Improving Multi-frame Data Association with Sparse Representations for Robust Near-online Multi-object Tracking

Loïc Fagot-Bouquet[1(✉)], Romaric Audigier[1], Yoann Dhome[1], and Frédéric Lerasle[2,3]

[1] CEA, LIST, Vision and Content Engineering Laboratory, Point Courrier 173, 91191 Gif-sur-Yvette, France
{loic.fagot-bouquet,romaric.audigier,yoann.dhome}@cea.fr
[2] CNRS, LAAS, 7, Avenue du Colonel Roche, 31400 Toulouse, France
[3] Université de Toulouse, UPS, LAAS, 31400 Toulouse, France
lerasle@laas.fr

Abstract. Multiple Object Tracking still remains a difficult problem due to appearance variations and occlusions of the targets or detection failures. Using sophisticated appearance models or performing data association over multiple frames are two common approaches that lead to gain in performances. Inspired by the success of sparse representations in Single Object Tracking, we propose to formulate the multi-frame data association step as an energy minimization problem, designing an energy that efficiently exploits sparse representations of all detections. Furthermore, we propose to use a structured sparsity-inducing norm to compute representations more suited to the tracking context. We perform extensive experiments to demonstrate the effectiveness of the proposed formulation, and evaluate our approach on two public authoritative benchmarks in order to compare it with several state-of-the-art methods.

Keywords: Multiple Object Tracking · Tracking by detection · Multiple frame data association · Sparse representation · MCMC sampling

1 Introduction

Multiple Object Tracking (MOT) aims to estimate the trajectories of several targets in a scene. It is still a challenging problem in computer vision and has a large number of potential applications from video-surveillance to embedded systems. Thanks to the recent advances in object detection, MOT community has strongly focused on the *tracking-by-detection* technique where object detections are grouped in order to estimate the correct tracks. However, despite this data

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46484-8_47) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part VIII, LNCS 9912, pp. 774–790, 2016.
DOI: 10.1007/978-3-319-46484-8_47

association formulation of the problem, tracking multiple objects remains a challenging problem due to frequent occlusions and interactions of targets, similar appearances between targets, pose variations, and object detection failures.

In the literature, the problem is addressed by a large variety of approaches, from *online* (or single-scan) techniques [1–4] where only the previous frames are considered, to *offline* approaches using past and future frames. Among offline techniques, global approaches perform the data association over all the frames simultaneously or by batch [5–15], whereas *sliding window* (a.k.a. multi-scan, near-online, or online with delay) methods optimize only a few recent frames at the same time [16–20].

The large variety of approaches in the literature is justified by the variety of contexts and applications that encounters the MOT problem. Online approaches are well-suited for time-critical applications but are more prone to specific errors such as identity switches. On the other hand, global tracking approaches offer the advantage of dealing with all the available information at the cost of a major temporal delay. Finally, sliding window approaches offer an interesting compromise, having a relative time to understand the situation at the cost of a slight temporal delay. By delaying the final tracking results by only a few frames, these methods are able to correct association mistakes occurring inside the sliding window and generally yield more robust results with fewer identity switches and fragmented tracks.

Recently, many online or sliding window approaches have gained in performances by incorporating more complex appearance models [1,4,17]. These models, inspired by the recent improvements in Single Object Tracking (SOT), can be updated online to take into account changes in appearance or pose variations and help better distinguish targets, for more robust tracking results.

In particular, *sparse representation*-based models have been employed successfully in SOT [21–26]. The main idea is to model the target appearance in a linear subspace defined by a small number of templates grouped in a dictionary. Each candidate for the new target location is then represented by a sparse linear combination of the dictionary elements, the best reconstruction error being used as the selection criterion. However, only a few recent methods have considered extending these models for online MOT systems [3,27,28].

We propose two contributions in this paper. The first one consists of improving multi-frame data association by using sparse representation-based appearance models. To the best of our knowledge, we are the first to combine such concepts and so their aforementioned advantages. Our second contribution is to use structured sparse representations, derived from a weighted $l_{\infty,1}$ norm, that are more suited in this context. Comparisons with the l_1 norm and more basic appearance models without sparse representations support the effectiveness of this approach. Our method was evaluated on two public benchmarks and compares well with recent state-of-the-art approaches.

2 Related Work

2.1 Object Tracking with Sparse Representations

Appearance models based on sparse representations were first proposed by [21] in a SOT framework before being extended by many other authors [26]. In contrast to standard approaches that use a dictionary composed solely by target views, some approaches tried to handle occlusions by better discriminating the target from its background. To this end, they considered a dictionary incorporating boundary elements that mix object and its surrounding background [24]. Others employed a description based on local patches of the target and used spatial considerations when reconstructing the patches from a candidate location [23]. Initially, these tracking methods induced a significant CPU cost until optimization techniques based on accelerated proximal gradient descent led to real-time approaches [22].

Due to their success in SOT context, these appearance models have been recently used in a few MOT frameworks. In [28], such models are used in an online tracking method based on a particle filter. However, as many specific and independent models as the number of targets are necessary. In contrast, in [3,27], a single dictionary is shared by all targets and collaborative representations are used to better discriminate them. All these MOT approaches are using a two-frame data association in an online fashion and thus cannot reconsider wrong associations when further information comes and contradicts them.

In this work, we propose a new approach that improves a standard sliding window method by exploiting sparse representations of the detections. Our approach is inspired by [3,27,28], but instead of relying on sparse representations induced by the standard l_1 norm, we design a sparsity-inducing norm, based on a weighted $l_{\infty,1}$ norm, more suited for a multi-frame data association problem.

2.2 Multi-frame Data Association

Offline MOT approaches consider the data association either globally over the whole sequence [5–15] or over a sliding window dealing with a few frames [16–20]. In all cases, this leads to formulate a multi-frame data association problem solved most of the time by an energy minimization procedure.

