IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 24, 2020, accepted April 18, 2020, date of publication April 27, 2020, date of current version May 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990333

Activities of Daily Living Monitoring via a Wearable Camera: Toward Real-World Applications

ALEJANDRO CARTAS[1], PETIA RADEVA[1], AND MARIELLA DIMICCOLI[2]
[1]Mathematics and Computer Science Department, University of Barcelona, 08007 Barcelona, Spain
[2]Institut de Robòtica i Informàtica Industrial, CSIC-UPC, 08028 Barcelona, Spain

Corresponding author: Alejandro Cartas (alejandro.cartas@ub.edu)

This work was supported in part by the TIN2018-095232-B-C21, in part by the SGR-2017 1742, in part by the Nestore ID: 769643, in part by the Validithi EIT Health Program, in part by the CERCA Programme/Generalitat de Catalunya, in part by the Spanish Ministry of Economy and Competitiveness, and in part by the European Regional Development Fund (MINECO/ERDF, EU) through the program Ramon y Cajal. The work of Alejandro Cartas was supported by a doctoral fellowship from the Mexican Council of Science and Technology (CONACYT) under Grant 366596.

ABSTRACT Activity recognition from wearable photo-cameras is crucial for lifestyle characterization and health monitoring. However, to enable its wide-spreading use in real-world applications, a high level of generalization needs to be ensured on unseen users. Currently, state-of-the-art methods have been tested only on relatively small datasets consisting of data collected by a few users that are partially seen during training. In this paper, we built a new egocentric dataset acquired by 15 people through a wearable photo-camera and used it to test the generalization capabilities of several state-of-the-art methods for egocentric activity recognition on unseen users and daily image sequences. In addition, we propose several variants to state-of-the-art deep learning architectures, and we show that it is possible to achieve 79.87% accuracy on users unseen during training. Furthermore, to show that the proposed dataset and approach can be useful in real-world applications, where data can be acquired by different wearable cameras and labeled data are scarcely available, we employed a domain adaptation strategy on two egocentric activity recognition benchmark datasets. These experiments show that the model learned with our dataset, can easily be transferred to other domains with a very small amount of labeled data. Taken together, those results show that activity recognition from wearable photo-cameras is mature enough to be tested in real-world applications.

INDEX TERMS Daily activity recognition, visual lifelogs, domain adaptation, wearable cameras.

I. INTRODUCTION

Activity recognition through wearable devices has been largely investigated in the past fifteen years [1]. While early works were mostly based on the use of simple wearable sensors such as accelerometers and heart monitors, during the last decade, a wide variety of sensors have been incorporated into different and more sophisticated types of wearable devices, ranging from motion to radar sensors.

The use of wearable cameras in the context of activity recognition began only very recently. Being small and lightweight, wearable cameras are ubiquitous and can autonomously record data without human intervention during long periods of time. Unlike other wearable sensors, they capture external and directly interpretable information, such as places, objects, and people around the user. With respect to fixed cameras, wearable ones can daily gather large amounts of human-centric data in a naturalistic setting, hence offering rich contextual information about the activities of the user. As a consequence, activity recognition from wearable cameras has several important applications as assistive technology, in particular in the field of rehabilitation and preventive medicine. Examples include self-monitoring of ambulatory activities of elderly people [2], [3], monitoring patients suffering dementia [4], [5], determining sedentary behavior of a user based on their spent time watching TV [6].

However, the opportunities for activity recognition from wearable cameras come along with several challenges as well.

The associate editor coordinating the review of this manuscript and approving it for publication was Peng Liu.

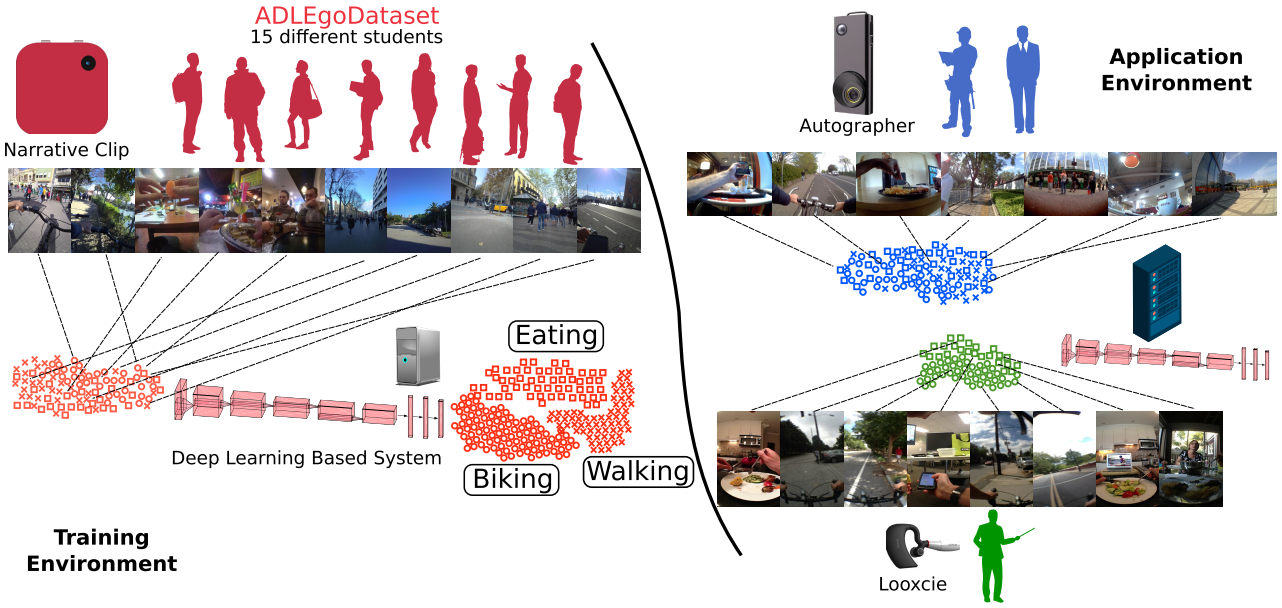

FIGURE 1. Our activity of daily living monitoring system via a wearable camera: A deep neural network architecture is trained on images coming from a specific camera and, more importantly, from a given group of people (ADLEgoDataset). We show that this deep neural network can be successfully used on pictures captured by distinct cameras and/or unseen people with different lifestyles (jobs/hobbies/cultures), needing just a very small amount of new labeled data; consequently, the system might be deployed in real-world applications, where typically the training data distribution differs substantially from the target data distribution.

The main one is to predict the user activities based not on the observation of camera wearer himself (with his body-pose, gestures, etc.), but on her/his context: the objects he/she is manipulating, other people around he/she is interacting with, and the environment itself. Additionally, first-person (egocentric) images suffer huge intra-class variation, due to the camera user not being static and also acting in a large variety of real-world scenarios. Even more, the lighting conditions are not fixed since the camera can be worn in indoor and outdoor settings at different times of the day. Lifelogging photo-cameras present yet another specific challenge with respect to wearable video-cameras. They continuously take pictures at regular intervals of 20-30 seconds instead of videos with a high number of frames per second, generating image sequences with a low frame rate, typically called *visual lifelogs* or *photo-streams*. Therefore, motion estimation, that is useful to describe the scene [7] and disambiguate actions/activities [8], become infeasible on such data.

Besides these technical challenges, recent work has shown very good performance on the task of activity recognition from visual lifelogs. This has been achieved mainly by leveraging deep learning architectures aiming at capturing the temporal evolution of semantic features over time, together with their contextual information [9].

However, as noticed in [10], these methods would need a more extensive validation on a larger scale dataset and on unseen users before being deployed in real-world applications. Indeed, in real scenarios, the distribution of the training, also called *source*, typically differs from the distribution of new data, also called *target*. For instance, this is always the case when the target data are acquired by a different wearable camera than the source data, or when the target data have been collected by people having a very different lifestyle than those who collected the source data, i.e. having different jobs/hobbies and living in different countries. In addition, new data can be unlabeled or scarcely labeled. Therefore, ensuring performance on unseen users from the same domain does not assure that the model could be employed in real-world applications. In addition, to guarantee the robustness of the method, performance should keep stable on larger and more varied datasets. To the best of our knowledge, currently, there are not large scale dataset of activity recognition from visual lifelogging. This is mainly due to several difficulties to be handled: bystander and user privacy concerns during data collection, the huge effort of the tedious manual annotation process, the lack of a standardized action/activity vocabulary, and the inherent ambiguity of the data annotation itself.

To cope with all these needs for real-world deployment, we first collected a large egocentric dataset acquired through a wearable photo-camera and we used it to validate for the first time the generalization capabilities of five existing methods for egocentric activity recognition on unseen users. In addition, we quantified the effectiveness of using together images from different domains in the same training/test setup. Furthermore, we show that the model trained on our dataset can be easily transferred to other domains (i.e. datasets collected by other wearable cameras, different users, etc.), achieving competitive performance with a small amount of labeled images. An overview of the above described capabilities of our system is given in Fig. 1.

More specifically, our contributions in this paper are three-fold:

(*i*) The collection, annotation, and release of a large egocentric dataset of Activity of Daily Living (ADLEgo-Dataset[a]) consisting of 102,227 images, from 15 users, with an average of 6,682 images per user.

(*ii*) The ranking of state-of-the-art algorithms in dealing not only with unseen full day sequences but also unseen users during training. This ranking also provides a strong baseline for our newly introduced dataset.

(*iii*) A set of experiments using the correlation alignment (CORAL) adaptation method [11], [12] showing that the model learned with our dataset can be easily and successfully transferred to other existing datasets acquired by two different wearable cameras (i.e. NTCIR-12 [13], [14], and Castro *et al.*'s [15] datasets), providing competitive results with a very little amount of labeled data.

The rest of the paper is organized as follows. First, in Section II we review the related work. Next, in Section III, we introduced our ADLEgoDataset. In Section IV, we present a classification baseline using state-of-the-art algorithms on our dataset. In Section V, we present our model and experiments on how to transfer the learned model to other domains. Finally, we present our conclusions in Section VI.

II. RELATED WORK

A. ACTIVITY RECOGNITION FROM FIXED CAMERAS

The standard pipeline of human action recognition was introduced in the seminal work of Yamato *et al.* [16]. This pipeline consists of first extracting feature vectors from a sequence of frames and then predicting an action based on them by using a classifier. This general approach has been extensively used in the past by varying the hand-crafted features and the type of classifier, and it is still in use nowadays [17]. This Computer Vision task, along with several others, has made great strides since the introduction of deep convolutional neural networks (CNNs) [18]. These networks learn feature representation from images and their classification in an end-to-end fashion. Over the last seven years, new architectures that improve their efficiency and accuracy have been presented [19]–[23]. Although CNNs do not model the temporal order of frames from the sequence, temporal learning mechanisms have been used on the top of them, i.e. fusion mechanisms [24], three-dimensional convolutional layers [25], and long short-term memory (LSTM) units [26], [27]. Specific deep architectures for action recognition have combined optical flow as an additional stream [28], and, later on, multimodal information such as audio [29]. In this work, our attention is set on activity recognition from wearable cameras. Its main difficulty is that the person himself is only partially visible in the images through his hands. Although the approaches detailed above have been adapted to this kind of camera, other methods have been proposed that rely solely on the user interactions with objects, other people, and the scene. These methods are described below.

[a]The dataset is publicly available at http://www.ub.edu/cvub/adlegodataset

B. ACTIVITY RECOGNITION FROM EGOCENTRIC VIDEOS

Several works on first-person action recognition from videos have focused on exploiting egocentric features. These features include the location of hands [30]–[32], the interaction with active/passive objects [33]–[38], the head motion [39], [40], the gaze [41]–[45], or a combination of them [46]–[48]. Other methods have explored egocentric contexts like social interactions [49] and the temporal structure of the activities [50]–[52]. Additionally, some approaches have adapted deep third-person action recognition methods [53]–[55] and developed new ones based on reinforcement learning [7]. In this work, we focus on activity recognition from visual lifelogs. In contrast with egocentric videos, they cover longer time periods with a low temporal resolution, hence being suitable for several applications of assistance technology [2]–[6]. Nevertheless, most of the approaches described above cannot be used on visual lifelogs because motion and gaze based features cannot be reliably estimated on such data.

C. ACTIVITY RECOGNITION FROM VISUAL LIFELOGS

Initial work on first-person action recognition from visual lifelogs was presented by Castro *et al.* [15]. Their approach, based on a late fusion strategy applied at frame-level, combines the output of a CNN with color histograms and timestamps. These additional contextual features are justified by the fact that a person typically performs activities such as *cooking* in the same place and about the same time per day. However, this approach has been tested on a dataset acquired by a single user and makes sense only for a single user or several users having the same lifestyle (similarly working hours, same job, etc). A generalized version of this method was proposed in [60], where the outputs of different layers from a CNN were combined to extract more general contextual information. More recent work [9] modeled lifelogs as sequences instead of a set of unrelated images and proposed two methods based on LSTMs for exploiting the temporal evolution of contextual features over time. Recently, information from different wearable devices, including a camera, was integrated using multimodal approaches for activity recognition. While these methods are promising and are tested on unseen users, they typically rely on off-the-shelf architectures for the visual modality [61], [62]. In this work, we provide a solid proof of the generalization capabilities of several state-of-the-art architectures for activity recognition from visual lifelogs by validating them on a new, large visual lifelog dataset.

