Generic Object Crowd Tracking by Multi-Task Learning

Wenhan Luo
http://www.iis.ee.ic.ac.uk/~whluo
Tae-Kyun Kim
http://www.iis.ee.ic.ac.uk/~tkkim

Department of Electrical and Electronic Engineering, Imperial College, London, UK

Abstract

We address Multiple Object Tracking (MOT) in crowds, where the type of target objects is generic and not limited to pedestrians as in most previous work. Following the popular tracking-by-detection strategy, we decompose this problem into two main tasks, detection and tracking, and formulate them under the Multiple Task Learning (MTL) framework. A binary detector is learnt to detect objects in images, whilst multiple trackers are learnt on top of the detector by MTL to trace detected objects in subsequent frames. The detector is utilised to anchor the trackers, helping them not drift away from targets. The trackers are jointly learnt by sharing common features. To further improve the performance, we use a smoothness term which considers all labelled and unlabelled data globally. Experiments on challenging new generic object sequences as well as a publicly available sequence show that the proposed method significantly outperforms the state-of-the-art methods.

1 Introduction

Multiple Object Tracking (MOT) is an important topic in the computer vision community. It is relatively easy when objects are isolated or can be clearly distinguished from background and other objects. However, in crowd scenarios, there are frequent occlusions and interactions among objects and many objects have similar appearance, leading to confusion. A large volume of studies have tackled these challenges. Owing to the great success in object detection (especially human or pedestrian detection), most current approaches take the tracking-by-detection strategy for MOT problems, and good results are reported on some public data sets. However, existing methods for MOT mainly rely on the pedestrian detector and thus have been applied to sequences of pedestrians only, rather than sequences of general type objects.

In this paper, we propose a method for tracking multiple objects of a general type by the tracking-by-detection strategy. Similar to multiple pedestrian tracking, we need a detector which is aware of objects of a generic type, and multiple trackers which can track these discovered objects individually. From the methodological perspective, this is a problem composed of two stages. In the first stage, we treat it as a binary classification problem, which has a goal of distinguishing objects from background. In the second stage, each object

is discriminated from other objects via tracking, thus it can be considered as a multi-class classification problem.

In the aforementioned two-stage problem, each sample has two kinds of labels. For detection, the label is "object" or "background". For tracking, its label is "object ***i***" or not. This problem differs from the traditional MOT problem, where target objects (pedestrians), despite being of the same type, have quite different appearances due to e.g. clothes. In our problem, target objects are of the same type and visually more alike (see Fig. 1). Similar objects can be jointly modeled effectively, and this motivates us to formulate our problem as a Multiple Task Learning (MTL) problem.

In the MTL literature, it has been proven helpful to learn related tasks jointly rather than individually. The relevance among the tasks is typically encoded by sharing a common part of features or embedding the learners in a low rank subspace. We treat objects commonly for detection and individually for tracking in our MOT problem, and we therefore consider the problem composed of two main tasks: detection and tracking. The main tracking task is further partitioned into multiple sub-tasks, each for tracking one object. For the detection task, we train a detector to distinguish all the objects from background. For multiple sub-tasks of tracking, we train multiple trackers. In the proposed method called the Mean-Regularised Joint Feature Learning, the two main tasks are associated in the manner that the trackers are learnt not to deviate much from the mean i.e. the detector, while the multiple sub-tasks of tracking are associated by sharing common features.

Most previous methods for MOT train a detector off-line and then classify each testing sample i.e. a scan-window independently (thus locally). In contrast, contextual information such as similarities among samples can help learn a better detector. We employ the Laplacian SVM [3] which includes a smoothness constraint among all labelled and unlabelled samples at present (thus globally) for object detection. The smoothness term is also incorporated into tracking, yielding better trackers.

The main contribution of this paper is threefold:

• We consider objects of a general type rather than pedestrians for MOT in crowds. To the best of our knowledge, this is the first attempt to tackle the detection and tracking of multiple generic objects in crowds.

• The MOT problem is formulated into MTL, which is our original idea. We propose the novel Mean Regularised Joint Feature Learning method. In the method, the detection and tracking of the general type multiple objects are linked using the detector as the mean to regularise the multiple trackers. Also we learn to select sharable features among the different trackers to better relate one tracking task to the others for our MOT problem.

• We derive formulations for a linear Laplacian SVM classifier for detection. The smoothness term in the modified linear Laplacian SVM enables us to view the candidates globally. The linear classifier is easy to incorporate into the MTL framework. We also introduce a smoothness term into the multiple trackers.