In some approaches, the multi-frame data association problem has been formulated in a more specific class of problems, like for example minimum cost flow problems [5–7,12], binary integer programming [14], maximum weighted clique [13] or independent set [15]. The main advantage of such approaches is that efficient optimization methods designed for these problems can be directly employed to find the data association solution. However, particular constraints must be satisfied by the energy formulation which makes it difficult to correctly model important aspects of the MOT problem like target interactions and dynamics.

On the other hand, some state-of-the-art approaches focused on designing more complex energies that better model the MOT problem. However, the non-convex energy formulation puts out of reach any possibility of global minimization. It is still possible to get approximate solutions using non-exact optimization

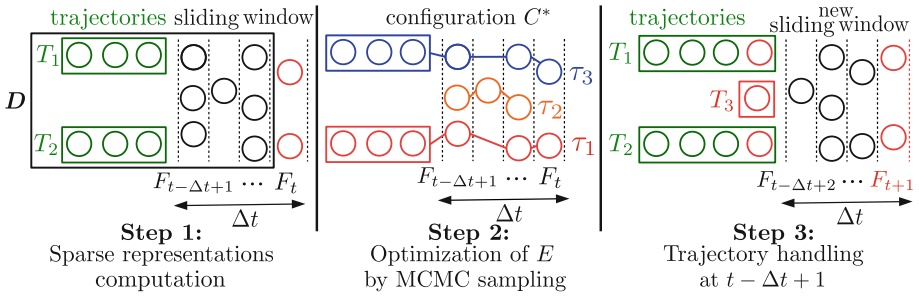

Fig. 1. Steps followed by the proposed approach. Firstly, sparse representations of the detections (symbolized by circles) from the last frame are computed. Then, the global energy E is optimized by MCMC sampling, yielding a configuration C^*. Finally, the trajectories (symbolized by rectangles) are definitively estimated in the first frame of the sliding window, following configuration C^*.

techniques that do not require a specific energy formulation, as done in Multiple Hypothesis Tracking [17] using a breadth-first search with branch pruning or in Markov Chain Monte Carlo Data Association (MCMCDA) with MCMC sampling [20,29]. Despite the non-optimality of the found solution, these methods can fully exploit the use of more appropriate interaction and dynamic models and can therefore cope with more difficult tracking issues.

In this work, we formulate a multi-frame data association with an energy that exploits sparse representations through its appearance model and that can be minimized efficiently using an MCMCDA approach.

3 System Overview

We propose a MOT system based on a sliding window and tracking-by-detection mechanisms. At each new frame, we seek for the best association between the detections over the current sliding window and the already estimated trajectories beyond this window. This multi-frame data association problem is formulated as an energy minimization solved by an MCMCDA approach in the vein of [29]. We design an energy function E assigning low values to solutions with tracks that are both close to the given detections and consistent with some appearance, motion and interaction models.

In the case of visually distinctive targets, taking into account appearances can lead to a significant improvement of the tracking performances. To this aim, we propose an appearance model that considers sparse representations of the detections in the sliding window. The main concept behind our work is that a target should be best represented by the detections of its own track rather than using detections from other targets. Our appearance model is thus formulated to promote the solutions that are the most consistent with these representations.

Our system performs the following steps (cf. Fig. 1). Firstly, sparse representations of the detections from a new frame are computed over a dictionary

that includes all the detections inside a sliding window of Δt frames and some from the latterly estimated trajectories. Secondly, the data association problem is solved using an MCMCDA approach that yields an approximate solution C^*. Thirdly, this solution is used to propagate the trajectories at the first frame of the sliding window, possibly initializing new ones or terminating some of them. Finally, the sliding window is shifted by one frame. While the associations remaining inside the sliding window can still be modified, the ones beyond it are definitive. Therefore, the proposed method outputs results with a slight delay limited to Δt frames.

4 Multi-frame Data Association Formulation

4.1 Notations

We consider a sliding window over the last Δt frames, $\{F_{t-\Delta t+1}, F_{t-\Delta t+2}, ..., F_t\}$. At each frame F_t the detector yields a set of n_t detections $\{d_t^1, d_t^2, ..., d_t^{n_t}\}$. Each detection d is associated to a specific bounding box x_d, with height h_d and width w_d, and a detection score s_d. The trajectories, definitively fixed beyond the sliding window and still active, are denoted by $T_1, T_2, ..., T_N$.

The multi-frame data association requires to find a set $\{\tau_1, \tau_2, ..., \tau_M\}$ of tracks where each track τ is composed by the detections and, possibly, the trajectory related to the same target. A feasible solution for the multi-frame data association is called a configuration. A configuration C is a set of tracks in which (i) each detection and trajectory is included in at most one track, (ii) each track includes at most a single trajectory, and (iii) a single detection by frame. Furthermore, two consecutive detections d and d' linked in a track, spaced by δt frames, have to satisfy (i) $\delta t \leq \delta t_l$, (ii) $dist(x_d, x_{d'}) \leq (1 + \delta t) \frac{w_d + w_{d'}}{2} d_l$, and (iii) $|h_d - h_{d'}| \leq (1 + \delta t) \frac{h_d + h_{d'}}{2} h_l$, where $dist(x_d, x_{d'})$ is the Euclidean distance between the two bounding box centers and δt_l, d_l, h_l are fixed parameters.

For each track τ, we denote by x_τ the set of bounding boxes resulting from a linear interpolation between the detections in τ. Therefore, $x_\tau(t)$ stands for the location of the track τ at time t which is either a bounding box from a detection in τ or one resulting from a linear interpolation to fill a gap between two consecutive detections in τ. We denote respectively by b_τ and e_τ the time of the first and last element in the track τ.

4.2 Proposed Energy

The proposed energy is formulated as a linear combination of four terms:

$$E(C) = \theta_{Ob} Ob(C) + \theta_{App} App(C) + \theta_{Mot} Mot(C) + \theta_{Int} Int(C)\,. \qquad (1)$$

Each one of these terms handles a specific aspect of the MOT problem while the θ values allow to ponderate them.

