HAL
archives-ouvertes.fr

ONLINE MULTI-PERSON TRACKING BASED ON GLOBAL SPARSE COLLABORATIVE REPRESENTATIONS

Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle

▶ **To cite this version:**

Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, Frédéric Lerasle. ONLINE MULTI-PERSON TRACKING BASED ON GLOBAL SPARSE COLLABORATIVE REPRESENTATIONS. International Conference on Image Processing, Sep 2015, Québec, Canada. hal-01763151

HAL Id: hal-01763151
https://hal.laas.fr/hal-01763151

Submitted on 10 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ONLINE MULTI-PERSON TRACKING BASED ON GLOBAL SPARSE COLLABORATIVE REPRESENTATIONS

Loïc Fagot-Bouquet[a], Romaric Audigier[a], Yoann Dhome[a], Frédéric Lerasle[b,c]

[a]CEA, LIST, Vision and Content Engineering Laboratory,
Point Courrier 173, F-91191 Gif-sur-Yvette, France
[b]CNRS, LAAS, 7, Avenue du Colonel Roche, F-31400 Toulouse, France
[c]Univ de Toulouse, UPS, LAAS, F-31400 Toulouse, France
loic.fagot-bouquet@cea.fr

ABSTRACT

Multi-person tracking is still a challenging problem due to recurrent occlusion, pose variation and similar appearances between people. Inspired by the success of sparse representations in single object tracking and face recognition, we propose in this paper an online tracking by detection framework based on collaborative sparse representations. We argue that collaborative representations can better differentiate people compared to target-specific models and therefore help to produce a more robust tracking system. We also show that despite the size of the dictionaries involved, these representations can be efficiently computed with large-scale optimization techniques to get a near real-time algorithm. Experiments show that the proposed approach compares well to other recent online tracking systems on various datasets.

***Index Terms*—** Online tracking, multi-person tracking, sparsity, collaborative representations.

1. INTRODUCTION

Object tracking is an important topic of interest in Computer Vision since it is needed for many practical applications. Existing techniques can be divided into single object tracking (SOT) or multi-object tracking approaches (MOT), and despite a lot of improvements in these fields it remains a challenging problem to design an efficient multi-object tracker due to similar appearances and occlusions of the targets.

Multi-object tracking can be performed either offline [1, 2], using past and future frames usually in a batch setting, or online [3, 4, 5, 6]. Online algorithms are more suitable for time critical applications and remain competitive with offline methods, as shown by some papers [7, 8]. Recent online approaches mainly rely on the tracking-by-detection scheme, in which an offline learned detector yields target locations in each frame. Detections from the current frame are then associated to existing tracks to reconstruct the trajectories across time. Most often, target-specific models are learned online and used to evaluate an appearance score for a track-detection pair, and recent works have proposed more complex and robust models to differentiate similar objects [5, 7, 8].

Employing more sophisticated appearance models has also led to significant improvements in SOT over the past years. In particular, sparse representations have been used to produce generative and discriminative appearance models [9, 10, 11]. Even though the first trackers based on sparse representations could not fulfill real time constraints, this problem has been handled with compressed sensing techniques as shown in [12]. Inspired by these methods, sparse representations have been used recently in MOT for designing target-specific models [3].

Sparse representations have also been successfully employed earlier in Computer Vision for face recognition. In [13], a sparse representation based classification considers a collaborative representation among all classes and assigns the represented sample to the class which achieves the smallest residual error. This technique achieved striking results at the time and has led to many extensions, involving especially dictionary learning methods. It has also been shown in [14] that the collaborative property of the representations has an essential role in the discriminative ability of this approach.

We propose in this paper an online multi-person tracking algorithm based on collaborative sparse representations between individuals. We argue that collaborative representations can better differentiate people compared to target-specific representations (as done in [3]) without any complex features and help to reduce the number of false associations between detections and tracks. Even though it implies computing sparse representations on large dictionaries, we will show that it can be rapidly and efficiently approximated with large-scale optimization techniques involving active sets as used for example in [15], leading to a near real-time method.

This paper is organized as follows. In Sections 2 and 3 we introduce respectively the proposed tracking approach and the optimization process. Experimental results are then provided in Section 4 while Section 5 concludes our paper.

2. PROPOSED APPROACH

2.1. System overview

At each frame, a person detector yields a set of detections $\mathcal{D}^t$ which are linked to existing tracks, thanks to specific appearance and motion models, or used to initialize new tracks. Tracks that have a high association rate are considered confident, and those with a low rate are declared lost and are later terminated. The tracker management is inspired from [4].

