Egocentric Daily Activity Recognition via Multi-task Clustering

Yan Yan, Elisa Ricci, Gaowen Liu, and Nicu Sebe, *Senior Member, IEEE*

Abstract—Recognizing human activities from videos is a fundamental research problem in computer vision. Recently, there has been a growing interest in analyzing human behavior from data collected with wearable cameras. First-person cameras continuously record several hours of their wearers' life. To cope with this vast amount of unlabeled and heterogeneous data, novel algorithmic solutions are required. In this paper, we propose a multi-task clustering framework for activity of daily living analysis from visual data gathered from wearable cameras. Our intuition is that, even if the data are not annotated, it is possible to exploit the fact that the tasks of recognizing everyday activities of multiple individuals are related, since typically people perform the same actions in similar environments (*e.g.* people working in an office often read and write documents). In our framework, rather than clustering data from different users separately, we propose to look for clustering partitions which are coherent among related tasks. Specifically, two novel multi-task clustering algorithms, derived from a common optimization problem, are introduced. Our experimental evaluation, conducted both on synthetic data and on publicly available first-person vision datasets, shows that the proposed approach outperforms several single task and multi-task learning methods.

Index Terms—Egocentric Activity Recognition, Multi-task Learning, Activity of Daily Living Analysis

1 Introduction

Research in wearable sensor-based activity recognition leverages the data automatically collected from sensors embedded into mobile devices to predict the user daily activities in real-time. RFID, GPS and accelerometers represent the most popular wearable sensors and several works [1, 2] have already proposed to exploit them for inferring people behaviors. Nowadays, wearable cameras are becoming increasingly common among consumers. Wearable cameras can be employed in many different applications, such as life-logging, ambient assisted living, personal security and drivers' assistance. It is intuitive that, while GPS and inertial sensors may suffice for detecting simple activities (*i.e.* running, walking), only by analyzing visual informations from wearable cameras more complex behaviors can be inferred.

Activity of Daily Living (ADL) analysis has attracted considerable attention in the computer vision and image processing communities [3–6]. Analyzing visual streams recorded from video surveillance cameras to automatically understand *what people do* is a challenging task [7]. It implies not only to infer the activities of a single individual, but also to recognize the environment where he/she operates, the people with whom he/she interacts, the objects he/she manipulates and even his/her future intentions. While much progress has been made in this area, recent works [8] have demonstrated that the traditional "third-person" view perspective (*i.e.* employing fixed cameras monitoring the user environment) may be insufficient for recognizing user activities and intentions and that wearable cameras provide a valid alternative.

In this paper, we consider the problem of ADL analysis from a first-person vision (FPV) perspective. Among the many challenges arising in this context, one particular issue is related to the fact that wearable cameras are intended to record the entire life of a person. Thus, a huge amount of visual data is automatically generated. Moreover, labels are usually not available since the annotation would require a massive human effort. As a consequence, algorithms which are both scalable and able to operate in an unsupervised setting are required. To face these challenges, we propose to cast the problem of egocentric daily activity recognition within a Multi-Task Learning (MTL) framework. When considering the tasks of inferring everyday activities of several individuals, it is natural to assume that these tasks are related. For example, people working in an office environment typically perform the same activities (*e.g.* working in front of a personal computer, reading and writing documents). Similarly, people at home in the morning usually make breakfast and brush their teeth. In this paper we argue that, when performing activity recognition, learning from data of several targets simultaneously is advantageous with respect to considering each person separately. For example, if there are limited data for a single person, typical clustering methods may fail to discover the correct clusters and leveraging auxiliary sources of information (*e.g.* data from other people) may improve the performance. However,

- *Y. Yan is with with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy, and Advanced Digital Sciences Center, UIUC, Singapore. (E-mail: yan@disi.unitn.it)*
- *G. Liu, and N. Sebe are with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy. (E-mail: gaowen.liu@unitn.it, sebe@disi.unitn.it)*
- *E. Ricci is with Department of Engineering, University of Perugia, Perugia, and Fondazione Bruno Kessler, Trento, Italy. (E-mail: eliricci@fbk.eu)*

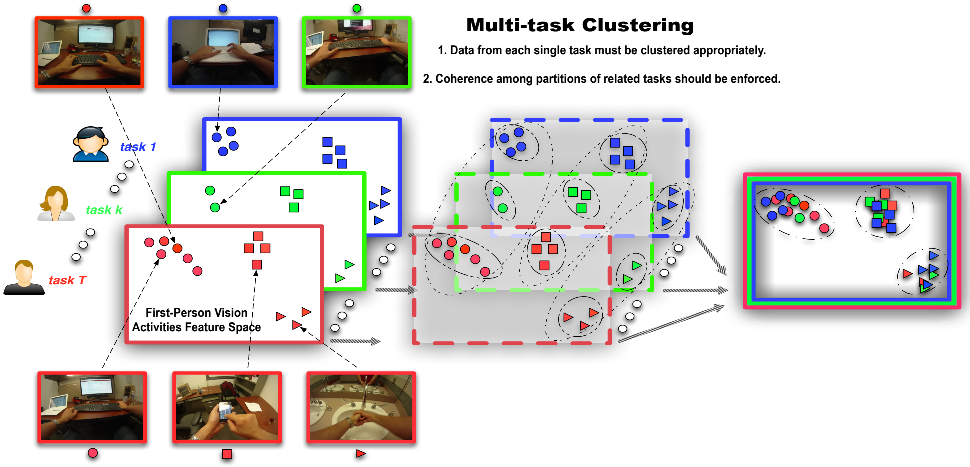

Multi-task Clustering
1. Data from each single task must be clustered appropriately.
2. Coherence among partitions of related tasks should be enforced.
task 1
task k
task T
First-Person Vision Activities Feature Space

Fig. 1. Overview of our multi-task clustering approach for FPV activity recognition (Figure best viewed in color).

simply combining data from different people together and applying a traditional clustering approach does not necessarily increase accuracy, because the data distributions of single tasks can be different (*i.e.* visual data corresponding to different people may exhibit different features).

To address these problems, we propose a novel Multi-Task Clustering (MTC) framework from which we derive two different algorithms. Our approach ensures that the data of each single task are clustered appropriately and simultaneously enforces the coherence between clustering results of related tasks (Fig. 1). To demonstrate the validity of our method we first conduct experiments on synthetic data and compare it with state-of-the-art single task and multi-task learning algorithms. Then, we show that our approach is effective in recognizing activities in an egocentric setting and we consider two recent FPV datasets, the FPV activity of daily living dataset [9] and the coupled ego-motion and eye-motion dataset introduced in [10]. This paper extends our previous work [11].

To summarize, the contributions of this paper are the following: (i) To our knowledge, this is the first work proposing a multi-task clustering framework for FPV activity recognition. Most papers on MTL for human activity analysis [12, 13] focus on video collected from fixed cameras and mostly rely on supervised methods. (ii) This paper is one of the few works presenting an *unsupervised* approach for MTL. The proposed MTC methods are novel and two efficient algorithms are derived for solving the associated optimization problems. (iii) Our learning framework is general and many other computer vision and pattern recognition applications can benefit from using it.

The paper is organized as follows. Section 2 reviews related work on first person vision activity recognition and supervised/unsupervised multi-task learning. In Section 3 our MTC framework for FPV activity recognition is described in details. The experimental results are reported in Section 4. We then conclude in Section 5.

2 Related Work

In this section, we review prior works in (i) FPV activity analysis, (ii) supervised MTL and (iii) multi-task clustering.

2.1 First-person Vision Activity Analysis

Automatically analyzing human behavior from videos is a widely researched topic. Many previous works have focused on recognizing everyday activities [3–5]. In [3] features based on the velocity history of tracked keypoints are proposed for detecting complex activities performed in a kitchen. A kitchen scenario is also analyzed by Rohrbach *et al.* [5] and an approach for fine-grained activity recognition is presented. More recently, RGB-D sensors are exploited for ADL analysis [4] and improved performance is obtained with respect to approaches based on traditional cameras. However, all these works consider a “third-person” view perspective, *i.e.* they are specifically designed to analyse video streams from fixed cameras.

The challenges of inferring human behavior from data collected by wearable cameras are addressed in [9, 10, 14–18]. Aghazadeh *et al.* [14] proposed an approach for discovering anomalous events from videos captured from a small camera attached to a person’s chest. In [19] a video summarization method targeted to FPV is presented. Fathi *et al.* [15] introduced a method for individuating social interactions in first-person videos collected during social events. Some recent works have focused on FPV-ADL analysis considering different scenarios (*e.g.* kitchen, office, home) [9, 10, 16–18]. In [9] Pirsi *et al.* introduced some features based on the output of multiple object detectors. In [10] the task of recognizing egocentric activities in an office environment is considered and motion descriptors extracted from an outside looking camera are combined with features describing the user eye movements captured by an inside looking camera. In [16] activity recognition in a kitchen scenario (*i.e.* multiple subjects preparing different recipes) is considered. A codebook learning framework is proposed in order to alleviate the problem of the large within-class data variability due to the different execution styles and speed among different subjects. Ryoo *et al.* [20] investigated multi-channel kernels to integrate global and local motion information and presented a new activity recognition methodology that explicitly models the temporal structures of FPV data. In [21] an approach for temporal segmentation of egocentric videos into twelve hierarchical classes is presented. Differently from these previous works, in this paper we address the problem of FPV ADL analysis proposing a multi-task learning framework.