The objective of the *observation model* is to keep the tracks close to both the given detections and the trajectories already estimated outside the sliding window. To that end, our observation model is written as:

$$Ob(C) = -\sum_{\tau \in C} \sum_{d \in \tau} [\alpha_{Ob} + \beta_{Ob} s_d] - \sum_{\tau \in C} \sum_{T \in \tau} \gamma_{Ob}\,, \tag{2}$$

where α_{Ob}, β_{Ob} and γ_{Ob} are fixed positive parameters. The first term of Eq. 2 rewards the inclusion of detections with a high detection score s_d in the tracks while the second favors the extension of the latterly estimated trajectories.
Our *appearance model* $App(C)$ uses sparse representations of the detections and promotes the configurations in which each detection achieves a small residual error over its own track. More details on this term are given in Sect. 5.

Assuming a constant velocity model, we consider the here below *motion model*:

$$Mot(C) = \sum_{\tau \in C} \sum_{t=b_\tau+1}^{e_\tau - 1} ||x_\tau(t+1) + x_\tau(t-1) - 2x_\tau(t)||_2^2\,. \tag{3}$$

This term favors smooth and constant motion by penalizing the acceleration over the tracks. A constant velocity model, despite its simplicity, already helps limit identity switches in the case of occlusions or collisions between targets.

Lastly, our *interaction model* takes the following form:

$$Int(C) = \sum_{\tau_1 \in C} \sum_{\tau_2 \in C \setminus \{\tau_1\}} \sum_{t=max(b_{\tau_1}, b_{\tau_2})}^{min(e_{\tau_1}, e_{\tau_2})} IOU(x_{\tau_1}(t), x_{\tau_2}(t))^2\,. \tag{4}$$

This term avoids collisions between estimated targets, using a two bounding box Intersection-Over-Union (IOU) criterion.

4.3 MCMC Optimization and Trajectory Handling

Inspired by some recent works [20,29] we use an MCMC sampling method based on the Metropolis-Hastings approach. It finds a good approximate solution of our energy minimization problem by exploring efficiently the space of possible configurations. Such an approach estimates the probability distribution:

$$\pi(C) = \frac{1}{Z} e^{-E(C)/\sigma^2}\,, \tag{5}$$

where Z is a normalization constant, not necessary to compute as only probability ratios are considered in the Metropolis-Hastings approach, and where σ can be chosen to make the distribution more or less peaked. In practice, a suited σ makes an appropriate trade-off between the exploration of the search space and the exploitation of the current state in the Markov Chain, and thus avoids being kept inside a local minimum/maximum of E/π respectively.

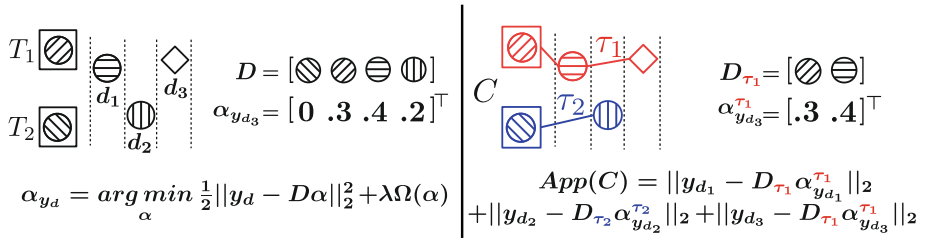

Fig. 2. Proposed appearance model with sparse representations. Left: current sliding window and sparse representations computed for detections in the new frame. Right: configuration C considered and related appearance model value $App(C)$.

We use the approach proposed in [29] with minor differences. In our method, the types of moves are limited to the following ones: birth and death, merge and split, update and switch. We allow these moves to be done not only forward in time, as in [29], but also in a backward manner in order to explore more efficiently the space of configurations.

This method gives an approximate solution C^* of the minimization problem of the energy E. Once this configuration is found, any trajectory T_i that belongs to a track τ in C^* is extended to the first frame of the sliding window accordingly to τ (cf. Fig. 1, Step 3). Any trajectory not included in C^* is terminated while a track τ at the beginning of the sliding window with no associated trajectory possibly leads to the creation of a new trajectory. A new trajectory is indeed created if we are confident enough on the track τ, requiring that τ includes at least N_c detections with a mean detection score value above s_c.

5 Sparse Representations Using an $l_{\infty,1}$ Penalty

5.1 Proposed Appearance Model

We define here the appearance model $App(C)$ that we use in the energy E described previously (Eq. 1). Our approach model takes benefit from the efficient sparse representation-based models in SOT [26].

We propose an appearance model which exploits sparse representations of the detections in the sliding window. Each detection d_t^i is associated to a normalized feature vector $y_{d_t^i}$ and we use a dictionary D that includes all the feature vectors of the detections in the current sliding window. The dictionary D also includes the feature vectors of the N_{tr} last detections assigned to each trajectory T_i. A sparse representation for a given detection d is defined by:

$$\alpha_{y_d} = \arg\min_{\alpha} \frac{1}{2}||y_d - D\alpha||_2^2 + \lambda\Omega(\alpha)\,, \tag{6}$$

where $\Omega(\alpha)$ is a penalty that promotes solutions α with a few non-zero elements.

When one needs to perform multiclass classification and assign a label or a class L^* to the vector y_d, sparse representations can be used to estimate this class based on its related residual error:

$$L^* = \arg\min_L ||y_d - D_L \alpha_{y_d}^L||_2 \,, \tag{7}$$

where D_L is the restriction of D to its elements from class L, and $\alpha_{y_d}^L$ is the restriction of α_{y_d} to the dimensions related to those elements [30]. In SOT, a common approach is to classify a candidate location either in a target or background class [24]. We propose an appearance model for multi-object tracking based on the same technique. This leads to consider:

$$App(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - D_\tau \alpha_{y_d}^\tau||_2 \,, \tag{8}$$

where $||y_d - D_\tau \alpha_{y_d}^\tau||_2$ is the residual error of detection d with respect to track τ. This model promotes the configurations C that achieve the smallest residual errors for all the detections with respect to the assigned tracks (cf. Fig. 2).