2 Related Work

Generally, methods for MOT form two categories: one using information only from the previous frames, and one using information from both the past and the future frames. Methods belonging to the former category derive robust appearance models [18], delicate motion models and interaction models [32, 42] and develop a cost function considering multiple types of information up to the current frame and estimate the lowest cost state [6, 14, 24,

36, 46]. Approaches [2, 4, 8, 19] considering both the past and the future information typically require low-level observations such as foreground, tracklet, or trajectory, etc. These types of low level observations can be obtained by background modelling [35] (to acquire foreground), or by associating confident responses of a human detector, head detector or part-based detector into tracklets [7, 12, 17, 25, 33, 43, 44, 45] (this is the most popular approach since significant progress has been made in the detection field [13, 16]), or by estimating trajectories based on the KLT tracker [36] or Kalman Filter [12]. Then, these types of low level observations are associated by optimisation methods, such as Markov Chain Monte Carlo (MCMC) [35], Dynamic Programming, Hungarian algorithm [33, 43], greedy bipartite algorithm [34], network flow [41] and K-Shortest Paths (KSP) algorithm [5].

As an effective method, MTL [9] performs better than single task learning as it learns multiple related tasks simultaneously rather than independently. These tasks are related by several strategies such as Mean-Regularised MTL [15], embedded feature selection [27], low-rank subspace learning [20], clustered MTL [49], tree structured MTL [23], and graph structured MTL [11]. In [37], MTL is combined with the boosting framework to learn the features shared by multiple classes to conduct multi-class detection, avoiding constructing a specialised classifier for each class. MTL is also utilised to handle single object tracking in [47] by treating representation of multiple particles based on the collected templates as multiple tasks.

With regard to the generality of objects' types, our work is related to multi-class object detection [29, 37, 40] to some extent. However, for multi-class object detection, the classes of objects are known in advance and there are sufficient training samples to train good classifiers. In our case, we do not know the type of objects and we can only collect training data online. The most relevant work to ours for tracking multiple objects of a general type is [48]. It employs the detector in the Tracking-Learning-Detection (TLD) framework [21] to detect similar objects, but it still focuses on pedestrian tracking.

3 Approach

3.1 Overview

In our approach, the tracking-by-detection strategy is employed for multiple object tracking in crowds. Given the initial bounding box of an arbitrary target object, we train a classifier to discriminate all target objects from background. For each of the detected objects, we form a tracker using the corresponding object as a positive sample, and other objects around it and random background patches as negative samples. Then we use the trackers to follow those objects in the subsequent frames respectively. After processing every few frames, we select the objects which are tracked confidently to retrain the detector. When the detector is aware of new objects or disappearance of existing objects, we form new trackers or delete the existing trackers. Fig. 1 illustrates the overview of our approach.

3.2 Generic Detector

For detection, we generate candidates using the sliding window strategy [38]. Like the previous work, we can reject most of the candidates confidently. However, unlike pedestrian detection, we do not know the type of objects, thus we are not given deliberately designed features with a high false positive rate and a low false negative rate in advance. To tackle this

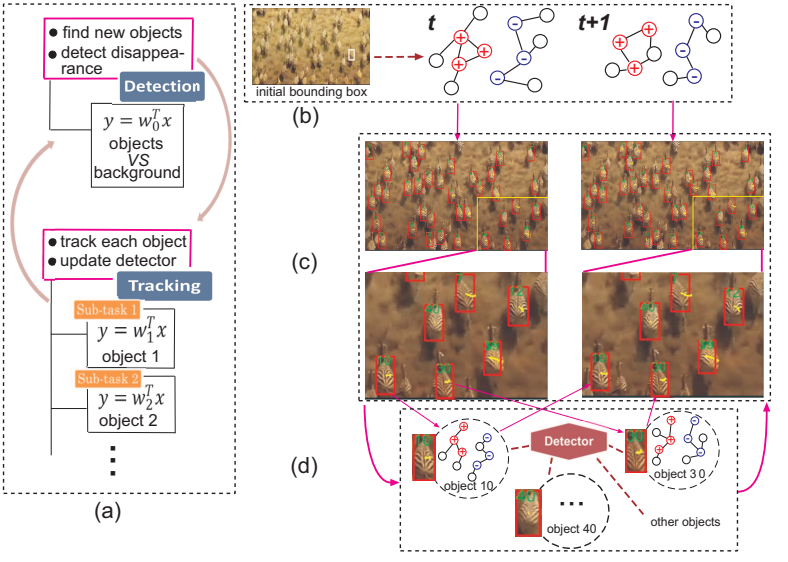

Figure 1: An overview of our approach. (a) Problem decomposition within the MTL framework. (b) Detection by the linear Laplacian SVM (given one initial bounding box). We show graphs of two continuous frames here. (c) Tracking results of continuous two frames. We zoom in a part of the image to give a clear view. (d) Tracking by the detector regularised multiple trackers (see Section 3.3). Each object is associated with a graph. The dotted line between each tracker and the detector indicates their association. This figure is best viewed in colour.

problem, we employ some efficient criterions in the following to measure the objectness of candidate windows and then reject candidates which are not likely to be objects.

Region Variance. This criterion computes the variance of the pixels within a candidate region as $RV = \frac{1}{N_R - 1}\sum_i (g_i - \bar{g})$, where N_R is the number of pixels in the region, g_i is the gray intensity of pixel i and $\bar{g}$ is the mean intensity of all the pixels in this region. This criterion can reject some candidates from the background such as grass or sky.

