Multiple Feature Fusion in the Dempster-Shafer Framework for Multi-Object Tracking

Dorra Riahi, Guillaume-Alexandre Bilodeau
LITIV Lab, Dept of Comp. and Software Eng.
École Polytechnique de Montréal
Montréal, QC, Canada
Email: dorra.riahi@polymtl.ca, gabilodeau@polymtl.ca

***Abstract*—This paper presents a novel multiple object tracking framework based on multiple visual cues. To build tracks by selecting the best matching score between several detections, a set of probability maps is estimated by a function integrating templates using a sparse representation and color information using locality sensitive histograms. All people detected in two consecutive frames are matched with each other based on similarity scores. This last task is performed using the comparison of two models (sparse apparence and color models). A score matrix is then obtained for each model. Those scores are combined by Dempster-Shafer's combination rule. To obtain an optimal selection of the best candidate, a data association step is achieved using a greedy search algorithm. We validated our tracking algorithm on challenging publicly available video sequences and we show that we outperform recent state-of-the-art methods.**

***Keywords*-Sparse appearance model; Dempster-Shafer combination rule; multi-object tracking**

I. Introduction

Visual tracking is considered as an important issue in many computer vision applications like surveillance, human computer interaction, and augmented reality. Multi-target tracking is a significant and challenging problem for many reasons:

- The tracking result depends on the quality of the detection of the objects of interest, especially when there is many false negatives and false positives detections;
- There is a matching ambiguity between similar targets (in appearance and shape) and for occluded targets;
- It is difficult to assign a unique and consistent label to all tracked targets.

Multiple object tracking (MOT) can be formulated as a data association task of the target objects in order to find all targets trajectory. In this paper, we deal with the problem of tracking multiple objects whose trajectories may intersect during a period of time. Furthermore, we focus particularly in reducing the matching ambiguities by strengthening the object model. We propose a novel tracking framework using multi-cues fusion. The selected visual cues are assumed to be independent so that they can be combined optimally. Our approach unites the strengths of a sparse representation and a color-based representation. As persons enter and exit a scene frequently, manual initialization is impractical. Therefore, the target tracks are automatically initializes and to be invariant to the target appearance changes, we propose to update the target model incrementally. After that, our framework incorporates each feature score into a Dempster-Shafer structure for distinguishing each tracked target. Finally, a global matching solution between two consecutive frames is found by applying a greedy search algorithm.

Our method contrasts with previous works on the following aspects. Approaches that incorporate multiple visual cues in their tracking algorithm often use a weight or coefficient for each cue. In contrast, in our approach, scores for each feature are fused into a global score by applying the Dempster-Shafer rule of combination [1]. Furthermore, previous approaches that exploit sparse representation use it for single target tracking. In contrast, our algorithm develops this technique under a robust multi-target tracking for distinguishing the objects in the scene. For sparse representation, we apply the method developed by Bao et al. [2]. Unlike the typical L1-tracker, in the proposed method, we use the sparse representation on the detected blob as an observation model and use it as a shape constraint in combination with an additional geometrical constraint for distinguishing candidate detections. Our three main contributions are:

- We adapt the sparse representation model for the case of multiple object tracking and we propose a novel probability estimation for the L1-tracker using a geometrical constraint.
- We propose an assignment of tracked targets that uses a function integrating multiple cues fusion.
- We formulate the fusion of the scores of the multiple cues under Dempster-Shafer framework.

II. Background and related works

Tracking algorithms can be classified into two categories based on the appearance models used:

- Generative model-based tracking algorithm that extracts an appearance model in feature space then locates the best target in the candidate set by optimizing similarity.

- Discriminative model-based tracking where object tracking is formulated as a binary classification problem between a target frame and the background. This second approach is not typically used in multiple object tracking.

Hybrid strategies for object tracking were recently proposed. Many tracking methods are interested in combining tracking and detection [3] [4] [5]. They link detection responses with particle filtering tracking results into trajectories. In [4], Xu Yan et al. proposed an ensemble framework that integrates independent trackers and a detector at each frame. The output from the tracker and the detector are directly used for associating candidates. To find the optimal selection, a hierarchical data association step is achieved. The use of a detector and a tracker output adds robustness against unreliable detections. Similarly, another work was presented by Breitenstein et al [6] that differs from the previous by the use of instance-specific classifiers to create an observation model for each target. The observation model is associated with pedestrian detectors.