In practice, evaluating the value of $App(C)$ for each state of an MCMC sampling framework is computationally expensive due to the estimation of a significant number of residual errors. Instead of using residual errors, some approaches in classification and SOT, as for example in [23], directly use:

$$L^* = \arg\max_L \sum_i \alpha_{y_d}^L(i) \,, \tag{9}$$

where the summation takes into account all coefficients $\alpha_{y_d}^L(i)$ of the vector $\alpha_{y_d}^L$. In order to speed up the MCMC sampling, we use this same approach and finally consider as appearance model:

$$App(C) = \sum_{\tau \in C} \sum_{d \in \tau} [1 - \sum_i \alpha_{y_d}^\tau(i)] \,. \tag{10}$$

5.2 Desired Sparsity Structure

In Eq. 6, a large number of penalties $\Omega(\alpha)$ can be employed to favor different sparsity structures in the representations. A simple choice is to consider $\Omega(\alpha) = ||\alpha||_1$, promoting a strict sparsity with an l_1 norm. More complex sparsity structures can be induced, notably by considering groups of dictionary elements. For example, an $l_{1,2}$ or $l_{1,\infty}$ norm can easily promote representations where only a few groups are non-zero with a uniform participation of the elements inside these groups. These penalties have been used in SOT approaches to produce sparse representations more suited to efficiently handle multiple features or to consider jointly all candidate locations.

This leads us to wonder which penalty will be the most appropriate for the MOT problem. Ideally, all detections should be represented by elements from their own trajectories. Therefore, a well-suited sparsity structure should promote

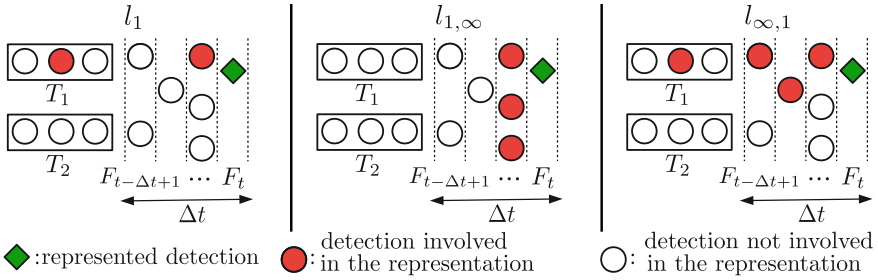

Fig. 3. Sparsity structures induced by different penalties over the sliding window.

a few non-zero elements in each frame, as two detections in a frame F_j cannot be related to the same target. It should as well favor the participation of only a few elements from trajectories $T_1, ..., T_N$ as a detection should be related to a single trajectory at most. Thus, considering for $i = [1...\Delta t - 1]$ a group G_i composed of the elements related to frame F_{t-i} and a group $G_{\Delta t}$ that includes all elements from trajectories $T_1, ..., T_N$, we want to impose a strict sparsity within each individual group. As a target should be located at each frame, we also want to promote a uniform participation of these groups. In this way, each detection should be represented by all the other detections relative to the same target.

Neither the l_1 norm nor group norms like the $l_{1,2}$ or $l_{1,\infty}$ norms induce the described structure. So we propose to use instead a weighted $l_{\infty,1}$ defined by:

$$||\alpha||^w_{\infty,1} = \max_{i=1..\Delta t} w_i ||\alpha^{G_i}||_1 \,, \tag{11}$$

where α^{G_i} is the restriction of α to the elements related to G_i. The values w are positive weights balancing the participation of the groups. We use in practice $w_{\Delta t} = \frac{1}{\Delta t - 1}$ and $w_i = 1$ for $i < \Delta t$ in order to allow a greater participation of the elements inside the trajectories in $G_{\Delta t}$. This norm induces the desired sparsity structure, as it imposes a strict sparsity inside the groups while favoring that all the groups are involved in the representation (cf. Fig. 3).

5.3 Computing $l_{\infty,1}$-based Sparse Representations

Computing sparse representations induced by a weighted $l_{\infty,1}$ norm requires to solve:

$$\alpha_y = \arg\min_{\alpha} \frac{1}{2} ||y - D\alpha||_2^2 + \lambda ||\alpha||^w_{\infty,1} \,. \tag{12}$$

This is a convex and non-differentiable problem, which can be efficiently solved using an accelerated proximal gradient descent (APG or FISTA) algorithm described by Algorithm 1. This method achieves a global optimization with a quadratic rate of convergence [31] but relies on a proximal operator defined by:

$$prox_{\lambda||.||^w_{\infty,1}}(u) = \arg\min_{v} \frac{1}{2} ||u - v||_2^2 + \lambda ||v||^w_{\infty,1} \,. \tag{13}$$

input: D, y, w
$k = 1$, $\alpha_{k-1} = \alpha_k = 0$;
repeat
  $\mu_k = \frac{k}{k+3}$;
  $\beta = \alpha_k + \mu_k(\alpha_k - \alpha_{k-1})$;
  find ρ by line search [31];
  $\gamma = \beta - \rho D^\top (D\beta - y)$;
  $\alpha_{k+1} = prox_{\rho\lambda||.||^w_{\infty,1}}(\gamma)$;
  $k = k + 1$;
until *convergence*;
return α_k;

Algorithm 1. FISTA optimization for $l_{\infty,1}$-based sparse representation.

input: D, y, w
$\mathcal{A} = \varnothing$, $\alpha_{\mathcal{A}} = 0$;
repeat
  $\mathcal{S} = \mathcal{S}(\alpha_{\mathcal{A}})$;
  using $\alpha_{\mathcal{A}}$ as a warm start, find the optimal solution $\alpha_{\mathcal{A}\cup\mathcal{S}}$ of the problem Eq. 12 restricted to $\mathcal{A} \cup \mathcal{S}$;
  $\mathcal{A} = \mathcal{A} \cup \mathcal{S}$;
until $||D^\top(D\alpha_{\mathcal{A}} - y)||^{1/w}_{1,\infty} \leq \lambda$;
return $\alpha_{\mathcal{A}}$;

Algorithm 2. Active set strategy for $l_{\infty,1}$-based sparse representation.

