Noname manuscript No.
(will be inserted by the editor)

An Adversarial Semi-Supervised Approach for Action Recognition from Pose Information

George Pikramenos · Evaggelos Spyrou · Eirini Mathe · Eleanna Vali · Ioannis Vernikos · Antonios Papadakis · Phivos Mylonas

Received: date / Accepted: date

Abstract The collection of video data for action recognition is very susceptible to measurement bias; the equipment used, camera angle and environmental conditions are all factors that majorly affect the distribution of the collected dataset. Inevitably, training a classifier that can successfully generalize to new data becomes a very hard problem, since it is impossible to gather general enough training sets. Recent approaches in the literature attempt to solve this problem by augmenting a given training set, with synthetic data, so as to better represent the global distribution of the covariates. However, these approaches are limited because they essentially involve hand-crafted data synthesizers, which are typically hard to implement and problem specific. In this work, we propose a different approach to tackling the above issues, which relies on the combination of two techniques: pose extraction, and domain adaptation as a means to improve the generalization capabilities of classifiers. We show that adapted skeletal representations can be retrieved automatically in a semi-supervised setting and these help to generalize classifiers to new forms

G. Pikramenos · E. Mathe · I. Vernikos · E. Spyrou
Institute of Informatics and Telecommunications, National Center for Scientific Research - “Demokritos,” Athens, Greece
E-mail: {george.pikramenos, emathe, ivernikos, espyrou}@iit.demokritos.gr

G. Pikramenos · A. Papadakis
Department of Informatics, University of Athens, Athens, Greece
E-mail: apapadakis@di.uoa.gr

E. Mathe · Ph. Mylonas
Department of Informatics, Ionian University, Corfu, Greece
E-mail: fmylonas@ionio.gr

I. Vernikos · E. Spyrou
Department of Informatics and Telecommunications, University of Thessaly, Lamia, Greece

E. Vali
School of Electrical and Computer Engineering, National Technical University of Athens, Athens, Greece
E-mail: el14032@central.ntua.gr

of measurement bias. We empirically validate our approach for generalizing across different camera angles.

Keywords Action Recognition · Domain Adaptation · Adversarial Neural Networks

1 Introduction

During the last decades, human activity recognition from video has appeared as one of the most challenging computer vision tasks [2]. It combines ideas from the research areas of pattern recognition and machine learning and has several applications such as surveillance, assisted living, human-machine interaction and affective computing. A categorization of related tasks has been proposed in [31], dividing them into gesture, action, and group activity recognition and differentiating them based on duration, involved body parts and number of interacting actors and/or objects. More specifically and according to the aforementioned categorization, gestures are instant activities, involving at most a couple of body parts. Actions require a larger amount of time and may involve more body parts. An interaction is performed between two "actors" or between an actor and an object. Finally, a group activity may be some combination of the above. Current open research challenges include the representation, analysis and recognition of actions [3].

During the early years of the previous decade, training of recognition algorithms was typically based on extracted features from raw activity visual data [24]. In short, the main drawback of such approaches was an observable drop in performance for large-scale datasets, i.e., comprised of examples performed by a large number of actors and for a large number of classes. Moreover, these early approaches are not robust to viewpoint changes and both these drawbacks make them insufficient for real-life applications. Although research towards feature extraction continued for several years, it was not until recently that such limitations have started to be surpassed. The keys to this, were first the development of low-cost RGB cameras that also extract depth information, such as the Microsoft Kinect[1] or the Asus Xtion,[2] and later the availability of modern graphics processing units (GPUs), which enabled fast training of deep neural network architectures [11]. The extra depth modality has been incorporated to a plethora of research efforts, since it is insensitive to illumination changes, while it offers an enhanced 3D structural information regarding a scene. Moreover, the combination of RGB and depth data has allowed for the extraction and tracking of 3D position of human joints [26].

Recent examples of large-scale action recognition datasets [20,18] have enabled training of deep architectures with partial success. Notably, several datasets comprised of raw (RGB) and depth sequences of video data depicting actions, wherein each example has been recorded by more than one viewpoints

[1] `https://developer.microsoft.com/en-us/windows/kinect`

[2] `https://www.asus.com/gr/3D-Sensor/Xtion_PRO/`

in an effort to provide some realistic variation to the quality of performed actions [20]. However, one may criticize such datasets, in the sense that the data collection process is very susceptible to measurement bias. Factors that may negatively affect data distribution within a collection process are, among others, the equipment used, the camera viewpoint or even the environment (e.g., illumination conditions). Therefore, in case of training a model for recognizing actions (e.g., a classifier), a common problem one may encounter is lack of generalization to unseen data examples, since the collection of sufficiently "general" datasets is inarguably an impossible task.

Contemporary approaches typically aim to surpass this problem through augmentation [29] of the available training data, i.e., by constructing "synthetic" data, in a way that the global distribution of the covariates is better represented e.g., by adding some kind of noise, translating/rotating the input data, cropping the scene or by using some domain specific heuristic [5,17]. Of course, such hand-crafted data synthesizers are typically hard to implement, while in most cases they are problem-specific or domain-dependent. One could argue that for any given application, a large corpus of data can be collected ad hoc in order to perform supervised training. However, the collection process is time-consuming and expensive, with the main bottleneck being the process of data annotation. Typically, a realistic use case is to have very few labeled data in an otherwise unlabeled dataset.

In previous work [21] we presented an approach for the recognition of human actions targeting at activities of daily living (ADLs) [14]. Skeletal information was used to create images capturing the motion of joints in 3D space. These images were then transformed to the spectral domain using well-known image transforms. A deep Convolutional Neural Network was trained to classify those images. Skeletal data and pose extraction are a good way to generalize across measurement biases such as environmental conditions but are still problematic when changes in viewpoint occur. In [22] we applied a geometric transform for augmentation of the available data in order to better generalize across viewpoint changes.

In this work, we propose a different approach for addressing the aforementioned issues, which is based on adversarial domain adaptation algorithms. More specifically, we show that adapted representations can be retrieved *automatically* to perform inference on a *sparsely* labeled dataset, using a model that has been trained on a related labeled dataset. That is, we introduce a technique for tackling, in the semi-supervised setting, the problem of classifying actions from video obtained with one form of measurement bias using a model trained on data where a different form of measurement bias is present. As such, our technique can be utilized to generalize classifiers to new forms of measurement bias. We perform experiments where we demonstrate that generalization between different viewpoints can be achieved without data augmentation.

Our approach combines ideas from domain adaptation and action representation from video data using skeletal features to provide a novel method for mitigating covariate shift in the form of measurement biases. In particular, different measurement settings are subject to distributional discrepancies that can prove catastrophic for the generalization ability of classification models. To the best of our knowledge, our work is the first to consider regularizing model training using a domain confusion term in the objective for producing classifiers that are robust to altered measurement environments and subjects for action recognition from video data. In addition, we apply our proposed approach to tackle different viewpoint scenarios. This is an application that is very important in assisted living environments since ensuring similar viewpoints for different sites is impossible.

The rest of this paper is as follows: in section 2 we present related work in the field of human action recognition, focusing on a) publicly available datasets; b) human recognition tasks; c) research works that are based on image representations of skeletal joints; and d) transfer learning for action recognition. Then, in section 3 we present the motivation for our approach and also a brief overview. Next, in section 4, we present the proposed approach which consists of two steps, i.e., classification and semi-supervised domain adaptation. Experimental results and technical details are presented in section 5, while conclusions are drawn in section 6, wherein plans for future works are also presented.