D. DOMAIN ADAPTATION

Domain adaptation (DA), also known as the *dataset shift problem* [63] and mathematically formalized in [64], deals with scenarios where a model trained on a source distribution does not generalize well in the context of a different

TABLE 1. Comparative overview of existing egocentric lifelogs datasets for action recognition. The activities are grouped into categories according to [1]. The highest attribute values are highlighted in bold.

Dataset	Camera	Body Location	#Frames	#Days	#People	#Classes	#Annotated Frames	Activity Groups
NTCIR-12 [13], [14]	Autographer	Chest	90k	90	3	6 [13]	13.8k [13]	Ambulation, Transportation
						21 [14]	44.9k [14]	Ambulation, Transportation, Device Usage, Daily Activities
NTCIR-13 [56]	Narrative Clip	Chest	**110k**	90	2	4	5.8k	Ambulation, Transportation
ImageCLEF Lifelog 2018 [57]	Narrative Clip	Chest	80k	60	1	2	5.1k	Ambulation, Transportation
NTCIR-14 [58]	Autographer	Chest	81k	43	2	2	8.9k	Ambulation, Transportation
Castro, et al. [15], [59]	Looxcie	Ear	40k	182	1	17	39.1k	Transportation, Daily Activities, Exercise/Fitness
ADLEgoDataset	Narrative Clip	Chest	105k	**191**	**15**	**35**	**100k**	Ambulation, Transportation, Device Usage, Daily Activities, Exercise/Fitness

(but related) target distribution. Two of the currently most predominant approaches to address the DA problem are based on the two-stream deep architecture first presented in [65]. Each of the streams represents the source and target model, respectively. A carefully designed domain regularization loss is employed to adapt the source to the target domain. One approach is to reduce the shift between domains using a discrepancy metric such as the maximum mean discrepancy (MMD) [65]–[68], the central moment discrepancy [69], [70], the correlation alignment (CORAL) function [12], and the Wasserstein metric [71]. Inspired by [72], another successful approach is to find a common feature space using adversarial training [73]–[76]. For example, in [74], a source encoder CNN is trained and its weights are subsequently fixed to train a target encoder. The adversarial training of the target encoder aims to deceive a domain discriminator between samples from both domains. Along with the same approach, Ganin and Lempitsky [73] simultaneously trained a generator and a discriminator by inverting the gradients using a special layer.

In this work, with the aim of measuring the effectiveness of our model on data acquired by different cameras and people, and hence having a different distribution with respect to the training data, we use a DA technique on our proposed dataset, i.e. the source domain, and two other available datasets [13]–[15], i.e. the target domains. Although DA is characterized by not having labeled data on the target domain, we consider it in a semi-supervised context, where different amounts of labeled target examples are taken into account.

III. ACTIVITIES OF DAILY LIVING EGOCENTRIC DATASET

A. RELATED DATASETS

Although several egocentric datasets for action recognition have been published in the last years [77], [78], most of them were recorded using video cameras. Since these devices have much higher energy consumption than lifelog cameras, each video in these datasets do not cover actions from whole days but capture up to a few hours. Furthermore, considering the obtrusiveness of the cameras, that are typically mounted on the head, most of these datasets only include actions in specific, often indoor, environments. For instance, several existing datasets have focused on tasks like cooking [41], [43], [50], [78], [79], interacting with a toy in a laboratory [49], working [80], [81], or performing indoor daily activities [35], [82]. Only a few datasets captured outdoor activities such as basketball [2] or ambulatory activities [83].

During the last five years, a reduced number of egocentric visual lifelog datasets for action recognition has been introduced. Unlike the egocentric video datasets described above, these datasets cover full-day activities performed in a larger variety of settings. Both characteristics made the lifelogging data collection more difficult to acquire. First, it requires longer recording times that also makes the process more expensive. Second, recording several locations and people during a day has more privacy restrictions than indoor locations. One of the first datasets was introduced in [15] and released in [59]. It describes the life of only one graduate student using 19 different activities, therefore it does not allow to test generalization capabilities on other users. Several other datasets have been presented in the context of image retrieval challenges [13], [56]–[58]. Although they capture images from several weeks, the number of originally annotated classes and images is low and mostly describes transportation and ambulation activities. The life of three unrelated subjects was presented in the NTCIR-12 challenge [13]. Although it was independently annotated with 21 daily activity labels by [14], it only considers three people. Another dataset for image retrieval consisting of the annotated moments of two people [56] was released and further labeled in terms of four different activities. This dataset was further used for another image retrieval task in [57]. Finally, a dataset consisting of two subjects and two distinct activities was introduced for the NTCIR-14 challenge [58]. The characteristics of the above described datasets and ours are summarized in Table 1. This Table not only considers the number of people, annotated classes, and images; but also the

number of day lifelogs. This latter number is relevant since a robust performance evaluation on frame-sequence data must be done on full sequences and not only frames. Moreover, Table 1 also highlights the diversity of the activities of our dataset as having classes belonging to more activity groups among the ones proposed in [1]. The main difficulties with existing visual lifelog datasets are: (*i*) the small number of users and lifelog sequences, that prevent to thoroughly test the generalization capabilities of machine learning methods for activity recognition; (*ii*) the limited number of activity categories and their diversity.

Here, we introduce our ADLEgoDataset, a collection of 105,529 images describing the lifestyle of fifteen post-graduate students. In comparison with previous visual lifelog datasets, the activities are not constrained to a specific domain and occurred in a wide variety of indoor and outdoor locations of a city. The set of activity labels is based on previous works [14], [15], [59] and further expanded to 35 activities, thus adding 14 more categories. Moreover, the number of users is greater than in existing datasets by 12 people as seen in Table 1, allowing them to perform a generalization test.

B. DATA COLLECTION

The data was collected by fifteen computer science post-graduates students who wore a lifelogging camera using a lanyard hanging around the neck. The number of female and male participants were 3 and 12, respectively. The collected pictures depict different outdoor and indoor locations across one city. The common place for all the participants was the university where they work or study.

The participants were instructed to perform their daily activities while wearing the camera during whole days. However, they were allowed to put away the camera on situations that they considered private, e.g. using the toilet. They were asked to use the camera in a minimum period of 10 days. For privacy concerns, all participants were allowed to discard pictures that they considered sensitive, even images from whole days.

We used the first and second versions of the Narrative Clip camera, but only two people wore the first version. Both cameras automatically take a picture at $\approx$ 30 seconds rate, but their main difference is that the latter has a wider field of view and an 8 megapixels resolution instead of 5. They can operate in a period of 10 to 12 hours without a battery recharge, thus allowing to capture between 1,200 and 1,900 images per day.

The selected categories for the dataset are general activities from five different egocentric groups [1], as seen on Table 2. The activity labels were based on previous works [14], [15], [59], but they were not specifically targeted to model the student lifestyle and were selected after the recording. As an illustration, the original number of categories proposed for annotation included a broader set of activities such as *child rearing*, *praying*, *painting* or *meditating*. However, they were not chosen by any participant during the annotation process.

TABLE 2. Distribution of the 35 activity categories in our dataset according to the proposed groups in [1].

Activity Group	Label	Number of Images
Ambulation	Stairclimbing	277
	Walking indoors	9,093
	Walking outdoors	9,994
Transportation	Airplane	930
	Bus	1,042
	Car (not driving)	1,267
	Cycling	894
	Driving	1,332
	Train/Metro	2,506
Device Usage	Using tablet/cellphone	5,150
Daily activities	Attending a Seminar	2,079
	Cleaning	154
	Cooking	976
	Dishwashing	429
	Drinking (alone)	823
	Drinking (not alone)	1,122
	Eating (alone)	2,290
	Eating (not alone)	6,458
	Formal meeting	1,467
	Going to a bar	1,024
	Hobby	552
	Hygiene	467
	Laundry	72
	Paying	181
	Playing an instrument	110
	Pets	212
	Reading	953
	Relaxing	1,086
	Shopping	1,919
	Talking	5,499
	Using a computer	38,425
	Watching TV	915
	Writing	266
Exercise/Fitness	Gym	450
Other	Undetermined	5,115

C. ANNOTATION PROCESS

Most of the participants were also involved in the annotation process since they are the best judges to determine not only what they were doing, but also when an activity started and ended. The correct activity boundaries in a lifelog sequence are important because the temporal context of between frames provides more information in the case of occlusions from single frames. For instance, in a *cycling* sequence a frame might not show the bicycle steering wheel and could be classified as *walking outside*. We used the batch-based annotation tool introduced in [60]. Finally, each recognizable face of people not directly involved in the data collection was manually blurred.

D. DATASET DETAILS

We collected over 105,529 pictures from 15 college students and researchers, covering in total 191 days and 35 activities. These activities belong to five of the seven categories presented in [1], as seen on Table 2. The young student lifestyle is implicitly reflected on the number of instances of each activity, for instance, the times the labels *used a computer* and *gone to a bar* frequently appear. The only location in common

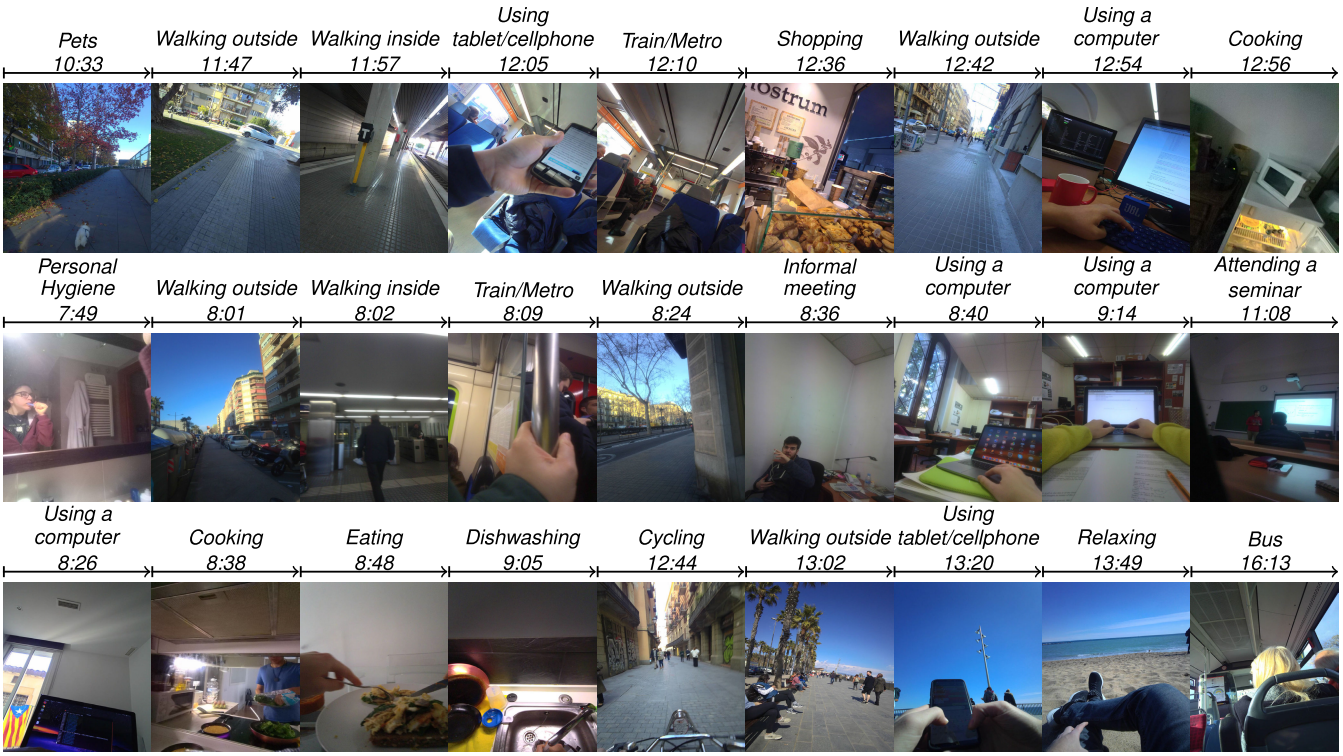

FIGURE 2. **Sampled pictures captured by a wearable photo-camera during a day. Each row depicts images from a different person annotated with their corresponding activity and time.**

for all volunteers was the university and most of the time they did not meet while wearing the camera. Fig. 2 illustrates different settings and activity sequences of three of the users. Each participant wore the camera a different number of days and times, and the mean number of days that the participants wore the camera was 14.6, resulting in 6,816.73 images on average.

IV. ACTIVITY RECOGNITION FROM LIFELOGS

In this section, we aim at ranking the generalization capability of state-of-the-art algorithms for activity recognition from visual lifelogs. Since our focus is only on visual information, we selected algorithms that do not rely on additional data from other sensors as [61], [62]. Our baseline considers two classification approaches for visual lifelogs: still images and image sequence based approaches. The first scenario consists in determining the activity a person is doing from a single frame; whereas the second scenario takes as input images from a full-day sequence that typically covers several daily activities.

We selected two still image classification methods as a baseline. The first is a convolutional neural network (CNN) that serves as a backbone for the rest of the algorithms. Specifically, we used ResNet-50 as backbone network. The other method is a late fusion ensemble that was introduced by Castro *et al.* [15] and further generalized by Cartas *et al.* [60]. Their approach consists of combining different output layers from a CNN using a random forest (RF) as a final classifier, thus named CNN+RF. Concretely, we combined the outputs of the average pooling and the fully-connected layers.