When Ω is a norm, its proximal can be derived from a Euclidean projection on the unit ball of its dual norm Ω^* [31]:

$$prox_{\lambda\Omega}(u) = u - \lambda\Pi_{\Omega^*\leq 1}(u/\lambda)\,. \tag{14}$$

In fact, the dual norm of the $l_{\infty,1}$ norm is exactly the $l_{1,\infty}$ norm. In the case of a weighted $l_{\infty,1}$ norm, the dual norm is also a weighted $l_{1,\infty}$ norm (see supplementary material for detail):

$$||\alpha||^w_{\infty,1}{}^* = ||\alpha||^{1/w}_{1,\infty} = \sum_{i=1..\Delta t} \frac{1}{w_i}||\alpha^{G_i}||_\infty\,. \tag{15}$$

Therefore, Eq. 12 reduces to compute the Euclidean projection on the unit ball of a weighted $l_{1,\infty}$ norm:

$$prox_{\lambda||.||^w_{\infty,1}}(u) = u - \lambda\Pi_{||.||^{1/w}_{1,\infty}\leq 1}(u/\lambda)\,. \tag{16}$$

An efficient algorithm for computing Euclidean projections on the unit ball of the $l_{1,\infty}$ norm was proposed in [32] and can be easily extended to handle the case of weighted $l_{1,\infty}$ norms. We use the implementation given on the authors' website to compute those projections for the proximal operators.

This optimization process can be sped up by using an active set strategy as explained in [33]. A necessary condition, based on the dual norm, for a representation α to be an optimal solution of Eq. 12 is:

$$||D^\top(D\alpha - y)||^{1/w}_{1,\infty} = \sum_{i=1..\Delta t} \frac{1}{w_i}||D_{G_i}{}^\top(D\alpha - y)||_\infty \leq \lambda. \tag{17}$$

An active set strategy optimizes Eq. 12 on a small set of active variables $\mathcal{A}$, yielding a solution $\alpha_{\mathcal{A}}$, and makes it progressively grow by adding a set of non-active variables $\mathcal{S}(\alpha_{\mathcal{A}})$ until the condition Eq. 17 is satisfied. This process, described in Algorithm 2, yields a global solution of Eq. 12 [33]. In practice, we set $\mathcal{S}(\alpha_{\mathcal{A}})$ to K non-active variables that have the highest $|d_i^\top(D\alpha_{\mathcal{A}} - y)|$ value with at most one variable by group to avoid focusing on a single one.

6 Evaluations and Discussion

6.1 Benchmarks, Metrics, and Parameter Tuning

We use the *MOTChallenge* benchmarks, *2DMOT2015* and *MOT16* [34,35], to evaluate the performances of the proposed approach. These benchmarks are composed of training and testing sets [36–41] with public detections, given by Aggregate Channel Features (ACF) pedestrian detector [42] in the case of the *2DMOT2015* and a Deformable Part Model (DPM) [43] for the *MOT16*.

The metrics employed by these benchmarks are based on the widely accepted CLEARMOT metrics [44]. MOT accuracy (MOTA) takes jointly into account false positives (FP), false negatives (FN) and identity switches (IDS). MOT precision (MOTP) measures the overlap distance between the found pedestrians' locations and the ground truth. We also indicate track fragmentations (FM), false alarms by frame (FAF) and the mostly tracked and mostly lost targets percentages (MT and ML). Furthermore, we report the IDS ratio (IR), defined by $\frac{IDS}{Recall}$, to measure the IDS more independently from the false negatives (FN).

As our method depends on several parameters, notably in the formulation of the energy E, manual tuning of these free parameters on the training set is out of reach. We use a hyper-optimization procedure (see the public implementation of [45]) to explore efficiently the space of parameters within 1000 runs of our algorithm. Thus, we automatically find the best set of parameters by optimizing the MOTA value, which is the main metric used to compare MOT approaches.

6.2 Comparison with $l1$ Norm and Basic Appearance Models

To validate our approach with $l_{\infty,1}$-based sparse representations, we implement three variants that only differ by considering different appearance models $App(C)$. We denote by LINF1 the proposed approach using $App(C)$ defined by Eq. 10 with $l_{\infty,1}$-based sparse representations. A first variant, called L1, uses the model $App(C)$ defined by Eq. 10 with l_1-based sparse representations to demonstrate the effectiveness of the weighted $l_{\infty,1}$ norm compared to the l_1 norm.

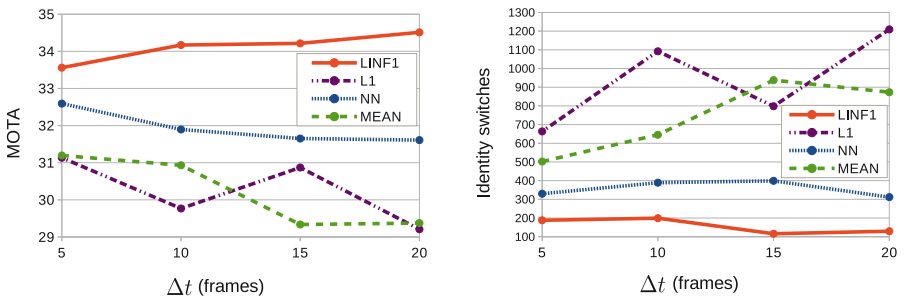

Fig. 4. MOTA (best higher↑) and identity switches (best lower↓) of our approach (LINF1) and other appearance models for different windows of size Δt frames, evaluated on the *2DMOT2015* training set.