In order to associate detections to tracks, we compute an affinity matrix A where A_{ij} stands for the affinity score between the i^{th} track and the j^{th} detection in $\mathcal{D}^t$. The association between detections and tracks is then formulated as a maximum matching problem in a bipartite graph, which can be solved with the Hungarian or a greedy algorithm. Like [4], we first associate detections to confident tracks and match the remaining detections and tracks in a second time.

Our approach mainly differs from previous ones in the way to compute the affinity scores, as explained in the following section.

2.2. Affinity scores

At frame t, we denote by $\mathcal{T}^t = \{T_1, ..., T_k\}$ the set of all existing tracks and by $\mathcal{D}^t = \{d_1, ..., d_l\}$ the set of detections given by the pedestrian detector. y_d stands for the feature vector related to a detection d. The affinity score A_{ij} between T_i and d_j is given by

$$A_{ij} = \begin{cases} a(i,j) & if \ \ (T_i, d_j) \in \mathcal{L} \\ -\infty & otherwise \end{cases}$$

where $a(i,j)$ is an appearance based term and $\mathcal{L}$ includes all track-detection pairs (T_i, d_j) that can be linked together by considering two criteria, one based on the distance between d_j and the estimated location of T_i in the current frame, and the second one based on their shapes. $\mathcal{L}$ is defined as

$$\mathcal{L} = \{(T_i, d_j), \ \ dist_{T_i,d_j} < R_i \ \ et \ \ \frac{|h_i - h_j|}{h_i} < S_i\}$$

where $dist_{T_i,d_j}$ is the Euclidean distance between T_i and d_j, and h_j (resp. h_i) is the height related to d_j (resp. T_i). The values R_i and S_i are estimated for each track and are increasing when one is lost in order to allow a wider search area.

Sparse representations are used to estimate the $a(i,j)$ values. Specifically, we denote by D_i the template dictionary related to the i^{th} track (which includes the most recent views of its associated person) and for any set $I = \{i_1, .., i_l\} \in [1..k]$, $D_I = [D_{i_1}, .., D_{i_l}]$ stands for the joint dictionary of the tracks $T_{i_1}, ..., T_{i_l}$. Each detection d_j is associated to a sparse code $\alpha_{y_{d_j}}$ defined by

$$\alpha_{y_{d_j}} = \underset{\alpha}{argmin} \ [\frac{1}{2}||y_{d_j} - D_{I_{d_j}}\alpha||_2^2 + \lambda||\alpha||_1] \qquad (1)$$

where I_{d_j} is a set of indexes specific to d_j and λ determines a trade-off between the reconstruction error $||y_{d_j} - D_{I_{d_j}}\alpha||_2^2$ and the sparsity constraint $||\alpha||_1$. The appearance affinity $a(i,j)$ is then defined as

$$a(i,j) = \begin{cases} -\frac{1}{2}||y_{d_j} - D_{I_{d_j}}\alpha^i_{y_{d_j}}||_2^2 & if \ \ i \in I_{d_j} \\ -\infty \ \ otherwise \end{cases}$$

where $\alpha^i_{y_{d_j}}$ is derived from $\alpha_{y_{d_j}}$ by setting all coordinates not related to the i^{th} track to zero, and $\frac{1}{2}||y_{d_j} - D_{I_{d_j}}\alpha^i_{y_{d_j}}||_2^2$ stands for the residual error of the i^{th} track.

2.3. Local and global collaborative representations

We have considered two possible options for designing the set I_{d_j}. The first one consists in defining $I_{d_j} = [1..k]$ which means that $\alpha_{y_{d_j}}$ involves the specific dictionaries of all existing tracks and is therefore denoted as the global sparse collaborative representation of d_j (GSC). The second option is $I_{d_j} = \{i, (T_i, d_j) \in \mathcal{L}\}$. In this setting, $\alpha_{y_{d_j}}$ is only computed using the specific dictionaries from the tracks that can be associated to d_j and this representation will be considered as the local sparse collaborative representation of d_j (LSC).

At first glance it seems useless to involve the existing tracks that cannot be associated with d_j and it also requires to optimize (1) over a much larger dictionary. However using the local representations means the affinities are defined as the residual errors of sparse codes found on unbalanced dictionaries and it is not obvious that these terms are truly comparable.

3. OPTIMIZATION

As the number of simultaneous tracks can be significant (up to 30 in some sequences), computing the global representations may require to solve the equation (1) over a large dictionary. In order to achieve a reasonable runtime, usual optimization techniques cannot be directly applied. We explain in this section how this problem can be efficiently solved using some large-scale optimization techniques.