Edge Density. This criterion calculates the density of edge pixels within a region as $ED = \frac{1}{N_R}\sum_i 1\{i \in \mathcal{E}_\mathcal{R}\}$, where $1\{\cdot\}$ is the indicator function, $\mathcal{E}_\mathcal{R}$ is the set of pixels which belong to edge. This criterion helps to reject candidates which are too smooth. Note that in [1] the Edge Density is also a cue to measure the objectness, but here we use different methods to calculate the edge density.

Colour Contrast. We borrow the Colour Contrast cue $CC(\theta_{CC}) = \chi(h_{Region}, h_{Surr(\theta_{CC})})$ in [1] to measure the objectness of a window. h_{Region} is the colour histogram of the region and $h_{Surr(\theta_{CC})}$ is the colour histogram of the surrounding of the region (θ_{CC} measures how large the surrounding is), and $\chi(\cdot,\cdot)$ is the chi-square distance function. Although this criterion is used in [1] that there is only one object in the image scene, it is also helpful to reject some windows in our case.

Typically the number of sliding windows is greater than $30,000$, and the number of windows survived from these three rejecters is about 1000. This enables us to adopt an elaborate detector. We treat the survived windows as unlabelled samples and write them as $\mathbf{X}_u = [\mathbf{x}_1, ..., \mathbf{x}_{n_u}]$, where $\mathbf{x}_i \in \mathbb{R}^d$, d is the dimension of the feature space. As we have

been given an initial bounding box as a target object, we augment the positive sample set by adding some slight disturbance to it. At the same time, we sample instances in a further distance (between r_1 and r_2) as negative data. The corresponding labels of them are $y_i \in \{1,-1\}, i = 1,\cdots,n_l$, where $y_i = 1$ means x_i is object and $y_i = -1$ corresponds to non-object (background). Along with the unlabelled candidates, we write all the n samples as $\mathbf{X} = [\mathbf{X}_l,\ \mathbf{X}_u] \in \mathbb{R}^{d\times n}$.

Let us define the detector as $f(\mathbf{x}) = \mathbf{w}_0^T\mathbf{x}$, where $\mathbf{w}_0 \in \mathbb{R}^d$. To tackle the detection problem, we minimise the following objective function:

$$\mathcal{L}_p = \gamma_1\|\mathbf{w}_0\|^2 + \gamma_2\mathbf{w}_0^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{w}_0 + \gamma_3\sum_{i=1}^{n_l}[1 - y_i f(\mathbf{x}_i)] \tag{1}$$

In Eq. 1, the first term is the regularisation of the classifier to improve the generalisation ability, the second term is the smoothness among all the samples and the third term is the fitting error of the labelled samples. $\mathbf{L}$ is the Laplacian matrix calculated from the graph constructed based on all the samples. It is notable that this objective function has the same form as Laplacian SVM [3]. However, here we modify the original Laplacian SVM to the linear case.

Introducing the slack variables ε_i we have the primal problem as:

$$\begin{aligned} \min_{\mathbf{w}_0,\varepsilon_i} \quad & \gamma_1\|\mathbf{w}_0\|^2 + \gamma_2\mathbf{w}_0^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{w}_0 + \gamma_3\sum_{i=1}^{n_l}\varepsilon_i \\ s.t. \quad & y_i\mathbf{w}_0^T\mathbf{x} \geq 1 - \varepsilon_i,\ i = 1,2,...,n_l \\ & \varepsilon_i \geq 0,\ i = 1,2,...,n_l \end{aligned} \tag{2}$$

Following the primal-dual formulation, we have:

$$\begin{aligned} \max_{\alpha\in\mathbb{R}^{n_l}} \quad & \sum_{i=1}^{n_l}\alpha_i - \frac{1}{2}\alpha^T\mathbf{Q}\alpha \\ s.t. \quad & 0 \leq \alpha_i \leq \gamma_3,\ i = 1,2,...,n_l \end{aligned} \tag{3}$$

where $\mathbf{Q} = \mathbf{Y}^T\mathbf{J}^T\mathbf{X}^T(2\gamma_1\mathbf{I} + 2\gamma_2\mathbf{X}\mathbf{L}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{J}\mathbf{Y}$, $\mathbf{J} = [\mathbf{I}\ \mathbf{0}]^T$ is a $n \times n_l$ matrix with $\mathbf{I}$ as the $n_l \times n_l$ identity matrix, $\mathbf{Y} = diag(y_1,...,y_{n_l}) \in \mathbb{R}^{n_l\times n_l}$ and $\alpha = [\alpha_1,...,\alpha_{n_l}]^T \in \mathbb{R}^{n_l}$ are Lagrangian multipliers.