In the case of image sequences, we evaluated the two temporal training approaches presented in [9]. These approaches extract the contextual features from a CNN and use LSTMs as a sequence learning mechanism. The difference between these approaches consists in their training strategy. The first approach trains directly over the *full-day image sequence*. The second approach trains using a fixed number of LSTM units and sampling a day sequence in a *sliding window* fashion. Specifically, we tested both LSTM training strategies using as input feature extractors the CNN and CNN+RF methods described above. In order to make a fair comparison between the features extracted from CNN and CNN+RF, the CNN weights were frozen during the training of LSTM.

In addition to these image sequence approaches, we also consider an LSTM variant as a temporal learning mechanism. Namely, we combined the encoding produced by a CNN with a Bidirectional LSTM (BLSTM) [84]. This kind of Recursive Neural Network (RNN) evaluates a sequence in forward and backward order and merges the result. Thus, it captures patterns that might have been missed by the unidirectional version and that can lead to potentially more robust representations. We implemented the CNN+BLSTM and CNN+RF+BLSTM methods using the same training approaches described above. All our ranking baseline models are depicted in Fig. 3.

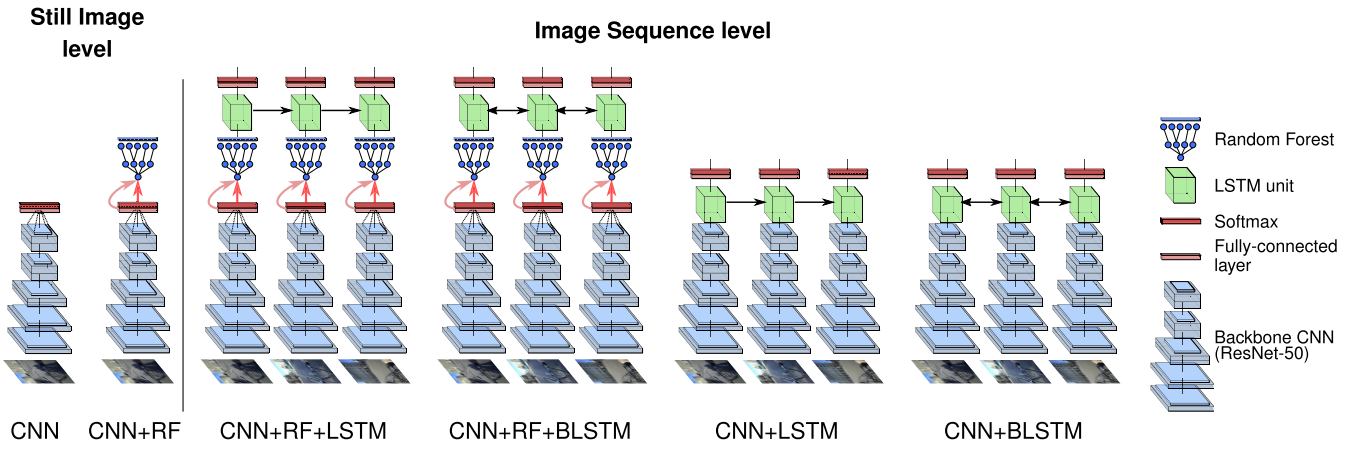

FIGURE 3. Ranking baseline models. All models used ResNet-50 as backbone convolutional network [20]. In order to have a fair comparison, after fine-tuning the CNN baseline model, it was used as a feature extractor for the rest of the evaluated methods.

With the aim of having a more realistic testing setting than previous works [9], [15], [60], we performed a special split on the ADLEgoDataset. Specifically, the test split not only considers multiple seen users and their unseen day sequences, but also different unseen users during training.

We first detail the dataset split in Section IV-A. Next, in Section IV-B, we outline the implementation details of the state-of-the-art algorithms and their bidirectional counterparts. Finally, we discuss the evaluation metrics and results of our experiments in Section IV-C.

A. DATASET SPLIT

Our goal in doing the training and testing partitions was to make possible the evaluation of generalization capabilities of several state-of-the-art algorithms on the ADLEgoDataset. In comparison with previous works [15], [59], we did not randomly and proportionally split each category of the data. Indeed, this kind of training/testing partition is not reliable on sequential data since consecutive frames depicting similar information might be present in both partitions. Therefore, instead of hiding single random frames from the training split, we selected in a test split full-day sequences from *seen users* during training. This selection was made as proportional as possible with respect to the categories since it had to be representative of the dataset. In contrast with [14], we considered that this kind of partition is not enough to assess the generalization performance, because similar days might depict similar activities in the same context of a person. Consequently, we made another test split consisting of *unseen users* during training. This test split was not constrained to be representative of the training split. The data percentage of the seen and unseen test users was around 10% and 5%, respectively. Moreover, in this experiment we discarded the activity categories that had less than 200 instances or that were performed by only one user, except for four categories (*airplane*, *cleaning*, *gym*, and *pets*). These categories were also considered for further comparisons on the experiments in Section V.

We first created the *unseen users* split because it reduced the complexity of the *seen users* split. The procedure is detailed as follows:

1) UNSEEN USERS SPLIT

First, we calculated all the possible combinations of unseen users from the 15 users (i.e. 32,767) by using the Twiddle algorithm [85]. Then we calculated the total number of images for each combination, and filtered the ones that did not have between 4.5% and 5% of images from the total amount of images in the ADLEgoDataset. Finally, we selected the combination with the lowest number of participants.

2) SEEN USERS SPLIT

This split is focused on separating complete days of images (or *full-day sequences*) from users, rather than separating users. A full-day sequence is composed of several images with different activity labels from one user. The objective of this test split is to separate full-day sequences from the training that maintains a similar category distribution as the whole dataset and thus being representative of what it is intended to learn. We measure the similarity between category distributions using the Bhattacharyya distance.

After removing the unseen users from the dataset, the remaining number of users is 9 and their number of full-day sequences is 103. By counting the number of images from each full-day sequence, 10% of the dataset for the split is obtained by selecting between 6 and 32 full-day sequences. We considered that the most representative full-day sequences are the ones with the closest category distribution with respect to the whole dataset. Consequently, finding it involves comparing the category histograms between the whole dataset and all possible combinations of full-day sequences. Although the number of test days is low, the search is prohibitively expensive as is characterized by combinatorial growth. For instance, the number of test sets considering 6 days out of the 103 is $\approx 1.42 \times 10^9$, but for 32 days out of the 103 is $\approx 4.42 \times 10^{26}$.

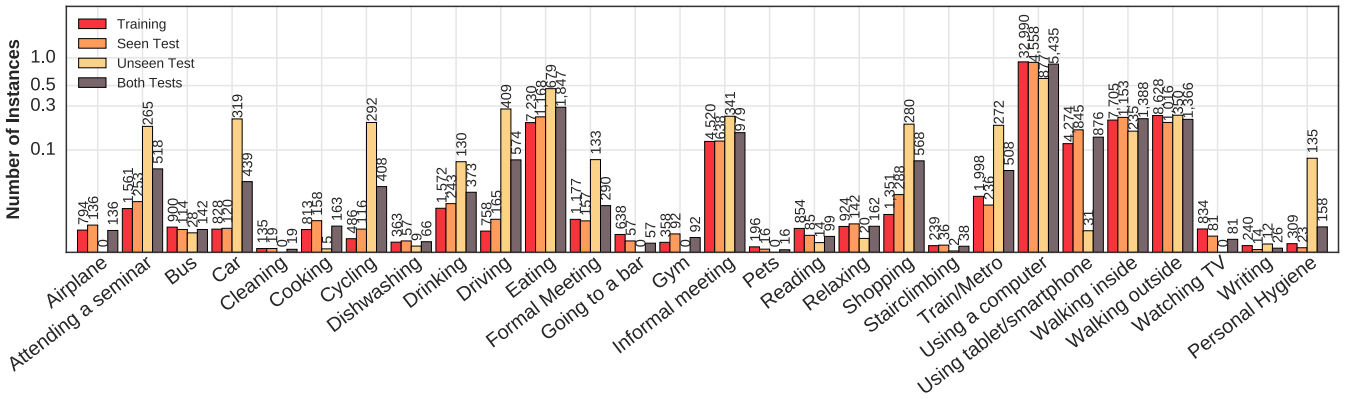

FIGURE 4. Training and test sets summary. Our splitting method generated data splits with similar distribution shapes, except for the unseen users split. Note that the distributions are normalized and their vertical axis has a logarithm scale.

Finding the best full-day sequences for the split was performed as a two-step optimization search using heuristics. With the goal of reducing the search space, instead of dealing with single full-day sequences, we first grouped them into *bins* of one or more full-day sequences. This was modeled as a bin packing problem, where the objects were full-day sequences, and their weight was its number of images. To further reduce the search space by half, these bins were matched in pairs with similar category distributions. The idea is that one bin was destined for the training set and the other for the test set. The resulting number of bin pairs was 32 containing between one and two full-day sequences.

The second step evaluated all test split candidates to find the most similar to the ADLEgoDataset distribution. A test split candidate is a combination of bin pairs that contains all activity categories and its number of images is approximately 10% of the data. The distributions of all our final splits are depicted in Fig. 4. It shows that all split distributions have a similar shape, except for the unseen users split because it considered random users as described above.

B. RANKING IMPLEMENTATION

1) STILL IMAGE LEVEL

We trained the following two models for static image level classification:

1) *CNN*. We used ResNet-50 [20] as CNN network and replaced the top layer with a fully-connected layer of 28 outputs. The fine-tuning procedure used Stochastic Gradient Descent (SGD) and a class-weighting scheme based on [86] to handle class imbalance. Moreover, the last ResNet block and the only fully connected (FC) layer were unfrozen. The CNN initially used the weights of a pre-trained network on ImageNet [87]. It was trained during 7 epochs using a learning rate $\alpha = 1 \times 10^{-2}$, a learning rate decay of 5×10^{-4}, a momentum $\mu = 0.9$, and a weight decay equal to $\alpha = 1 \times 10^{-3}$.
2) *CNN+RF*. Two random forests were trained using the output of different layers from the previously described ResNet-50 network. Specifically, the first RF was trained using as input the features extracted from the average pooling layer. The other RF uses the average pooling layer plus the concatenation of the FC layer. The number of trees was set to 500 and used the Gini impurity criterion [88].

2) IMAGE SEQUENCE LEVEL

The following outlined models take into account temporal information and use as backbone the previously trained models. We used as temporal architectures the LSTM and BLSTM networks. Following [9], the training of each model was performed in two ways by treating differently an input day lifelog sequence. The first training strategy operates directly over a day lifelog, i.e. over the full day sequence. The second training strategy truncates a day lifelog sequence into fixed-size subsequences in a *sliding window* fashion.

With the purpose of making a fair comparison, their weights and outputs of the backbone models were frozen during training. All the models were trained using the SGD optimization algorithm using different learning rates but the same momentum $\mu = 0.9$, weight decay equal to $\alpha = 5 \times 10^{-6}$, batch size of 1, and a timestep of 5.

1) *CNN+LSTM* and *CNN+BLSTM*. These models removed the top layer of the ResNet-50 network and respectively added a LSTM and BLSTM layer having 256 units, followed by a fully-connected layer of 28 outputs. For both models, the learning rates of the *full sequence* and the *sliding window* training were $\alpha = 1 \times 10^{-2}$ and $\alpha = 1 \times 10^{-3}$, correspondingly.
2) *CNN+RF+LSTM* and *CNN+RF+BLSTM*. Both models were trained using as input the prediction of the best *CNN+RF* model, namely the combination of the avg. pooling and the FC layers. These models respectively added an LSTM and BLSTM layer having 30 units, followed by a fully-connected layer of 28 outputs. The learning rate for both models and types of training was $\alpha = 1 \times 10^{-3}$.

TABLE 3. Activity classification performance metrics for all models. Best result per measure is shown in **bold**.

		STILL IMAGE LEVEL			IMAGE SEQUENCE LEVEL							
		CNN	CNN+RF		CNN+RF+ LSTM		CNN+RF+ BLSTM		CNN+LSTM		CNN+BLSTM	
	Measure	ResNet-50	Avg. pool	Avg. pool+pred.	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window
SEEN USERS	Accuracy	79.46	78.70	78.71	79.34	78.11	77.61	79.01	79.93	80.23	79.13	**80.64**
	mAP	63.06	66.08	66.05	64.35	55.79	59.63	63.84	69.74	**70.62**	68.21	69.96
	Macro precision	67.83	67.41	67.59	66.35	56.49	**73.04**	71.67	68.56	67.91	67.17	68.16
	Macro recall	65.04	60.36	60.50	58.11	48.78	48.45	57.84	64.44	67.53	62.74	**68.55**
	Macro F1-score	64.22	61.92	62.07	59.19	49.27	53.51	60.95	64.37	65.60	63.27	**66.85**
UNSEEN USERS	Accuracy	75.44	73.71	73.79	75.40	74.29	69.00	73.44	77.88	**79.87**	76.00	78.05
	mAP	53.42	55.71	56.05	52.03	50.28	46.27	51.20	59.02	**62.01**	58.03	59.03
	Macro precision	41.24	48.78	47.98	52.58	40.41	48.51	**59.75**	55.89	53.01	52.14	48.28
	Macro recall	43.74	43.11	43.23	46.46	40.61	40.71	46.85	49.47	**49.63**	48.71	46.44
	Macro F1-score	39.52	40.97	40.99	44.09	37.70	38.28	45.76	47.07	**47.30**	44.94	43.45
ALL	Accuracy	78.30	77.26	77.29	78.21	77.01	75.13	77.41	79.34	**80.12**	78.23	79.90
	mAP	59.02	62.92	62.97	61.27	54.71	56.01	60.47	66.52	**67.45**	65.08	66.96
	Macro precision	64.74	66.32	66.39	66.63	57.36	**72.60**	71.33	67.96	67.79	66.51	67.25
	Macro recall	60.95	56.53	56.67	54.86	47.33	44.82	53.98	60.97	64.16	59.06	**64.25**
	Macro F1-score	60.80	58.89	58.97	57.91	48.65	49.75	57.91	61.84	63.71	60.83	**64.01**

C. RANKING EVALUATION

The model performance was evaluated using the accuracy, the mean average precision (mAP), and macro metrics for precision, recall, and F1-score. Using the accuracy as the only classification metric might be misleading under the class imbalance present in both test splits. The purpose of using these macro metrics is to offer a more solid comparison baseline. Table 3 shows the performance of all the static and temporal models on the seen and unseen test partitions. The best models for the seen and unseen test splits were CNN+BLSTM (80.64%) and CNN+LSTM (79.87%), respectively. In both test splits, the *sliding window* training resulted in better performance. Although both models achieved a similar accuracy on the test splits, the rest of the metrics remained significantly different. This indicates that the CNN+BLSTM model suffers from overfitting on unseen users. Overall, the best model for both test splits was the CNN+LSTM achieving an 80.12% accuracy, as it had a similar performance on the seen users split, and better performance on the unseen users split.