2.2 Supervised Multi-task Learning

Multi-task learning methods [22] have recently proved to be particularly effective in many applications, such as complex event detection [23], object detection [24], head pose estimation [25], image classification [26], etc. The idea of MTL is simple: given a set of related tasks, by simultaneously learning the corresponding classification or regression models, improved performance can be achieved. Usually, the advantages of MTL over traditional approaches based on learning independent models are particularly pronounced when the number of samples in each task is limited.

To capture the tasks dependencies a common approach is to constrain all the learned models to share a common set of features. This motivates the introduction of a group sparsity term, *i.e.* the ℓ_1/ℓ_2-norm regularizer as in [27]. This approach works well in ideal cases. However, in practical applications, simply using a ℓ_1/ℓ_2-norm regularizer may not be effective since not every task is related to all the others. To this end, the MTL algorithm based on the dirty model is proposed in [28] with the aim to identify irrelevant (outlier) tasks. Similarly, robust multi-task learning is introduced in [29]. In some cases, the tasks exhibit a sophisticated group structure and it is desirable that the models of tasks in the same group are more similar to each other than to those from a different group. To model complex task dependencies several clustered multi-task learning methods have been introduced [30–32]. In computer vision, MTL have been previously proposed in the context of visual-based activity recognition from fixed cameras and in a supervised setting [33**?** , 34]. In this paper, we consider the more challenging FPV scenario where no annotated data are provided.

2.3 Multi-task Clustering

Many works on MTL focused on a supervised setting. Only few [35–37] have considered the more challenging scenario where the data are unlabeled and the aim is to predict the cluster labels in each single task. Gu *et al.* [35] presented an algorithm where a shared subspace is learned for all the tasks. Zhang *et al.* [37] introduced a MTC approach based on a pairwise agreement term which encourages coherence among clustering results of multiple tasks. In [36] the k-means algorithm is revised from a Bayesian nonparametric viewpoint and extended to MTL. None of these works have focused on the problem of visual-based activity recognition.

In this paper, we propose two novel approaches for multi-task clustering. The first one is inspired by the work in [37] but it is based on another objective function and thus on a radically different optimization algorithm. Furthermore, in the considered application, it provides superior accuracy with respect to [37]. Our second approach instead permits to easily integrate prior knowledge about the tasks and the data of each task (*e.g.* temporal consistency among subsequent video clips). Moreover, it relies on a convex optimization problem, thus avoids the issues related to local minima of previous methods [35–37].

3 Multi-task Clustering for First-person Vision Activity Recognition

In this section, we first introduce the motivation behind our approach, together with an overview of the proposed framework. Then, two different MTC algorithms, namely Earth Mover's Distance Multi-Task Clustering (EMD-MTC) and Convex Multi-task Clustering (CMTC), and their application to the problem of FPV ADL recognition are described.

3.1 Motivation and Overview

We consider the videos collected from wearable cameras of several people performing daily activities. No annotation is provided. We only assume that people perform about the same tasks, a very reasonable assumption in the context of ADL analysis.

To discover people activities, we consider T related tasks corresponding to T different people[1] and we introduce a MTC approach. For each task (person) t, a set of samples $X^t = \{\mathbf{x}_1^t, \mathbf{x}_2^t, ..., \mathbf{x}_{N_t}^t\}$ is available, where $\mathbf{x}_j^t \in \boldsymbol{R}^d$ is the d-dimensional feature vector describing the j-th video clip and N_t is the total number of samples associated to the t-th task. We want to segment the entire video clip corresponding to user t into parts, *i.e.* we want the data in the set X^t to be grouped into K_t clusters. Furthermore, as we assume the tasks to be related, we also require that the resulting partitions are consistent with each other. This is a reasonable assumption in the context of everyday activity recognition where people perform about the same activities. Note that the number of required partitions K_t can be different for different tasks, as different people can perform slightly different types of activities. Our assumptions are verified in the context of ADL recognition. For example, typical activities in the morning are preparing breakfast, eating and brushing teeth. Therefore, when analyzing video streams collected by wearable cameras of different users, it is reasonable to expect that the recordings will capture the same or at least very similar activities. To automatically discover these activities, we formulate the following optimization problem corresponding to multi-task clustering:

$$\min_{\boldsymbol{\Theta}_t} \sum_{t=1}^{T} \Lambda(X^t, \boldsymbol{\Theta}^t) + \lambda \sum_{t=1}^{T} \sum_{s=t+1}^{T} \Omega(\boldsymbol{\Theta}^t, \boldsymbol{\Theta}^s) \qquad (1)$$

The term $\Lambda(\cdot)$ represents a reconstruction error which must be minimized by learning the optimal task-specific model parameters $\boldsymbol{\Theta}^t$ (*i.e.* typically the cluster centroids and the associated assignment matrix), while $\Omega(\cdot)$ is an "agreement" term imposing that, since the multiple tasks are related, also the associated model parameters should be similar. Under

[1] This is not a constraint. In this paper we focus on detecting activities for each user by exploiting related information from other users.

this framework, in this paper we propose two different algorithms for MTC.

To stress the generality of our framework, we apply the proposed algorithms in two different FPV scenarios: an office environment where people are involved in typical activities such as browsing the web or writing documents and a home environment where a chest mounted camera records users' activities such as opening a fridge or preparing tea. To perform experiments we use two publicly available datasets, corresponding to the scenarios described above: the FPV office dataset introduced in [10] and the FPV ADL dataset described in [9]. Both datasets contain visual streams recorded from an outside-looking wearable camera while the office dataset also has information about eye movements acquired by an inside-looking camera. In the following subsections we describe the proposed MTC algorithms and the adopted feature descriptors.

Notation: In the following $\mathbf{A}_{i.}$, $\mathbf{A}_{.j}$ denote respectively the i-th row and the j-th column of the matrix $\mathbf{A}$. We also denote with $(\cdot)'$ the transpose operator, $N = \sum_{t=1}^{T} N_t$ is the total number of datapoints, while $\mathbf{X} \in \mathbb{R}^{N \times d}$, $\mathbf{X} = [\mathbf{X}^{1'} \ \mathbf{X}^{2'} \ \ldots \ \mathbf{X}^{T'}]'$ is the data matrix obtained by concatenating the task specific matrices $\mathbf{X}^t \in \mathbb{R}^{N_t \times d}$, $\mathbf{X}^t = [\mathbf{x}_1^t \ \mathbf{x}_2^t \ ... \ \mathbf{x}_{N_t}^t]'$.

3.2 Earth Mover's Distance Multi-task Clustering

Given the task data matrices $\mathbf{X}^t$, we are interested in finding the centroid matrices $\mathbf{C}^t \in \mathbb{R}^{K_t \times d}$, and the cluster indicators matrices $\mathbf{W}^t \in \mathbb{R}^{N_t \times K_t}$ by solving the following optimization problem:

$$\min_{\mathbf{C}^t, \mathbf{W}^t} \sum_{t=1}^{T} \|\mathbf{X}^t - \mathbf{W}^t\mathbf{C}^t\|_F^2 + \lambda \sum_{t=1}^{T} \sum_{s=t+1}^{T} \Omega_E(\mathbf{C}^t, \mathbf{W}^t, \mathbf{C}^s, \mathbf{W}^s)$$

The first term in the objective function is a relaxation of the traditional k-means objective function for T separated data sources. The agreement term $\Omega_E(\cdot)$ is added to explore the relationships between clusters of different data sources and it is defined as follows:

$$\begin{aligned}
\Omega_E(\mathbf{C}^t, \mathbf{W}^t, \mathbf{C}^s, \mathbf{W}^s) = \quad & \min_{f_{ij}^{st} \geq 0} \quad \sum_{i=1}^{K_t} \sum_{j=1}^{K_s} f_{ij}^{st} (\mathbf{C}_{i.}^t - \mathbf{C}_{j.}^s)'(\mathbf{C}_{i.}^t - \mathbf{C}_{j.}^s) \\
& \text{s.t.} \quad \sum_{j=1}^{K_s} f_{ij}^{st} = \sum_{n=1}^{N_t} \mathbf{W}_{ni}^t \quad \forall t, i \\
& \qquad \sum_{i=1}^{K_t} f_{ij}^{st} = \sum_{n=1}^{N_s} \mathbf{W}_{nj}^s \quad \forall s, j \\
& \qquad \sum_{i=1}^{K_t} \sum_{j=1}^{K_s} f_{ij}^{st} = 1 \quad \forall s, t
\end{aligned}$$

It consists in the popular Earth Mover's Distance (EMD) [38] computed considering the signatures $\mathcal{T}$ and $\mathcal{S}$ obtained by clustering the data associated to task t and s separately, *i.e.* $\mathcal{T} = \{(\mathbf{C}_{1}^t., w_t^1), \ \ldots, \ (\mathbf{C}_{K_t}^t., w_t^{K_t})\}$, $w_t^i = \sum_{n=1}^{N_t} \mathbf{W}_{ni}^t$, and $\mathcal{S} = \{(\mathbf{C}_{1}^s., w_s^1), \ \ldots, \ (\mathbf{C}_{K_s}^s., w_s^{K_s})\}$, $w_s^i = \sum_{n=1}^{N_s} \mathbf{W}_{ni}^s$. In practice $\mathbf{C}_{i}^t.$ and $\mathbf{C}_{j}^s.$ are the cluster centroids and w_i^s, w_i^t denote the weights associated to each cluster (approximating the number of datapoints in each cluster). In practice $\Omega_E(\cdot)$ represents the distance between two distributions and minimizing it we impose the found partitions between pairs of related tasks to be consistent. The variables f_{ij}^{st} are flow variables as follows from the definition of EMD as a transportation problem [38].