This problem is a typical quadratic optimisation problem which can be solved by standard optimisation software. After α is obtained, we can acquire $\mathbf{w}_0$ in Eq. 4. For more details, please refer to [3].

$$\mathbf{w}_0 = (2\gamma_1\mathbf{I} + 2\gamma_2\mathbf{X}\mathbf{L}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{J}\mathbf{Y}\alpha \tag{4}$$

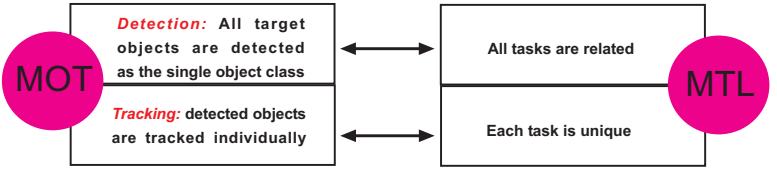

Figure 2: Formulation of the MOT problem into MTL. This figure is best viewed in colour.

3.3 Detector Regularised Trackers

As mentioned before, we maintain an individual tracker for each object. However, they are objects of the same type, which is confirmed by the detector. From this perspective, our MOT problem can be naturally formulated within the MTL framework. All the tasks in MTL are related, while all the objects are treated as the same type of objects in the detection stage. All the tasks in MTL are different from each other, while we treat objects differently when tracking them. Fig. 2 illustrates how the MOT and MTL problems are inherently linked.

Based on the above inspiration, we treat detection and tracking of multiple objects as two main tasks, and tracking of each object as a sub-task within the MTL framework. We denote the tracker for object t as $f_t(\mathbf{x}) = \mathbf{w}_t^T\mathbf{x}$. To relate the two main tasks, we penalise the deviation of each tracker from the detector $\mathbf{w}_0$ using the cost function as the following,

$$\sum_{t=1}^{T} \|\mathbf{w}_t - \mathbf{w}_0\|^2 \tag{5}$$

This regularisation term benefits the trackers in two aspects. Firstly, as $\|\mathbf{w}_t\|^2 = \|\mathbf{w}_t - \mathbf{w}_0 + \mathbf{w}_0\|^2 \leq \|\mathbf{w}_t - \mathbf{w}_0\|^2 + \|\mathbf{w}_0\|^2$, and we have minimised $\|\mathbf{w}_0\|^2$ in the detection stage, thus minimising $\|\mathbf{w}_t - \mathbf{w}_0\|^2$ equals minimising $\|\mathbf{w}_t\|^2$, further improving the generalisation ability of each tracker. Secondly, this term can prevent trackers from drifting to the background as we enforce each tracker to be close to the detector.

Furthermore, we encode the relatedness of multiple sub-tasks by learning the features jointly shared by all the trackers via a regularisation term $\|\mathbf{W}\|_{2,1}$, where $\mathbf{W} \in \mathbb{R}^{d\times T}$ is the matrix composed of all the trackers as $[\mathbf{w}_1, \cdots, \mathbf{w}_T]$. $\|\mathbf{W}\|_{2,1}$ is the $\ell_{2,1}$ norm of $\mathbf{W}$ which first computes the ℓ_2 norm of each row to obtain a column vector, then computes the ℓ_1 norm of the column vector. This regularisation term can result in that only some rows of $\mathbf{W}$ are non-zero, which correspond to the features shared by all sub-tasks.

In addition, we introduce a smoothness term for each tracker. The smoothness term enables the tracker to view the labelled and unlabelled samples (candidates) together. It has been applied to the MTL framework by Luo *et al*. [28] to handle semi-supervised learning. Here we introduce it to gain the smoothness property of all the trackers.

We sample the nearby instances and the farther instances in the current frame as positive data and negative data respectively, and we take the surrounding samples in the next frame as unlabelled data. Having obtained training data, we propose the Mean Regularised Joint Feature Learning algorithm which minimises the following objective function:

$$\min_{\mathbf{W}\in\mathbb{R}^{d\times T}} \frac{1}{2}\sum_{t=1}^{T} \|\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t - \mathbf{Y}_t\|^2 + \rho_1\|\mathbf{W}\|_{2,1} + \frac{\rho_2}{2}\sum_{t=1}^{T}\|\mathbf{w}_t - \mathbf{w}_0\|^2 + \frac{\rho_3}{2}\sum_{t=1}^{T}\mathbf{w}_t^T\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t \tag{6}$$

where $\mathbf{X}_t \in \mathbb{R}^{d\times(n_t^l+n_t^u)}$ is the combination of n_t^l labelled samples and n_t^u unlabelled samples for a sub-task t, $\mathbf{J}_t = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$ is a $(n_t^l+n_t^u)\times(n_t^l+n_t^u)$ matrix with $\mathbf{I}$ as the $n_t^l \times n_t^l$ identity matrix. $\mathbf{Y}_t \in \mathbb{R}^{(n_t^l+n_t^u)}$ is the label vector of the task t (we give the neutral label 0 to the unlabelled data). $\mathbf{L}_t$ is the Laplacian matrix associated with the graph of the task t, and ρ_1, ρ_2, ρ_3 are the trade-off parameters. The above objective function captures the relatedness of the multiple tasks from two perspectives. One lies in the feature level, which makes the tasks share a common set of features. The other one lies in the classifier level, which encodes that all of the trackers should not be too different from the detector.