2 Related Work

In this section we present related work focusing in the research areas where our work lies, i.e., human action recognition from visual data, adversarial domain adaptation and transfer learning for action recognition. More specifically, in subsection 2.1 we present a brief overview of human action recognition datasets. Then, in subsection 2.2 we define the two main recognition tasks. In subsection 2.3 we focus on image representations of skeletal joint motion, while research works regarding transfer learning for action recognition are presented in subsection 2.4.

2.1 Human Action Recognition Data Sets

Large-scale action recognition datasets can serve as a good starting point for tackling action classification, especially when coupled with other techniques such as feature adaptation or data augmentation. The early publicly available datasets were limited to a small number of simple actions; e.g., the KTH dataset [24] was limited to 6 actions such as *walking*, *running*, *hand clapping* and other similar. A few years later, several datasets targeted more complex actions; e.g., the Hollywood dataset [13] included actions such as *answer phone*, *get out of car*, *hand shake* and other similar. In less than a decade, more challenging datasets emerged, comprising of large numbers of more complex actions and even interactions with objects; e.g., in UCF101 [27] or HMDB [12], instances of actions such as *playing cello*, *horse riding*, *swing baseball bat*, *fencing* and other similar. Contemporary large-scale multimodal datasets such as PKU-MMD [18] or the NTU [25], are comprised of large numbers of training videos and skeletal and depth sequences.

2.2 Human Action Recognition Tasks

According to Wang et al. [31] human action recognition tasks may be divided into two major categories:

- **segmented recognition**: the given input video sequence contains *only* the action to be recognized. This means that any frame before/after the action, i.e., not depicting a part of the action, has been removed. In this case, Recurrent Neural Networks (RNNs) [7] or CNNs [15] are typically used.
- **continuous recognition**: the goal is to recognize actions within a given video; the video may or may not depict a single action. In that case, also known as "online" recognition, RNNs are typically used.

Note that our approach falls in the first category; we consider video segments depicting actions limited to those we aim to recognize.

2.3 Image Representations of Skeletal Joint Motion

Typically, when a CNN is used and the only available motion modality are skeletal data, an intermediate visual representation of skeletal sequences is required. This representation should capture both spatial and temporal information regarding the motion of joints, which can be encoded in its color and/or texture properties. In this section our goal is to present research works that are based on visual representations of sequences of 3D skeletal data of human actions and training deep networks, i.e., an intermediate hand-crafted feature extraction step is not included in the process. Skeletal data typically consist of a set of skeletal joints moving in 3D space over time, i.e., for each joint 3 1D signals are generated per action. The extraction of joints from video requires depth information. Several research efforts have been proposed, aiming to provide 2D pseudo-colored image representations of skeletal motion.

In the work of Du et al. [6], a color image representation of skeleton sequences has been proposed. More specifically, they create pseudo-colored images that have been generated by the x, y and z spatial coordinates of skeletons and corresponded to the R, G and B components, respectively. Then, they used them to feed a CNN. In order to preserve the spatial information, the set of joints is split into five subsets corresponding to arms, legs and the trunk. For evaluation, they used two publicly available datasets for human actions and gestures. To preserve temporal information, spatial representations were chronologically arranged. Moreover, Wang et al. [30] proposed a representation for preserving the spatio-temporal information of skeletal joint motion, namely “joint trajectory maps,” where the spatio-temporal information originated from 3D skeleton sequences is transformed into three 2D images by encoding the dynamics of joint trajectories. More specifically, maps are constructed by appropriately setting saturation and brightness, so that texture would ultimately correspond to motion magnitude; each is based on the projected trajectory of the skeleton to a Cartesian plane. For evaluation, they used four publicly available datasets for human actions and gestures. Similarly, Hou et al. [8] transformed the extracted skeleton joints into a representation called “skeleton optical spectra,” and used CNNs for classification. Their representation is based on the idea that hue changes should reflect to the temporal variation of skeletal motion and is composed of four steps: mapping of joint distribution, spectrum coding of joint trajectories, spectrum coding of body parts, and joint velocity weighted saturation and brightness. They claimed that it is possible to use a standard CNN to learn dynamic features from skeletal sequences, i.e., without having to train a huge number of parameters. They also maintained that an advantage of their approach is its ability to work using an insufficient corpus of training examples. They also evaluated their approach using publicly available gesture, action and interaction recognition datasets.

Li et al. [16] proposed another color texture image representation of 3D skeletal data called “joint distance maps,” and opted for encoding the pairwise joint distances in the 3 orthogonal 2D planes also using a fourth one to encode distances in the 3D space. Hue was used to encode distance variations. Each map was separately classified and a late fusion scheme was then adopted. They asserted that their representation is suitable for both single-view and cross-view action recognition and evaluated it using human action and interaction datasets. Liu et al. [19] attempted to compensate for the variation of the initial position and orientation of the skeleton and for changes of viewpoint that occur during its motion within an action or changes. First, they use a transform whose goal is to make the approach view-invariant. Then, they apply a visualization that creates a series of color images from the transformed skeleton joints. A joint was represented by its 3D space coordinates, while time and joint label were also added to create a 5D representation. Upon projection to a 2D image using two of the aforementioned dimensions, the remaining three were used as R, G, B values to form pseudo-colored images. This representation is used to capture the 3D joint spatio-temporal information and are followed by image processing and enhancement steps whose goal is to highlight patterns. For the extraction of discriminative features and they use CNNs and fuse action class scores. They evaluate their approach using publicly available action and gesture datasets. In the work of Ke et al. [10] a novel deep learning framework for 3D action recognition has been proposed. Contrary to the aforementioned works, Ke et al. did not rely on the extraction of 3D joint coordinates. Instead, they extracted translation, rotation and scale invariant features by subsets of joints as in [6]. From each, they extracted cosine distances and normalized magnitudes from vector representations generated from pairwise relative positions between joints. These representations were not treated as time series but were concatenated so as to form a 2D representation. Classification is performed by a deep network which is composed by two parts: the first works as a feature extractor, while the second is responsible for the generation of discriminative and compact representations. They evaluated their approach using publicly available action and interaction datasets.

2.4 Transfer Learning for Action Recognition

The notion of transfer learning [23], i.e., of storing knowledge which has been gained upon the solution of a problem and then of its use in a different, yet related one, has great potential in aiding the action recognition problem.

Xu et al. [38] propose the use of autoencoders on high-level representations obtained through semantic transfer. In particular, video data is represented by action bank features [39]. Action banks for source and target datasets are then brought to lie in the same space by applying PCA to a fixed number of components. A latent representation is computed by training the autoencoders to reconstruct the class centroid over both datasets for each instance in the source and target domain. Unlike our method, the training procedure requires labels for the target domain in order to be able to train the target autoencoder. Moreover, Yusuf and Koniusz [40] propose a CNN architecture on kernel based feature maps extracted from skeletal data and leverage a supervised domain adaptation technique to increase the robustness of the learnt classifier. Note that again, target labels are required and this method cannot be used for labeling an unlabeled or sparsely labeled dataset. The particular supervised adaptation technique utilized is the So-HoT algorithm [41] which attempts to align second order statistics across datasets.