In contrast with the results previously obtained in [14], our experiments indicate that the CNN+RF models decreased the overall accuracy of the ResNet-50 network. Considering both test splits, the macro precision improved whereas the macro recall decreased. Thus, indicating that the CNN+RF models are confident in their predictions, but they miss a large number of class samples. Consequently, both temporal models trained on top of this configuration (CNN+RF+LSTM and CNN+RF+BLSTM) have a decreasing score in all the considered metrics with respect to the CNN baseline. This is likely due to the fact that here we are using another dataset (NTCIR-12 [13], [14]) and an unseen users split in our test set.

The confusion matrices of the best CNN+BLSTM and CNN+LSTM models for the seen and unseen test splits are illustrated in Fig. 5. A straight comparison of all classes between each test split cannot be made, as the number of test samples is different and it might be misleading. For instance, not all categories appear on the unseen test split like *airplane* or *watching tv*. Additionally, the proportion of the number of test samples is less in some classes, e.g. *stairclimbing*.

Nevertheless, a comparison between the results of each temporal model and the CNN model can be done by calculating their difference, as shown at the right of each confusion matrix row in Fig. 5. Since the accuracy improvement with respect to the baseline is higher on the unseen than on the seen test split, there are more changes in its difference. Moreover, the plots show low performance on the CNN model for the categories *Cleaning*, *Relaxing*, *Drinking*, and *Writing*. They might be due to the large intra-class variability of the category (*Relaxing*), the social context ambiguity (*Formal and Informal meeting*), and to the fact that same activities occurs on very similar places (*Cleaning*, *Cooking* and *Dishwashing*). Further results containing the recall scores for each class on both test splits are reported in the Appendix A.

V. GENERALIZATION TO OTHER DOMAINS

In real-world applications, a system pretrained on a large scale dataset is typically used on new visual unseen lifelogs during training, belonging to previously unknown users.

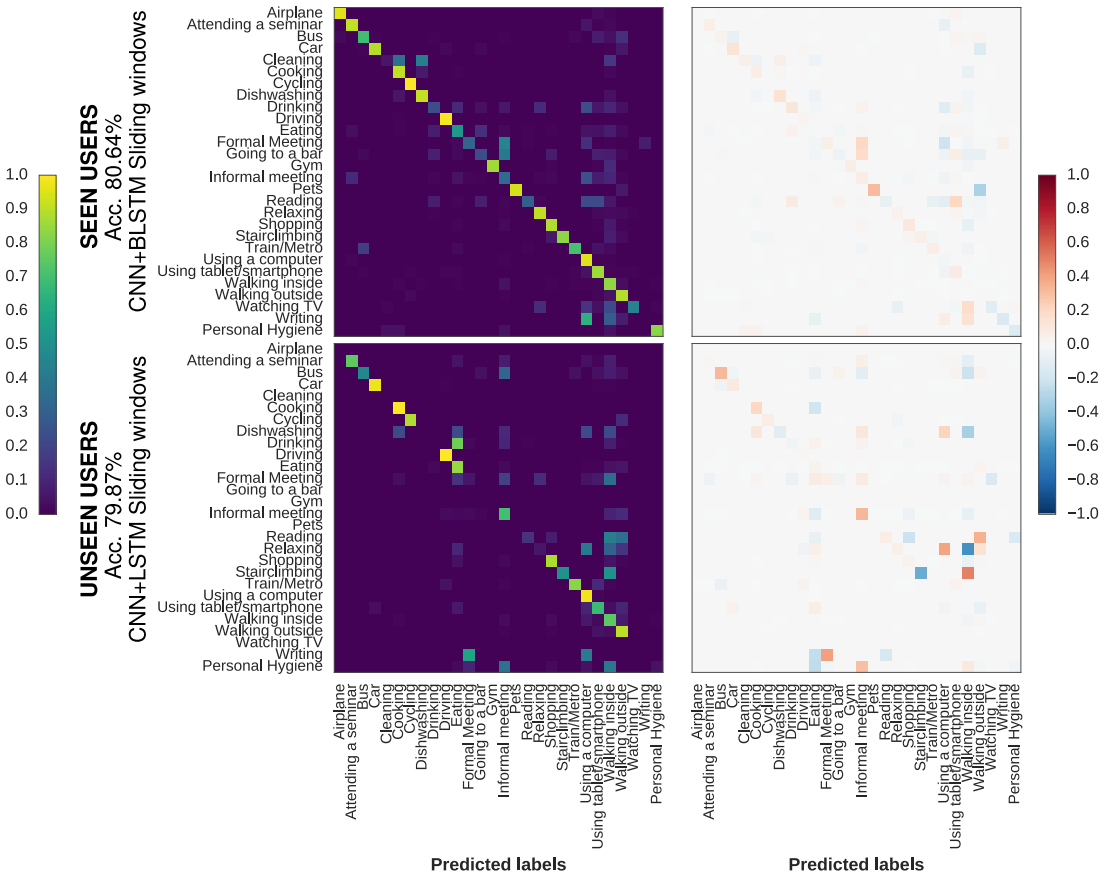

FIGURE 5. **Normalized confusion matrices of the best models for the seen and unseen test sets and their difference with respect to the CNN model. The increase and decrease of confidence is represented by the intensity of red and blue colors. Note that the classes *Airplane*, *Cleaning*, *Going to a bar*, *Gym*, and *Watching TV* do not appear on the unseen users test set.**

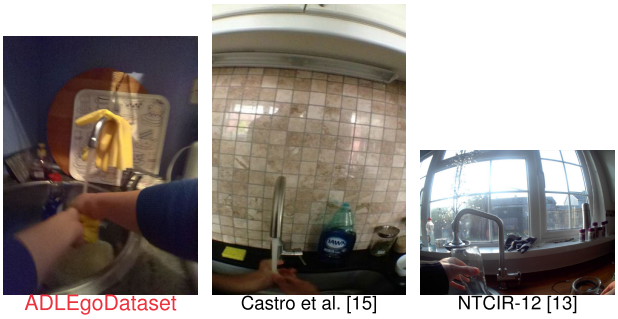

FIGURE 6. **Examples of people performing the same activity from different domain datasets. Below each image is its corresponding dataset, egocentric camera type, and body wearing location.**

The images composing these lifelogs might have been recorded from different cameras than the one used to capture the training dataset. For instance, Fig. 6 shows egocentric images of different people *washing the dishes* in their houses captured with three different wearable cameras. Besides the visual variability of tap and sinks in different kitchens, one can notice the contrast of fields of view and the angle distortion produced by different lenses. Due to the different nature of the source and target domains, performance on the target domain typically experiences a drop.

In this section, we aim at mitigating the performance drop by applying a semi-supervised learning technique, namely domain adaptation (DA). Our goal is to assess the performance between egocentric domains with and without transfer learning, rather than proposing a new adaptation method tailored at egocentric image sequences. Therefore, we strictly focus on a simple image-based DA method, the Deep Correlation Alignment (CORAL) regularization loss [12]. We perform two experiments using the ADLEgoDataset as the source domain, and the NTCIR-12 [13] and Castro *et al.* [15] datasets as target domains. These datasets were selected as target domains, as they are the closest to our dataset in number of activity categories and annotated images, and were recorded with different camera, as it can appreciated on Table 1. In the first experiment, we measure the performance of adding annotated images from different domains for training without using DA, and we quantify the difference between the target and the source domains. In the second experiment, we use the CORAL loss function as DA method on the target datasets and calculate the amount of labeled target data needed to achieve a good classification performance.

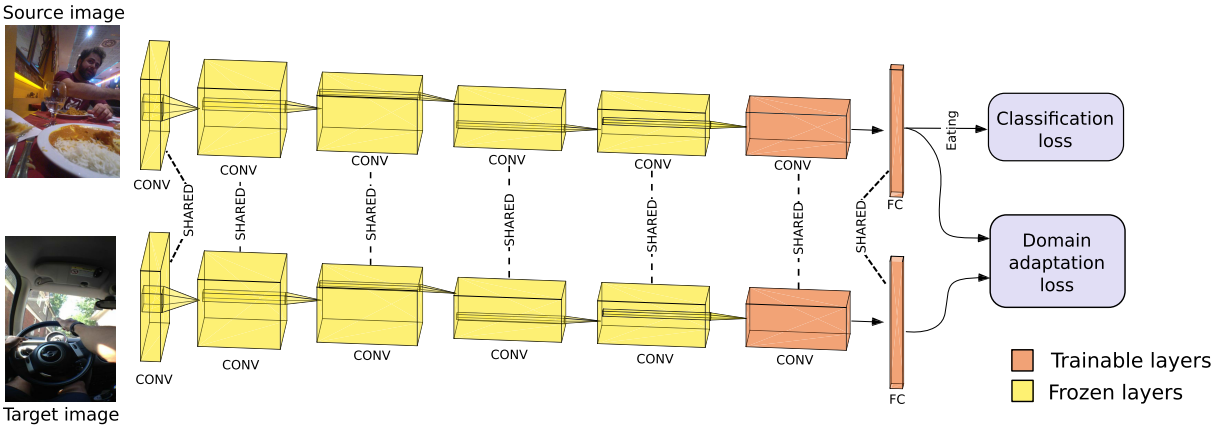

FIGURE 7. Domain adaptation training pipeline. During training, two CNNs with shared weights are used for the source and target data domains, respectively. Since the target domain labels are unknown, only the classification loss for the source CNN is evaluated. The adaptation from the source to the target domain comes from penalizing the discrepancy of their predictions using the domain adaptation loss. In this example, the discrepancy of both images should be high, because the source and target images correspond to the classes *eating* and *driving*.

In Section V-A, we detail the domain adaptation technique we used, namely the CORAL regularization loss. Next, in Section V-B we outline the datasets we used and their splits on the two experiments. In Section V-C, we thoroughly describe the implemented models. The experimental results evaluation and discussion are presented in Section V-D.

A. DOMAIN ADAPTATION USING A REGULARIZATION LOSS

Let $L_S = \{y_i\}$, $i \in \{1, \ldots, L\}$ be the labels from the source domain, and let us assume that the target domain has only unlabeled examples. During training, both domains have their own CNN architecture with shared weights, but only the source domain has a classification loss ℓ_{CLASS}. In order to adapt the learned model from the source to the target domain, a regularization loss ℓ_{DA} is used. This domain regularization loss penalizes the discrepancy between the output distributions from two single feature layers having a dimension d. This is a common setting used in [12], [65], [70] and it is illustrated in Fig. 7, where a single DA loss is penalizing the output of the fully-connected (FC) layers. The training loss function can be expressed as:

$$\ell = \ell_{CLASS} + \sum_{i=1}^{n} \lambda_i \ell_{DA} \tag{1}$$

where n is the number of DA regularization layers in the network and λ denotes the hyperparameter that trades off the adaptation with classification accuracy. Since our CNNs only had one FC layer, we only used one DA loss.

Specifically, we used the CORAL regularization loss [11], [12]. One of its advantages is that only the hyperparameter λ requires to be set. In this context, the output features of the source and target layers are said to come from the source domain $\mathcal{D}_S = \{\mathbf{x}_i\}$, $\mathbf{x} \in \mathbb{R}^d$ and the target domain $\mathcal{D}_T = \{\mathbf{u}_i\}$, $\mathbf{u} \in \mathbb{R}^d$, respectively. Then the CORAL regularization loss can be defined as:

$$\ell_{CORAL} = \frac{1}{4d^2} \| C(\mathcal{D}_S) - C(\mathcal{D}_T) \|_F^2 \tag{2}$$

where $\| \cdot \|_F^2$ denotes the squared matrix Frobenius norm and C is the covariance of $\mathcal{D}$ given by:

$$C(\mathcal{D}) = \frac{1}{m}(\mathcal{D}^\top \mathcal{D} - \frac{1}{m}(\mathbf{1}^\top \mathcal{D})^\top (\mathbf{1}^\top \mathcal{D})) \tag{3}$$

where m is the number of data in the domain $\mathcal{D}$ and $\mathbf{1}$ is a column vector with all elements equal to 1. The CORAL loss penalizes the discrepancy between domain features, so that when the source and target images correspond to different classes the penalty is high.