Algorithm 1: Algorithm for solving (2).

```
Input: the data matrices $\mathbf{X}^1, \mathbf{X}^2$, the numbers of clusters $K_1$, $K_2$,
       the parameter $\lambda$.
1: Initialize $\mathbf{F}$ as an identity matrix.
2: Initialize $\mathbf{W} > 0$ with $l_1$ normalized columns and $\mathbf{P} > 0$ with
   $l_1$ normalized rows.
3: repeat
     Given $\mathbf{W}^k$, $\mathbf{P}^k$, compute $\mathbf{F}^{k+1}$ solving (4).
     Given $\mathbf{F}^{k+1}$, $\mathbf{P}^k$, compute:
     $\mathbf{W}^{k+1} = \max(0, \mathbf{W}^k - \alpha_k \nabla_{\mathbf{W}} \Delta(\mathbf{P}^k, \mathbf{W}^k, \mathbf{F}^{k+1}))$.
     Given $\mathbf{F}^{k+1}$, $\mathbf{W}^{k+1}$, compute:
     $\mathbf{P}^{k+1} = \max(0, \mathbf{P}^k - \alpha_k \nabla_{\mathbf{P}} \Delta(\mathbf{P}^k, \mathbf{W}^{k+1}, \mathbf{F}^{k+1}))$.
     Normalize $\mathbf{P}$ by $\mathbf{P}_{ij}^{k+1} \leftarrow \frac{\mathbf{P}_{ij}^{k+1}}{\sum_j \mathbf{P}_{ij}^{k+1}}$.
   until convergence;
Output: the optimized matrices $\mathbf{W}, \mathbf{P}$.
```

In the proposed optimization problem there are no constraints on the $\mathbf{C}_t$ values. In this paper we define the matrix $\mathbf{C} \in \mathbb{R}^{K \times d}$, $\mathbf{C} = [\mathbf{C}^{1'} \ldots \mathbf{C}^{T'}]'$, $K = \sum_{t=1}^{T} K_t$, and we impose that the columns of $\mathbf{C}$ are a weighted sum of certain data points, *i.e.* $\mathbf{C} = \mathbf{P}\mathbf{X}$ where $\mathbf{P} = \text{blkdiag}(\mathbf{P}^1, \ldots, \mathbf{P}^T)$, $\mathbf{P} \in \mathbb{R}^{K \times N}$. In the following, for the sake of simplicity and easy interpretation, we consider only two tasks. The extension to T tasks is straightforward. Defining $\mathbf{F} = \text{diag}(f_{11} \ldots f_{K_1K_2})$, $\mathbf{F} \in \mathbb{R}^{K_1K_2 \times K_1K_2}$ and the block diagonal matrix $\mathbf{W} = \text{blkdiag}(\mathbf{W}^1, \mathbf{W}^2)$, $\mathbf{W} \in \mathbb{R}^{N \times K}$, we formulate the following optimization problem:

$$\Delta(\mathbf{P}, \mathbf{W}, \mathbf{F}) = \min_{\mathbf{P}, \mathbf{W}, \mathbf{F} \geq 0} \{\|\mathbf{X} - \mathbf{W}\mathbf{P}\mathbf{X}\|_F^2 + \lambda \text{tr}(\mathbf{M}\mathbf{P}\mathbf{X}\mathbf{X}'\mathbf{P}'\mathbf{M}'\mathbf{F})\} \tag{2}$$

$$\begin{aligned}
\text{s.t.} \quad & \|\mathbf{P}_{i.}^t\|_1 = 1, \quad \forall i = 1, \ldots, K \ \ \forall \ t = 1, 2 \\
& \text{tr}(\mathbf{I}_j \mathbf{F}) = \sum_{i=1}^{N} \mathbf{W}_{ij}, \quad \forall j = 1, ..., K \qquad (3) \\
& \text{tr}(\mathbf{F}) = 1
\end{aligned}$$

where: $\mathbf{I}_j \in \mathbb{R}^{K_1K_2 \times K_1K_2}$ and $\mathbf{M} \in \mathbb{R}^{K_1K_2 \times K}$ are appropriately defined selection matrices.

To solve the proposed optimization problem we develop an iterative optimization scheme described below. It is worth noting that our method can be kernelized, defining a feature mapping $\phi(\cdot)$ and the associated kernel matrix $\mathbf{K}_{\mathbf{X}} = \phi(\mathbf{X})\phi(\mathbf{X})'$. The objective function of (2) becomes:

$$\begin{aligned}
& \|\phi(\mathbf{X}) - \mathbf{W}\mathbf{P} \ \phi(\mathbf{X})\|_F^2 + \lambda \text{tr}(\mathbf{M}\mathbf{P}\phi(\mathbf{X})\phi(\mathbf{X})'\mathbf{P}'\mathbf{M}'\mathbf{F}) = \\
& \text{tr}(\mathbf{K}_{\mathbf{X}} - \mathbf{2}\mathbf{K}_{\mathbf{X}}\mathbf{P}'\mathbf{W}' + \mathbf{W}\mathbf{P}\mathbf{K}_{\mathbf{X}}\mathbf{P}'\mathbf{W}' + \lambda \mathbf{M}\mathbf{P}\mathbf{K}_{\mathbf{X}}\mathbf{P}'\mathbf{M}'\mathbf{F})
\end{aligned}$$

The update rules of the kernelized version of our method can be easily derived similarly to the linear case presented below using $\mathbf{K}_{\mathbf{X}}$ instead of $\mathbf{X}'\mathbf{X}$.

Algorithm 2: Algorithm for solving (5).

```
Input: The data matrix $\mathbf{X}$, $\mathbf{E},\mathbf{B}$, the parameter $\lambda_2$.
1: Set $\mathbf{Q} = \rho\mathbf{E}'\mathbf{E} + 2\mathbf{I} + 2\lambda_2\mathbf{B}$.
2: Compute Cholesky factorization of the matrix $\mathbf{Q}$.
3: for j=1:d do
     repeat
         Set $\mathbf{b}^k = \rho\mathbf{E}'\mathbf{q}^k - \mathbf{E}'\mathbf{p}^k + 2\mathbf{X}_{.j}$
         Update $\mathbf{\Pi}_{.j}$
           Solve $\mathbf{Q}[\mathbf{\Pi}_{.j}]^{k+1} = \mathbf{b}^k$
         Update $\mathbf{q}$ using a soft thresholding operator
           $\mathbf{q}^{k+1} = ST_{1/\rho}(\mathbf{E}[\mathbf{\Pi}_{.j}]^{k+1} + \frac{1}{\rho}\mathbf{p}^k)$
         Update $\mathbf{p}$
           $\mathbf{p}^{k+1} = \mathbf{p}^k + \rho(\mathbf{E}[\mathbf{\Pi}_{.j}]^{k+1} - \mathbf{q}^{k+1})$
     until convergence;
Output: The final centroid matrix $\mathbf{\Pi}$.
```

3.2.1 Optimization

To solve (2), we first note that the optimal solution can be found by adopting an alternating optimization scheme, *i.e.* optimizing separately first with respect to $\mathbf{P}$ and then with respect to $\mathbf{W}$ and $\mathbf{F}$ jointly. In both cases, a non-negative least square problem with constraints arises, for which standard solvers can be employed. However, due to computational efficiency, in this paper we consider an approximation of (2), replacing the constraints (3) with $\text{tr}(\mathbf{I}_j\mathbf{F}) = \mathbf{e}$, where $\mathbf{e} \in \boldsymbol{R}^{K_1K_2}$, $\mathbf{e}_i = \frac{1}{K_1}$, if $i \leq K_1$, $\mathbf{e}_i = \frac{1}{K_2}$ otherwise. This approximation implies that for each task the same number of datapoints is assigned to all the clusters. In this way a more efficient solver can be devised. Specifically, we adopt an alternating optimization strategy, *i.e.* we optimize (2) separately with respect to $\mathbf{F}$, $\mathbf{W}$ and $\mathbf{P}$ until convergence, as explained in the following:

Step 1: Fixed $\mathbf{W},\mathbf{P}$, optimize $\mathbf{F}$ solving:

$$\begin{aligned} \min_{\mathbf{F}>\mathbf{0},\ \text{tr}(\mathbf{F})=1} \quad & \text{tr}(\mathbf{MPXX'P'M'F}) \\ \text{s.t.} \quad & \text{tr}(\mathbf{I}_j\mathbf{F}) = \mathbf{e}, \quad \forall j = 1, ..., K_1 + K_2 \end{aligned} \tag{4}$$

This is a simple linear programming problem. It can be solved efficiently with standard solvers.