Another interesting work attempts to transfer knowledge for the video action recognition task from still images, which are typically more available. Zhang et al. [42] manipulate feature extracted through kernel PCA to effectively transfer knowledge between heterogeneous domains. The video data is analyzed into key frames using a shot boundary detection algorithm. The method benefits from both labeled and unlabeled video data and thus lies within the frame of semi-supervised learning as does our method. Finally, similarly to our work, Hachiya et al. [43] tackle homogeneous adaptation for action recognition in the semi-supervised setting. However, rather than video, they use accelerometer data and extract a representation based on orientation invariant statistics. Moreover, while we use a feature learning approach for adapting classifiers (adversarial discriminative adaptation), in the aforementioned work, importance sampling based adaptation is utilized.

3 Motivation and Overview

To assess possible applications of the domain adaptation approach towards the aforementioned goal, consider the following scenario: In an assisted living environment one would typically monitor the behaviour and activities of an actor/user (e.g., an elderly resident) to infer her/his emotional state, or her/his health. Typically, cameras would be placed in the user's environment and the collected data would need to be analyzed using a trained model, e.g., a classifier. As we have discussed, obtaining such a classifier to work on-the-fly for any user and for any environment is not a trivial task.

The primary goal of this work is to experimentally prove whether an adversarial domain adaptation approach may be applied in a human activity recognition for adaptation of samples originating from different viewpoints, so as to improve the classification accuracy. The secondary goal of our work is to investigate whether our methodology may still be applied in intense viewpoint changes. We assume that action instances are available for all viewpoints and for all classes. Note that our methodology is not limited to the classification approach adopted. Herein we demonstrate a case with image representations of 3D skeletal data and classification using Convolutional Neural Networks, yet other representations and architectures may be also used.

To the best of our knowledge, this is the first work to utilize ideas from domain adaptation and pose estimation to create robust classifiers that can generalize across different viewpoints and subjects. Furthermore, we extend an existing framework for unsupervised domain adaptation through adversarial networks to the semi-supervised setting, where supervision signal from the target domain helps guide the underlying distribution alignment process.

Our approach could be deployed in such a real-life setting to alleviate this issue. A model could be trained on generic action recognition datasets or on data collected from multiple users. The skeletal information extraction and adaptation procedures could then be used to improve the generalization capabilities of the model on data collected from a new user/environment. In addition, relocating the monitoring equipment for an existing user could hinder a trained model's utility. Our method could again readily be applied in such a scenario to boost performance.

In Fig. 1 we illustrate an overview of the proposed approach. In brief, given two camera viewpoints (i.e., Left and Right), raw visual data are captured, while e.g., an actor performs a given action. From the RBG and depth data captured, skeletal sequences may be extracted. Upon concatenation and interpolation of those sequences, a signal image is formed. Using the Discrete Sine Transformation, an activity image is formed. The collection of captured action instances form two datasets, namely D_s and D_t. The former corresponds to the source domain, while the latter to the target domain and is further split to a labelled (D_t^l) and an unlabelled (D_t^u) subset. D_s is used for training the source model S, consisting of the source representer M_s and the classifier C. M_s is the input of the viewpoint adaptation procedure, which is a process combining domain confusion and supervised training. The former utilizes D_s and D_t^u while the latter uses D_t^l. The output of the viewpoint adaptation is the target representer model M_T. Now, when the target camera captures a raw data sample, following the aforementioned process its activity image is

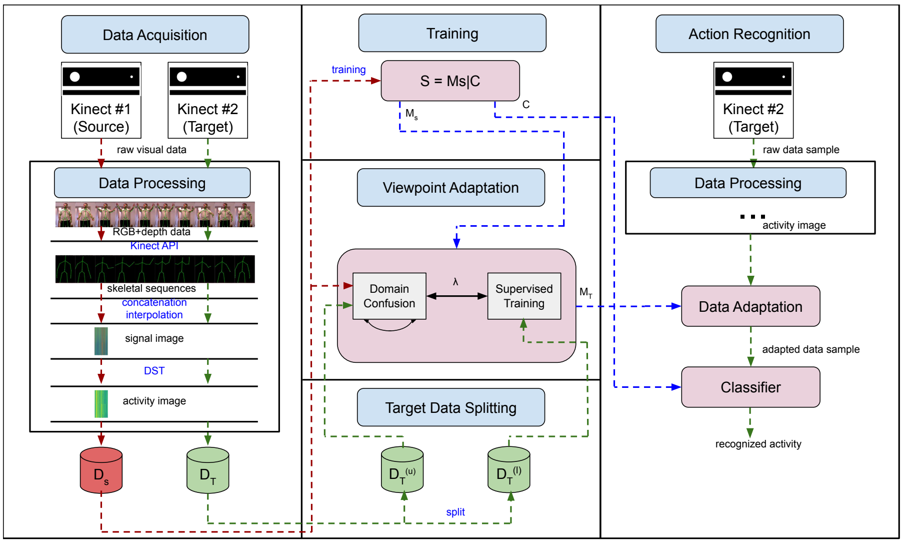

Fig. 1 A visual overview of the proposed approach.

adapted using M_T and the action is recognized by C, fed with the adapted sample.

4 Methodology

As it has already been discussed, our approach builds on ideas from the areas of pose extraction and domain adaptation. We now give some technical details regarding: a) the extraction of skeletal joint information from video; b) the representation used for capturing spatial and temporal properties of the aforementioned information during the performance of some activity; c) the classification approach we follow; and finally, d) the adversarial domain adaptation we propose.

4.1 Classification

As it has been already mentioned, for human activity recognition tasks human motion is typically captured by depth cameras, which extract both RGB video and depth maps per video frame, i.e., an extra video channel where the value of each pixel is related to the depth of the corresponding object to the image plane. Our approach utilizes the modality that corresponds to the motion of joints in 3D space. More specifically, we require as input 3D trajectories of skeletal joints (i.e., x, y and z coordinates at each frame for each) during an action.

We work with 3D skeletal data that have been captured with the Microsoft Kinect v2 sensor. These data consist of 25 human joints per skeleton. The set

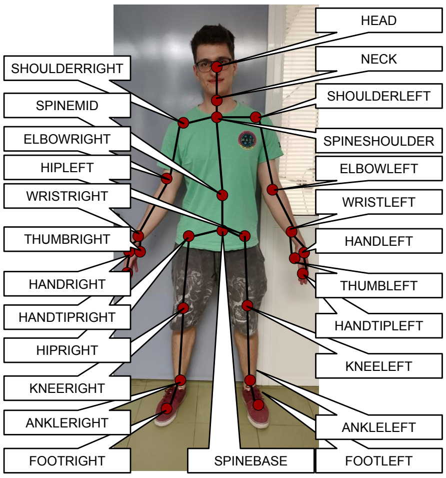

Fig. 2 Extracted human skeleton 3D joints using the Kinect SDK.

of skeletal joints is illustrated in Fig. 2. Up to 6 skeletons can be simultaneously extracted in real time using the Kinect SDK. Therein, a human skeleton corresponds to a graph; nodes correspond to body parts such as arms, legs, head, neck and so on, while edges follow the body structure. Moreover, a parent-child relationship is implied. For example, the joint "HEAD" is parent of "NECK," while the "NECK" is the parent of "SPINE_SHOULDER," and so on. Each joint consists a 3D signal capturing its 3D position over time. Equivalently, this signal may be seen as 3 1D signals; each corresponding to a coordinate. Therefore, 75 such 1D signals can be obtained from the set of 25 joints and for any given video sequence. Note that their duration may vary, since different actions may require different amounts of time. Also different subjects may perform the same action with similar, yet not equal duration. To address the aforementioned problem of temporal variability between actions and between users, an interpolation step is necessary. Upon experimenting with several duration values, we ended up setting the duration of all videos to be 159 frames by performing a linear interpolation step.