3.1. Accelerated proximal gradient

Proximal methods have been used to solve the problem (1) and applied to several domains, especially single object tracking [16]. These techniques can be seen as a generalization of usual first order optimization methods and are of particular interest when the objective function can be expressed as the sum of two closed proper convex function f and g with f differentiable. The objective function in (1) can be decomposed this way by choosing $f(\alpha) = \frac{1}{2}||y - D\alpha||_2^2$ and $g(\alpha) = \lambda||\alpha||_1$.

The accelerated proximal gradient method is able to find the minimum of $f + g$ with a convergence rate of $O(1/i^2)$, i

being the iteration index [17]. This method requires to compute at each iteration the gradient $\bigtriangledown f(\alpha_i)$ and a few proximal values and evaluations of the function f at several points during a line search step. The main computation bottleneck is actually the computation of $\bigtriangledown f(\alpha_i)$ and the evaluations of f, each of these indeed requires $O(nm)$ operations when dictionary D has n elements of size m.

As shown in [17], one can compute the Gram matrix $D^T D$, which requires $O(n^2 m)$ operations, in order to perform each iteration in $O(n^2)$. This leads to a significant speed-up in the optimization process when $n \ll m$. However, it is not relevant in our case because the condition $n \ll m$ is not satisfied for the involved dictionaries (especially for the global representations).

3.2. Active set strategy

Some approaches have been proposed to handle the case of large dictionaries. In fact, optimality condition for (1) indicates the solution α satisfies $\alpha_i = 0$ if and only if $|D_i^T(D\alpha - y)| \leq \lambda$, where D_i stands for the i^{th} column (or element) of D. Therefore one can use an active set strategy (as done in [15]) which consists in solving (1) on a subset of elements of D and progressively adding elements to this subset.

In details, equation (1) is solved on a subset $\mathcal{A}$ of elements of D which are grouped in a dictionary $D_{\mathcal{A}}$. This yields a solution $\alpha_{\mathcal{A}}$ and then the elements of D which have the largest values $|D_i^T(y - D_{\mathcal{A}}\alpha_{\mathcal{A}})|$ above λ are added to $\mathcal{A}$. In practice we only perform a few number of iterations before the next selection step. This process converges to the optimal solution, and it is even possible to compute the Gram matrix in order to speed up the iterations when optimizing over $\mathcal{A}$ since $\mathcal{A}$ has only a few elements.

This optimization process is detailed in algorithm 1, in which K elements are added to $\mathcal{A}$ at each selection step.

Data: D, y, K
Result: α_y
$\alpha_y = \emptyset$, $\mathcal{A} = \emptyset$, $r = -y$;
repeat
  for $k = 1$ ***to*** K **do**
    $i = \underset{j \notin \mathcal{A}}{argmax} \; |D_j^T r|$;
    if $|D_i^T r| > \lambda$ **then**
      $\mathcal{A} = \mathcal{A} \cup \{i\}$;
    end
  end
  $G = D_{\mathcal{A}}^T D_{\mathcal{A}}$;
  using α_y as starting point and G find
  $\alpha_y = \underset{\alpha}{argmin} \; \frac{1}{2}||y - D_{\mathcal{A}}\alpha||_2^2 + \lambda||\alpha||_1$;
  $r = D_{\mathcal{A}}\alpha_y - y$;
until *convergence*;

Algorithm 1: Active set strategy

4. EXPERIMENTS

4.1. Implementation

Our approach is implemented in C++ and tested on a laptop computer using a single core at 2.7 GHz. We compare two variants of our approach, one using global sparse collaborative representations (GSC) and a second one using local sparse collaborative representations (LSC) in order to compute the affinity scores. We also compare a last method in which the values $a(i, j)$ in the affinity matrix are defined as the opposite of the reconstruction error of y_{d_j} over the specific dictionary D_i related to T_i. This last approach is denoted by TSS (target-specific sparse representations).

Kalman filters are used to estimate the locations of the targets in the next frame. The association of detections to tracks, formulated as a maximum matching problem, is solved with a greedy algorithm. We do not consider any complex features and directly use RGB intensity values of the templates (resized to 30x30 pixels). All parameters are empirically fixed and remained unchanged for all the sequences. Particularly, the size of the specific dictionaries is fixed to 30 elements, and the parameter λ in (1) is set to 0.1.