While numerous algorithms for object tracking have been proposed, modeling the target appearance over time remains a key challenge in visual tracking because of appearance variations and occlusions. The most popular appearance features used to model target are color, texture, and motion. In [7], authors used weighted color histograms to represent the targets. Many tracking approaches assume that the appearance of the target is not fixed over the time. So online learning algorithm have been developed. The target model is updated dynamically. In [8], a new algorithm of multiple-instance learning is developed based on random forests. These tracking algorithms allow achieving long-term persistence of model appearance even if the object model looks different in at different point in time.

The use of hybrid appearance model has drawn increasing attention in recent literature. In [9], Avidan presented a discriminative tracking algorithm that use the HOG (histogram of oriented gradients) and RGB space within an Adaboost framework. A confidence map of pixels is then created using an ensemble of weak classifiers. In paper [10], a new approach of tracking people in sports videos is proposed using combination of a particle filter with a confidence map of detectors, appearance features and motion information.

In this work, we used a generative model. Among these models, sparse representation techniques have been used recently to locate targets based on their appearance [2] [11] [12] . The tracking process is defined as finding a sparse representation of the tracking target in a template space, so that each target is approximated by a combination of few templates (limit number of templates). An L1-norm minimization problem has to be solved for each frame to locate the tracked target. Typically, this representation is used within the particle filter framework and is considered as a standard approach for object tracking. A large number of particles are sampled to localize the target. These particles are distributed in a neighbouring region around the target. In this work, we will use a sparse representation and L1 minimization in a MOT framework. But here, instead of sampling around the previous object position to obtain candidate targets, we consider all the objet detections in the scene as candidate targets (somewhat like particles in the particle filter framework). The sparse representation is thus effectively used to distinguish between different targets. Motivated by the success of using hybrid appearance model, in addition to using a sparse representation and L1 minimization, we are also using color information using locality sensitive histograms [13]. The benefit of using many features for the object model is that it adds robustness for the tracker, e.g. color and shape cues are not sensitive to object deformations and lighting changes at the same time during tracking.

III. Proposed framework

A. System Overview

The main goal of this paper is to reduce the matching ambiguities by strengthening the object model in multiple object tracking (MOT). We explore a new combination of similarity score that can be used for improving the robustness of matching detections over time. Figure 1 illustrates the steps of our proposed method. Given a video sequence, all the persons (tracking targets) are obtained by a person detector. Initially, all detections in the first frame are used to initialize our tracker, so that, these detection responses build the appearance model for each tracked target. For each feature, a score matrix is calculated by comparing the similarity between detections in two consecutive frames. After collecting a score vector from each feature (color and sparse representation) for each target, a combined probability map is obtained using Dempster-Shafers rule of combination. To find the global maximum assignment for each target for a pair of consecutive frames, a data association step in achieved. We apply a simple greedy-search on the score matrix. Then the appearance model for each target is update based on the assignation result obtained in the current frame. To handle the entry and exit of targets, a special process is applied that will be described later on III-D

B. Multi-visual cues model

MOT is defined as an assignment problem that finds the best matches between detected blobs in each frame. In this paper, our focus is in improving the matching process. Our matching step is based on two appearance models: a sparse appearance model and a color model.

1) Sparse appearance model : Sparse appearance models have attracted attention in many research areas recently. We adopted the method proposed in [2] for creating a sparse appearance model of each target. As illustrate in figure 2, each candidate is sparsely projected in a template space.