The representation used in this work has been partially inspired by the one of Jiang and Yin [9], who concatenated raw signal measurements collected by the inertial measurement units of mobile phones and then extracted the 2D Discrete Fourier Transform (DFT) of the concatenated signal. Similarly, we first create a 2D image by concatenating the aforementioned 75 1D signals corresponding to the joint motion in 3D space. In what follows, this representation will be referred to as "signal" image. Then, from each signal image we create an "activity" image, by applying the 2D Discrete Sine Transform (DST).

Note that in previous work [21] we conducted classification experiments using 2D DFT, 2D Fast Fourier Transform (FFT), 2D Discrete Cosine Transform (DCT) and 2D DST. The latter was chosen in this work because it showed best

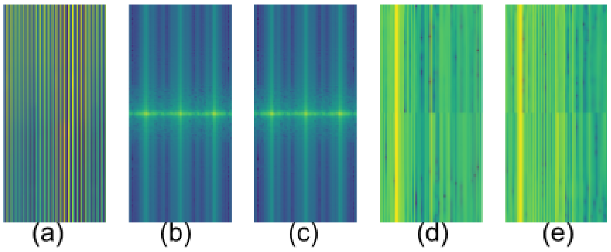

Fig. 3 (a) A signal image; activity image resulting upon (b) DFT; (c) FFT; (d) DCT; (e); DST. Action is *playing with phone/tablet*. DFT and FFT images have been processed with log transformation for visualization purposes. Figure best viewed in color.

accuracy in most settings. From the 2D DST we preserve only the magnitude, i.e. we discard the phase, and also normalize using the orthonorm. Obviously, the result of this processing is a 2D signal, which we treat as a 2D image. We herein remind that we do not extract any hand-crafted features at any step of the proposed methodology. In Fig. 3 we illustrate an example signal image and the corresponding activity image.

We herein remind that the goal of this work is limited to action classification. Therefore, it belongs to the category of segmented recognition (see section 2), since it does not perform a temporal segmentation step. As we shall see in section 5, we work using pre-segmented video sequences, aiming to only recognize the performed actions within each segment and under the hypothesis that each segment contains exactly one action.

The architecture of the proposed CNN is presented in detail in Fig. 10. In brief, the first convolutional layer filters the 159×75 input activity image with 32 kernels of size 3×3. The first pooling layer uses "max-pooling" to perform 2×2 sub-sampling. The second convolutional layer filters the 76×34 resulting image with 64 kernels of size 3×3. A second pooling layer uses "max-pooling" to perform 2×2 sub-sampling. A third convolutional layer filters the 36×15 resulting image with 128 kernels of size 3×3. A third pooling layer uses "max-pooling" to perform 2×2 sub-sampling. Then, a flatten layer transforms the output image of the last pooling to a vector, which is then used as input to a dense layer using dropout. Finally, a second dense layer produces the output of the network. To avoid overfitting, the most popular approach which is also adopted in this work is the use of the dropout regularization technique [28]: at each training stage several nodes are "dropped out" of the network. This way overfitting is reduced or even prevented, since complex co-adaptations on training data are prevented. In addition, we use a validation set to monitor the validation loss and we utilize the early stopping technique.

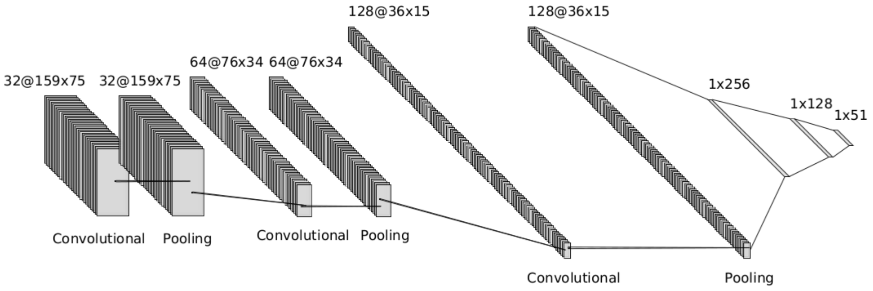

Fig. 4 The proposed CNN architecture.

4.2 Implemented Adversarial Domain Adaptation

Domain adaptation [32] is a technique for automatically decreasing the test error of a classifier when it is trained on data sampled from a different distribution than the test set. In this context, the train set is called the *source* data and the test set is called the *target* data. In particular, the underlying assumption in the domain adaptation setting, is that the distribution of the covariates in source and target domains is different but the conditional distribution of the label random variable given the covariate values is the same. There is a wide range of domain adaptation methods [33], however in this work we are limited solely to adversarial neural network algorithms [34], [35], [36], which offer flexibility and are considered state-of-the-art. The main idea in such algorithms is similar to the idea in Generative Adversarial Networks [37].

More specifically, the goal of such algorithms is to learn two embeddings M_S, M_T for source and target data, respectively, into some latent space $\mathcal{L}$, such that the distributions of target and source data in $\mathcal{L}$ are the same. We further require that the representation of data in $\mathcal{L}$ is rich enough to support classification of source instances. Note that typically no labelled data for the target domain is available during training, while the source domain data is annotated.

Latent space distribution alignment is typically achieved through the use of a domain discriminator network D which is trained to discriminate the domains in latent space for fixed M_S and M_T. In turn, gradients from the discriminator are reversed and used to update the parameters of M_T, keeping D fixed. This process is repeated until an equilibrium is reached. Under mild assumptions on the capacity of the involved networks, it can be shown that at equilibrium, the distributions of source and target latent representations are aligned (similar to [37], [35]). A visual illustration of the herein adopted domain adaptation approach is provided in Fig. 5. Note that in some approaches, a target and a source model are trained concurrently while in other approaches the source model is trained in a separate phase before the adaptation procedure and its parameters are kept fixed while the target model is being trained.

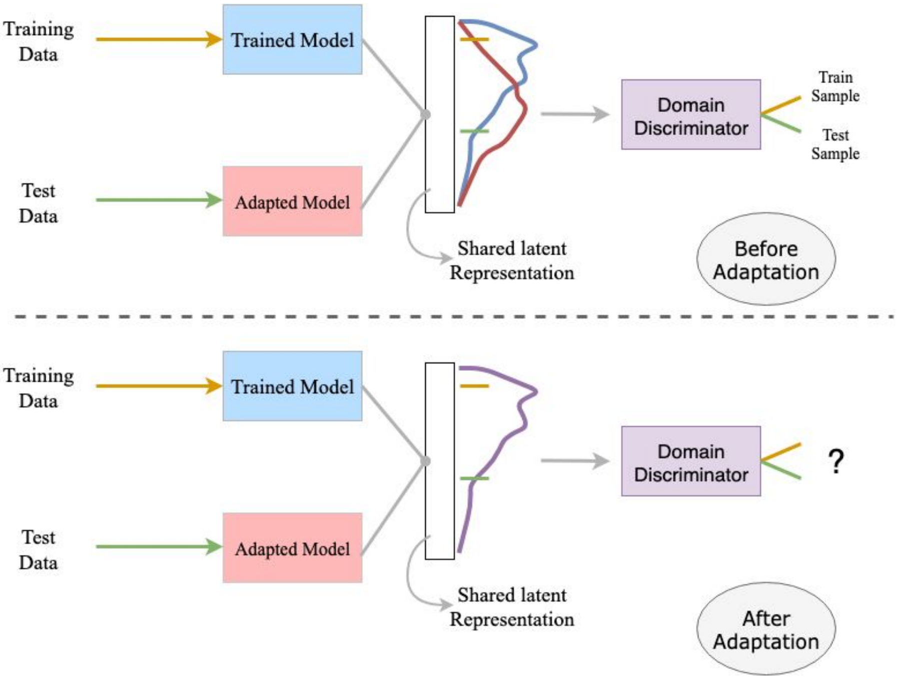

Fig. 5 Visual illustration of the effects of domain adaptation. On top the adapted model is equal to the trained model. Because source and target data have different distributions, the resulting distributions in latent space are different and the domain discriminator can be trained to classify instances in latent space. After the adaptation procedure is finished, the adapted model has been trained so that the distribution of its output, when its input is distributed as the test data, is equal to the source data distribution in latent space. The domain discriminator can no longer discriminate the domains.