4.2. Experimental setting

We evaluate our approach on various datasets: PETS S2L1 and S2L2, TownCenter and Parking Lot. These sequences differ in terms of number of people, point of view and frame rate. Like [4] we do not use any 3D camera calibration. Two different sets of detections are used for the PETS and TownCenter sequences (same detections as in [4]) to show the robustness of our tracker over the employed detector. In order to yield a fair comparison we compare our approach against other state-of-the-art online trackers [3, 4, 5, 6] on the sequences for which the detections used are available.

We use the CLEARMOT metrics, detailed in [18], which are evaluated using the public implementation from [4] (with a standard overlap threshold of 0.5 instead of the unusual one used in [4]). The given trajectories from the authors' website are used when available [4, 6].

4.3. Results

The CLEARMOT metrics for our approach and some other trackers are shown in table 3, and some samples of estimated trajectories are shown in figure 2. First of all, our strategies with collaborative representations (GSC or LSC) give better results compared with target specific representation (TSS). Since the tracked people are quite similar in appearance, it seems that specific dictionaries cannot really discriminate them as shown in figure 1 where we plot the residual error of true detection-track matches and false ones. Even though using more complex features could improve the discriminative

Data	S2L1		S2L2		Town Center		Parking Lot
Opt. \ Det.	[2]	[4]	[2]	[4]	[4]	[6]	[5]
Naive (fps)	1.9	3.4	0.23	0.44	0.51	0.35	0.92
Active sets (fps)	24	29	5.1	8.7	8.4	6.5	10
Speed-up	12x	8.5x	22x	19x	16x	18x	10x

Table 1. Computation times (with given detections).

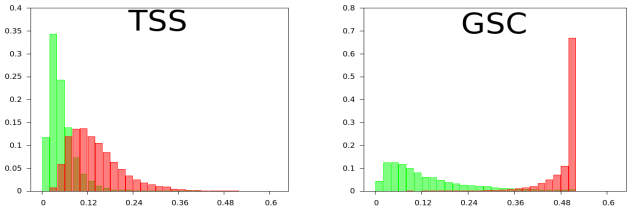

Fig. 1. Affinity costs distribution for true (in green) and false (in red) detection-track pairs on the PETS S2L2 sequence.

Metric	Method	S2L1		S2L2		Town Center		Parking Lot
		[2]	[4]	[2]	[4]	[4]	[6]	[5]
MOTA (%)	TSS	68.8	71.1	38.3	42.1	60.7	65.1	**85.7**
	LSC	**70.2**	71.2	40.5	42.7	60.6	65.5	**85.7**
	GSC	69.5	**71.3**	**41.3**	**43.9**	**61.3**	**66.1**	85.6
IDS	TSS	37	**18**	215	214	211	225	18
	LSC	29	22	**214**	210	214	210	18
	GSC	**25**	19	225	**194**	**192**	**201**	**17**

Table 2. MOTA and IDS values for the proposed approaches (best values in bold and red). Second row: detection sets used.

Data	Det.	Method	MOTA	IDS	MOTP	FP	MS
S2L1	[2]	[4]	**69.9**	35	**71.2**	805	**557**
		GSC	69.5	**25**	65.6	**757**	631
	[4]	[4]	70	21	71.7	543	**827**
		GSC	**71.3**	**19**	**73.2**	**457**	852
S2L2	[2]	[4]	**43.1**	347	**69.5**	**1318**	**4189**
		GSC	41.3	**225**	66	1502	4291
	[4]	[4]	39.3	287	69	1416	4536
		GSC	**43.9**	**194**	**71.1**	**1044**	**4514**
Town Center	[4]	[4]	60.7	212	71.2	7295	**20549**
		GSC	**61.3**	**192**	**71.6**	**3983**	23476
	[6]	[4]	63.4	446	74.5	9359	16302
		[6]	61.3	318	**80.5**	12309	**14982**
		GSC	**66.1**	**201**	74.8	**6682**	17309
Parking Lot	[5]	[3]*	84.5	**4**	73.2	-	-
		[5]*	79.3	-	**74.1**	-	-
		GSC	**85.6**	17	71.3	266	773

Table 3. CLEARMOT metrics on various sequences (best values in bold and red for MOTA and IDS). The symbol * means the associated scores have been directly reported.