Solving Eq. 6. We adopt the Accelerated Gradient Method (AGM) [30] to solve this composite optimisation problem. Compared to the traditional gradient method, the AGM has the convergence speed of $\mathcal{O}(\frac{1}{k^2})$ (i.e. it achieves the solution with $\mathcal{O}(\frac{1}{k^2})$ residual from the optimal solution after k iterations), which is the optimal among the first order methods. For the sake of convenience, we write Eq. 6 as a combination of a smooth component $\mathcal{L}(\mathbf{W}) = \frac{1}{2}\sum_{t=1}^{T}\|\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t - \mathbf{Y}_t\|^2 + \frac{\rho_2}{2}\sum_{t=1}^{T}\|\mathbf{w}_t - \mathbf{w}_0\|^2 + \frac{\rho_3}{2}\sum_{t=1}^{T}\mathbf{w}_t^T\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t$ and a non-smooth component $\Omega(\mathbf{W}) = \rho_1\|\mathbf{W}\|_{2,1}$. The AGM here iterates by using a linear combination of previous two points as the search point, rather than the latest point in the traditional gradient method. Each AGM iteration is composed of two steps: (1) Generalised Gradient Mapping which updates $\mathbf{W}^{(k+1)}$ given the search point $\mathbf{W}_S^{(k)}$, (2) Updating the current search point $\mathbf{W}_S^{(k)}$ by combining the previous two points.

(1) Generalised Gradient Mapping: given the current search point $\mathbf{W}_S^{(k)}$, the estimation $\mathbf{W}^{(k+1)}$ can be obtained by solving Eq. 7

$$\mathbf{W}^{(k+1)} = \arg\min_{\mathbf{W}} \frac{\gamma}{2}\|\mathbf{W} - (\mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)}))\|_F^2 + \Omega(\mathbf{W}) \tag{7}$$

where γ is a step parameter and $\nabla\mathcal{L}(\mathbf{W})$ is the gradient of $\mathcal{L}(\mathbf{W})$. Each column of $\nabla\mathcal{L}(\mathbf{W})$ is:

$$\mathbf{X}_t\mathbf{J}_t(\mathbf{J}_t^T\mathbf{X}_t^T\mathbf{w}_t - \mathbf{Y}_t) + \rho_2(\mathbf{w}_t - \mathbf{w}_0) + \rho_3\mathbf{X}_t\mathbf{L}_t\mathbf{X}_t^T\mathbf{w}_t, \quad t = 1, \cdots, T \tag{8}$$

Considering the computation procedure of $\ell_{2,1}$ norm, Eq. 7 can be decoupled as d disjoint sub-problems in Eq. 9 (one for each row vector $\mathbf{W}_i$),

$$\mathbf{W}_i^{(k+1)} = \arg\min_{\mathbf{W}_i} \frac{1}{2}\|\mathbf{W}_i - \mathbf{U}_i\|_2^2 + \lambda\|\mathbf{W}_i\|_2, \quad i = 1, \cdots, d \tag{9}$$

where $\mathbf{U} = \mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)})$, $\mathbf{U}_i$ is the i*th* row of $\mathbf{U}$ and $\lambda = \rho_1/\gamma$. Following [10, 47], the solution to Eq. 9 is the following:

$$\mathbf{W}_i^{(k+1)} = \max(1 - \frac{\lambda}{\|\mathbf{U}_i\|_2}, 0)\mathbf{U}_i, \quad i = 1, \cdots, d \tag{10}$$

(2) Updating the current search point as a linear combination of the previous two points:

$$\mathbf{W}_S^{(k+1)} = (1+\alpha)\mathbf{W}^{(k+1)} - \alpha\mathbf{W}^{(k)} \tag{11}$$

where $\alpha = (t^{(k)} - 1)/t^{(k+1)}$ and $t^{(k+1)} = \frac{1}{2}(1 + \sqrt{1 + 4(t^{(k)})^2})$. The algorithm terminates when the change of the function is lower than a threshold or the number of iterations has achieved the maximum. Our Mean Regularised Joint Feature Learning algorithm is summarised in Algorithm 1. Note that we implement this algorithm based on the code from the MALSAR package [50].

After we obtain the solution $\mathbf{W}$, each column $\mathbf{w}_t$ is the tracker for each sub-task (each object). And we select the most confident candidate as the estimation of each object, i.e.,

$$\mathbf{x}_t^* = \arg\max_{\mathbf{x}\in\mathbf{X}_t^u} \mathbf{w}_t^T\mathbf{x} \tag{12}$$

where $\mathbf{X}_t^u$ is the unlabelled part of $\mathbf{X}_t$.