Our method is inspired from Adversarial Discriminative Domain Adaptation (ADDA) [35]. In particular, we adapt ADDA to the semi-supervised setting, i.e., we assume that a small part of the target domain is labelled and, for the rest of it, the labels are unknown. This assumption is in line with a lot of problems that need to be resolved in practice. As discussed earlier, annotating a dataset is the main bottleneck in constructing training sets but labeling a small portion of a given dataset is considered feasible. Our aim is to boost the generalization capabilities of a source data classifier given a large corpus of sparsely labeled data from the target domain. Our method is complementary to source classification; We build on top of a source classifier obtained through any supervised learning approach on the source data.

The target domain labeled data provide supplementary information to the model during training and in general lead to better models compared to unsupervised adaptation. The classifier C, remains fixed with pre-computed parameters (e.g. obtained through standard supervised learning on the source domain), while the target representer M_t is trained.

In contrast to ADDA, the loss function used to train the target representer network $\mathcal{L}_{M_t}$ is given by

$$\mathcal{L}_{M_t} = -E_{x_t \sim X_T}[\log(D(\mathbf{M}_t(x_t)))] - \lambda E_{x_t, y_t \sim X_T}[\log(C(\mathbf{M}_t(x_t)))] \quad (1)$$

where the second expectation is replaced with the empirical average over the labeled subset of the target domain. In (1), λ is a trade-off parameter; for $\lambda = 0$ we get the typical ADDA loss, while for $\lambda \to \infty$ we obtain a typical supervised optimization task. Note that a very small portion of the target data set is labeled and as such simple supervised learning is not a robust approach to tackle the classification problem. In practice, λ is selected according to some empirical validation technique. The aforementioned domain adaptation methodology is presented as pseudocode in Algorithm 1.

The described method was used to generalize a source model across viewpoint changes. In addition, we attempted to generalize a model across different subjects. In particular, video data featuring 10 subjects were selected from the PKU dataset and 10 different source-target data pairs were tested. For each test, the actions performed by one subject served as a target set and the rest of the actions (i.e. by the other subjects) as a source set. However, we found that source and target errors were similar without any adaptation indicating that no covariate shift is present in the cross subject setting. This indicates that covariate shift due to different subjects has been effectively mitigated through the use of skeletal data and interpolation.

We should herein note that the domain adaptation step introduces a significant overhead in any classification methodology, since it is a time consuming task. However, it consists an offline step, therefore, it may still be used in real-life applications, since it does not have any effect in a given deep neural network architecture in terms of complexity.

5 Experiments

5.1 Dataset

The experimental evaluation of the proposed approach has been performed using part of the well-known PKU-MMD data set [18]. As it has already been mentioned, PKU-MMD comprises a large-scale benchmark focusing on human action understanding as well as on multi-modal action analysis. It consists of approximately 21.5K action instances from 51 action categories. The aforementioned instances span into approximately 5.4M video frames. Each video may contain several actions and lasts about 3–4 min. Indicative actions of PKU-MMD include *drinking*, *waving hand*, *putting on the glasses* and so on. Moreover, 10 interactions are included, such as *hugging*, *shaking hands* and so on.

The dataset has been recorded with 66 human subjects participating in the data collection process. Each subject is part of 4 action and 2 interaction videos. Each action has been recorded by 3 camera angles (tagged left (L),

Algorithm 1: *Semi-Supervised Discriminative Domain Adaptation*

```
Input: A source dataset $D_S$, a target dataset $D_T$ separated into a labeled subset
  $D_T^{(l)}$ and an unlabelled subset $D_T^{(u)}$, a source representer $M_S$, a source classifier
  $C$, a hyperparameter vector $\vec{\lambda}$.
Output: A target model $C(M_T)(\cdot)$.
INITIALIZE():
$M_T \leftarrow M_S$; $D \leftarrow$ random_initialization();
$C$.freeze();
TRAIN():
for $\vec{\lambda}(MAXITER)$ do
    $M_T$.freeze(); $D$.unfreeze();
    for $\vec{\lambda}(D\_iter\_per\_cycle)$ do
        $D$.train_on_batch($\lceil 0.5\vec{\lambda}(\text{batch\_size})\rceil, D_S$);
        $D$.train_on_batch($\lfloor 0.5\vec{\lambda}(\text{batch\_size})\rfloor, D_T^u$);
    end
    $D$.freeze(); $M_T$.unfreeze();
    $D \circ M_T$.get_batch_gradients($\vec{\lambda}(\text{batch\_size}), D_T^{(u)}$);
    $C \circ M_T$.get_batch_gradients($\vec{\lambda}(\text{batch\_size}), D_T^{(l)}$);
    $M_T$.update_weights($\vec{\lambda}(\lambda)$);
end
return: $C \circ M_T$.
```

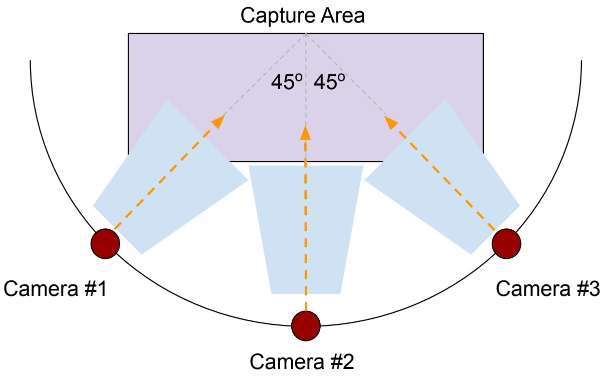

Fig. 6 Camera setup of the PKU-MMD dataset. Cameras #1, #2, #3 correspond to L, M, R, respectively.

right (R) and middle (M)) using the Microsoft Kinect v2 camera. The camera setup is illustrated in Fig. 6. Moreover, in Fig. 7, we illustrate a skeleton as seen by the three cameras. For each action example, raw RGB video sequences, depth sequences, infrared radiation sequences and extracted 3D positions of skeletons are the modalities provided.