Two variants without sparse representations are also evaluated to verify that using appropriate sparse representations effectively increases performances compared to more basic appearance models. These two baselines, denoted by NN and $MEAN$, differ from the proposed approach by respectively using the appearance models $App_{NN}(C)$ and $App_{MEAN}(C)$ defined by:

$$App_{NN}(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - NN_\tau(y_d)||_2 \,, \tag{18}$$

$$App_{MEAN}(C) = \sum_{\tau \in C} \sum_{d \in \tau} ||y_d - y_\tau||_2 \,, \tag{19}$$

where $NN_\tau(y_d)$ stands for the nearest neighbor of y_d among the features of the other detections in track τ, and y_τ stands for the mean of the features of all detections in τ.

6.3 Comparison Between the Proposed Variants

Our approach and the three variants described previously are compared on the *2DMOT2015* training set, with sliding windows of size $\Delta t \in \{5, 10, 15, 20\}$. Similarly to [21–24], we do not use any complex features and simply use for y_d color intensity values of the templates resized to 32×32. To fairly compare these variants, we use for each variant and Δt value the hyper-optimization procedure discussed previously to find the best set of parameters.

MOTA values and IDS are indicated in Fig. 4. First of all, they show that the proposed LINF1 variant outperforms the other variants both in terms of MOTA

Table 1. Results of our approach, with windows of Δt frames, on the *2DMOT2015* training set (best values in bold and red, second best ones underlined in blue).

Method	Δt	MOTA (%)↑	IDS ↓	IR ↓	FM ↓	FAF ↓	MOTP (%) ↑	FP ↓	FN ↓	MT (%) ↑	ML (%) ↓
LINF1	5	33.6	188	4.4	413	**0.6**	**72.8**	**3346**	22980	16.5	57.8
	10	34.2	199	4.5	444	0.7	72.6	3740	22330	17.8	59.7
	15	34.2	**116**	**2.6**	410	0.7	72.7	3829	22307	18.4	58.8
	20	**34.5**	129	2.9	**385**	0.7	**72.8**	3931	22073	18.0	57.1
	25	34.1	163	3.3	470	1.1	72.4	6200	**19942**	**22.2**	**50.6**
	30	33.8	155	3.3	446	1.0	72.6	5260	20997	20.8	53.2
	35	33.6	141	3.0	446	1.0	72.7	5455	20909	20.4	51.4

Table 2. Best parameter set for LINF1, with a sliding window of 20 frames, found on the *2DMOT2015* and *MOT16* training set using a hyper-optimisation procedure.

Benchmark	θ_{Ob}	θ_{App}	θ_{Mot}	θ_{Int}	α_{Ob}	β_{Ob}	γ_{Ob}	σ	λ	N_c	s_c	δt_l	d_l	h_l	N_{tr}
2DMOT2015	0.50	0.39	0.77	0.08	0.60	0.004	0.99	0.14	3.1	5	29	6	0.28	0.24	11
MOT16	0.33	0.40	0.77	0.41	1.3	0.99	1.3	0.15	7.7	5	0.13	18	0.29	0.30	18

and IDS. L1 variant performs poorly in our multi-frame data association context, especially concerning IDS. When using these representations, each detection is represented by only a few similar detections. It leads to promote short tracks of highly similar detections rather than long tracks through the whole sliding window. The two other appearance models, App_{NN} and App_{MEAN}, yield more acceptable results. However, they rapidly deteriorate in performance when the number of frames in the sliding window increases.

The proposed approach, LINF1, is the only one able to leverage a larger sliding window, gaining slightly in MOTA and track fragmentations, and reducing more significantly the number of IDS (cf. Table 1). As it promotes representations where each frame is involved, even distant ones, the $l_{\infty,1}$ norm succeeds in efficiently exploiting the additional information provided by larger sliding windows. The optimal sliding window size is about 20 frames and the performances deteriorate slightly for larger windows. The search space for the MCMCDA is rapidly growing with the sliding window size, making the optimization more difficult, which possibly explains this slight decrease in performances.

6.4 Evaluations on the MOTChallenge Benchmarks

The results of the proposed LINF1 approach on the *2DMOT2015* test dataset are shown in Table 3 with all the other published methods that use the public detections given by the benchmark. Following the benchmark policy, we use the best set of parameters found on the training set, as indicated in Table 2.

In terms of MOTA, our method is superior or comparable to most of recent approaches. However, our method distinguishes itself by achieving *the smallest number of IDS* on the benchmark. This indicates a greater ability in discriminating similar targets, especially compared to methods achieving a similar or greater false negative number (FN) as increasing this number can naturally lead to decrease the number of IDS. IDS ratios (IR) can be considered to compare IDS more independently of the false negative number, and the proposed method is still *the best one in terms of IDS ratios*. Our approach is also *the first one in terms of false alarm by frame (FAF) and false positive (FP)*, and is *the second one in terms of track fragmentations (FM)*. The proposed method yields very confident results due to the use of $l_{\infty,1}$-based sparse representations. Indeed, these representations are still sparse over the elements of a same frame and thus exhibit a high discriminative power to differentiate the targets, leading to a small number of IDS. Furthermore, inducing a sparsity structure that promotes the participation of all the frames creates more links between temporally distant elements and helps handle occlusions or gaps between detections, reducing again the number of IDS and track fragmentations. Our approach is therefore well-suited for applications where the precision is a more important concern than the recall and where maintaining the identities of the targets is a crucial need.

Our method can process the *2DMOT15* benchmark around 7.5 fps using a 8 cores CPU at 2.7 GHz, running near real-time. Some results are shown in Fig. 5, and entire trajectories are visible on the benchmark website.