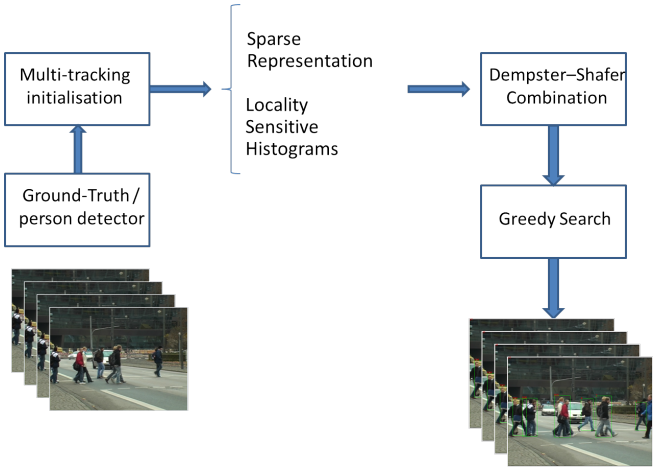

Figure 1: Steps of our tracking system

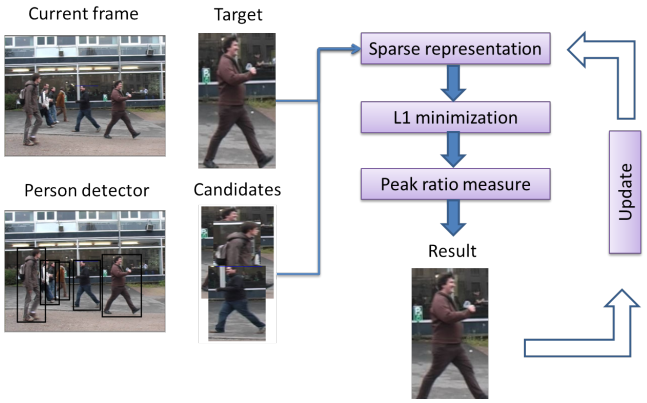

Figure 2: Overview of the creation of a sparse appearance model

Then, a vector of coefficients is obtained and it reflects the approximate error of the sparse representation projection. The candidate with the smallest target template projection error is selected as the prediction result of the target in the current frame.

In the original version of the L1 tracker that we use, the candidates are generated using a particle filtering approach. For each tracked object, a set of equally weighted particles is recursively constructed. The posterior probability reflects the similarity between the target and the particle. In our approach, we proposed a modification of this process. The detections in each frame are used as observation candidates. Thus, for each target, we should calculate the similarity between the candidate detections and the target based on a sparse representation.

First of all, a template dictionary is obtained by applying successive translations around the initial position of the target. Let $T = \{t_1, t_2, ..., t_n\} \in \mathbb{R}^{d \times n}$ be a set of n target template, let x_t describes the appearance model of the target at frame t and let $Y = \{y_1, y_2, ..., y_m\}$ denote the corresponding candidate. A candidate y_i is sparsely projected in the template space using:

$$y_i = Ta = [a_1 t_1 + a_2 t_2 + ... + a_n t_n], \quad (1)$$

where $a = (a_1, a_2, ..., a_n)^T \in \mathbb{R}^n$ is a target coefficient vector. In practice, a target is corrupted by noise or partial occlusions. So, equation 1 is modified to:

$$y_i = Bc, \quad (2)$$

where $B = [T, I, -I] \in \mathbb{R}^{d \times (n+2d)}$is a set of non-trivial and trivial templates and $c = [a, e^+, e^-] \in \mathbb{R}^{n+2d}$ is a non-negative coefficient vector.

The nonnegative constraints imposed add robustness on the L1 tracker [2]. After doing the sparse projection of each candidate, the L1-norm minimization problem should be solved to get the prediction tracking result:

$$\min \frac{1}{2} \|y_i - Bc\|_2^2 + \lambda \|c\|_1, \quad (3)$$

where $||.||_1$ and $||.||_2$ denote the l_1 and l_2 norms. The observation likelihood of the candidate y_i is :

$$p(y_i|x_t) = \frac{1}{\tau} \exp[-\alpha \|y_i - Tc\|_2^2, \quad (4)$$

where α is a constant and τ is a normal factor and c is the solution of the last equation.

The best candidate match corresponds to the one with the largest probability (with the minimum reconstruction error in the template space). To obtain the final best candidate match, we added a geometrical constraint on the candidate depending on the last position of the target. In fact, in an environment with multiple persons, the grayscale templates of person can be similar (e.g. if the two persons are in the same posture). That is why, the L1 tracker fails in some case. As such, it is better to reject improbable candidates with the hypothesis that a person posture does not change a lot between two consecutive frames. To handle this problem, we use a confidence measure. This technique is used in various stereo reconstruction problems. It reflects the measure of the uncertainty of the sparse-based appearance model. We calculate the peak ratio measure that is the ratio between the first two matching costs (in our case, the matching cost is the likelihood between the target and the candidate). If this ratio is smaller than a threshold, it means that two candidates could equally be the predicted as the target. In such case, we include a geometrical constraint to choose the candidate that it is closest to the target in term of Euclidean distance.