B. SOURCE AND TARGET DATASETS DETAILS

In our experiments, we used the ADLEgoDataset as the source domain dataset, and the NTCIR-12 [13] and Castro *et al.* [15] as target domain datasets. Both datasets were selected as target domains since they used different cameras and have more annotated categories and images than other lifelogging datasets, as can be appreciated in Table 1. Additionally, the domain visual difference with respect to our dataset can be appreciated in Fig. 6. We did not consider using the NTCIR-12 [13] and Castro's datasets as source domains since they have fewer people, half of the images, and fewer activity categories. Since their labels correspond to a different set of activity categories than ours, we manually mapped the matching categories. More categories would have required an automatic matching between words. The resulting categories and data distributions are shown in Fig. 8. The corresponding number of images of the source and the target for the NTCIR-12 was 96,632 and 44,902, and for the Castro's dataset was 68,507 and 39,166. The specific data splits for each experiment are detailed below.

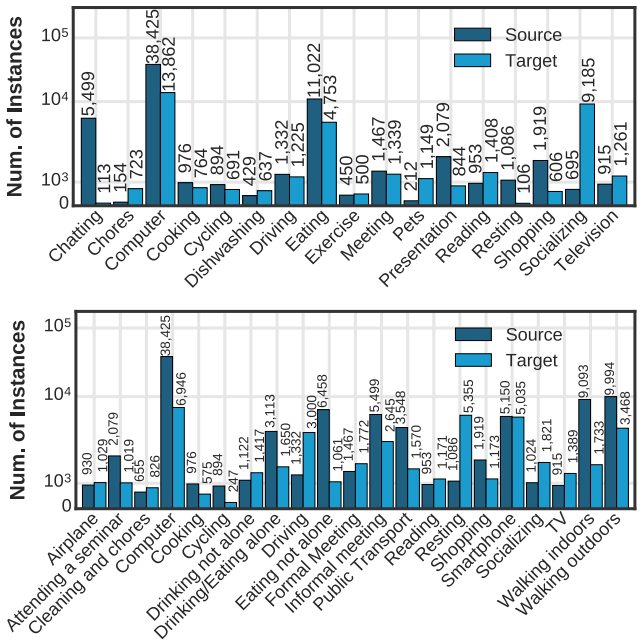

FIGURE 8. Categories mapping between the source and target domains with their data distributions. The source domain corresponds to our dataset, whereas the target domains are the Castro's dataset (top) and the NTCIR-12 dataset (bottom). Note that the vertical axis has a logarithm scale.

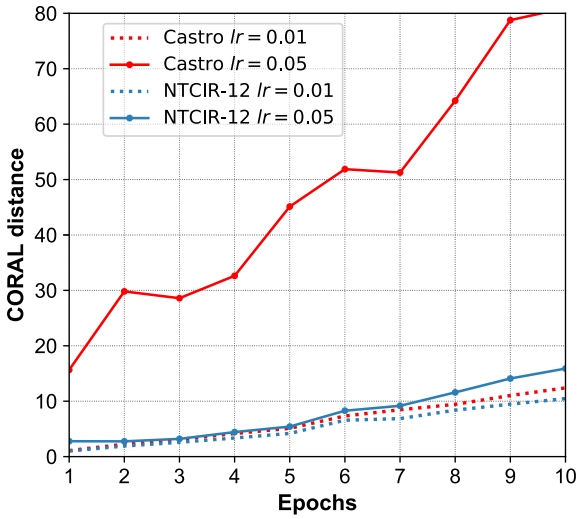

FIGURE 9. Sensitivity of the CORAL distance due to different learning rates using the Xception network. These results were obtained by only measuring the CORAL distance and not penalizing it (i.e. by having a fixed $\lambda = 0$).

1) TRAINING WITHOUT DOMAIN ADAPTATION

The goal of this experiment was to measure the performance of adding images from two different domains only during training without using DA. Therefore, we combined the source dataset with each target dataset for the training and validation splits, but the testing split only considered images from the source domain. Explicitly, we used the same splits for the source images as described in Section IV-A. The images from the target domains were randomly stratified in a 90/10% proportion for the training/validation splits.

2) DOMAIN ADAPTATION ON THE TARGET DATASETS

The objective of this experiment was to (*i*) use transfer learning in a practical setting and (*ii*) determine the required amount of labeled data from the target domain to obtain a good classification performance. The initial setting of the experiment considered that only the source domain data was labeled, but later different proportions of labeled data from the target domain were added.

First, we randomly stratified the source data into training and validation sets, and the target data into training and testing sets. Throughout the experiment, the proportion of training and validation data of the source images was fixed and set to 90/10%, whereas the proportion of training and testing data of the target images was initially set to 85/15%. Subsequently, different proportions (10, 20, . . . , 50%) of images were randomly and incrementally removed from the target training split. These images were added to the training/validation splits of the source domain while maintaining their original 90/10% proportion.

C. GENERALIZATION EXPERIMENTS IMPLEMENTATION

The following paragraphs describe the training settings for each experiment.

1) TRAINING WITHOUT DOMAIN ADAPTATION

We used a ResNet-50 [20] network as a CNN model and replaced its top layer with a fully-connected (FC) layer of 28 outputs. In order to have comparative results with the classification baseline of Section IV, we explicitly used the same network. It was trained using Stochastic Gradient Descent (SGD) with its weights initialized on ImageNet [87]. The last ResNet block and the FC layer were unfrozen during fine-tuning procedure. The training parameters were a learning rate $\alpha = 1 \times 10^{-2}$, a learning rate decay of 5×10^{-4}, and a momentum $\mu = 0.9$. Since we used two validation splits, the training was stopped when their epoch losses were not further improved. The number of epochs for the target datasets NTCIR-12 and Castro's one were 6 and 9, respectively.

Domains Discrepancy: As a means to quantify the difference between the source dataset and the target datasets, we calculated the maximum mean discrepancy (MMD) [89] between them for each shared category. First, we sampled between 500 and 1,000 images per category that were both in the source and the target datasets. These sampled images took into account all users and all days. Then, for each sampled image, we extracted a feature vector from the last pooling layer of a ResNet-50 CNN pre-trained on ImageNet [87]. Finally, we calculated the MMD between the sets of feature vectors of the source and target datasets using a Gaussian kernel with a $\sigma = 0.1$.

2) DOMAIN ADAPTATION ON THE TARGET DATASETS

We initially used two CNN architectures, i.e. Xception [19] and ResNet-50, as they are more robust and have better

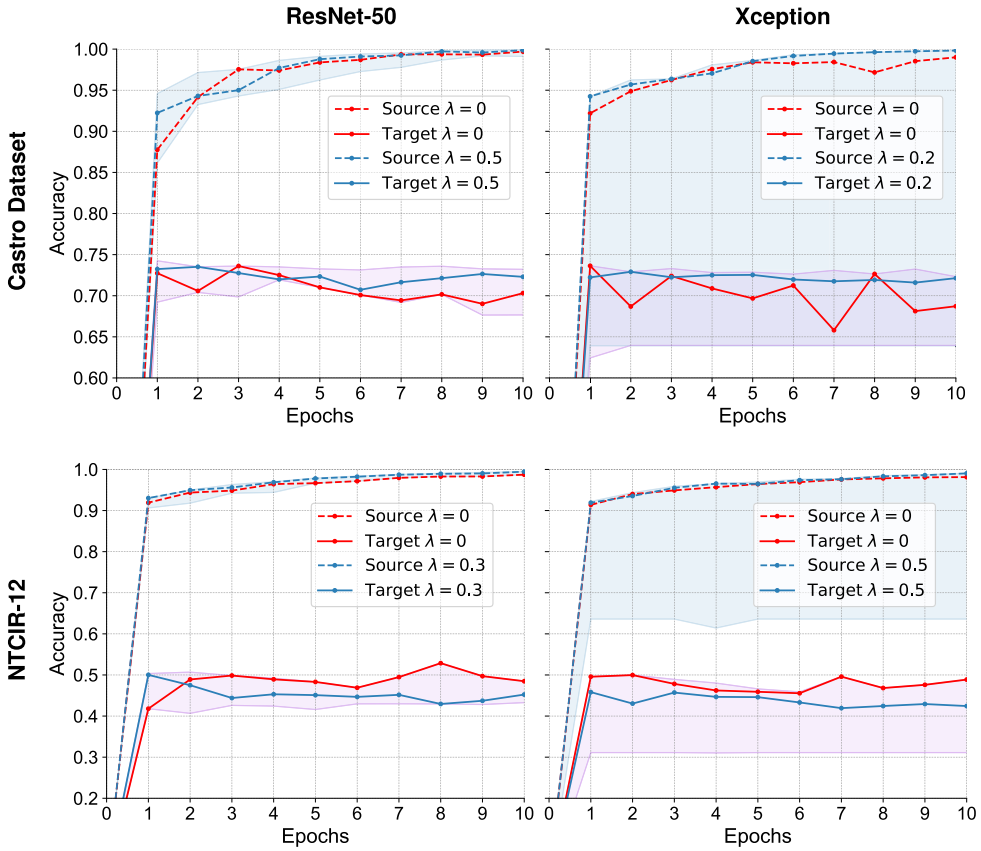

FIGURE 10. Validation accuracy for the ResNet-50 and Xception on the transfer learning to the Castro's dataset (top) and the NTCIR-12 dataset (bottom). Each plot shows the validation accuracy obtained with the best value of λ and without using domain adaptation (λ = 0). Additionally, the blue and violet areas represent the range between the minimal and maximal values of the accuracy for λ = 0.1, 0.2, . . . , 1.0 on the source and target domains, correspondingly.

performance than AlexNet [18], the original network used in [12], [65], [70].

a: ARCHITECTURE SETUP

In comparison with AlexNet, the Xception and ResNet-50 architectures have only one FC layer, making it the only layer suitable for the CORAL loss. The weights of this FC layer were initialized with $\mathcal{N}(0, 0.005)$ and its learning rate was set ten times bigger than the other layers, as stated in [12]. The rest of the layers were initialized using pre-trained weights on ImageNet [90]. We initially kept frozen all the layers except the classification layer, but it had a negative impact on the performance in the target domain. Hence, the layers from the last ResNet block of the ResNet-50 architecture and the exit flow block of the Xception architecture were unfrozen. We used SGD as an optimization method for both networks.

b: LEARNING RATE α TUNING

We experimentally found that an *adequate* learning rate α had to be high enough to produce a significant CORAL distance between the source and the target domain, but not so high that it did not converge. In order to find it, we first varied the learning rates while maintaining the other parameters constant and setting $\lambda = 0$. In other words, the training was performed without penalizing the discrepancy between domains, but measuring their distance. For instance, Fig. 9 illustrates significantly different CORAL distances for two different learning rates on both target datasets. In both cases, the highest learning rates were used as their training converged. Additionally, in our experiments, the lower learning rate did not produce higher accuracy scores for the training split of the target domain.

The final training parameters for ResNet-50 were a learning rate of $\alpha = 5 \times 10^{-3}$, a batch size of 60, a momentum equal to 0.9, and a weight decay equal to $5 \times 10{-}4$. Additionally, the training parameters for the Xception network were a learning rate of $\alpha = 5 \times 10^{-2}$, a batch size of 40, a momentum equal to $\mu = 0.9$, and a weight decay equal to 5×10^{-4}.

c: CORAL LOSS WEIGHT λ TUNING

After finding an adequate learning rate, we trained the ResNet-50 and Xception networks for $\lambda = 0, 0.1, \ldots, 1$. The best value of λ was obtained considering only the highest validation accuracy of the source domain, as the target data is supposed to be unknown. The best values of λ for ResNet-50 were 0.3 and 0.5 on the NTCIR-12 and Castro's datasets, respectively; whereas the best values of λ for Xception were

TABLE 4. Activity classification performance results obtained by adding the Castro's [15], [59] and NTCIR-12 [13], [14] datasets for training without domain adaptation. Best result per measure is shown in bold. Note that not all categories appeared on the unseen users test set.