Step 2: Fixed $\mathbf{F},\mathbf{P}$, optimize $\mathbf{W}$ solving:

$$\min_{\mathbf{W}>\mathbf{0}} \|\mathbf{X} - \mathbf{WPX}\|_F^2$$

Following [39], we update $\mathbf{W}$ using a projected gradient method for bound-constrained optimization, *i.e.* $\mathbf{W}^{k+1} = \max(0, \mathbf{W}^k - \alpha_k\nabla_{\mathbf{W}}\Delta(\mathbf{P}^k, \mathbf{W}^k, \mathbf{F}^{k+1}))$, where $\nabla_{\mathbf{W}}\Delta(\mathbf{P},\mathbf{W},\mathbf{F}) = \mathbf{WPXX'P'} - \mathbf{XX'P'}$.

Step 3: Fixed $\mathbf{W},\mathbf{F}$, optimize $\mathbf{P}$ solving:

$$\begin{aligned} & \min_{\mathbf{P}>\mathbf{0}} \|\mathbf{X} - \mathbf{WPX}\|_F^2 + \lambda\text{tr}(\mathbf{MPXX'P'M'F}) \\ & \text{s.t.} \quad \|\mathbf{P}^t_{i.}\|_1 = 1, \ \forall i \quad \forall\ t = 1, 2 \end{aligned}$$

Similarly to step 2, we update $\mathbf{P}$ using a projected gradient method for bound-constrained optimization, *i.e.* $\mathbf{P}^{k+1} = \max(0, \mathbf{P}^k - \alpha_k\nabla_{\mathbf{P}}\Delta(\mathbf{P}^k, \mathbf{W}^{k+1}, \mathbf{F}^{k+1}))$, where $\nabla_{\mathbf{P}}\Delta(\mathbf{P},\mathbf{W},\mathbf{F}) = \mathbf{W'WPXX'} - \mathbf{W'XX'} + \lambda\mathbf{M'FMPXX'}$. To account for constraints at each iteration we also normalize each row of $\mathbf{P}$, following the normalization invariance approach in [40].

The algorithm for solving (2) is summarized in Algorithm 1. Regarding the computational complexity, the cost of solving (2) with the iterative approach outlined in Algorithm 1 is dominated by the first step, *i.e.* by the linear programming problem in (4) which can be solved in polynomial time.

3.3 Convex Multi-task Clustering

Given the task specific training sets X^t, we propose to learn the sets of cluster centroids $\Pi^t = \{\boldsymbol{\pi}^t_1, \boldsymbol{\pi}^t_2, ..., \boldsymbol{\pi}^t_{N_t}\}, \boldsymbol{\pi}^t_i \in \boldsymbol{R}^d$, by solving the following optimization problem:

$$\min_{\boldsymbol{\pi}^t_i} \ \{\sum_{t=1}^{T}\sum_{i=1}^{N_t} \|\mathbf{x}^t_i - \boldsymbol{\pi}^t_i\|_2^2 + \lambda_t \sum_{t=1}^{T} \sum_{\substack{i,j=1 \\ j>i}}^{N_t} w^t_{ij}\|\boldsymbol{\pi}^t_i - \boldsymbol{\pi}^t_j\|_1 + \lambda_2\mathbf{\Omega}_C(\Pi^t)\} \tag{5}$$

where:

$$\mathbf{\Omega}_C(\Pi^t) = \sum_{\substack{t,s=1 \\ s>t}}^{T} \gamma_{st} \sum_{i=1}^{N_t}\sum_{j=1}^{N_s} \|\boldsymbol{\pi}^t_i - \boldsymbol{\pi}^s_j\|_2^2$$

In (5) the first two terms guarantee that the data of each task are clustered: specifically with $\lambda_t = 0$ the found centroids are equal to the datapoints while as λ_t increases the number of different centroids $\boldsymbol{\pi}^t_i$ reduces. The last term $\mathbf{\Omega}_c(\Pi^t)$ instead imposes the found centroids to be similar if the tasks are related. The relatedness between tasks is modeled by the parameter γ_{st} which can be set using an appropriate measure between distributions. We consider the Maximum Mean Discrepancy [41], defined as $\mathcal{D}(X^t, X^s) = \|\frac{1}{N_t}\sum_{i=1}^{N_t}\phi(\mathbf{x}^t_i) - \frac{1}{N_s}\sum_{i=1}^{N_s}\phi(\mathbf{x}^s_i)\|^2$ and we compute it using a linear kernel. We set $\gamma_{st} = e^{-\beta\mathcal{D}(X^t,X^s)}$ with β being a user-defined parameter ($\beta = 0.1$ in our experiments). The parameters w^t_{ij} are used to enforce datapoints in the same task to be assigned to the same cluster and can be set according to some *a-priori* knowledge or in a way such that the found partitions structure reflects the density of the original data distributions.

3.3.1 Optimization

To solve (5) we propose an algorithm based on the alternating direction method of multipliers [42]. We consider the matrix $\mathbf{\Pi} = [\mathbf{\Pi}^{1'}\ \mathbf{\Pi}^{2'}\ \ldots\ \mathbf{\Pi}^{T'}]'$, $\mathbf{\Pi} \in \boldsymbol{R}^{N\times d}$, obtained concatenating the task-specific matrices $\mathbf{\Pi}^t = [\boldsymbol{\pi}^t_1\ \boldsymbol{\pi}^t_2\ ...\ \boldsymbol{\pi}^t_{N_t}]'$. The problem (5) can be solved considering d separate minimization subproblems (one for each column of $\mathbf{X}$) as follows:

$$\begin{aligned} \min_{\mathbf{q},\ \mathbf{\Pi}_{.j}} \quad & \{\|\mathbf{X}_{.j} - \mathbf{\Pi}_{.j}\|_2^2 + \|\mathbf{q}\|_1 + \lambda_2\|\mathbf{B\Pi}_{.j}\|_2^2\} \\ \text{s.t.} \quad & \mathbf{E\Pi}_{.j} - \mathbf{q} = 0 \end{aligned} \tag{6}$$

where $\mathbf{E}$ is a block diagonal matrix defined as $\mathbf{E} = \text{blkdiag}(\mathbf{E}^1, \mathbf{E}^2, \ldots, \mathbf{E}^T)$ and $\mathbf{E}^t \in \boldsymbol{R}^{|\mathcal{E}_t|\times N_t}$ is a matrix with $|\mathcal{E}_t| = \frac{N_t(N_t-1)}{2}$ rows. Each row is a vector of all zeros except in the position i where it has the value $\lambda_t w^t_{ij}$ and in the position j where it has the value $-\lambda_t w^t_{ij}$. Similarly the matrix $\mathbf{B} \in \boldsymbol{R}^{|\mathcal{B}|\times N}$, where $|\mathcal{B}| = \frac{T(T-1)}{2}$, imposes smoothness between the parameters of related tasks. A row of the matrix $\mathbf{B}$ is a vector with all zeros except in the terms corresponding to datapoints of the t-th task which are set

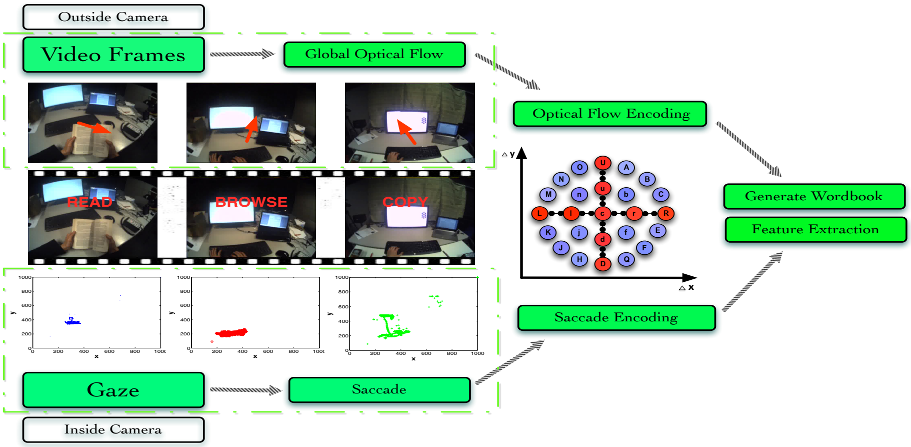


Fig. 2. Feature extraction pipeline on the FPV office dataset. Some frames corresponding to the actions *read*, *browse* and *copy* are shown together with the corresponding optical flow features (top) and eye-gaze patterns depicted on the 2-D plane (bottom). It is interesting to observe the different gaze patterns among these activities.

to γ_{st} and to the terms corresponding to datapoints of the s-th task which are all set to $-\gamma_{st}$. To solve (6) we consider the associated lagrangian:

$$L_\rho(\mathbf{\Pi}_{.j},\mathbf{q},\mathbf{p}) = \|\mathbf{X}_{.j}-\mathbf{\Pi}_{.j}\|_2^2 + \|\mathbf{q}\|_1 + \lambda_2\|\mathbf{B}\mathbf{\Pi}_{.j}\|_2^2$$
$$+\mathbf{p}'(\mathbf{E}\mathbf{\Pi}_{.j}-\mathbf{q}) + \frac{\rho}{2}\left\|\mathbf{E}\mathbf{\Pi}_{.j}-\mathbf{q}\right\|_2^2$$

with $\mathbf{p}$ being the vector of augmented Lagrangian multipliers and ρ being the dual update step length. We devise an algorithm where three steps, corresponding to the update of the three variables $\mathbf{\Pi}_{.j},\mathbf{q},\mathbf{p}$, are performed.