Finally, the set of templates should be updated depending on the tracking result. A template will be replaced if the predicted candidate is not similar to it. Furthermore, the weight of each template is increased if the similarity is higher than a threshold. A matrix score of similarity based on comparison of sparse appearance model is obtained after this step.

2) Color model: A locality sensitive histogram is used to construct the color model [13]. The local sensitive histogram takes into account contributions from each pixel in the target area. In the conventional image histogram, the frequency of occurrences of intensity value is incremented when the intensity value belongs to bin b for each pixel of the target area. In contrast, a floating-point value proportional to the distance to the pixel location is added to the corresponding bin for each occurrence of intensity pixel in the case of locality sensitive histograms. Mathematically, the locality sensitive histogram at pixel p can be written as:

$$H_p^E(b) = \sum_{q=1}^{W} \alpha^{|p-q|.Q(I_q,b),b=1,...,B}, \tag{5}$$

where $\alpha \in [0,1]$ is a control parameter, q is a neighbouring pixel, W is the number of pixels in the region E (the target or candidate area) and $Q(I,b)$ is the occurrence of the pixel intensity belonging to bin b. After calculating the locality sensitive histogram for each candidate, the similarity between candidate detections and target model is measured as:

$$d(H_1, H_2) = \sum_{b=1}^{B} |H_1(b) - H_2(b)|, \tag{6}$$

where H_1 and H_2 are the locality sensitive histograms of candidate and target and B is the number of bin in the histogram. Finally, we obtain a matrix of scores where each value defines the comparison between histograms.

3) Improving matching accuracy by improving target to candidate alignment: In our algorithm, we are using the output of a people detector to create the target model. In addition to the problem of misdetections and false positive detections, a detected person does not always have the same size as the target model. Thus, this can be problematic when attempting to compare templates like in a sparse representation where a good alignment and similar image region size are required to evaluate properly the similarity with templates. To address this problem, our method tries various target to candidate alignment by breaking up a target region into several equal patches in the following way: 1) If the candidate region (the detected person) is bigger than the target model, then we split the detected region into multiple equal patches, and 2) Otherwise, if the candidate region (the detected person) is smaller than the target model, then we split the tracked region into multiple equal patches. In order to decide which candidate patch should be chosen, we calculate the color and sparse appearance model for each patch of each candidate like described in the previous section. Then, the candidate that gives the highest scores (score for each candidate patches) will be selected as the current prediction of the target.

C. Dempster-Shafer's rule of combination

The second task in our multiple objects tracking framework is the combination of similarity scores based on the sparse appearance model and color model. We have chosen the Dempster-Shafer theory to combine multiple decisions [14]. The advantage of this theory is that it is robust for combining uncertain scores. To apply this theory, there are two assumptions: 1) The combined masses should be independent and 2) The mass function should be in $[0,1]$ and $\sum m(A) = 1$, where A is called body of evidence and m is a mass function. In our framework, the body of evidence is the candidates and the evidence items are the two score matrices of similarity. So the combination mass can be written as:

$$(m_1 \oplus m_2)(A) = \begin{cases} 0 & if A = \varnothing \\ \frac{1}{1-K} \sum_{B \cap C = A} m_1(B) m_2(C) & if A \neq \varnothing \end{cases} \tag{7}$$

with

$$K = \sum_{B \cap C = \varnothing} m_1(B) m_2(C) \tag{8}$$

where m_1 is the sparse appearance probability, m_2 is the similarity between locality sensitive histograms and A is the list of candidates. Let denote $O = [o_1, o_2, ..., o_n]$, the list of tracked people and $D = [d_1, d_2, ..., d_m]$ indicating the set of all candidates (detections) in the current frame. The fusion similarity score is then saved into an $n \times m$ matrix:

$$\begin{pmatrix} s_1^1 & s_1^2 & \dots & s_1^n \\ s_2^1 & s_2^2 & \dots & s_2^n \\ \vdots & \vdots & \vdots & \vdots \\ s_m^1 & s_m^2 & \dots & s_m^n \end{pmatrix}$$

where s_i^j is the fusion similarity score between o_i and d_j. It is obtained by combination of similarity score based on sparse appearance and color models comparison between all combinations of the candidates in two consecutives frames.