	SEEN			UNSEEN			ALL		
Measure	**ADLEgoDataset**	**ADLEgoDataset+Castro**	**ADLEgoDataset+NTCIR**	**ADLEgoDataset**	**ADLEgoDataset+Castro**	**ADLEgoDataset+NTCIR**	**ADLEgoDataset**	**ADLEgoDataset+Castro**	**ADLEgoDataset+NTCIR**
Accuracy	**79.46**	67.87	67.44	**75.44**	58.95	55.02	**78.30**	65.31	63.87
mAP	**63.06**	43.08	43.03	**53.42**	36.99	37.59	**59.02**	40.17	40.08
Macro precision	**67.83**	47.93	54.31	**41.24**	31.50	32.21	**64.74**	45.69	49.99
Macro recall	**65.04**	52.34	45.71	**43.74**	30.61	33.79	**60.95**	47.40	41.97
Macro F1-score	**64.22**	48.71	45.80	**39.52**	28.59	27.89	**60.80**	44.67	41.09

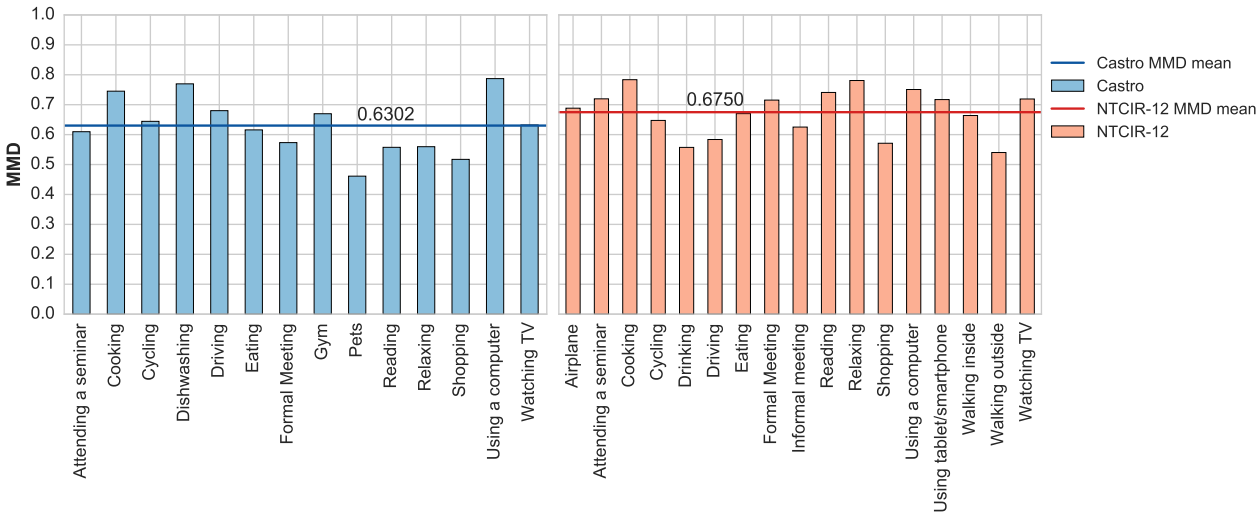

FIGURE 11. Maximum mean discrepancy (MMD) between the categories from the source and target datasets. The closer the value to zero the more similar the domains for that category.

0.5 and 0.2 on the NTCIR-12 and Castro's datasets, correspondingly.

The validation accuracy plots for ResNet-50 and Xception networks on both datasets are shown in Fig. 10. Two observations can be made from these plots. First, the areas between the minimal and maximal values of the accuracy obtained using the different values of λ suggest that the training of Xception network is more unstable than the ResNet-50 network. Consequently, no further experiments were implemented using the Xception network. Second, the difference between the target accuracy of both datasets ($\approx$ 73.22% for Castro and $\approx$ 47.92% for NTCIR-12) shows that a good performance is not always achieved using the CORAL loss alone. Therefore, more data from the target domain is needed to be labeled during training.

d: ADDITION OF TARGET LABELED DATA TO THE SOURCE DOMAIN

After fine-tuning the hyperparameters, we separately trained the ResNet-50 network adding different percentages of random target labeled data to the source domain. The considered percentages of target data were 0, 10%, . . . , 50% and were selected as described in Section V-B.

D. GENERALIZATION EXPERIMENTS EVALUATION

1) TRAINING WITHOUT DOMAIN ADAPTATION

The objective of this experiment was to (*i*) measure the activity classification performance when mixing the source and target datasets during training without DA method and (*ii*) estimate how different were the source and target domains. Given the class imbalance present in the dataset and for comparative purposes, we used the same performance metrics as the experiments presented in Section IV-A. The discrepancy between shared categories of the source and target domains was calculated using the MMD as described in Section V-C.

The classification results of separately adding Castro's and NTCIR-12 datasets for training are presented in Table 4. It shows that the addition of labeled data from the target domains diminished all the evaluated performance metrics; in particular, the accuracy was lower by 13.71% on average.

TABLE 5. Action recognition accuracy for the domain shifts from the ADLEgoDataset dataset using ResNet-50.Best result per measure is shown in bold.

Domain shift	Percentage of random target data used during training											
	0%		10%		20%		30%		40%		50%	
	Mean	Median	Mean	Median	Mean	Median	Mean	Median	Mean	Median	Mean	Median
Castro	58.34±0.2	72.47	46.70±0.4	75.58	78.75±0.0	79.04	79.50±0.0	79.73	**80.07±0.0**	81.18	65.83±0.3	**81.88**
NTCIR-12	45.21±0.0	45.14	77.85±0.0	78.46	82.60±0.0	81.81	84.91±0.0	84.90	87.34±0.0	87.40	**87.77±0.0**	**87.76**

TABLE 6. Seen users classification recall for all models in the baseline. Best results are shown in bold.

	CNN	CNN+RF		CNN+RF+ LSTM		CNN+RF+ BLSTM		CNN+LSTM		CNN+BLSTM	
Activity	ResNet-50	Avg. pool	Avg. pool+pred.	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window
Airplane	94.85	86.03	86.76	80.88	**95.59**	63.97	72.79	87.50	92.65	83.09	**95.59**
Attending a seminar	84.98	83.40	84.19	77.08	90.51	65.61	73.91	92.09	**93.68**	84.19	90.51
Bus	65.79	50.00	50.00	53.51	50.00	26.32	46.49	55.26	**69.30**	43.86	**69.30**
Car	75.83	72.50	71.67	66.67	70.83	60.83	65.83	82.50	79.17	85.00	**89.17**
Cleaning	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	**05.26**
Cooking	82.91	80.38	81.01	82.91	82.28	83.54	89.24	**91.14**	90.51	90.51	90.51
Cycling	**100.00**	**100.00**	99.14	96.55	79.31	92.24	**100.00**	98.28	**100.00**	**100.00**	**100.00**
Dishwashing	75.44	84.21	84.21	78.95	03.51	64.91	84.21	87.72	**91.23**	89.47	**91.23**
Drinking	11.52	13.17	13.99	11.93	13.58	10.29	10.70	23.05	18.11	21.81	**23.46**
Driving	96.97	97.58	97.58	96.97	**100.00**	96.97	99.39	98.79	99.39	98.79	99.39
Eating	54.54	54.37	53.77	**56.25**	55.14	55.22	54.11	52.23	50.17	50.77	52.65
Formal Meeting	26.11	21.66	23.57	21.02	19.11	13.38	17.83	**31.85**	29.94	28.66	**31.85**
Going to a bar	28.07	14.04	14.04	01.75	03.51	00.00	08.77	**29.82**	**29.82**	15.79	22.81
Gym	79.35	78.26	79.35	81.52	39.13	31.52	75.00	79.35	**95.65**	76.09	85.87
Informal meeting	27.43	31.82	32.13	33.39	33.07	**36.05**	32.13	31.35	32.76	33.70	34.48
Pets	62.50	68.75	68.75	43.75	00.00	31.25	56.25	**93.75**	**93.75**	68.75	**93.75**
Reading	**36.47**	21.18	17.65	28.24	12.94	05.88	07.06	08.24	22.35	05.88	29.41
Relaxing	86.62	76.76	76.76	73.94	76.76	57.04	73.24	88.73	**92.25**	90.85	90.85
Shopping	77.78	75.00	74.31	85.42	78.82	63.19	78.12	82.29	**89.93**	78.47	88.19
Stairclimbing	77.78	66.67	69.44	47.22	00.00	11.11	55.56	58.33	69.44	**83.33**	**83.33**
Train/Metro	63.14	68.64	69.92	**75.00**	68.64	65.25	70.34	66.10	72.88	53.39	70.76
Using a computer	97.32	96.64	96.64	97.39	96.77	97.39	**98.03**	95.35	96.29	97.02	95.15
Using tablet/smartphone	77.16	78.58	79.05	80.83	80.24	84.62	79.41	84.97	82.60	84.73	**85.92**
Walking inside	83.78	84.56	84.22	83.61	83.35	89.33	86.73	**89.51**	82.13	83.52	83.87
Walking outside	89.17	88.58	88.48	89.76	**90.85**	90.45	89.57	87.01	89.07	85.14	89.17
Watching TV	**55.56**	32.10	32.10	25.93	41.98	20.99	29.63	39.51	40.74	45.68	44.44
Writing	**14.29**	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00
Personal Hygiene	**95.65**	65.22	65.22	56.52	00.00	39.13	65.22	69.57	86.96	78.26	82.61

The overall classification performance of adding Castro's dataset was better than when adding the NTCIR-12 dataset. This is also reflected in their calculated discrepancy with respect to the source domain. Fig. 11 shows the MMD for each shared category between each target and source domains, and a horizontal line representing its mean. The MMD mean for the Castro's dataset is lower than for the NTCIR-12 dataset, thus meaning that it is more similar to the ADLEgoDataset. This difference in discrepancy also is reflected in the performance of domain adaptation as describe below. Supplementary results containing the recall scores for each class are reported in the Appendix B.

2) DOMAIN ADAPTATION ON THE TARGET DATASETS

The objective of this experiment was to use transfer learning on a practical setting and to determine the required amount of labeled data from the target domain to obtain a good classification performance. As in previous works [12], [65], [70], [73], [91], [92], we use the prediction accuracy as evaluation metric for five different training runs. Our results only consider ResNet-50 architecture, since the training of the Xception network was unstable as discussed above. The summarized results are shown in Table 5 and plot in Fig. 12.

The results in Table 5 show that ResNet-50 was also susceptible to instability during training, producing a high variance in some training runs. This instability only affected Castro's dataset and can be visually seen in the plot of Fig. 12. Therefore, the accuracy median was also considered to measure performance improvement.

The results confirm that performing domain adaptation without using labeled target data does not necessarily achieve

TABLE 7. Unseen users classification recall for all models in the baseline. Best results are shown in bold. Note that not all categories appeared on this test set.

	CNN	CNN+RF		CNN+RF+ LSTM		CNN+RF+ BLSTM		CNN+LSTM		CNN+BLSTM	
Activity	ResNet-50	Avg. pool	Avg. pool+pred.	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window	Day sequence	Sliding window
Attending a seminar	73.21	48.68	48.68	48.68	61.13	22.26	37.36	73.58	**75.09**	53.96	63.02
Bus	14.29	14.29	14.29	03.57	03.57	00.00	10.71	14.29	**46.43**	17.86	21.43
Car	87.46	78.37	79.31	80.56	94.04	56.74	72.41	87.15	**97.18**	88.09	90.28
Cooking	80.00	60.00	60.00	40.00	40.00	40.00	40.00	40.00	**100.00**	**100.00**	60.00
Cycling	80.48	68.49	68.49	71.92	39.04	41.78	66.44	78.42	**88.01**	79.45	82.88
Dishwashing	**11.11**	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00
Drinking	**03.08**	00.00	00.00	00.00	00.00	00.00	00.00	00.77	00.00	01.54	00.00
Driving	98.53	97.56	97.56	97.31	98.78	97.31	98.04	**100.00**	**100.00**	99.51	**100.00**
Eating	80.85	82.03	81.89	83.21	**85.27**	82.77	82.92	84.39	84.24	81.74	82.92
Formal Meeting	00.00	01.50	00.75	00.00	00.00	00.00	00.00	03.01	05.26	**06.77**	03.01
Informal meeting	36.07	61.58	61.58	68.91	69.21	70.67	69.21	66.86	69.50	**76.54**	73.90
Reading	07.14	07.14	07.14	07.14	00.00	00.00	07.14	07.14	14.29	**21.43**	**21.43**
Relaxing	00.00	05.00	05.00	00.00	00.00	00.00	00.00	**15.00**	05.00	00.00	00.00
Shopping	83.57	78.57	78.93	86.43	77.14	68.21	80.00	80.71	**87.50**	80.36	86.07
Stairclimbing	**100.00**	50.00	50.00	50.00	00.00	00.00	50.00	50.00	50.00	50.00	50.00
Train/Metro	81.99	76.84	75.74	79.04	70.59	68.75	75.37	78.68	82.35	59.56	**83.09**
Using a computer	97.95	96.92	97.38	98.06	98.63	97.83	**98.86**	97.38	98.40	97.95	96.69
Using tablet/smartphone	67.74	77.42	**80.65**	**80.65**	67.74	74.19	70.97	**80.65**	67.74	**80.65**	74.19
Walking inside	78.30	80.85	80.85	80.43	79.57	82.98	78.72	**85.96**	74.89	79.15	79.15
Walking outside	**91.14**	88.86	88.86	89.71	90.00	90.57	90.29	89.43	90.29	87.71	87.71
Writing	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00	00.00
Personal Hygiene	**08.15**	03.70	03.70	02.96	00.00	01.48	02.22	04.44	04.44	06.67	05.19

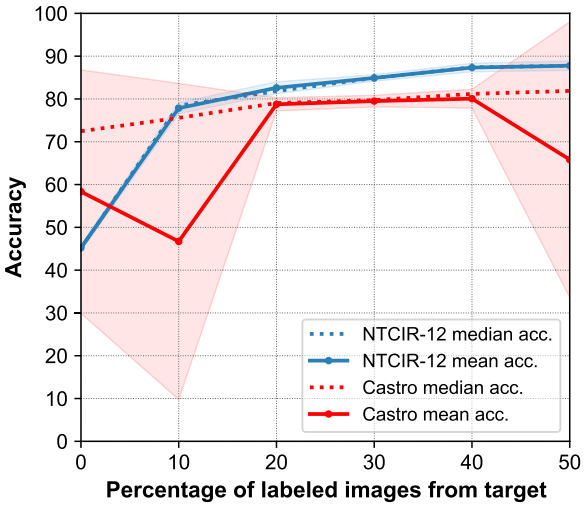

FIGURE 12. Mean test accuracy with respect to different percentages of added target labeled images to the training/validation source data using ResNet-50. The colored areas denotes the mean value ± standard deviation.

a good performance on all target datasets. Specifically, the median accuracy of the NTCIR-12 was 45.14% whereas for Castro's dataset was 72.58%. This low performance was improved by adding a small subset of labeled target data to the training. The largest increment in median accuracy was obtained by adding 10-20% of labeled data, i.e. for the NTCIR-12 it improved by 33.32% when adding 10% and for Castro's dataset it improved by 6.57% after adding 20%. The most benefited dataset was the NTCIR-12 since their initial discrepancy was higher as shown by the previous experiment. The mean and median accuracy curves from Fig. 12 show a decreasing increment that settles around 40%. Although a straight comparison with previous works cannot be made [14], [15], the mean accuracy values at 40% of added data are competitive. Originally, the accuracy obtained for Castro's and NTCIR-12 datasets were 83.07% and 94.08%, correspondingly.