Algorithm 1 Mean Regularised Joint Feature Learning for MOT

Input: $\mathbf{X}_t, \mathbf{Y}_t, \mathbf{w}_0, t = 1, \cdots, T$.
Output: $\mathbf{W}$
1. *Initialisation*: each column of $\mathbf{W}^{(0)}$ and $\mathbf{W}^{(1)}$ is $\mathbf{X}_t * \mathbf{Y}_t$, $t^{(0)} = 0$, $t^{(1)} = 1$, $k = 1$,$\alpha = (t^{(0)} - 1)/t^{(1)}$, $\mathbf{W}_S^{(1)} = (1+\alpha)\mathbf{W}^{(1)} - \alpha\mathbf{W}^{(0)}$
2. *While* not converged, *do*
3. Obtain $\mathbf{U} = \mathbf{W}_S^{(k)} - \frac{1}{\gamma}\nabla\mathcal{L}(\mathbf{W}_S^{(k)})$,
4. Solve Eq. 9 via Eq. 10 to acquire $\mathbf{W}_i^{(k+1)}$, $i = 1, \cdots, d$
5. Update the search point as $\mathbf{W}_S^{(k+1)} = (1+\alpha)\mathbf{W}^{(k+1)} - \alpha\mathbf{W}^{(k)}$
6. $k \leftarrow k+1$, $t^{(k+1)} \leftarrow \frac{1}{2}(1+\sqrt{1+4(t^{(k)})^2})$, $\alpha \leftarrow (t^{(k)} - 1)/t^{(k+1)}$.
7. *End*

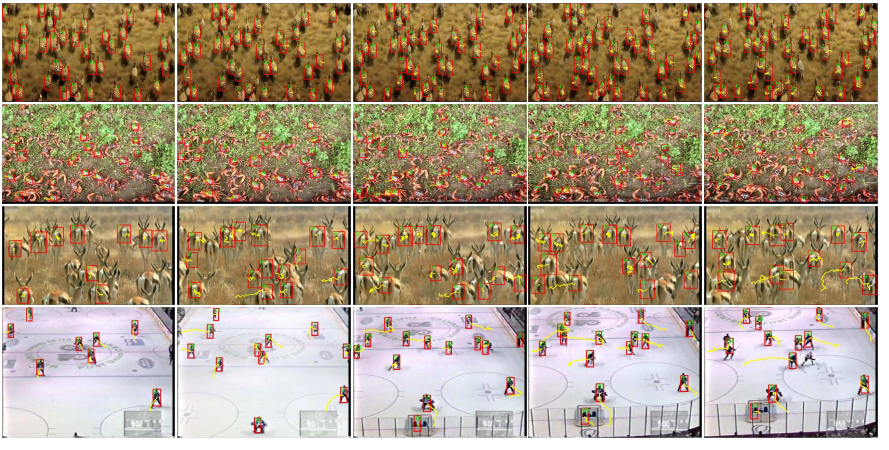

Figure 3: Images excerpted from the sequences. From top to bottom they are Zebra, Red crab, Antelope and UBC Hockey sequences. The number attached to each bounding box is the object's ID and the yellow line is its estimated trajectory. This figure is best viewed in colour.

4 Experiments

Feature. We compute HOG [13], LBP [39] and the Colour Histogram as features and concatenate these feature vectors to represent a window. The joint feature learning in our algorithm will select the useful features for MOT.

Tracking Management. At runtime, we maintain a list to save the objects. If the detector discovers a new object, we assign it a weight and buffer it. In the following once it is detected we increase its weight and once it is not detected the weight is decreased. If the weight is greater than a threshold τ, a tracker is initialised for it. For the objects existed in the list, we have the opposite process to delete objects when they disappear from the image scene.

Parameters. Here we note the setting of some parameters. For the Colour Contrast cue, we use the default parameters as in [1] except θ. As we do not have enough training examples to learn it we empirically set it as 60, and it works well on our data sets. When constructing graph, we employ the *10-NN* and the *rbf* kernel to calculate the adjacency matrix.