Step 1: Update $\mathbf{\Pi}_{.j}$, given $\mathbf{q},\mathbf{p}$ fixed, by solving:

$$\min_{\mathbf{\Pi}_{.j}} \quad \|\mathbf{X}_{.j}-\mathbf{\Pi}_{.j}\|_2^2 + \|\mathbf{q}\|_1 + \lambda_2\|\mathbf{B}\mathbf{\Pi}_{.j}\|_2^2$$
$$+\mathbf{p}'(\mathbf{E}\mathbf{\Pi}_{.j}-\mathbf{q}) + \frac{\rho}{2}\left\|\mathbf{E}\mathbf{\Pi}_{.j}-\mathbf{q}\right\|_2^2$$

Imposing the gradient with respect to $\mathbf{\Pi}_{.j}$ equal to 0, the update step is formulated as:

$$\mathbf{Q}[\mathbf{\Pi}_{.j}]^{k+1} = \mathbf{b}^k$$

where $\mathbf{Q} = \rho\mathbf{E}'\mathbf{E} + 2\mathbf{I} + 2\lambda_2\mathbf{B}$ and $\mathbf{b}^k = \rho\mathbf{E}'\mathbf{q}^k - \mathbf{E}'\mathbf{p}^k + 2\mathbf{X}_{.j}$. The computation of $\mathbf{\Pi}_{.j}$ involves solving a linear system. To solve it efficiently, we use Cholesky factorization and decompose $\mathbf{Q} = \mathbf{\Sigma}'\mathbf{\Sigma}$. In practice, at each iteration, we solve two linear systems: $\mathbf{\Sigma}'\mathbf{g} = \mathbf{b}^k$ and $\mathbf{\Sigma}\mathbf{\Pi}_{.j} = \mathbf{g}$. Since $\mathbf{\Sigma}$ is an upper triangular matrix, solving them is typically very efficient.

Step 2: Update $\mathbf{q}$, given $\mathbf{\Pi}_{.j},\mathbf{p}$ fixed, by solving:

$$\min_{\mathbf{q}} \quad \|\mathbf{q}\|_1 - \mathbf{p}'\mathbf{q} + \frac{\rho}{2}\left\|\mathbf{E}\mathbf{\Pi}_{.j}-\mathbf{q}\right\|_2^2$$

Neglecting the constant terms, the update step is:

$$\mathbf{q}^{k+1} = \arg\min_{\mathbf{q}} \frac{1}{2}\left\|\mathbf{q} - \mathbf{E}[\mathbf{\Pi}_{.j}]^{k+1} - \frac{1}{\rho}\mathbf{p}^k\right\|_2^2 + \frac{1}{\rho}\|\mathbf{q}\|_1$$

This equation has a closed-form solution. Defining the soft thresholding operator $ST_\lambda(x) = \text{sign}(x)\max(|x|-\lambda, 0)$ the update step becomes:

$$\mathbf{q}^{k+1} = ST_{1/\rho}(\mathbf{E}[\mathbf{\Pi}_{.j}]^{k+1} + \frac{1}{\rho}\mathbf{p}^k)$$

Step 3: Update $\mathbf{p}$, given $\mathbf{\Pi}_{.j},\mathbf{q}$ fixed, with the equation:

$$\mathbf{p}^{k+1} = \mathbf{p}^k + \rho(\mathbf{E}[\mathbf{\Pi}_{.j}]^{k+1} - \mathbf{q}^{k+1})$$

We summarize our approach in Algorithm 2. Regarding the computational complexity of Algorithm 2, the most computationally expensive step is the Cholesky matrix factorization ($O(N^3)$). However, the Cholesky factorization is performed only once. In the inner loop, for each dimension $j = 1,\ldots,d$, each iteration involves solving one linear system ($O(N^2)$) and a soft-thresholding operation ($O(\sum_t|\mathcal{E}^t|)$).

3.4 Features Extraction in Egocentric Videos

The growing interest in the vision community towards novel approaches for FPV analysis has motivated the creation of several publicly available datasets (see [43] for a recent survey). In this paper we consider two of them, the FPV office dataset [10] and the FPV home dataset [9].

Due to the large variability of visual data collected from wearable cameras there exist no standard feature descriptors. While in some situations extracting simple motion information, *e.g.* by computing the optical flow, may suffice [10], in other cases motion patterns may be too noisy and other kind of information (*e.g.* presence/absence of objects) must be exploited. In this paper we demonstrate that, independently from the employed feature descriptors, MTC is an effective strategy for recognizing everyday activities. We now describe the adopted feature representations respectively for the considered office and home scenarios.

3.4.1 *FPV office dataset*

The FPV office dataset [10] consists of five common activities in an office environment (*reading a book, watching a video, copying text from screen to screen, writing sentences on paper* and *browsing the internet*). Each action was performed by five subjects, who were instructed to execute each task for about two minutes, while 30 seconds intervals of void class were placed between target tasks. To provide a natural experimental setting, the void class contains a wide variety of actions such as conversing, singing and random head motions. The sequence of five actions was repeated twice to induce interclass variance. The dataset consists of over two hours of data, where the video from each subject is a continuous 25-30 minutes video.

We follow [10] and extract features describing both the eye motion (obtained by the inside-looking camera) and the head and body motion (computed processing the outside camera's stream). To calculate the eye motion features, we consider the gaze coordinates provided in the dataset and smooth them applying a median filter. Then the continuous wavelet transform is adopted for saccade detection separately on the x and y motion components [44]. The resulting signals are quantized according to magnitude and direction and are coded with a sequence of discrete symbols. To analyze the streams of the output camera, for each frame the global optical flow is computed by tracking corner points over consecutive frames and taking the mean flow in the x and y directions. Then, the optical flow vectors are quantized according to magnitude and direction with the same procedure adopted in the eye motion case. The obtained sequences of symbols are then processed to get the final video clip descriptors. We use a temporal sliding window approach to build an n-gram dictionary over all the dataset. Then each video is divided into segments corresponding to 15 seconds, each of them representing a video clip. For each sequence of symbols associated to a video clip, a histogram over the dictionary is computed. The final feature descriptor $\mathbf{x}_i$ is calculated by considering some statistics over the clip histogram and specifically computing the maximum, the average, the variance, the number of unique n-grams, and the difference between maximum and minimum count. Fig.2 shows the feature extraction pipeline.

3.4.2 FPV home dataset

The FPV home dataset [9] contains videos recorded from chest-mounted cameras by 20 different users. The users perform 18 non-scripted daily activities in the house, like *brushing teeth, washing dishes,* or *making tea*. The length of the videos is in the range of 20-60 minutes. The annotations about the presence of 42 relevant objects (*e.g.* kettle, mugs, fridge) and about temporal segmentation are also provided.

In this paper we adopt the same object-centric approach proposed in [9], *i.e.* to compute features for each video clip we consider the output of several object detectors. We use the pre-segmented video clips and the active object models in [9]. Active object models are introduced to exploit the fact that objects may look different when being interacted with (*e.g.* open and close fridge). Therefore in [9] additional detectors are trained using a subset of training images depicting the object appearance when objects are used by people. To obtain object-centric features for each frame a score for each object model and each location is computed. The maximum scores of all the object models are used as frame features. To compute the final clip descriptor $\mathbf{x}_i$, two approaches are adopted: one based on "bag of features" (accumulating frame features over time) and the other based on temporal pyramids. The temporal pyramid features are obtained concatenating multiple histograms constructed with accumulation: the first is a histogram over the full temporal extent of a video clip, the next is the concatenation of two histograms obtained by temporally segmenting the video into two parts, etc.

4 Experimental Results

In this section, we first conduct experiments on synthetic data to demonstrate the advantages of the proposed MTC approach over traditional single task learning methods. Then, we apply our MTC algorithms to FPV data showing their effectiveness for recognizing everyday activities.

In the experiments, we compare our methods, *i.e.* EMD Multi-task Clustering with linear and gaussian kernel and Convex Multi-task Clustering (here denoted as EMD-MTC, KEMD-MTC and CMTC, respectively), with single task clustering approaches. Specifically we consider k-means (KM), kernel k-means (KKM), convex (CNMF) and semi-nonnegative matrix factorization (SemiNMF) [45]. We also consider recent multi-task clustering algorithms such as the SemiEMD-MTC proposed in [37], its kernel version KSemiEMD-MTC and the LS-MTC method in [35]. For all the methods (with the exception of CMTC which relies on convex optimization) ten runs are performed, corresponding to different initializations conditions. Averaging over multiple iterations is typical when considering non-convex optimization problems for clustering, such as in case of the popular k-means. The average results are shown. In CMTC the parameters λ_t are varied in order to obtain the desired number of clusters. The value of the regularization parameters of our approaches (λ for the methods based on EMD regularization and λ_2 for CMTC) are set in the range $\{0.01, 0.1, 1, 10, 100\}$. As evaluation metrics, we adopt the clustering accuracy (ACC) and the normalized mutual information (NMI), as they are widely used in the literature.