Note that as it has already been mentioned, our main focus was to assess whether the proposed approach may be used for real-life ambient assisted living scenarios and more specifically for the recognition of ADLs, instead of a more generic set of daily activities. In such scenarios, it is of major importance to accurately detect a few activities closely linked to the subject's quality

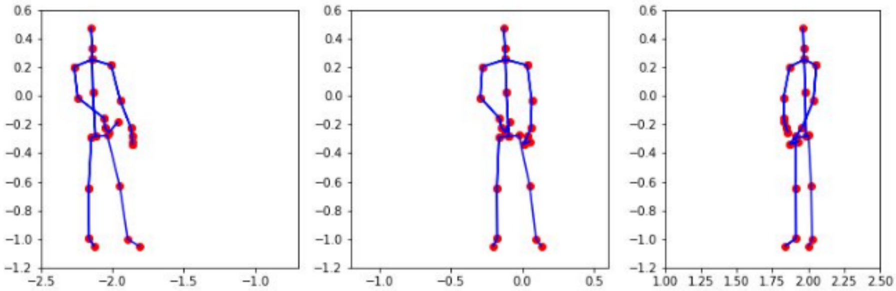

Fig. 7 A skeleton as seen by cameras #1 (L), #2 (M), #3 (R), from left to right, respectively. For illustrative purposes, depth information, i.e., z-coordinate has been discarded.

of life. Therefore, as in our previous work [21] we selected 11 out of the 51 classes of PKU-MMD, which we believe are the closest to ADLs or events in such a scenario. The selected classes are: *eat meal snack*, *falling*, *handshaking*, *hugging other person*, *make a phone call answer phone*, *playing with phone tablet*, *reading*, *sitting down*, *standing up*, *typing on a keyboard* and *wear jacket*. In Fig. 8 we illustrate sample signal and activity images from these 11 classes.

5.2 Part I: classification

The evaluation protocol we followed is as follows: we first performed experiments per camera position; in this case both training and testing sets are derived from the same viewpoint. Then, we performed cross-view experiments, where different viewpoints were used for training and testing. The goal of these experiments was to test the robustness of the proposed approach in terms of transformation (e.g., a translation and a rotation), which could correspond to abrupt viewpoint changes which typically occur in real-life situations. Finally, we performed cross-subject experiments, where subjects were split in training and testing groups, i.e., any actor "participated" only into one of the groups. As in previous work we found that our representation of skeletal motion suffices for cross-subject shift mitigation. The goal of this part of evaluation was to test the robustness of our approach into intra-class variations. In real-life situations this is expected to happen when a system is trained e.g., within a laboratory environment and is deployed into a real ambient-assistive living environment. Note that in all cases we measured classification accuracy. Detailed results are listed in Table 1 under source model ("S") for ADL action recognition with cross-view shift.

5.3 Part II: domain adaptation

For each experiment, we select one of the camera angles as source data and one as target data. We test the combinations $L \to R, L \to M, R \to M$, where $X \to$

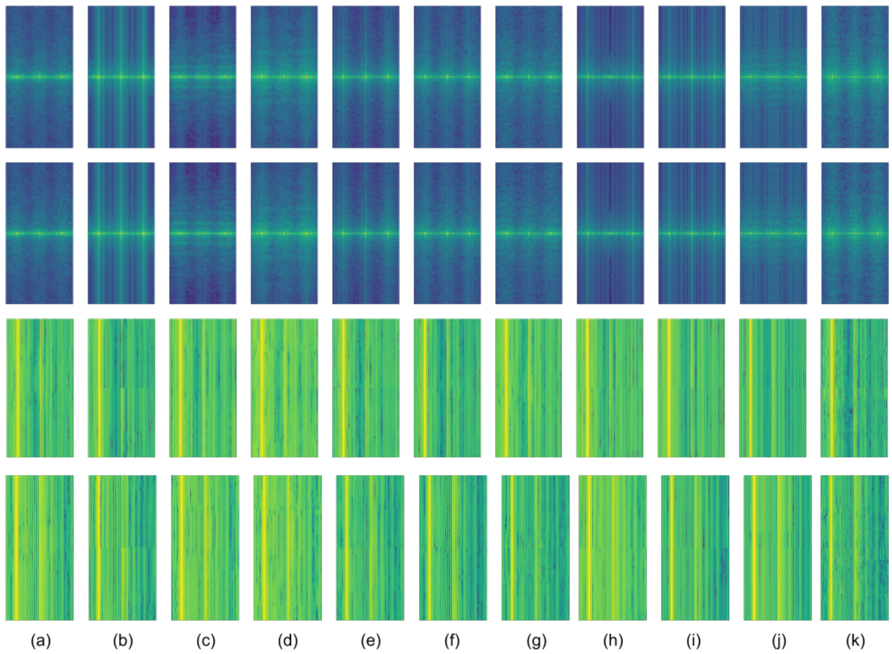

Fig. 8 Examples of activity images from 11 classes and for the 4 transforms used. 1st row: DFT; 2nd row: FFT; 3rd row: DCT; 4th row: DST. a) eat meal/snack; b) falling; c) handshaking; d) hugging other person; e) make a phone call/answer phone; f) playing with phone/tablet; g) reading; h) sitting down; i) standing up; j) typing on a keyboard; k) wear jacket. DFT and FFT images have been processed with log transformation for visualization purposes. Figure best viewed in color.

Y is typical notation in the domain adaptation literature denoting adaptation between source X and target Y. In addition, since we are considering the semi-supervised setting, a small subset is sampled without replacement from the target data to serve as the labeled target instances. This subset is utilized during training both for providing a supervision signal and for unsupervised adversarial training, but it is excluded when calculating validation scores.

For each combination of source and target domains, we vary the percentage of the target data set that is labeled between $0\%, 5\%$ and 10%. In all experiments we utilize a source model (S), which is trained in a standard supervised way on the source domain. For our method, we setup the standard adversarial network setting for domain adaptation using S and we modify the training procedure to include supervised information as in subsection 4.2. The trade-off parameter λ in (1) was set to 1 for all experiments. This resulted upon initial trial and error experiments to determine an appropriate interval. Upon this determination, grid search was performed. We additionally, train a model on the labeled subset of the target domain to serve as a benchmark. In particular, this is done for initial weights taken both randomly and from S yielding models T_{rand} and T_{w_S} respectively.

Table 1 Domain adaptation experimental results. Figures represent average accuracy percentages over ten runs. In each setup significant improvements are indicated with bold. Last column corresponds to the difference between S and DA.

		S	T_{w_S}	T_{rand}	**DA**	diff.
$L \rightarrow R$	0	41.28	-	-	**49.94**	+20.98%
	5	41.28	70.43	61.24	**77.87**	+88.64%
	10	41.28	74.56	70.58	**81.52**	+97.48%
$L \rightarrow M$	0	84.49	-	-	84.21	-0.33%
	5	84.49	86.09	72.56	**88.65**	+4.92%
	10	84.49	91.61	76.90	**92.03**	+8.92%
$R \rightarrow M$	0	84.82	-	-	84.90	+0.09%
	5	84.82	86.17	70.23	**88.32**	+4.13%
	10	84.82	90.54	76.83	**91.47**	+7.84%
$R \rightarrow L$	0	44.20	-	-	**53.41**	+20.84%
	5	44.20	76.36	64.77	**79.90**	+80.77%
	10	44.20	80.56	77.15	**86.78**	+96.54%
$M \rightarrow L$	0	82.68	-	-	82.99	+0.37%
	5	82.68	**86.48**	68.37	**86.78**	+4.96%
	10	82.68	90.14	79.36	**91.12**	+10.21%
$M \rightarrow R$	0	77.76	-	-	**82.66**	+6.30%
	5	77.76	84.95	69.71	**85.90**	+10.47%
	10	77.76	90.01	75.83	**91.14**	+17.21%

Table 2 Classification report for the source model on target data.