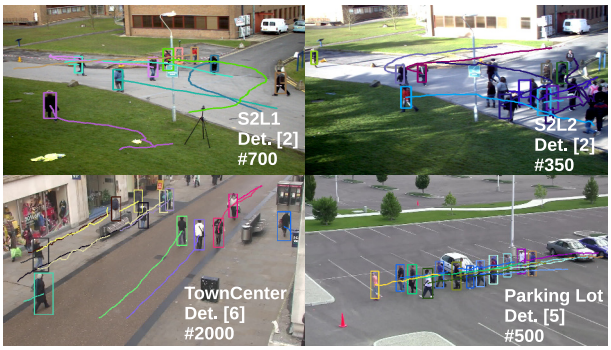

Fig. 2. Illustrations of some of our tracking results.

ability of the specific dictionaries, collaborative representations significantly improve the tracking results without any advanced features and seem to better differentiate targets.

Furthermore, global representations (GSC) yield better results than local ones (LSC) and we argue this result can be explained by the following reason. The global representations are computed at each frame on the same dictionary and therefore we avoid comparing residual errors from unbalanced dictionaries, as done with the local representations.

Our approach based on global representations (GSC) yields most often better results in terms of MOTA, and is still in any case comparable with other online trackers. Our system produces in overall far less ID-switches (IDS) thanks to the great discriminative ability of collaborative representations. Therefore, our approach is competitive to existing ones and robust with respect to the person detector.

The runtime of our approach relies mainly on the number of targets and the number of detections per frame. Computation times for the usual accelerated proximal gradient method and the one using active sets are shown in table 1 (for the GSC approach on a single core). This demonstrates that our approach can be near real-time using active sets and even be fully real-time by using a parallelized implementation.

5. CONCLUSION

We have proposed in this paper an online tracking by detection system based on global and collaborative sparse representations of all the tracked people. We have shown that these collaborative representations produce superior tracking results compared with those derived from target-specific representations and can be efficiently computed despite the possible size of the involved dictionary. An extensive evaluation of our system on several datasets confirms that this approach is robust and competitive to existing ones.

In future work, we plan to replace the sparse prior used to compute the representations by a prior more adapted to the tracking context. We also plan to explore the possibility of jointly computing the sparse representations and performing the association between tracks and detections.

6. REFERENCES

[1] X. Wang, E. Turetken, F. Fleuret, and P. Fua, "Tracking interacting objects optimally using integer programming," in *European Conference on Computer Vision (ECCV)*. IEEE, 2014, pp. 17–32.

[2] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," in *Transactions on Pattern Analysis and Machine Intelligence*. IEEE, 2014, vol. 36, pp. 58–72.

[3] M. A. Naiel, M. O. Ahmad, M.N.S. Swamy, Y. Wu, and M. Yang, "Online multi-person tracking via robust collaborative model," in *International Conference on Image Processing*. IEEE, 2014.

[4] J. Zhang, L. L. Presti, and S. Sclaroff, "Online multi-person tracking by tracker hierarchy," in *Advanced Video and Signal-Based Surveillance*. IEEE, 2012, pp. 379–385.

[5] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1815–1821.

[6] B. Benfold and I. Reid, "Stable multi-target tracking in real-time surveillance video," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2011, pp. 3457–3464.

[7] S. Bae and K. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2014, pp. 1218–1225.

[8] Z. Wu, J. Zhang, and M. Betke, "Online motion agreement tracking," in *British Machine Vision Conference*. 2013, BMVA Press.

[9] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," in *Transactions on Pattern Analysis and Machine Intelligence*. IEEE, 2011, vol. 33, pp. 2259–2272.

[10] W. Zhong, H. Lu, and M. Yang, "Robust object tracking via sparsity-based collaborative model," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012.

[11] X. Jia, H. Lu, and M. Yang, "Visual tracking via adaptive structural local sparse appearance model," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012.

[12] H. Li, C. Shen, and Q. Shi, "Real-time visual tracking using compressive sensing," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2011.

[13] J. Wright, A. Yang, A. Ganesh, S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," in *Transactions on Pattern Analysis and Machine Intelligence*. IEEE, 2009, vol. 31, pp. 210–227.

[14] L. Zhang, M. Yang, and F. Xiangchu, "Sparse representation or collaborative representation: Which helps face recognition?," in *International Conference on Computer Vision*. IEEE, 2011, pp. 471–478.

[15] Y. Mu, J. Wright, and S. Chang, "Accelerated large scale optimization by concomitant hashing," in *European Conference on Computer Vision (ECCV)*. IEEE, 2012, pp. 414–427.

[16] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust l1 tracker using accelerated proximal gradient approach," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1830–1837.

[17] N. Parikh and S. Boyd, "Proximal algorithms," in *Foundations and Trends in Optimization*, 2013.

[18] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The clear mot metrics," in *EURASIP J. Image and Video Processing*, 2008, vol. 2008.