Table 3. Results for LINF1 on the test set of the *2DMOT2015* and *MOT16* benchmarks (accessed on 14/03/2016), compared to other recent state-of-the-art methods (best values in bold and red, second best ones underlined in blue). Third column: method type with O standing for online, G for global and S for sliding window.

2DMOT2015			MOTA	IDS	IR	FM	FAF	MOTP	FP	FN	MT	ML
Method	Ref.	T.	(%) ↑	↓	↓	↓	↓	(%) ↑	↓	↓	(%) ↑	(%) ↓
NOMT	[16]	S	**33.7**	442	9.4	823	1.3	**71.9**	7762	32547	12.2	44.0
MHT_DAM	[17]	S	32.4	435	9.1	826	1.6	71.8	9064	**32060**	**16.0**	43.8
MDP	[1]	O	30.3	680	14	1500	1.7	71.3	9717	32422	13.0	**38.4**
LP_SSVM	[5]	G	25.2	646	16	849	1.4	71.7	8369	36932	5.8	53.0
ELP	[6]	G	25.0	1396	36	1804	1.3	71.2	7345	37344	7.5	43.8
LINF1	-	S	24.5	**298**	**8.6**	744	**1.0**	71.3	**5864**	40207	5.5	64.6
JPDA_m	[18]	S	23.8	365	11	869	1.1	68.2	6373	40084	5.0	58.1
MotiCon	[7]	G	23.1	1018	24	1061	1.8	70.9	10404	35844	4.7	52.0
SegTrack	[19]	S	22.5	697	19	**737**	1.4	71.7	7890	39020	5.8	63.9
DCO_X	[8]	G	19.6	521	14	819	1.8	71.4	10652	38232	5.1	54.9
CEM	[9]	G	19.3	813	19	1023	2.5	70.7	14180	34591	8.5	46.5
RMOT	[2]	O	18.6	684	17	1282	2.2	69.6	12473	36835	5.3	53.3
SMOT	[10]	G	18.2	1148	33	2132	1.5	71.2	8780	40310	2.8	54.8
ALExTR.	[46]	S	17.0	1859	53	1872	1.6	71.2	9233	39933	3.9	52.4
TBD	[11]	G	15.9	1939	45	1963	2.6	70.9	14943	34777	6.4	47.9
GSCR	[3]	O	15.8	514	18	1010	1.3	69.4	7597	43633	1.8	61.0
TC_ODAL	[4]	O	15.1	637	17	1716	2.2	70.5	12970	38538	3.2	55.8
DP_NMS	[12]	G	14.5	4537	105	3090	2.3	70.8	13171	34814	6.0	40.8
MOT16			MOTA	IDS	IR	FM	FAF	MOTP	FP	FN	MT	ML
Method	Ref.	T.	(%) ↑	↓	↓	↓	↓	(%) ↑	↓	↓	↑	↓
LINF1	-	S	**40.5**	**426**	**9.4**	953	1.4	74.9	8401	**99715**	**10.7**	56.1
DP_NMS	[12]	G	31.9	969	29	**941**	**0.2**	**76.4**	**1343**	121813	4.8	65.2
SMOT	[10]	G	29.2	3072	75	4437	3.0	75.2	17929	108041	4.9	**53.3**

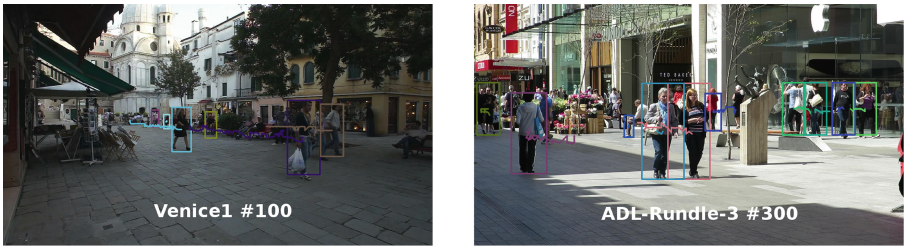

Fig. 5. Tracklets inferred by our approach on the *2DMOT2015* test set.

The results on the *MOT16* benchmark are also reported at the bottom of Table 3. As this benchmark was recently released, the results from only two other

tracking approaches are available. Our method outperforms these approaches with the best MOTA score and the lowest number of IDS.

7 Conclusion

In this paper, we have proposed a new MOT approach by combining a sparse representation-based appearance model with a sliding window tracking method. We have designed a weighted $l_{\infty,1}$ norm in order to induce a sparsity structure more suited to a MOT problem compared to the usual l_1 norm. Besides, we have proposed an efficient optimization to compute the $l_{\infty,1}$-based sparse representations using accelerated proximal gradient descent techniques. Combining $l_{\infty,1}$-based sparse representations with a sliding window approach results in a robust tracking method less prone to association errors like identity switches or track fragmentations due to its ability to efficiently correct previous association mistakes. Our method was tested on the *MOTChallenge* benchmarks, comparing well with the majority of competitors in terms of MOTA and achieving the best results in terms of identity switches and false alarms.

Several ideas developed in this paper can be extended as future work. For example, the representations are defined independently for each detection whereas one could consider computing them jointly with an appropriate penalty.