	$L \rightarrow R$	$L \rightarrow M$	$R \rightarrow M$	$R \rightarrow L$	$M \rightarrow L$	$M \rightarrow R$
Precision	41.45	81.43	83.56	44.90	81.39	78.91
Recall	42.31	88.93	87.70	44.67	86.07	79.07
F1	41.87	85.01	85.57	44.78	83.66	78.99

Table 3 Classification report for our method with no labelled target data.

	$L \rightarrow R$	$L \rightarrow M$	$R \rightarrow M$	$R \rightarrow L$	$M \rightarrow L$	$M \rightarrow R$
Precision	50.36	80.13	86.28	54.74	80.21	79.96
Recall	52.11	81.22	86.40	50.06	81.37	83.17
F1	51.22	80.67	86.34	52.29	80.78	79.47

Table 4 Classification report for our method with 5% labelled target data.

	$L \rightarrow R$	$L \rightarrow M$	$R \rightarrow M$	$R \rightarrow L$	$M \rightarrow L$	$M \rightarrow R$
Precision	75.44	82.83	82.12	84.74	83.69	82.66
Recall	77.93	89.77	90.39	75.06	86.08	83.83
F1	76.66	86.16	86.06	79.61	84.87	83.24

Our results are summarized in tables 1–5. More specifically, in Table 1 we present the accuracy in all 6 transfer scenarios and for the cases of the source model on target data and for the three cases of the proposed method, i.e., without any labelled target data and with 5% and 10% labelled target data. As it may be observed, the improvement is significant in the more demanding transfer scenarios, i.e., $L \rightarrow R$ and $R \rightarrow L$, where increase ranges between 21% and 98% and is improving as the percentage of labelled target data increases. In most other less demanding transfer scenarios, i.e., those involving M and one of

Table 5 Classification report for our method with 10% labelled target data.

	$L \rightarrow R$	$L \rightarrow M$	$R \rightarrow M$	$R \rightarrow L$	$M \rightarrow L$	$M \rightarrow R$
Precision	77.61	86.24	88.49,	88.19	95.12	94.47
Recall	79.69	90.07	86.98	87.16	93.58	89.22
F1	78.64	88.11	87.71	87.67	94.38	91.77

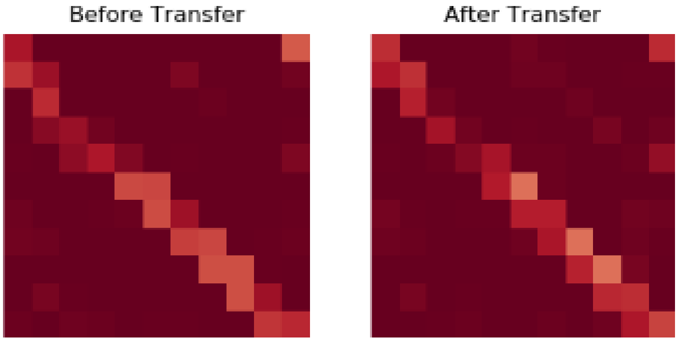

Fig. 9 Confusion matrix Right → Middle. Illustration of the confusion matrix for our classifier before and after transfer.

the other camera setups, the increase of accuracy ranges between 4% and 17% with 5% and 10% labelled target data, respectively. In three of the remaining scenarios, i.e., with no labelled target data, accuracy remains practically the same. Notably, in one case ($M \rightarrow R$), accuracy is increased by 6% even without any labelled training data. In tables 2–5 we present Precision, Recall and F1-scores for the aforementioned cases and for all scenarios. Similar observations may be made also for these metrics.

Moreover we provide, indicatively, color-coded visual representations of the confusion matrices before and after applying our method. These are shown in Figures 9 and 10. For brevity, we only included these for two transfer scenarios ($L \rightarrow R$, $R \rightarrow M$), which are representative of the other scenarios. Similar observations occur for $R \rightarrow L$ and $M \rightarrow R$, $M \rightarrow L$ and $L \rightarrow M$. It is apparent that our method greatly benefits the learnt classifier especially on domains that are less related (e.g. $L \rightarrow R$ as opposed to $R \rightarrow M$). Even with very few labelled target data, the model's performance is boosted significantly and overfitting is avoided due to the regularization that is introduced by the signal coming from the source domain through the adversarial training procedure.

5.4 Implementation Details

The experiments were performed on a personal workstation with an Intel$^{\text{TM}}$i7 5820K 12 core processor on 3.30 GHz and 16GB RAM, using NVIDIA$^{\text{TM}}$Geforce GTX 2060 GPU with 8 GB RAM and Ubuntu 18.04 (64 bit). The deep CNN architecture has been implemented in Python, using Keras 2.2.4 [4] with the

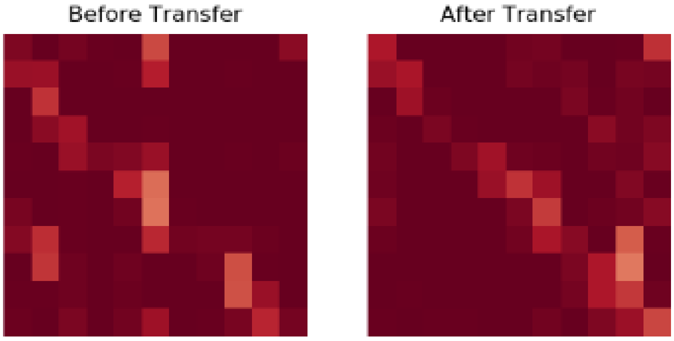

Fig. 10 Confusion matrix Right → Left. Illustration of the confusion matrix for our classifier before and after transfer.

Tensorflow 1.12 [1] backend. All data pre-processing and processing steps have been implemented in Python 3.6 using NumPy[3], SciPy[4] and OpenCV.[5]

6 Discussion and Conclusions

Our motivation for this work was to address the problem of generalizing classifiers for human action recognition to new forms of measurement bias. In particular, in real life applications obtaining annotated training samples from the measurement setup of interest may be unrealistic, since the annotation process is slow and expensive. As such we can only assume that a few labeled instances are available from our particular setup. In addition, different measurement biases make it hard for a classifier trained on a generic action recognition dataset to generalize well. For the aforementioned reasons we propose the use of domain adaptation algorithms in the semi-supervised setting as an effective approach to labelling a sparsely labeled test dataset, in the presence of covariate shift.

In our methodology, we extend previous work for classifying human actions in videos to a setting where training and test datasets are subject to different measurement biases. In particular, we leverage a novel representation of 3D skeletal motion which relies on spectral images obtained through DST and adversarial domain adaptation algorithms for automatically adapting the representation learnt by a deep neural network to a new form of measurement bias. We demonstrated the effectiveness of our proposed methodology on cross-subject and cross-view datasets shifts. We found that our skeletal data representation effectively tackles the cross-subject shift, while domain adaptation allows us to effectively tackle the cross-view shift.

[3] http://www.numpy.org/

[4] https://www.scipy.org/

[5] https://opencv.org/

We evaluated the proposed approach using a popular action recognition dataset as a source, which consists of skeletal sequences which have been captured by 3 Kinect v2 cameras, under different camera angles. The skeletal joints of the human actors involved had been extracted. We performed experiments involving either a single camera (single-view) or more than one (cross-view). We also performed cross-subject experiments to evaluate the robustness of the approach. We mainly focused on a subset of 11 actions which in our opinion are the most close to real-life ADLs. Our initial results indicate that the proposed approach may be successfully applied to human action recognition in real-like conditions, yet a drop of performance is expected when significant changes of viewpoint occur.