D. Data Association

First of all, an initialization step is achieved. All blobs detected in the first frame form our initial list of targets to be tracked, and an in/out region is selected manually (in the first frame) to define the area of exit and entry of persons. Each target is defined by its coordinates, an identifier, the frame number when it first appears in the scene, the number of times it was not matched and a state. The state field allows distinguishing a target that is occluded and to update our list of targets by adding the new entry and deleting the exiting object. The states can be defined by the labels $initial$, $entry$, $activate$, $occluded$ or $exit$. As it is described in the state machine (Figure 3), initially, a target blob is in the $initial$ state or in the $entry$ state if it is detected in

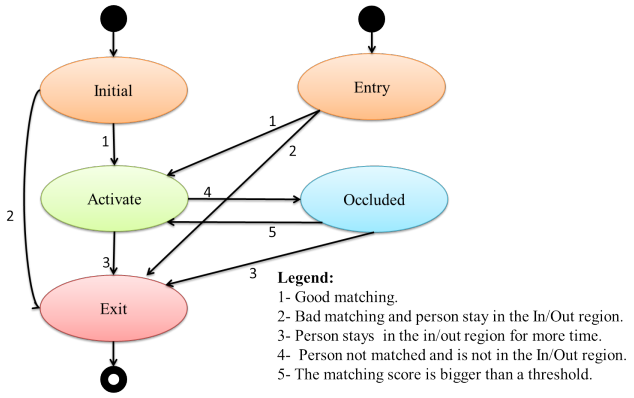

Figure 3: Person state machine

the in/out region. If after a few frames (we use 5 frames) a tracked person is not matched, it will be deleted from the list of person to be tracked. Else, the tracked person is considered in the *activate* state. A person in the *activate* state is marked as *occluded* if is not matched (using a greedy search algorithm). An *occluded* person will go back to the *activate* state only if the associated score is larger than a threshold.

To do the assignment of all candidates at each frame, we use a greedy search approach. The greedy search algorithm works in the following manner. First, a matching score matrix S is computed for each pair (o , d) for the target o and the current detection d (as described in the section III-C). Then the pair with maximum score (only the scores above a threshold are considered) is iteratively selected and the rows and columns belonging them are deleted from the matrix. This processes will repeated until no pair is available. Finally, only one of the detections is selected as a good match for the target. An updating phase is required to update person features (color and sparse appearance model). For each target, the patch belonging the selected detection is used to update the target model in the current frame only if this latter is not occluded or deleted.

A summary of the approach is presented in Algorithm 1:

IV. Experiments

Implementation and parameters For all experiments, we used the person detector of [3]. We apply a filtering algorithm to delete most false detections based on size below or above a threshold and based on the confidence of the detector. As the detected regions contain many background pixels, we removed some pixels on the sides (right and left side) of the detected region. All the parameters have been fixed experimentally. For the locality sensitive histogram, $nbin = 32$ and $\alpha = 0.15$ and for the sparse appearance model, we fixed the number the templates to 10 for all the experiments. The size of the template is sets at [42, 14]. The threshold of the greedy search algorithm depends on

Algorithm 1 Multi-Object Tracking

Require: Person detector output
Ensure: assignment matrix
Initialization
$O^0 = \{o_1, o_2, ..., o_n\}$ set of target belonging to the first frame ;
for each frame **do**
$D^t = \{d_1, d_2, ..., d_m\}$ set of detected blob in the current frame t ;
Calculate score matrix M for each pair (o_i, d_i);
$m_1(o_i, d_i) = \frac{1}{\tau} \exp[-\alpha \|y_i - Tc\|_2^2$
$m_2(o_i, d_i) = sum_{b=1}^{B} |H_1(b) - H_2(b)|$
$M(o_i, d_i) = m_1 \oplus m_2$
A = greedySearch(M, th)
for each target o_i **do**
if o_i is not matched **then**
o_i is marked as occluded;
end if
if o_i is detected in the in/out region **then**
$O^t = \{O^{t-1}\} \setminus o_i$;
end if
end for
for each detection d_i **do**
if d_i is not assigned and is detected in the In/Out region **then**
$O^t = \{O^{t-1}, d_i\}$;
end if
end for
Compute the new matrix M and A;
Update appearance and sparse models for only activate person;
end for

the video (for TUD campus is 0.37 and for TUD crossing is 0.53).