References

1. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: online multi-object tracking by decision making. In: ICCV (2015)
2. Yoon, J.H., Yang, M.H., Lim, J., Yoon, K.J.: Bayesian multi-object tracking using motion context from multiple objects. In: WACV (2015)
3. Fagot-Bouquet, L., Audigier, R., Dhome, Y., Lerasle, F.: Online multi-person tracking based on global sparse collaborative representations. In: ICIP (2015)
4. Bae, S.H., Yoon, K.J.: Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning. In: CVPR (2014)
5. Wang, S., Fowlkes, C.C.: Learning optimal parameters for multi-target tracking. In: BMVC (2015)
6. McLaughlin, N., Del Rincon, J.M., Miller, P.: Enhancing linear programming with motion modeling for multi-target tracking. In: WACV (2015)
7. Leal-Taix, L., Fenzi, M., Kuznetsova, A., Rosenhahn, B., Savarese, S.: Learning an image-based motion context for multiple people tracking. In: CVPR (2014)
8. Milan, A., Schindler, K., Roth, S.: Multi-target tracking by discrete-continuous energy minimization. TPAMI (2016). doi:10.1109/TPAMI.2015.2505309
9. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. TPAMI **36**(1), 58–72 (2014)
10. Dicle, C., Sznaier, M., Camps, O.: The way they move: tracking targets with similar appearance. In: ICCV (2013)
11. Geiger, A., Lauer, M., Wojek, C., Stiller, C., Urtasun, R.: 3D traffic scene understanding from movable platforms. TPAMI **36**(5), 1012–1025 (2014)
12. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR (2011)

13. Zamir, A.R., Dehghan, A., Shah, M.: GMCP-tracker: global multi-object tracking using generalized minimum clique graphs. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 343–356. Springer, Heidelberg (2012)
14. Dehghan, A., Assari, S.M., Shah, M.: GMMCP-tracker: globally optimal generalized maximum multi clique problem for multiple object tracking. In: CVPR (2015)
15. Brendel, W., Amer, M.R., Todorovic, S.: Multiobject tracking as maximum weight independent set. In: CVPR (2011)
16. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: ICCV (2015)
17. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: ICCV (2015)
18. Rezatofighi, S.H., Milan, A., Zhang, Z., Shi, Q., Dick, A.R., Reid, I.D.: Joint probabilistic data association revisited. In: ICCV (2015)
19. Milan, A., Leal-Taix, L., Schindler, K., Reid, I.: Joint tracking and segmentation of multiple targets. In: CVPR (2015)
20. Benfold, B., Reid, I.: Stable multi-target tracking in real-time surveillance video. In: CVPR (2011)
21. Mei, X., Ling, H.: Robust visual tracking and vehicle classification via sparse representation. TPAMI **33**(11), 2259–2272 (2011)
22. Bao, C., Wu, Y., Ling, H., Ji, H.: Real time robust L1 tracker using accelerated proximal gradient approach. In: CVPR (2012)
23. Jia, X., Lu, H., Yang, M.: Visual tracking via adaptive structural local sparse appearance model. In: CVPR (2012)
24. Zhong, W., Lu, H., Yang, M.: Robust object tracking via sparsity-based collaborative model. In: CVPR (2012)
25. Hong, Z., Mei, X., Prokhorov, D., Tao, D.: Tracking via robust multi-task multi-view joint sparse representation. In: ICCV (2013)
26. Zhang, S., Yao, H., Sun, X., Lu, X.: Sparse coding based visual tracking: review and experimental comparison. Pattern Recogn. **46**(7), 1772–1788 (2013)
27. Fagot-Bouquet, L., Audigier, R., Dhome, Y., Lerasle, F.: Collaboration and spatialization for an efficient multi-person tracking via sparse representations. In: AVSS (2015)
28. Naiel, M.A., Ahmad, M.O., Swamy, M.N.S., Wu, Y., Yang, M.: Online multi-person tracking via robust collaborative model. In: ICIP (2014)
29. Oh, S., Russell, S.J., Sastry, S.: Markov chain Monte Carlo data association for multi-target tracking. Trans. Autom. Control **54**(3), 481–497 (2009)
30. Wright, J., Yang, A., Ganesh, A., Sastry, S., Ma, Y.: Robust face recognition via sparse representation. TPAMI **31**(2), 210–227 (2009)
31. Parikh, N., Boyd, S.: Proximal algorithms. Found. Trends Optim. **1**(3), 123–231 (2013)
32. Quattoni, A., Carreras, X., Collins, M., Darrell, T.: An efficient projection for l1, infinity regularization. In: ICML (2009)
33. Bach, F., Jenatton, R., Mairal, J., Obozinski, G.: Optimization with sparsity-inducing penalties. Found. Trends Mach. Learn. **4**(1), 1–106 (2012)
34. Leal-Taixé, L., Milan, A., Reid, I., Roth, S., Schindler, K.: MOTChallenge 2015: Towards a Benchmark for Multi-Target Tracking. arXiv:1504.01942 [cs] (2015)
35. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: MOT16: A Benchmark for Multi-Object Tracking. arXiv:1603.00831 [cs] (2016)
36. Ess, A., Leibe, B., Gool, L.V.: Depth and appearance for mobile scene analysis. In: ICCV (2007)

37. Andriluka, M., Roth, S., Schiele, B.: Monocular 3D pose estimation and tracking by detection. In: CVPR (2010)
38. Andriluka, M., Roth, S., Schiele, B.: People-tracking-by-detection and people-detection-by-tracking. In: CVPR (2008)
39. Ferryman, J., Shahrokni, A.: Pets 2009: dataset and challenge. In: Performance Evaluation of Tracking and Surveillance (PETS-Winter) (2009)
40. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? The KITTI vision benchmark suite. In: CVPR (2012)
41. Benfold, B., Reid, I.: Guiding visual surveillance by tracking human attention. In: BMVC (2009)
42. Dollar, P., Appel, R., Belongie, S., Perona, P.: Fast feature pyramids for object detection. TPAMI **36**(8), 1532–1545 (2014)
43. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. TPAMI **32**(9), 1627–1645 (2010)
44. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **2008**(1), 1–10 (2008). doi:10.1155/2008/246309
45. Hutter, F., Hoos, H.H., Leyton-Brown, K.: Sequential model-based optimization for general algorithm configuration. In: Coello, C.A.C. (ed.) LION 2011. LNCS, vol. 6683, pp. 507–523. Springer, Heidelberg (2011)
46. Bewley, A., Ott, L., Ramos, F., Upcroft, B.: ALExTRAC: affinity learning by exploring temporal reinforcement within association chains. In: ICRA (2016)