Sequence	MOTA ↑	MOTP ↑	Rec. ↑	Prec. ↑	MT ↑	ML ↓
Zebra (eTLD)	58.73%	64.48%	60.37%	92.14%	15.94%	42.03%
Zebra (BS1)	72.31%	58.23%	78.58%	93.10%	30.43%	34.78%
Zebra (BS2)	69.40%	**67.12%**	75.30%	93.08%	33.33%	34.78%
Zebra	**77.69%**	66.78%	**80.30%**	**97.02%**	**43.48%**	**30.43%**
Red crab (eTLD)	6.77%	**64.62%**	21.49%	58.20%	4.85%	86.41%
Red crab (BS1)	26.95%	59.39%	47.63%	69.24%	8.74%	76.70%
Red crab (BS2)	32.70%	59.40%	45.25%	77.89%	**9.71%**	74.76%
Red crab	**39.06%**	60.00%	**51.50%**	**80.65%**	**9.71%**	**70.87%**
Antelope (eTLD)	8.76%	64.98%	29.08%	57.05%	23.53%	76.47%
Antelope (BS1)	24.46%	**67.29%**	65.31%	61.75%	35.29%	39.71%
Antelope (BS2)	23.62%	66.73%	65.55%	61.27%	**38.24%**	38.24%
Antelope	**35.58%**	63.31%	**73.97%**	**65.81%**	36.76%	**36.76%**
UBC Hockey (eTLD)	54.66%	64.66%	65.04%	84.25%	17.86%	25.00%
UBC Hockey [7]	79.7%	60.0%	80.5%	98.9%	-	-
UBC Hockey [6]	76.5%	57.0%	77.7%	98.8%	-	-
UBC Hockey [31]	67.8%	51.0%	68.7%	**100%**	-	-
UBC Hockey	**80.30%**	**69.09%**	**92.37%**	89.20%	**67.86%**	**10.71%**

Table 1: Quantitative results compared with the extended TLD, our baselines (BS1, BS2) and other MOT approaches. In the metrics with the upward arrow, the greater number indicates the better performance (and vice versa for the downward arrow). For each data sequence, the last line shows our result. The best accuracies are in bold. Note that [7][6][31] do not supply the MT and ML results.

We firstly test our approach on three challenging data sets named Zebra, Red crab and Antelope respectively. There are scale changes in the Zebra sequence and there are background clutter, scale variation and rotation in the Red crab sequence. For the Antelope sequence, there exist severe occlusions and out-of-plane rotation. For comparison, we extend the TLD framework [21] for MOT. The detector of the TLD framework is based on random ferns, which can detect non-specific type objects. The extended TLD selects the detected similar objects to track. At the same time, to verify the improvement from the joint feature learning term and the smoothness term, we form another two baselines to be compared with. The first one (BS1) is formed by only keeping the fitting error term and the mean-regularised term. The second one (BS2) is formed by that we incrementally add the jointly feature learning term to BS1 (still without the smoothness term).

To evaluate the tracking performance quantitatively, we employ the Multiple Object Tracking Accuracy (MOTA) and Multiple Object Tracking Precision (MOTP) metrics proposed in [22]. MOTA considers the false positive, missed objects and ID switches. MOTP simply calculates the average overlap between the ground truth and the estimated objects. We also compute Recall and Precision, the number of Mostly Tracked (MT), and Mostly Lost (ML) trajectories [26] for further comparison. Table 1 and Fig. 3 show the results. For more results, please see our supplementary videos.

The figures in Table 1 reveal that the extended TLD performs slightly worse than our approach on the Zebra sequence, but on the other two sequences its results are much worse than ours. That is because the crabs and the antelopes do not have evident patterns like the zebras and the backgrounds of the two sequences are cluttered (see Fig. 3). It is also easy to

observe the improvement from the jointly feature learning term and the smoothness term if our results are compared with those of our two baselines.

To further illustrate that our approach is effective, we also test it on a public data set named UBC Hockey [31]. We compare our results with some MOT approaches [6, 7, 31]. The purpose of comparison with other MOT approaches is to certify that our approach can also work well on some human data sets even if we do not have an elaborate human detector.

5 Conclusion

We have shown how generic object crowd tracking is formulated into the multiple task learning problem and have proposed the novel methods. We have decomposed the problem into two main tasks and represented their relation by the proposed Mean Regularised Joint Feature Learning algorithm. The optimisation functions of these two main tasks have the terms for the generalisation ability, the smoothness, the fitting errors and feature learning. Solving the optimisation problems yields the desired list of detected and tracked objects in frames. Experimental results on the challenging data sequences have confirmed the efficacy of our approach over the state-of-the-art ones.

References

[1] B. Alexe, T. Deselaers, and V. Ferrari. What is an object? In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[2] A. Andriyenko and K. Schindler. Multi-target tracking by continuous energy minimization. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[3] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[4] B. Benfold and I. Reid. Stable multi-target tracking in real-time surveillance video. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[5] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua. Multiple object tracking using k-shortest paths optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9):1806–1819, 2011.

[6] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *International Conference on Computer Vision (ICCV)*, 2009.

[7] W. Brendel, M. Amer, and S. Todorovic. Multiobject tracking as maximum weight independent set. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[8] G. Brostow and R. Cipolla. Unsupervised bayesian detection of independent motion in crowds. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[9] R. Caruana. Multitask learning. *Machine Learning*, 28(1):41–75, 1997.

[10] X. Chen, W. Pan, J. Kwok, and J. Carbonell. Accelerated gradient method for multi-task sparse learning problem. In *International Conference on Data Mining*, 2009.

[11] X. Chen, S. Kim, Q. Lin, J. Carbonell, and E. Xing. Graph-structured multi-task regression and an efficient optimization method for general fused lasso manuscript. 2010.

