp. 1–6.

[5] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, pp. 1157–1182, 2003.

[6] B. Han, S. W. Joo, and L. S. Davis, "Probabilistic fusion tracking using mixture kernel-based bayesian filtering," in *Int. Conf. Computer Vision*, 2007, pp. 1–8.

[7] A. Jepson, D. Fleet, and T. El Maraghi, "Robust online appearance models for visual tracking," *CVPR*, vol. 1, pp. 415–422, 2001.

[8] Z. Khan, T. Balch, and F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1805–1918, Nov. 2005.

[9] O. Lanz, "Approximate bayesian multibody tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1436–1449, 2006.

[10] Y. Li, H. Ai, T. Yamashita, S. Lao, and M. Kawade, "Tracking in low frame rate video: A cascade particle filter with discriminative observers of different lifespans," in *Proc. CVPR*, 2007.

[11] T. B. Moeslund, A. Hilton, and V. Kruger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understand.*, vol. 104, pp. 90–126, 2006.

[12] M. R. Morelande, C. M. Kreucher, and K. Kastella, "A bayesian approach to multiple target detection and tracking," *IEEE Trans. Signal Process.*, vol. 55, pp. 1589–1604, 2007.

[13] H. Nguyen and A. Smeulders, "Fast occluded object tracking by a robust appearance filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp. 1099–1104, August 2004.

[14] H. T. Nguyen, Q. Ji, and W. M. Smeulders, "Spatio-temporal context for robust multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 52–64, 2007.

[15] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *Proceedings of the ECCV, Volume 3021 of LNCS*. Berlin, Germany: Springer, 2004, pp. 28–39.

[16] E. Polat, M. Yeasin, and R. Sharma, "Robust tracking of human body parts for collaborative human computer interaction," *Computer Vision and Image Understanding*, vol. 89, pp. 44C69–, 2003.

[17] W. Qu, D. Schonfeld, and M. Mohamed, "Real-time distributed multi-object tracking using multiple interactive trackers and a magnetic-inertia potential model," *IEEE Trans. Multimedia*, vol. 9, pp. 511–519, 2007.

[18] P. Smets and B. Ristic, "Kalman filter and joint tracking and classification based on belief functions in the TBM framework," *Inform. Fusion*, vol. 8, pp. 16–27, 2007.

[19] K. Smith, D. Gatica-Perez, and J. Odobez, "Using particles to track varying numbers of interacting people," *CVPR05*, pp. 962–969, 2005.

[20] B. Stenger, A. Thayananthan, H. S. Torr, and R. Cipolla, "Model-based hand tracking using a hierarchical bayesian filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1372–1384, 2006.

[21] J. Vermaak, A. Doucet, and P. Perez, "Maintaining multi-modality through mixture tracking," in *Int. Conf. Computer Vision*, 2003, vol. 2, pp. 1110–1116.

[22] J. Wang, X. Chen, and W. Gao, "Online selecting discriminative tracking features using particle filter," *CVPR'05*, pp. 1037–1042, 2005.

[23] Y. D. Wang, J. K. Wu, A. A. Kassim, and W. M. Huang, "Tracking a variable number of human groups in video using probability hypothesis density," in *Int. Conf. Pattern Recognition*, 2006, pp. 1127–1130.

[24] T. Yu and Y. Wu, "Collaborative visual tracking of multiple identical targets," in *Proc. SPIE.*, 2005, vol. 5682.

[25] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environments," in *Computer Vision and Pattern Recognition*, Washington, DC, Jun. 2004.

[26] S. K. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Process.*, vol. 13, pp. 1491–1506, 2004.

Peng Cui received the B.E. degree in computer science in 2005 from the University of Science and Technology, Beijing, China. He is currently pursuing the Ph.D. degree in Computer Science Department, Tsinghua University, China. His research interests include motion tracking, visual event analysis, and video mining.

Li-Feng Sun (M'08) was born in Xi'an, China, in 1972. He received the B.S. Ph.D. degrees in system engineering in 1995 and 2000, respectively, from National University of Defense Technology, Changsha, Hunan, China.

He is now on the faculty of the Department of Computer Science and Technology at Tsinghua University. His professional interests lie in the areas of peer-to-peer video streaming, interactive multiview video, and video analysis. He has published over 50 papers in the above domains.

Dr. Sun is the secretary-general of the Multimedia Committee of China Computer Federation, Vice Director of Tsinghua-Microsoft Media and Network Key Lab of Ministry of Education (MOE). he served as TPC member of Pacific-Rim Conference on Multimedia (PCM) 2005, Pacific-Rim Conference on Multimedia (PCM) 2006, IEEE International MultiMedia Modelling Conference (MMM) 2007, IEEE International MultiMedia Modelling Conference (MMM) 2006, and IEEE MMSP 2008.

Fei Wang is pursuing the Ph.D. degree in the Department of Automation, Tsinghua University, Beijing, China. His main research interests include machine learning, data mining, information retrieval and pattern recognition. He has published over 40 papers in the top journal & conferences of the relevant field.

Shi-Qiang Yang (M'97–SM'08) was born in July 1952 and graduated from the Department of Computer Science and Technology of Tsinghua University, China, in 1977 and received the M.E. degree in 1983.

He is now a Professor at Tsinghua University. His research interests include multimedia technology and systems, video compression and streaming, content-based retrieval for multimedia information, multimedia content security and digital right management. He has published more than 100 papers and MPEG standard proposals.

Mr. Yang has organized many conferences as program Chair or TPC member, including PCM'05, PCM'06, ACM Multimedia'05, MMM'06, ICME'06, MMSP'05, ASWC'06, etc.