Evaluation protocol

The evaluation of our tracking algorithm was achieved on the challenging data set TUD [3]: TUD campus (71 frames containing 6 persons) and TUD crossing (201 frames containing 8 persons). The TUD dataset is very challenging due to heavy occlusion (the walking persons occlude each other frequently and their overlap differ). Also, the detections have different sizes.

To evaluate our tracking performance, we use the CLEAR MOT [15] metrics and Precision, and Recall :

- MOTP, MOT precision, is calculating the average distance between predicted targets and the ground truth.
- MOTA, MOT accuracy, is reflecting the performance of tracking algorithm. It is calculated as a function of false negative (FN), false positive (FP) and the number of ID switches (IDS).
- Precision is the percentage of correctly matched detec-

tions relative to the total number of detections in the tracking result.
- Recall is the percentage of correctly matched detections relative to the total number of detections in the ground-truth.

Results Table I and II show quantitative results for our tracking algorithm. It shows that it is competitive with the state-of-the-art. We compared our approach to four state-of-the-art tracking methods. Despite of using a public dataset and the well-known CLEAR MOT metrics, it is not easy to obtain comparisons as many tracking algorithms differ in the person detector or the detection algorithm used. To remove the effect of the detector in our comparison with the literature, and focus on the object model, the two works with which we are comparing ourselves use the same detector as us. For the two videos sequence, we managed to exceed 72% as performance of our tracking method (MOTA) with more than 74% of precision. For TUD Campus, our results outperform Breitenstein et al. tracking method [6]. On the TUD Crossing sequence which is similar to TUD Campus but contains heavy occlusions, the MOTA value reach 72% compared to 71% for Breitenstein et al. method [6]. Our improved performance is explained by the use of a combination of color feature and a sparse representation in our fusion based framework, instead of relying only on a motion model like in [6] and additional single object trackers. So even with a simple data association method, our stronger appearance model allows us to obtain better or similar performance to a method that uses a more complex data association that combines outputs from trackers and detectors.

To compare with other methods that use other metrics, we also calculated the Precision and Recall. As it can be seen in table III, our precision score for the two video sequences exceeds 97% while recall value of more than 73%. Our tracking algorithm is shown to improve the results in term of precision compared to other methods even though our method does not use a discriminative model (boosted classification in [16] and SVM classifier in [17]. Using a target model based on sparse appearance and color feature, our method outperforms Brendel et al. method [18] that use a learning approach including many features (color, motion, location)..

The results give in table I and II show that if we use the ground truth detections, our method gives higher value of CLEAR MOT. In fact, higher number of false detections gives more false assignments. So, the number of targets in the results is higher than what it is with the ground-truth detections.

Our tracking method may be affected by two phenomenons: false detections and missing detections. To investigate quantitatively the effect of the false detection errors, we plot a curve describing the variation of MOTA with respect to the false positive value. As illustrated in figure 4, we can conclude that our approach is somewhat sensitive to the false positive variations. The false positive detections are mainly caused by the failure of the person detector. These missing detections makes handling the occlusion more difficult as information are missing to sort them out. However, it is more a limit of our simple data association then our model.

Video Sequence	MOTA	MOTP	FP	FN	IDS
TUD Campus	**72**	74	2	25	1
TUD Campus*	83	100	0	13	12
TUD Campus [6]	67	73	-	-	2

Table I: CLEAR MOT evaluation results of TUD Campus video. Our results are in the top row. * CLEAR MOT results using ground-truth detections

Video Sequence	MOTA	MOTP	FP	FN	IDS
TUD Crossing	72	76	1	26	7
TUD Crossing*	98	100	0	2	9
TUD Crossing [6]	71	84	-	-	2

Table II: CLEAR MOT evaluation results of TUD Crossing video. Our results are in the top row. * CLEAR MOT results using ground-truth detections

We also study the qualitative evaluation of our tracker algorithm on the two TUD videos. As it can be seen in figure 5 and 6, all the detections are successfully matched. These competitive results the advantage of using multiple models to distinguish similar people (in color, in shape, in movements). Indeed, as it can be seen in the second row of figure 5, our method is able to maintain the id for target number 2 while it has been occluded in several frames. Although walking people look similar, the use of combined similarity scores from different feature spaces (the color feature and the sparse appearance feature) can handle the drawbacks of using each feature separately. Moreover, because of our state machine approach, our tracker method is able to handle heavy occlusions provided that the persons are properly detected.