4.1 Synthetic data experiments

In the synthetic data experiments we consider $T = 4$ different tasks. Each task contains 4 clusters as shown in Fig.3. The input data $\mathbf{x}_i^t \in R^d$ $(d = 2)$ for the four clusters are generated from multivariate normal distributions $\mathcal{N}(\mu, \sigma)$, as shown in Table 1, in order to obtain correlated clusters for the different tasks. For each task and each cluster 10 samples are generated for training and 20 are used to set the regularization parameters. For CMTC we set the weights $w_{ij}^t = e^{-\|\mathbf{x}_i^t - \mathbf{x}_j^t\|^2}$ if $e^{-\|\mathbf{x}_i^t - \mathbf{x}_j^t\|^2} \leq \theta$ and $w_{ij}^t = 0$ otherwise. This aims to enforce that the discovered partitions reflect the density of the original data distributions.

We compared the proposed methods with other state-of-the-art approaches. Fig.4 reports the average accuracy and NMI. The higher numbers indicate better performance. From Fig.4 it is evident that our multi-task approaches

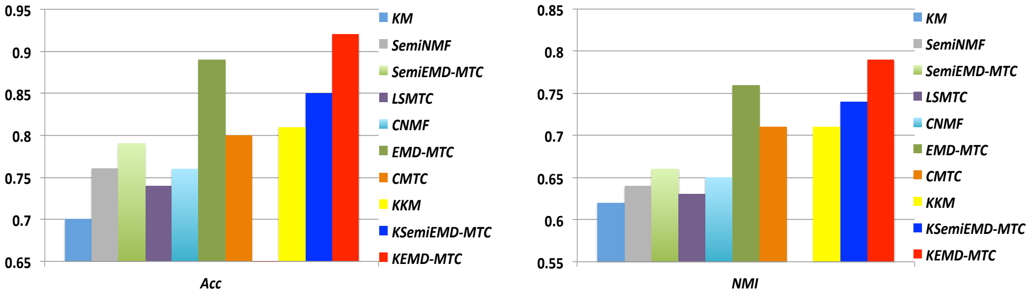


Fig. 4. Clustering results on synthetic data for different methods. Methods based on linear kernel are separated from those with gaussian kernel. (Figure is best viewed in color).

TABLE 1
Parameters used in the synthetic data experiments.

	Cluster 1	Cluster 2	Cluster 3	Cluster 4
	μ_1	μ_2	μ_3	μ_4
Task 1	(0, 0)	(1, 1)	(-1, 1)	(1, -1)
Task 2	(-0.2, -0.22)	(1, 1.04)	(-1, 0.95)	(1.2, -0.83)
Task 3	(0.02, 0)	(1.04, 1)	(-0.95, 1)	(1.03, -1)
Task 4	(-0.22, -0.22)	(1.04, 1.04)	(-0.95, 0.95)	(1.23, -0.83)
σ	(0.1, 0.1)	(0.2, 0.4)	(0.1, 0.2)	(0.4, 0.2)

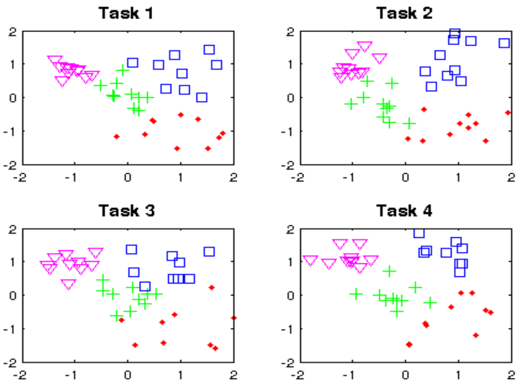


Fig. 3. Samples generated in the synthetic data experiments (different colors represent different clusters).

TABLE 2
FPV office dataset: comparison of different methods using saccade (S), motion (M) and S+M features.

	ACC			NMI		
	S	M	S+M	S	M	S+M
KM	0.230	0.216	0.257	0.029	0.021	0.045
SemiNMF [45]	0.320	0.303	0.358	0.149	0.131	0.166
SemiEMD-MTC [37]	0.371	0.349	0.415	0.229	0.209	0.259
LSMTC [35]	0.286	0.261	0.335	0.043	0.031	0.071
CNMF [45]	0.328	0.301	0.357	0.152	0.139	0.170
EMD-MTC	0.389	0.363	0.442	0.239	0.221	0.273
CMTC ($\lambda_2 = 0$)	0.367	0.346	0.413	0.224	0.209	0.244
CMTC	0.425	0.401	0.468	0.259	0.238	0.305
KKM	0.345	0.316	0.377	0.159	0.152	0.185
KSemiEMD-MTC [37]	0.387	0.359	0.432	0.241	0.228	0.287
KEMD-MTC	0.436	0.419	0.485	0.268	0.244	0.311

significantly outperform the single-task methods, both when a linear kernel is used (*e.g.* EMD-MTL and CMTC achieve higher accuracy than KM), and in the nonlinear case (KEMD-MTC outperforms KKM). The proposed algorithms also achieve higher accuracy than recent multi-task clustering methods, *i.e.* KSemiEMD-MTC [37] and LSMTC [35].

4.2 FPV Results

In this subsection, we present the experimental results on the FPV office dataset and the FPV home dataset, respectively.

4.2.1 FPV office dataset

We consider $T = 5$ tasks, as the FPV office dataset [10] contains videos corresponding to five people. As each datapoint corresponds to a video clip in this dataset, we set the parameters w^t_{ij} in CMTC in order to enforce temporal consistency, *i.e.* for each task t, $w^t_{ij} = 1$ if the features vectors $\mathbf{x}^t_i$ and $\mathbf{x}^t_j$ correspond to temporal adjacent video clips, otherwise $w^t_{ij} = 0$.

Table 2 compare different clustering methods when different types of features are employed, *i.e.* only saccade, only motion and saccade+motion features. The last three rows correspond to methods which employ a non-linear kernel. From Table 2, several observations can be made. First, independently on the adopted features representation, multi-task clustering approaches always perform better than single task clustering methods (*e.g.* SemiEMD-MTC outperforms SemiNMF, EMD-MTC provides higher accuracy than CNMF, a value of λ_2 greater than 0 leads to an improvement in accuracy and NMI in CMTC). Confirming the findings reported in [10], we also observe that combining motion and saccade features is advantageous with respect to considering each single feature representation separately. Noticeably, our methods are among the best performers, with KEMD-MTC reaching the highest values of accuracy and NMI. This is somehow expected probably due to both the use of kernels and the adoption of the multi-task learning paradigm. Moreover, CMTC outperforms EMD-MTC by up to 4% which means that incorporating information about temporal consistency in the learning process is beneficial. Furthermore, in this case the use of Maximum Mean Discrepancy may capture better the relationship among tasks with respect to EMD. Fig.5 shows some qualitative temporal segmentation results on the second sequence of subject-3. In this case for example the CMTC method outperforms all the other approaches and the importance of enforcing temporal consistency among clips is evident.

Finally, Fig.6 shows the confusion matrices associated

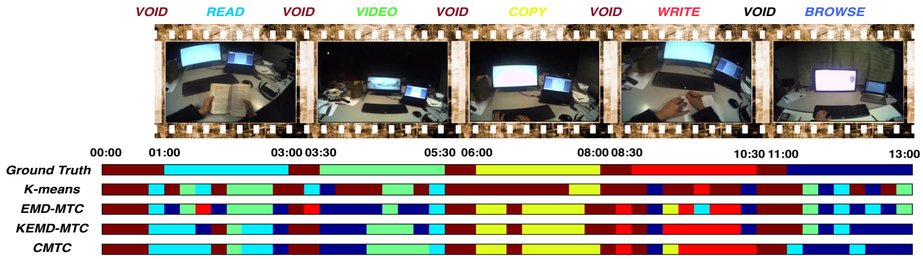


Fig. 5. FPV Office dataset. Temporal video segmentation on the second sequence of subject-3 (13 minutes): comparison of different methods. (Best viewed in color).

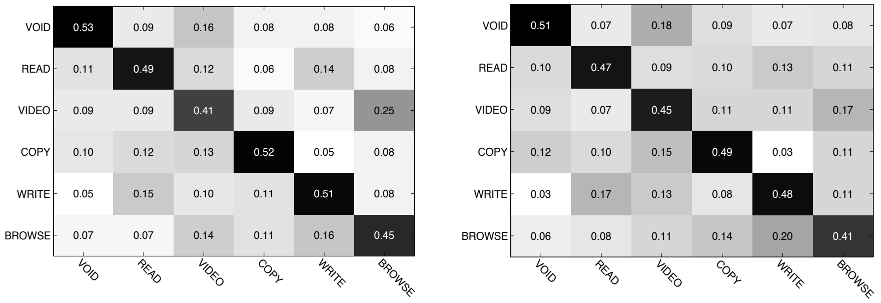


Fig. 6. FPV Office dataset. Confusion matrices using saccade+motion features obtained with (left) KEMD-MTC and (right) CMTC methods.

to our methods KEMD-MTC and CMTC. Examining the matrix associated to KEMD-MTC, we observe that the *void, copy* and *write* actions achieve relative high recognition accuracies compared with the *video* and *browse* actions. It is also interesting to note that 25% and 17% of the *video* actions are recognized as *browse* actions for KEMD-MTC and CMTC respectively, because of the similarity among motion and eye-gaze patterns.