VI. CONCLUSION

We introduced the so-far largest egocentric lifelog dataset of activities of daily living consisting of 105,529 annotated images, the ADLEgoDataset. It was recorded by 15 different participants wearing a Narrative Clip camera while performing 35 activities of daily life in a naturalistic setting during a total of 191 days. With respect to other available lifelog datasets, it contains many more categories, annotated images, users and types of activities, hence allowing to perform generalization tests on unseen users.

We presented a strong classification baseline on our dataset that considers a more realistic comparison by not only testing

TABLE 8. Activity classification recall by adding the Castro's [15], [59] and NTCIR-12 [13], [14] datasets without domain adaptation. Best result per measure is shown in bold. Note that not all categories appeared on the unseen users test set.

	SEEN			UNSEEN			ALL		
Activity	ADLEgoDataset	ADLEgoDataset+Castro	ADLEgoDataset+NTCIR	ADLEgoDataset	ADLEgoDataset+Castro	ADLEgoDataset+NTCIR	ADLEgoDataset	ADLEgoDataset+Castro	ADLEgoDataset+NTCIR
Airplane	**94.85**	61.76	79.41	-	-	-	**94.85**	61.76	79.41
Attending a seminar	84.98	**86.56**	65.22	**73.21**	10.19	16.60	**78.96**	47.49	40.35
Bus	**65.79**	30.70	17.54	**14.29**	00.00	00.00	**55.63**	24.65	14.08
Car	**75.83**	70.00	53.33	**87.46**	41.38	19.44	**84.28**	49.20	28.70
Cleaning	**00.00**	**00.00**	**00.00**	-	-	-	**00.00**	**00.00**	**00.00**
Cooking	82.91	79.75	**85.44**	80.00	00.00	**100.00**	82.82	77.30	**85.89**
Cycling	**100.00**	94.83	93.97	**80.48**	53.42	50.68	**86.03**	65.20	62.99
Dishwashing	**75.44**	**75.44**	31.58	11.11	**33.33**	00.00	66.67	**69.70**	27.27
Drinking	11.52	08.64	**18.11**	03.08	00.77	**22.31**	08.58	05.90	**19.57**
Driving	**96.97**	91.52	90.30	**98.53**	89.73	82.15	**98.08**	90.24	84.49
Eating	**54.54**	36.47	26.63	**80.85**	69.81	59.50	**64.21**	48.73	38.71
Formal Meeting	**26.11**	06.37	08.92	00.00	**11.28**	01.50	**14.14**	08.62	05.52
Going to a bar	28.07	**29.82**	08.77	-	-	-	28.07	**29.82**	08.77
Gym	**79.35**	67.39	55.43	-	-	-	**79.35**	67.39	55.43
Informal meeting	27.43	33.07	**33.86**	36.07	42.23	**44.28**	30.44	36.26	**37.49**
Pets	**62.50**	**62.50**	43.75	-	-	-	**62.50**	**62.50**	43.75
Reading	**36.47**	01.18	04.71	**07.14**	**07.14**	00.00	**32.32**	02.02	04.04
Relaxing	**86.62**	59.15	66.20	**00.00**	**00.00**	**00.00**	**75.93**	51.85	58.02
Shopping	77.78	66.32	**78.12**	**83.57**	68.57	**83.57**	80.63	67.43	**80.81**
Stairclimbing	**77.78**	52.78	22.22	**100.00**	**100.00**	**100.00**	**78.95**	55.26	26.32
Train/Metro	**63.14**	49.58	27.54	**81.99**	50.37	18.01	**73.23**	50.00	22.44
Using a computer	**97.32**	85.26	89.67	**97.95**	85.75	84.61	**97.42**	85.34	88.85
Using tablet/smartphone	**77.16**	65.33	57.16	**67.74**	45.16	51.61	**76.83**	64.61	56.96
Walking inside	**83.78**	66.09	67.39	**78.30**	60.85	57.87	**82.85**	65.20	65.78
Walking outside	**89.17**	85.24	86.52	**91.14**	81.14	86.29	**89.68**	84.19	86.46
Watching TV	**55.56**	43.21	50.62	-	-	-	**55.56**	43.21	50.62
Writing	**14.29**	00.00	00.00	**00.00**	**00.00**	**00.00**	**07.69**	00.00	00.00
Personal Hygiene	**95.65**	56.52	17.39	**08.15**	05.93	00.00	**20.89**	13.29	02.53

on unseen days but also on unseen users. This baseline was done using existing state-of-the-art algorithms on it, which also served as a ranking of their generalization capabilities. The best algorithm achieved an 80.12% of accuracy and was the CNN+LSTM trained in a sliding window fashion.

Moreover, we presented experiments of generalization in different domains. We first showed that the evaluated source and target datasets have a large discrepancy that diminished the classification performance by 13.71% on average. Finally, we used the CORAL loss function as a DA technique and showed that a good performance is not always achieved on different target datasets. Specifically, we obtained a median accuracy value of 72.47% and 45.14% on Castro's and the NTCIR-12 datasets. We also showed that the performance can improve by incorporating a small percentage of labeled target data to the training. In the case of the NTCIR-12 dataset, the performance improved to 78.46% by randomly adding 10% of target data.

We consider that further research lines using this dataset are twofold. First, taking into account the ambiguity of the context, the activity recognition problem from lifelogs could be posed more naturally as a multi-classification problem. For instance, a person might be *reading* a book while being on *train*. Second, we only considered full day sequences on the temporal classification algorithms, but splitting them into sub-sequences with higher temporal coherence could improve the classification accuracy. Moreover, we consider that activity recognition from wearable photo-cameras, in conjunction with information coming from more sensors, is mature enough to be tested in real-world applications. These applications could come from different domains, for instance, the assessment of several activities of daily living for the elderly or for monitoring the wellbeing of young people.

Although the people in our dataset have different lifestyles and hobbies, their activities reflect the life of computer science graduate students. We consider that a dataset captured

by users having different jobs would help to cope better with real-world scenarios. For instance, a construction worker would have a routine with different activities and settings.

APPENDIXES

APPENDIX A
CLASSIFICATION RECALL FROM DATASET BASELINE

In Tables 6 and 7 are shown the classification recall scores for ADLEgoDataset baseline from Section IV. These Tables reflect the results obtained measuring the macro metrics, i.e. the best performance for the seen users test split is obtained the CNN+BLSTM method, whereas for the unseen users test split is obtained the CNN+LSTM method. Additionally, both Tables show that the best training strategy is the *sliding window*.

APPENDIX B
CLASSIFICATION RECALL FROM GENERALIZATION WITHOUT DOMAIN ADAPTATION EXPERIMENT

In Table 8 is shown the classification recall scores for generalization experiments without domain adaptation from Section V. Although the performance was diminished in overall metrics, some categories were benefited. The improved categories for the NTCIR-12 dataset were *cooking*, *drinking*, *informal meeting*, and *shopping*. In the case of the Castro's dataset, only two categories improved their accuracy: *dish-washing* and *going to a bar*. The latter category was not present in any of the target images. This table also shows that overall performance of adding the Castro's dataset was better than the NTCIR-12 dataset.

APPENDIX C
ACKNOWLEDGMENT

The authors acknowledge the support of NVIDIA Corporation with the donation of Titan Xp GPUs.

REFERENCES

[1] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 3, pp. 1192–1209, 3rd Quart., 2013.

[2] G. Abebe, A. Cavallaro, and X. Parra, "Robust multi-dimensional motion features for first-person vision activity recognition," *Comput. Vis. Image Understand.*, vol. 149, pp. 229–248, Aug. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077314215002350

[3] K. Zhan, S. Faux, and F. Ramos, "Multi-scale conditional random fields for first-person activity recognition on elders and disabled patients," *Pervas. Mobile Comput.*, vol. 16, pp. 251–267, Jan. 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1574119214001850

[4] S. Karaman, J. Benois-Pineau, R. Megret, V. Dovgalecs, J.-F. Dartigues, and Y. Gaestel, "Human daily activities indexing in videos from wearable cameras for monitoring of patients with dementia diseases," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 4113–4116.

[5] S. Karaman, J. Benois-Pineau, V. Dovgalecs, R. Mégret, J. Pinquier, R. André-Obrecht, Y. Gaëstel, and J.-F. Dartigues, "Hierarchical hidden Markov model in detecting activities of daily living in wearable videos for studies of dementia," *Multimedia Tools Appl.*, vol. 69, no. 3, pp. 743–771, Apr. 2014, doi: 10.1007/s11042-012-1117-x.

[6] Y. C. Zhang and J. M. Rehg, "Watching the TV watchers," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 2, no. 2, pp. 1–27, Jul. 2018, doi: 10.1145/3214291.

[7] S. Z. Bokhari and K. M. Kitani, "Long-term activity forecasting using first-person vision," in *Proc. 13th Asian Conf. Comput. Vis.*, Taipei, Taiwan, Nov. 2016, pp. 346–360, doi: 10.1007/978-3-319-54193-8_22.

[8] S. Gella, M. Lapata, and F. Keller, "Unsupervised visual sense disambiguation for verbs using multimodal embeddings," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 182–192. [Online]. Available: https://www.aclweb.org/anthology/N16-1022

[9] A. Cartas, M. Dimiccoli, and P. Radeva, "Batch-based activity recognition from egocentric photo-streams," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 2347–2354.

[10] M. Dimiccoli, A. Cartas, and P. Radeva, "Activity recognition from visual lifelogs: State of the art and future challenges," in *Multimodal Behavior Analysis in the Wild*. Amsterdam, The Netherlands: Elsevier, 2019, pp. 121–134.

[11] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 2058–2065. [Online]. Available: http://dl.acm.org/citation.cfm?id=3016100.3016186

[12] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Computer Vision—ECCV Workshops*, G. Hua and H. Jégou, Eds. Cham, Switzerland: Springer, 2016, pp. 443–450.

[13] C. Gurrin, H. Joho, F. Hopfgartner, L. Zhou, and R. Albatal, "Overview of ntcir-12 lifelog task," Tech. Rep., 2016, pp. 354–360. [Online]. Available: http://eprints.gla.ac.uk/119206/

[14] A. Cartas, J. Marín, P. Radeva, and M. Dimiccoli, "Batch-based activity recognition from egocentric photo-streams revisited," *Pattern Anal. Appl.*, vol. 21, no. 4, pp. 953–965, Nov. 2018, doi: 10.1007/s10044-018-0708-1.

[15] D. Castro, S. Hickson, V. Bettadapura, E. Thomaz, G. Abowd, H. Christensen, and I. Essa, "Predicting daily activities from egocentric images using deep learning," in *Proc. ACM Int. Symp. Wearable Comput. ISWC*, 2015, pp. 75–82.

[16] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden Markov model," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 1992, pp. 379–385.

[17] S. Herath, M. Harandi, and F. Porikli, "Going deeper into action recognition: A survey," *Image Vis. Comput.*, vol. 60, pp. 4–21, Apr. 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0262885617300343

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-conv%olutional-neural-networks.pdf

[19] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[21] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 7132–7141.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: https://arxiv.org/abs/1409.1556

[23] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," 2014, *arXiv:1409.4842*. [Online]. Available: http://arxiv.org/abs/1409.4842

[24] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1725–1732.

[25] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 221–231, Jan. 2013.

[26] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2625–2634.

[27] J. Y.-H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4694–4702.

[28] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 568–576.