Video Sequence	Precision(%)	Recall(%)
TUD Campus	**97**	75
TUD Crossing	**99**	73
TUD Crossing [17]	94	78
TUD Crossing [18]	73	-
TUD Crossing [16]	71	-

Table III: Precision- Recall evaluation results of TUD data set videos using a person detector

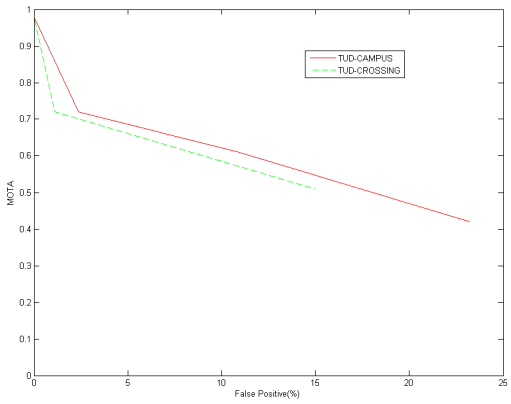

Figure 4: Results for MOTA for increasing false positives for TUD-CAMPUS and TUD-CROSSING videos.

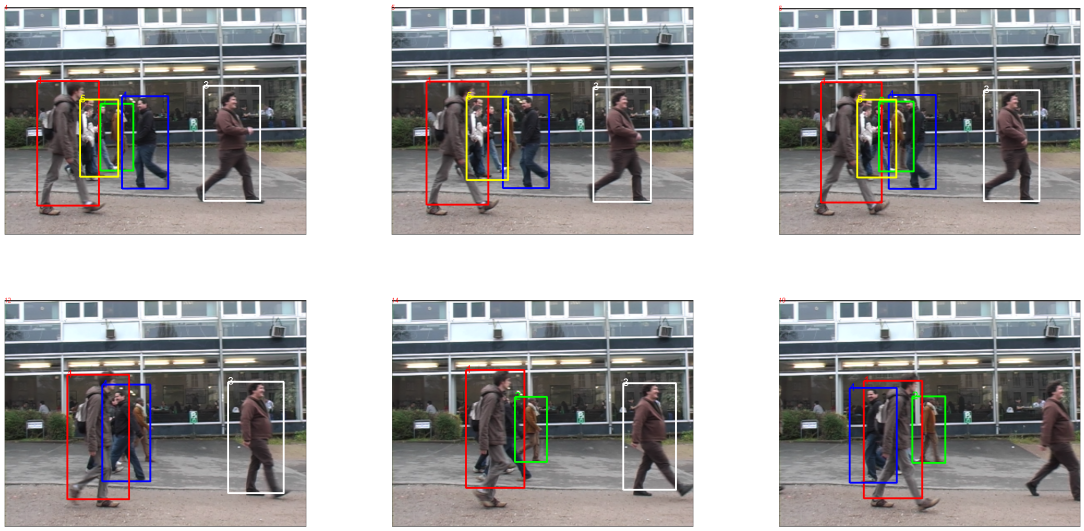

Figure 5: Screenshots result of the persons over TUD Campus video frames with serious occlusion (in the second row).

V. Conclusion

In this paper, we presented a novel approach for object multi-tracking that integrates the fusion of two heterogeneous features that are a sparse representation and locality sensitive histograms. First of all, a score matrix is calculated using the comparison of object models between all the detections in two consecutive frames. The two similarity scores are then fused under a Dempster-Shafer framework in order to ameliorate the similarity scores for each feature. Finally, a data association step is achieved using a greedy search algorithm. Using the ground-truth detection, our model is shown to be robust to distinguish different targets. For detection with a person detector, our method shown promising abilities as it obtains competitive results compared to state-of-the-art methods.

Acknowledgment

This work was supported by NSERC discovery grant No. 311869-2010.