4.2.2 FPV home dataset

In the FPV home dataset [9] there are 18 different non-scripted activities. Since each person typically performs a small subset of the 18 activities, in our experiments we consider a series of three tasks problems, selecting videos associated to three randomly chosen users but imposing the condition that videos corresponding to the three users should have at least three activities in common. We perform 10 different runs. In this series of experiments, we did not cluster video clips of fixed size as in the office dataset, but we consider the pre-segmented clips as provided with the dataset. In this scenario, it does not make sense to set w_{ij}^t as in CMTC to model temporal consistency. Therefore, as for in the synthetic data experiments, we set $w_{ij}^t = e^{-\|\mathbf{x}_i^t-\mathbf{x}_j^t\|^2}$ if $e^{-\|\mathbf{x}_i^t-\mathbf{x}_j^t\|^2} \leq \theta$ and $w_{ij}^t = 0$ otherwise.

Fig.7 shows the results (average accuracy) obtained with different clustering methods for both the bag-of-words and the temporal pyramid features representation. From Fig.7 it is evident that the MTC approaches outperforms their single task version (*e.g.* CMTC outperforms CMTC with $\lambda_2 = 0$, EMD-MTC outperforms CNMF, SemiEMD-MTC outperforms SemiNMF). On the other hand, our algorithms based on EMD regularization and CMTC achieve a considerably higher accuracy with respect to all the other methods. Fig.10 shows some temporal segmentation results on a sequence of the FPV home dataset comparing KM with the proposed methods. As discussed above, pre-segmented clips of different duration are considered here.

Finally, we investigate the effect of different values of the regularization parameters λ in (2) for EMD-MTC, λ_t and λ_2 in (5) for CMTC on clustering performance. As shown in Fig.8, independently from the adopted feature representation, the accuracy values are sensitive to varying λ. Fig.8 shows that choosing a value of $\lambda = 0.1$ in EMD-MTC and KEMD-MTC always leads to similar or superior performance with respect to adopting a single-task clustering approach ($\lambda = 0$). The value $\lambda = 0.1$ corresponds to the results reported in Fig.7. This clearly confirms the advantage of using a MTC approach for FPV analysis. Similar observations can be drawn in the case of CMTC. In Fig.9 we analyze how the accuracy changes at varying λ_t and λ_2. Note that in our previous experiments the parameters λ_t are fixed independently for each task according to the desired number of clusters. In this experiment instead we show that, independently from the chosen values for λ_t (*i.e.* the number of clusters) the best performance is typically obtained for $\lambda_2 \geq 0.1$, *i.e.* when the coherence between partitions of different tasks is enforced. For example, for temporal pyramid features, the higher accuracy is usually given by $\lambda_2 = 1$.

4.3 Discussion

In this paper we address the problem of automatically discovering activities of daily living from first-person videos. Currently, few datasets are publicly available for this task and, according to the recent survey in [43], the two datasets we consider [9, 10] are the only ones suitable. The other

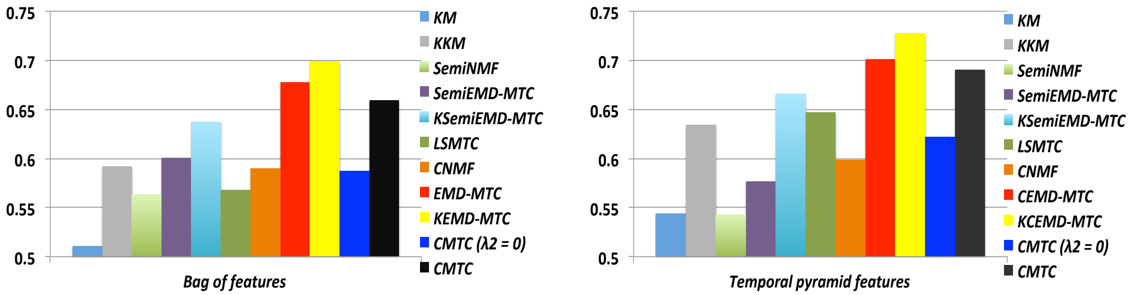


Fig. 7. Comparison of different methods using (left) bag of features and (right) temporal pyramid features on FPV home dataset. (Figure is best viewed in color).

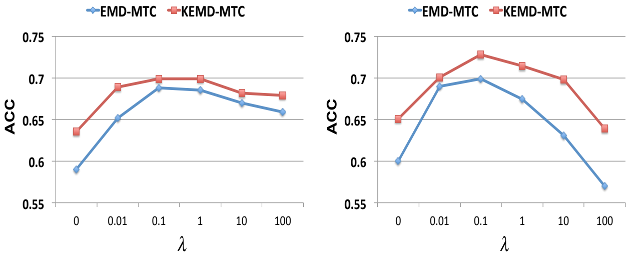


Fig. 8. FPV home dataset: performance variations of EMD-MTC and KEMD-MTC at different values of λ using (left) bag of features and (right) temporal pyramid features.

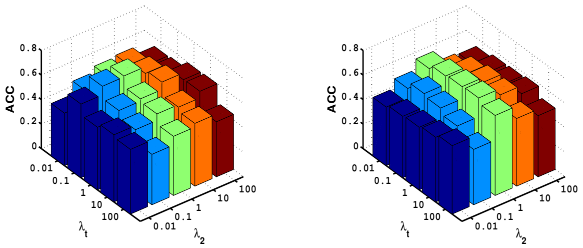


Fig. 9. Sensitivity study of parameters λ_t and λ_2 in CMTC using (left) bag of features and (right) temporal pyramid features.

datasets focus on different applications, *e.g.* food preparation or analysis of social interactions, and often do not have videos recorded from multiple users, as required by the proposed framework.

Regarding previous works using the same datasets [9, 10], it is worth noting that we consider an unsupervised setting. Previous works focused on a supervised scenario and therefore use different evaluation metrics. While a direct comparison is not possible, it is reasonable to expect that their methods are more accurate than our approach since they use labeled data for learning. However, recognizing everyday activities in absence of annotated data is especially important to automatically analyze videos recorded from wearable cameras.

As stated in the introduction, the proposed multi-task clustering approach is general and can be used in other applications. For example, our framework naturally applies to the problem of activity of daily living analysis when traditional cameras are used as an alternative to wearable sensors [3–5, 46].

5 Conclusions and Future Work

In this paper, we proposed a multi-task clustering framework to tackle the challenging problem of egocentric activity recognition. Oppositely to many previous works, we focused on the unsupervised setting and we presented two novel MTC algorithms: Earth Movers Distance Multi-Task Clustering and Convex Multi-task Clustering. We extensively evaluated the proposed methods on synthetic data and on two real world FPV datasets, clearly demonstrating the advantages of sharing informations among related tasks over traditional single task learning algorithms. Comparing the proposed methods, KEMD-MTC achieves the best performance, while CMTC is particularly advantageous when some *a-priori* knowledge about the data relationship is available. For example, in this paper we consider embedding temporal information about video clips but the CMTC method also permits to integrate other information about task dependencies by defining an appropriate matrix **B** (*e.g.* people performing the same activities in the same rooms may correspond to closely related tasks with respect to people operating in different rooms). Future work will focus on improving our MTC algorithms (*e.g.* by detecting outlier tasks) and on testing the effectiveness of the proposed methods for other vision applications.

Acknowledgments:

This work was partially supported by the MIUR Cluster project Active Ageing at Home, the EC project xLiMe and A*STAR Singapore under the Human-Centered Cyber-physical Systems (HCCS) grant.

References

[1] E. M. Tapia, S. S. Intille, and K. Larson, “Activity recognition in the home using simple and ubiquitous sensors,” in *Pervasive Computing*, 2004, pp. 158–175.

[2] P. Casale, O. Pujol, and P. Radeva, “Human activity recognition from accelerometer data using a wearable device,” in *Pattern Recognition and Image Analysis*. Springer, 2011, pp. 289–296.

[3] R. Messing, C. Pal, and H. Kautz, “Activity recognition using the velocity histories of tracked keypoints,” in *IEEE International Conference on Computer Vision*, 2009.