[29] Z. Wu, Y.-G. Jiang, X. Wang, H. Ye, and X. Xue, "Multi-stream multi-class fusion of deep networks for video classification," in *Proc. ACM Multimedia Conf. MM*, New York, NY, USA: ACM, 2016, pp. 791–800, doi: 10.1145/2964284.2964328.
[30] S. Bambach, S. Lee, D. J. Crandall, and C. Yu, "Lending a hand: Detecting hands and recognizing activities in complex egocentric interactions," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1949–1957.
[31] G. Garcia-Hernando, S. Yuan, S. Baek, and T.-K. Kim, "First-person hand action benchmark with RGB-D videos and 3D hand pose annotations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 409–419.
[32] W. W. Mayol and D. W. Murray, "Wearable hand activity recognition for event summarization," in *Proc. 9th IEEE Int. Symp. Wearable Comput. (ISWC)*, Oct. 2005, pp. 122–129.
[33] I. González Díaz, V. Buso, J. Benois-Pineau, G. Bourmaud, and R. Megret, "Modeling instrumental activities of daily living in egocentric vision as sequences of active objects and context for alzheimer disease research," in *Proc. 1st ACM Int. Workshop Multimedia Indexing Inf. Retr. Healthcare MIIRH*, New York, NY, USA: ACM, 2013, pp. 11–14, doi: 10.1145/2505323.2505328.
[34] K. Matsuo, K. Yamada, S. Ueno, and S. Naito, "An attention-based activity recognition for egocentric video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 565–570.
[35] H. Pirsiavash and D. Ramanan, "Detecting activities of daily living in first-person camera views," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2847–2854.
[36] T. McCandless and K. Grauman, "Object-centric spatio-temporal pyramids for egocentric activity recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2013, p. 3.
[37] D. Surie, T. Pederson, F. Lagriffoul, L.-E. Janlert, and D. Sjölie, "Activity recognition using an egocentric perspective of everyday objects," in *Ubiquitous Intelligence and Computing*, J. Indulska, J. Ma, L. T. Yang, T. Ungerer, and J. Cao, Eds. Berlin, Germany: Springer, 2007, pp. 246–257.
[38] S. Sudhakaran and O. Lanz, "Attention is all we need: Nailing down object-centric attention for egocentric activity recognition," in *Proc. Brit. Mach. Vis. Conf.*, Newcastle, UK: Northumbria Univ., Sep. 2018, p. 229. [Online]. Available: http://bmvc2018.org/contents/papers/0756.pdf
[39] K. M. Kitani, T. Okabe, Y. Sato, and A. Sugimoto, "Fast unsupervised ego-action learning for first-person sports videos," in *Proc. CVPR*, Jun. 2011, pp. 3241–3248.
[40] Y. Poleg, C. Arora, and S. Peleg, "Temporal segmentation of egocentric videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2537–2544.
[41] A. Fathi, Y. Li, and J. M. Rehg, "Learning to recognize daily actions using gaze," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 314–327.
[42] I. Hipiny and W. Mayol-Cuevas, "Recognising egocentric activities from gaze regions with multiple-voting bag of words," Dept. Comput. Sci., Univ. Bristol, Bristol, U.K., Tech. Rep. CSTR-12-003, 2012.
[43] Y. Li, M. Liu, and J. M. Rehg, "In the eye of beholder: Joint learning of gaze and actions in first person video," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 619–635.
[44] K. Ogaki, K. M. Kitani, Y. Sugano, and Y. Sato, "Coupling eye-motion and ego-motion features for first-person activity recognition," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 1–7.
[45] Y. Shiga, A. Dengel, T. Toyama, K. Kise, and Y. Utsumi, "Daily activity recognition combining gaze motion and visual features," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput. Adjunct Publication UbiComp Adjunct*, 2014, pp. 1103–1111.
[46] A. Behera, D. C. Hogg, and A. G. Cohn, "Egocentric activity monitoring and recovery," in *Proc. Asian Conf. Comput. Vis.*, 2012, pp. 519–532.
[47] Y. Li, Z. Ye, and J. M. Rehg, "Delving into egocentric actions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 287–295.
[48] C. Yu and D. H. Ballard, "Understanding human behaviors based on eye-head-hand coordination," in *Biologically Motivated Computer Vision*, H. H. Bülthoff, C. Wallraven, S.-W. Lee, and T. A. Poggio, Eds. Berlin, Germany: Springer, 2002, pp. 611–619.
[49] M. S. Ryoo and L. Matthies, "First-person activity recognition: Feature, temporal structure, and prediction," *Int. J. Comput. Vis.*, vol. 119, no. 3, pp. 307–328, Sep. 2016.
[50] A. Fathi, A. Farhadi, and J. M. Rehg, "Understanding egocentric activities," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 407–414.
[51] A. Fathi and J. M. Rehg, "Modeling actions through state changes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 2579–2586.
[52] H. F. M. Zaki, F. Shafait, and A. Mian, "Modeling sub-event dynamics in first-person action recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1619–1628.
[53] M. Ma, H. Fan, and K. M. Kitani, "Going deeper into first-person activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1894–1903.
[54] S. Singh, C. Arora, and C. V. Jawahar, "First person action recognition using deep learned descriptors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2620–2628.
[55] S. Song, V. Chandrasekhar, B. Mandal, L. Li, J.-H. Lim, G. S. Babu, P. P. San, and N.-M. Cheung, "Multimodal multi-stream deep learning for egocentric activity recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2016, pp. 24–31.
[56] C. Gurrin, H. Joho, F. Hopfgartner, L. Zhou, R. Gupta, R. Albatal, N. Dang, and T. Duc, "Overview of ntcir-13 lifelog-2 task," in *Proc. 13th NTCIR Conf. Eval. Inf. Access Technol.*, 2017, pp. 6–11.
[57] D.-T. Dang-Nguyen, L. Piras, M. Riegler, L. Zhou, M. Lux, and C. Gurrin, "Overview of ImageCLEFlifelog 2018: Daily living understanding and Lifelog moment retrieval," in *Proc. CLEF Working Notes (CEUR)*, Avignon, France, Sep. 2018, pp. 1–19, [Online]. Available: CEUR-WS.org<http://ceur-ws.org>
[58] C. Gurrin, H. Joho, F. Hopfgartner, L. Zhou, V.-T. Ninh, T.-K. Le, R. Albatal, D.-T. Dang-Nguyen, and G. Healy, "Advances in lifelog data organisation and retrieval at the ntcir-14 lifelog-3 task," in *NII Testbeds and Community for Information Access Research*, M. P. Kato, Y. Liu, N. Kando, and C. L. A. Clarke, Eds. Cham, Switzerland: Springer, 2019, pp. 16–28.
[59] M. Dimiccoli, J. Marín, and E. Thomaz, "Mitigating bystander privacy concerns in egocentric activity recognition with deep learning and intentional image degradation," *Proc. ACM Interact., Mobile, Wearable Ubiquitous Technol.*, vol. 1, no. 4, pp. 1–18, Jan. 2018.
[60] A. Cartas, J. Marín, P. Radeva, and M. Dimiccoli, "Recognizing activities of daily living from egocentric images," in *Proc. Pattern Recognit. Image Anal.*, vol. 10255, May 2017, pp. 87–95.
[61] H. Yu, W. Jia, Z. Li, F. Gong, D. Yuan, H. Zhang, and M. Sun, "A multisource fusion framework driven by user-defined knowledge for egocentric activity recognition," *EURASIP J. Adv. Signal Process.*, vol. 2019, no. 1, p. 14, Dec. 2019, doi: 10.1186/s13634-019-0612-x.
[62] H. Yu, G. Pan, M. Pan, C. Li, W. Jia, L. Zhang, and M. Sun, "A hierarchical deep fusion framework for egocentric activity recognition using a wearable hybrid sensor system," *Sensors*, vol. 19, no. 3, p. 546, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/3/546
[63] A. Storkey, "When training and test sets are different: Characterising learning transfer," in *Dataset Shift in Machine Learning*, J. Quionero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, Eds. Cambridge, MA, USA: MIT Press, 2009, pp. 3–28, ch. 1.
[64] H. Daumé and D. Marcu, "Domain adaptation for statistical classifiers," *J. Artif. Int. Res.*, vol. 26, no. 1, pp. 101–126, May 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1622559.1622562
[65] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. 32nd Int. Conf. Mach. Learn.*, vol. 37, F. Bach and D. Blei, Eds. Lille, France: PMLR, Jul. 2015, pp. 97–105. [Online]. Available: http://proceedings.mlr.press/v37/long15.html
[66] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, New York, NY, USA: Curran Associates, 2016, pp. 136–144. [Online]. Available: http://dl.acm.org/citation.cfm?id=3157096.3157112
[67] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo, "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2272–2281.
[68] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 2208–2217. [Online]. Available: http://dl.acm.org/citation.cfm?id=3305890.3305909
[69] W. Zellinger, T. Grubinger, E. Lughofer, T. Natschläger, and S. Saminger-Platz, "Central moment discrepancy (CMD) for domain-invariant representation learning," in *Proc. 5th Int. Conf. Learn. Representations, ICLR*, Toulon, France, Apr. 2017. [Online]. Available: https://openreview.net/forum?id=SkB-_mcel

[70] W. Zellinger, B. A. Moser, T. Grubinger, E. Lughofer, T. Natschläger, and S. Saminger-Platz, "Robust unsupervised domain adaptation for neural networks via moment alignment," *Inf. Sci.*, vol. 483, pp. 174–191, May 2019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0020025519300301

[71] C.-Y. Lee, T. Batra, M. H. Baig, and D. Ulbricht, "Sliced wasserstein discrepancy for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10285–10295.

[72] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds. New York, NY, USA: Curran Associates, 2014, pp. 2672–2680. [Online]. Available: http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf

[73] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, vol. 37, F. Bach and D. Blei, Eds. Lille, France: PMLR, Jul. 2015, pp. 1180–1189. [Online]. Available: http://proceedings.mlr.press/v37/ganin15.html

[74] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2962–2971, doi: 10.1109/CVPR.2017.316.

[75] K. Saito, Y. Ushiku, T. Harada, and K. Saenko, "Adversarial dropout regularization," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–14. [Online]. Available: https://openreview.net/forum?id=HJIoJWZCZ

[76] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3723–3732.

[77] M. Bolanos, M. Dimiccoli, and P. Radeva, "Toward storytelling from visual lifelogging: An overview," *IEEE Trans. Human-Machine Syst.*, vol. 47, no. 1, pp. 77–90, Oct. 2017.

[78] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The epic-kitchens dataset," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 720–736.

[79] F. De la Torre, J. Hodgins, A. Bargteil, X. Martin, J. Macey, A. Collado, and P. Beltran, "Guide to the carnegie mellon university multimodal activity (cmu-mmac) database," Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-RI-TR-08-22, Apr. 2008.

[80] G. Abebe, A. Catala, and A. Cavallaro, "A first-person vision dataset of office activities," in *Multimodal Pattern Recognition of Social Signals in Human-Computer-Interaction*, F. Schwenker and S. Scherer, Eds. Cham, Switzerland: Springer, 2019, pp. 27–37.

[81] D. Damen, T. Leelasawassuk, O. Haines, A. Calway, and W. Mayol-Cuevas, "Discovering task relevant objects and their modes of interaction from multi-user egocentric video," in *Proc. Brit. Mach. Vis. Conf.*, 2014.

[82] G. A. Sigurdsson, A. Gupta, C. Schmid, A. Farhadi, and K. Alahari, "Charades-ego: A large-scale dataset of paired third and first person videos," 2018, *arXiv:1804.09626*. [Online]. Available: http://arxiv.org/abs/1804.09626

[83] Y. Poleg, A. Ephrat, S. Peleg, and C. Arora, "Compact CNN for indexing egocentric videos," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2016, pp. 1–9.

[84] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, nos. 5–6, pp. 602–610, Jul. 2005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0893608005001206

[85] P. J. Chase, "Algorithm 382: Combinations of m out of n objects [G6]," *Commun. ACM*, vol. 13, no. 6, p. 368, Jun. 1970, doi: 10.1145/362384.362502.

[86] G. King and L. Zeng, "Logistic regression in rare events data," *Political Anal.*, vol. 9, no. 2, pp. 137–163, 2001.

[87] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[88] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification Regression Trees*. Boca Raton, FL, USA: CRC Press, 1984.

[89] R. Fortet and E. Mourier, "Convergence de la répartition empirique vers la répartition théorique," *Annales Scientifiques de l'École Normale Supérieure*, vol. 70, no. 3, pp. 267–285, 1953.

[90] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[91] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 32, no. 1. Bejing, China: PMLR, Jun. 2014, pp. 647–655. [Online]. Available: http://proceedings.mlr.press/v32/donahue14.html

[92] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2066–2073.

ALEJANDRO CARTAS received the B.S. degree in computer engineering from the Instituto Tecnológico Autónomo de México (ITAM), Mexico, in 2009, and the M.Sc. degree in artificial intelligence from the University of Edinburgh, U.K., in 2011. He is currently pursuing the Ph.D. degree in mathematics and informatics with the University of Barcelona, Spain.

PETIA RADEVA received the B.S. degree in applied mathematics from the University of Sofia, Bulgaria, in 1989, and the M.S. and Ph.D. degrees in image processing, computer graphics, and artificial intelligence from the Universitat Autónoma de Barcelona, Spain, in 1993 and 1996, respectively.

Since 2019, she has been a Full Professor and P.I. of the Consolidated Research Group Computer Vision and Machine Learning, University of Barcelona. She is currently a Senior Researcher with the Computer Vision Center, Universitat Autónoma de Barcelona. Her current research interests include food intake monitoring, domain adaptation and multitask learning, uncertainty modeling, and lifelogging and egocentric vision.

Dr. Radeva was awarded IAPR Fellow, since 2015, ICREA Academia assigned to the 30 best scientists in Catalonia for her scientific merits, since 2014, and received several international awards (Aurora Pons Porrata of CIARP and Prize Antonio Caparrós for the best technology transfer of UB). She is an Associate Editor of *Pattern Recognition* journal and *International Journal of Visual Communication and Image Representation*.

MARIELLA DIMICCOLI received the M.S. degree in computer engineering from the Polytechnic University of Bari, Italy, in 2004, and the MASt and Ph.D. degrees in signal theory and communications from the Technical University of Catalonia, Spain, in 2006 and 2009, respectively.

After finishing her Ph.D. degree, she spent a year as a Postdoctoral Researcher at the Laboratory of Physiology of Perception and Action, CNRS-Collège de France. From 2011 to 2013, she was a Postdoctoral Researcher with the Laboratory of Applied Mathematics, Paris Descartes University, France. In 2013, she was a Visiting Professor with the Image Processing Group, University Pompeu Fabra, Spain. In 2014, she joined the Computer Vision Center and the University of Barcelona with a Beatriu de Pinós grant. She is currently a Ramón y Cajal Fellow with the Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Spain.