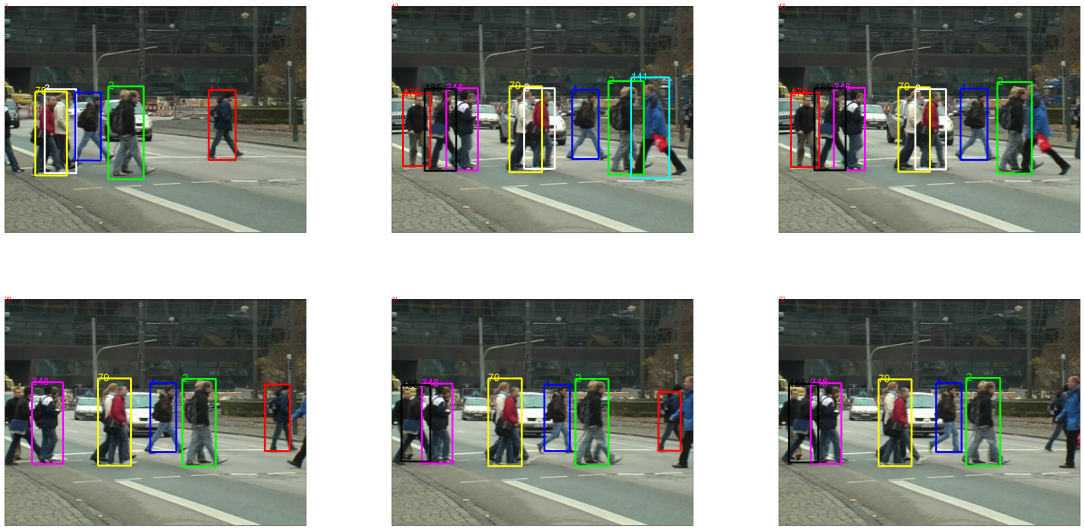

Figure 6: Screenshots result of the persons over TUD Crossing video frames with serious occlusion (in the second row).

REFERENCES

[1] G. Shafer, *A mathematical theory of evidence*. Princeton university press Princeton, 1976, vol. 1.

[2] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *CVPR*, 2012, pp. 1830–1837.

[3] M. Andriluka, S. Roth, and B. Schiele, "People-tracking-by-detection and people-detection-by-tracking," in *CVPR 2008*, 2008, pp. 1–8.

[4] X. Yan, X. Wu, I. Kakadiaris, and S. Shah, "To track or to detect? an ensemble framework for optimal selection," in *ECCV 2012*, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Springer Berlin Heidelberg, 2012, vol. 7576, pp. 594–607.

[5] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *ECCV 2008*, ser. Lecture Notes in Computer Science, D. Forsyth, P. Torr, and A. Zisserman, Eds. Springer Berlin Heidelberg, 2008, vol. 5303, pp. 788–801.

[6] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *PAMI*, vol. 33, no. 9, pp. 1820–1833, 2011.

[7] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *PAMI*, vol. 25, no. 5, pp. 564–577, 2003.

[8] C. Leistner, A. Saffari, and H. Bischof, "Miforests: Multiple-instance learning with randomized trees," in *ECCV 2010*, ser. Lecture Notes in Computer Science, K. Daniilidis, P. Maragos, and N. Paragios, Eds. Springer Berlin Heidelberg, 2010, vol. 6316, pp. 29–42.

[9] S. Avidan, "Ensemble tracking," *PAMI*, vol. 29, no. 2, pp. 261–271, 2007.

[10] A. Yao, D. Uebersax, J. Gall, and L. Gool, "Tracking people in broadcast sports," vol. 6376, pp. 151–161, 2010.

[11] X. Mei and H. Ling, "Robust visual tracking using l1 minimization," 2009, pp. 1436–1443.

[12] X. Mei, H. Ling, Y. Wu, E. Blasch, and L. Bai, "Minimum error bounded efficient l1 tracker with occlusion detection," in *CVPR*, 2011, pp. 1257–1264.

[13] S. He, Q. Yang, R. W. Lau, J. Wang, and M.-H. Yang, "Visual tracking via locality sensitive histograms," *Proceedings of IEEE*.

[14] S. Tulyakov, S. Jaeger, V. Govindaraju, and D. Doermann, *Review of Classifier Combination Methods*, ser. Studies in Computational Intelligence, S. Marinai and H. Fujisawa, Eds. Springer Berlin Heidelberg, 2008, vol. 90.

[15] B. Keni and S. Rainer, "Evaluating multiple object tracking performance: the clear mot metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, 2008.

[16] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Robust tracking-by-detection using a detector confidence particle filter," pp. 1515–1522, Sept 2009.

[17] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth, and B. Schiele, "Learning people detectors for tracking in crowded scenes," *ICCV13*, 2013.

[18] W. Brendel, M. Amer, and S. Todorovic, "Multiobject tracking as maximum weight independent set," pp. 1273–1280, June 2011.