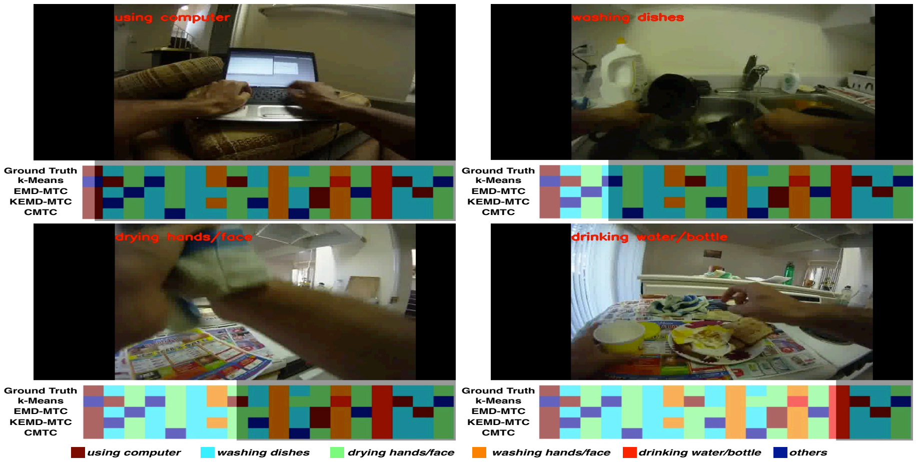


Fig. 10. Temporal video segmentation on a sequence of the FPV home dataset. (The edge of the shaded area at the bottom of each subfigure indicates the current frame).

[4] J. Lei, X. Ren, and D. Fox, “Fine-grained kitchen activity recognition using RGB-D,” in *ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2012.

[5] M. Rohrbach, S. Amin, M. Andriluka, and B. Schiele, “A database for fine grained activity detection of cooking activities,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[6] M. A. As’ari and U. U. Sheikh, “Vision based assistive technology for people with dementia performing activities of daily living (adls): an overview,” in *Int. Conf. on Digital Image Processing*, 2012.

[7] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, “Machine recognition of human activities: A survey,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[8] T. Kanade and M. Hebert, “First-person vision,” *Proceedings of the IEEE*, vol. 100, no. 8, pp. 2442–2453, 2012.

[9] H. Pirsiavash and D. Ramanan, “Detecting activities of daily living in first-person camera views,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[10] K. Ogaki, K. M. Kitani, Y. Sugano, and Y. Sato, “Coupling eye-motion and ego-motion features for first-person activity recognition,” in *CVPR Workshop on Egocentric Vision*, 2012.

[11] Y. Yan, E. Ricci, G. Liu, and N. Sebe, “Recognizing daily activities from first-person videos with multi-task clustering,” in *Asian Conference on Computer Vision*, 2014.

[12] B. Mahasseni and S. Todorovic, “Latent multitask learning for view-invariant action recognition,” in *IEEE International Conference on Computer Vision*, 2013, pp. 3128–3135.

[13] Y. Yan, E. Ricci, R. Subramanian, G. Liu, and N. Sebe, “Multi-task linear discriminant analysis for multi-view action recognition,” *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5599–5611, 2014.

[14] A. Omid, S. Josephine, and C. Stefan, “Novelty detection from an egocentric perspective,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[15] A. Fathi and J. M. Rehg, “Social interactions: A first-person perspective,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[16] E. Taralova, F. De la Torre, and M. Hebert, “Source constrained clustering,” in *IEEE International Conference on Computer Vision*, 2011.

[17] A. Fathi, Y. Li, and J. M. Rehg, “Learning to recognize daily actions using gaze,” in *European Conference on Computer Vision*, 2012.

[18] A. Fathi, A. Farhadi, and J. M. Rehg, “Understanding egocentric activities,” in *IEEE International Conference on Computer Vision*, 2011.

[19] Z. Lu and K. Grauman, “Story-driven summarization for egocentric video,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[20] M. S. Ryoo and L. Matthies, “First-person activity recognition: What are they doing to me?” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[21] Y. Poleg, C. Arora, and S. Peleg, “Temporal segmentation of egocentric videos,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[22] R. Caruana, “Multitask learning,” *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[23] Y. Yan, Y. Yang, D. Meng, G. Liu, W. Tong, A. Hauptmann, and N. Sebe, “Event oriented dictionary learning for complex event detection,” *IEEE Transactions on Image Processing*, vol. 24, no. 6, pp. 1867–1878, 2015.

[24] R. Salakhutdinov, A. Torralba, and J. Tenenbaum, “Learning to share visual appearance for multiclass object detection,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[25] Y. Yan, E. Ricci, R. Subramanian, O. Lanz, and N. Sebe, “No matter where you are: Flexible graph-guided multi-task learning for multi-view head pose classification under target motion,” in *IEEE International Conference on Computer Vision*, 2013.

[26] Y. Luo, D. Tao, B. Geng, C. Xu, and S. Maybank, “Manifold regularized multitask learning for semi-supervised multilabel image classification,” *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 523–536, 2013.

[27] A. Argyriou, T. Evgeniou, and M. Pontil, “Multi-task feature learning,” in *Conference on Advances in Neural Information Processing Systems*, 2007.

[28] A. Jalali, P. Ravikumar, S. Sanghavi, and C. Ruan, “A dirty model for multi-task learning.” in *Conference on Advances in Neural Information Processing Systems*, 2010.

[29] J. Chen, J. Zhou, and J. Ye, “Integrating low-rank and group-sparse structures for robust multi-task learning,” in *ACM SIGKDD International conference on Knowledge discovery and data mining*, 2011.

[30] L. Jacob, F. Bach, and J. Vert, “Clustered multi-task learning: A convex formulation,” in *Conference on Advances in Neural Information Processing Systems*, 2008.

[31] Y. Zhang and D. Yeung, “A convex formulation for learning task relationships in multi-task learning,” in *Conference on Uncertainty in Artificial Intelligence*, 2010.

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

[33] B. Mahasseni and S. Todorovic, “Latent multitask learning

for view-invariant action recognition." in *IEEE International Conference on Computer Vision*, 2013.

[34] C. Yuan, W. Hu, G. Tian, S. Yang, and H. Wang, "Multi-task sparse learning with beta process prior for action recognition." in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[35] Q. Gu and J. Zhou, "Learning the shared subspace for multi-task clustering and transductive transfer classification," in *IEEE International Conference on Data Mining*, 2009.

[36] B. Kulis and M. I. Jordan, "Revisiting k-means: New algorithms via bayesian nonparametrics," in *International Conference on Machine Learning*, 2012.

[37] J. Zhang and C. Zhang, "Multitask bregman clustering," *Neurocomputing*, vol. 74, no. 10, pp. 1720–1734, 2011.

[38] Y. Rubner, C. Tomasi, and L. J. Guibas, "A metric for distributions with applications to image databases." in *IEEE International Conference on Computer Vision*, 1998.

[39] C.-J. Lin, "Projected gradient methods for non-negative matrix factorization," *Neural Computation*, vol. 19, pp. 2756–2779, 2007.

[40] J. Eggert and E. Korner, "Sparse coding and NMF," *Neural Networks*, vol. 4, pp. 2529–2533, 2004.

[41] K. Borgwardt, A. Gretton, M. Rasch, H.-P. Kriegel, B. Schoelkopf, and A. Smola, "Integrating structured biological data by kernel maximum mean discrepancy," *Bioinformatics*, vol. 22, no. 14, pp. 1–9, 2006.

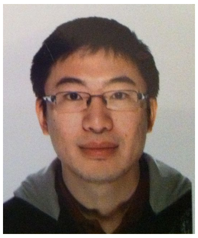
[42] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2011.

[43] S. Song, V. Chandrasekhar, N.-M. Cheung, S. Narayan, L. Li, and J.-H. Lim, "Activity recognition in egocentric life-logging videos," in *Int. Workshop on Mobile and Egocentric Vision, Asian Conference on Computer Vision*, 2014.

[44] A. Bulling, J. A. Ward, H. Gellersen, and G. Troster, "Eye movement analysis for activity recognition using electrooculography," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 4, pp. 741–753, 2011.

[45] C. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 1, pp. 45–55, 2010.

[46] C. Wolf, E. Lombardi, J. Mille, O. Celiktutan, M. Jiu, E. Dogan, G. Eren, M. Baccouche, E. Dellandréa, C.-E. Bichot *et al.*, "Evaluation of video activity localizations integrating quality and quantity measurements," *Computer Vision and Image Understanding*, vol. 127, pp. 14–30, 2014.



Yan Yan received the PhD from the University of Trento in 2014. Currently, he is a Postdoctoral researcher in the MHUG group at the University of Trento. His research interests include machine learning and its application to computer vision and multimedia analysis.

Elisa Ricci is an assistant professor at University of Perugia and a researcher at Fondazione Bruno Kessler. She received her PhD from the University of Perugia in 2008. During her PhD she was a visiting student at University of Bristol. After that she has been a post- doctoral researcher at Idiap, Martigny and the Fondazione Bruno Kessler, Trento. Her research interests are mainly in the areas of computer vision and machine learning.

Gaowen Liu received B.S. degree in Automation from Qingdao Univerisity, China, in 2006, M.S. degree in System Engineering from Nanjing University of Science and Technology, China, in 2008. She is currently a Ph.D. candidate in MHUG group at the University of Trento, Italy. Her research interests include machine learning and its application to computer vision and multimedia analysis.

Nicu Sebe is a professor at University of Trento, Italy, leading the research in the areas of multimedia information retrieval and human behavior understanding. He was General Co- Chair of the IEEE FG Conference 2008 and ACM Multimedia 2013, and Program Chair of the International Conference on Image and Video Retrieval in 2007 and 2010 and ACM Multimedia 2007 and 2011. He is Program Chair of ECCV 2016 and ICCV 2017. He is a fellow of IAPR.