[12] W. Choi and S. Savarese. Multiple target tracking in world coordinate with single, minimally calibrated camera. In *ECCV*. 2010.

[13] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

[14] G. Duan, H. Ai, S. Cao, and S. Lao. Group tracking: exploring mutual relations for multiple object tracking. In *ECCV*. 2012.

[15] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004.

[16] P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645, 2010.

[17] J.F. Henriques, R. Caseiro, and J. Batista. Globally optimal solution to multi-object tracking with merged measurements. In *International Conference on Computer Vision (ICCV)*, 2011.

[18] W. Hu, X. Li, W. Luo, X. Zhang, S. Maybank, and Z. Zhang. Single and multiple object tracking using log-euclidean riemannian subspace and block-division appearance model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012.

[19] H. Izadinia, I. Saleemi, W. Li, and M. Shah. (mp)2t: Multiple people multiple parts tracker. In *ECCV*. 2012.

[20] S. Ji and J. Ye. An accelerated gradient method for trace norm minimization. In *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM, 2009.

[21] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-learning-detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1409–1422, 2012.

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

[23] S. Kim and E. Xing. Tree-guided group lasso for multi-task regression with structured sparsity. *Proceedings of the 27th Annual International Conference on Machine Learning*, 2010.

[24] L. Kratz and K. Nishino. Tracking with local spatio-temporal motion patterns in extremely crowded scenes. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[25] C. Kuo, C. Huang, and R. Nevatia. Multi-target tracking by on-line learned discriminative appearance models. In *Computer Vision and Pattern Recognition (CVPR)*, 2010.

[26] Y. Li, C. Huang, and R. Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[27] J. Liu, S. Ji, and J. Ye. Multi-task feature learning via efficient l 2, 1-norm minimization. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2009.

[28] Y. Luo, D. Tao, B. Geng, C. Xu, and S. Maybank. Manifold regularized multi-task learning for semi-supervised multi-label image classification. *IEEE Transactions on Image Processing*, 2013.

[29] K. Mikolajczyk, B. Leibe, and B. Schiele. Multiple object class detection with a generative model. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

[30] Y. Nesterov. Gradient methods for minimizing composite objective function. core discussion papers 2007076, université catholique de louvain. *Center for Operations Research and Econometrics (CORE)*, 2007.

[31] K. Okuma, A. Taleghani, N. de Freitas, J. Little, and D. Lowe. A boosted particle filter: Multitarget detection and tracking. In *ECCV*, pages 28–39. 2004.

[32] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *International Conference on Computer Vision (ICCV)*, 2009.

[33] Z. Qin and C. Shelton. Improving multi-target tracking via social grouping. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[34] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah. Part-based multiple-person tracking with partial occlusion handling. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[35] B. Song, T. Jeng, E. Staudt, and A. Roy-Chowdhury. A stochastic graph evolution framework for robust multi-target tracking. In *ECCV*. 2010.

[36] D. Sugimura, K. Kitani, T. Okabe, Y. Sato, and A. Sugimoto. Using individuality to track individuals: Clustering individual trajectories in crowds using local appearance and frequency trait. In *International Conference on Computer Vision (ICCV)*, 2009.

[37] A. Torralba, K. Murphy, and W. Freeman. Sharing visual features for multiclass and multiview object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(5):854–869, 2007.

[38] P. Viola and M. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[39] X. Wang, T. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *International Conference on Computer Vision (ICCV)*, 2009.

[40] B. Wu and R. Nevatia. Cluster boosted tree classifier for multi-view, multi-pose object detection. In *International Conference on Computer Vision (ICCV)*, 2007.

[41] Z. Wu, A. Thangali, S. Sclaroff, and M. Betke. Coupling detection and data association for multiple object tracking. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[42] K. Yamaguchi, A. Berg, L. Ortiz, and T. Berg. Who are you with and where are you going? In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[43] B. Yang and R. Nevatia. Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[44] B. Yang and R. Nevatia. An online learned crf model for multi-target tracking. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[45] B. Yang, C. Huang, and R. Nevatia. Learning affinities and dependencies for multi-target tracking using a crf model. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

[46] M. Yang, F. Lv, W. Xu, and Y. Gong. Detection driven adaptive multi-cue integration for multiple human tracking. In *International Conference on Computer Vision (ICCV)*, 2009.

[47] T. Zhang, B. Ghanem, S. Liu, and N. Ahuja. Robust visual tracking via multi-task sparse learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

[48] X. Zhao, D. Gong, and G. Medioni. Tracking using motion patterns for very crowded scenes. In *ECCV*. 2012.

[49] J. Zhou, J. Chen, and J. Ye. Clustered multi-task learning via alternating structure optimization. *Advances in Neural Information Processing Systems*, 2011.

[50] J. Zhou, J. Chen, and J. Ye. *MALSAR: Multi-tAsk Learning via StructurAl Regularization*. Arizona State University, 2011. URL http://www.public.asu.edu/~jye02/Software/MALSAR.