We experimented using four different setups. In the first we only performed classification using our novel skeletal motion representation and no adaptation. We empirically found that cross-subject shifts are effectively handled by this method since our target domain accuracy was the same as our test accuracy. In the other three setups we varied the percentage of labelled examples in the target domain to 0, 5 and 10 which consisted of data subject to cross-view shifts. The adaptation procedure provides a clear improvement in all three settings.

In particular, we experimented with a novel transfer scenario where we aim to improve classifier generalization capabilities across different viewpoints. Our method may be generally seen as a regularization technique which improves performance by incorporating knowledge learnt from another domain. It is evident that introducing a domain confusion term in the objective function leads to better generalizing classifiers when target labelled data are scarce. Our procedure is inspired by existing techniques (namely discriminative domain adaptation), which have been successfully used in unsupervised domain adaptation applications. The principle behind our methodology is to align covariate distributions in source and target domains using a target supervision signal to help avoid poor local minima. Our procedure allows us to build high utility systems in many applications, such as monitoring ADLs, by introducing an offline overhead in computation.

Among our plans for future are the following: a) investigation on methods for creating the signal image, possibly with the use of other types of sensor measurements such as wearable accelerometers, gyroscopes and so on; b) investigation on image processing methods for transforming the signal image to the activity image; c) exploitation of other types of visual modalities in the process, such as RGB and depth data; d) evaluation of the proposed approach on several other public datasets; e) application into a real-like or even real-life assistive living environment; and f) extend our approach to open set domain adaptation for applications where the target dataset contains previously unseen (in the source domain) classes.

7 Conflict of interest

The authors declare that they have no conflict of interest.

Acknowledgements This project has received funding from the Hellenic Foundation for Research and Innovation (HFRI) and the General Secretariat for Research and Technology (GSRT), under grant agreement No 273 (Funding Decision:ΓΓΕΤ122785/I2/19-07-2018). We also acknowledge support of this work by the project SYNTELESIS "Innovative Technologies and Applications based on the Internet of Things (IoT) and the Cloud Computing" (MIS 5002521) which is implemented under the "Action for the Strategic Development on the Research and Technological Sector", funded by the Operational Programme "Competitiveness, Entrepreneurship and Innovation" (NSRF 2014-2020) and co-financed by Greece and the European Union (European Regional Development Fund).

References

1. Abadi M. et al. (2016) TensorFlow: A system for Large-Scale Maching Learning. In Proc. of the USENIX Symposium on Operating Systems Design and Implementation (OSDI).
2. Aggarwal, Jake K. "Human activity recognition-A grand challenge." In Digital Image Computing: Techniques and Applications (DICTA'05), pp. 1-1. IEEE, 2005.
3. Berretti, Stefano, Mohamed Daoudi, Pavan Turaga, and Anup Basu. "Representation, analysis, and recognition of 3D humans: A survey." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 14, no. 1s (2018): 16.
4. Chollet, F. (2015) *Keras*, `https://github.com/fchollet/keras`.
5. Ding, Jun, Bo Chen, Hongwei Liu, and Mengyuan Huang. "Convolutional neural network with data augmentation for SAR target recognition." IEEE Geoscience and remote sensing letters 13, no. 3 (2016): 364-368.
6. Du, Yong, Yun Fu, and Liang Wang. "Skeleton based action recognition with convolutional neural network." In 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR), pp. 579-583. IEEE, 2015.
7. Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In 2013 IEEE international conference on acoustics, speech and signal processing (pp. 6645-6649). IEEE.
8. Hou, Yonghong, Zhaoyang Li, Pichao Wang, and Wanqing Li. "Skeleton optical spectra-based action recognition using convolutional neural networks." IEEE Transactions on Circuits and Systems for Video Technology 28, no. 3 (2016): 807-811.
9. Jiang, Wenchao, and Zhaozheng Yin. "Human activity recognition using wearable sensors by deep convolutional neural networks." In Proceedings of the 23rd ACM international conference on Multimedia, pp. 1307-1310. Acm, 2015.
10. Ke, Qiuhong, Senjian An, Mohammed Bennamoun, Ferdous Sohel, and Farid Boussaid. "Skeletonnet: Mining deep part features for 3-d action recognition." IEEE signal processing letters 24, no. 6 (2017): 731-735.
11. Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." In Advances in neural information processing systems, pp. 1097-1105. 2012.
12. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., & Serre, T. (2011). HMDB: a large video database for human motion recognition. In 2011 International Conference on Computer Vision (pp. 2556-2563). IEEE.
13. Laptev, I., Marszalek, M., Schmid, C., & Rozenfeld, B. (2008). Learning realistic human actions from movies. In 2008 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-8). IEEE.
14. Lawton, M. P., & Brody, E. M. (1969). Assessment of older people: self-maintaining and instrumental activities of daily living. The gerontologist, 9(3 Part 1), 179-186.
15. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

16. Li, Chuankun, Yonghong Hou, Pichao Wang, and Wanqing Li. "Joint distance maps based action recognition with convolutional neural networks." IEEE Signal Processing Letters 24, no. 5 (2017): 624-628.
17. Li, Bo, Yuchao Dai, Xuelian Cheng, Huahui Chen, Yi Lin, and Mingyi He. "Skeleton based action recognition using translation-scale invariant image mapping and multi-scale deep CNN." In 2017 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), pp. 601-604. IEEE, 2017.
18. Liu, Chunhui, Yueyu Hu, Yanghao Li, Sijie Song, and Jiaying Liu. "PKU-MMD: A large scale benchmark for continuous multi-modal human action understanding." arXiv preprint arXiv:1703.07475 (2017).
19. Liu, Mengyuan, Hong Liu, and Chen Chen. "Enhanced skeleton visualization for view invariant human action recognition." Pattern Recognition 68 (2017): 346-362.
20. Liu, Jun, Amir Shahroudy, Mauricio Lisboa Perez, Gang Wang, Ling-Yu Duan, and Alex Kot Chichung. "NTU RGB+ D 120: A Large-Scale Benchmark for 3D Human Activity Understanding." IEEE transactions on pattern analysis and machine intelligence (2019).
21. A. Papadakis, E. Mathe, I. Vernikos, A. Maniatis, E. Spyrou and Ph. Mylonas, Recognizing Human Actions using 3D Skeletal Information and CNNs. In Proc. of Int'l Conf. on Engineering Applications of Neural Networks (EANN), 2019
22. A. Papadakis, E. Mathe, E. Spyrou and Ph. Mylonas. A Geometric Approach for Cross-View Human Action Recognition using Deep Learning. In Proc. of Int'l Symposium on Image and Signal Processing and Analysis (ISPA), 2019
23. Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. IEEE Transactions on knowledge and data engineering, 22(10), 1345-1359.
24. Schuldt, Christian, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach." In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., vol. 3, pp. 32-36. IEEE, 2004.
25. Shahroudy, Amir, Jun Liu, Tian-Tsong Ng, and Gang Wang. "Ntu rgb+ d: A large scale dataset for 3d human activity analysis." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1010-1019. 2016.
26. Shotton, Jamie, Andrew Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. "Real-time human pose recognition in parts from single depth images." In CVPR 2011, pp. 1297-1304. Ieee, 2011.
27. Soomro, Khurram, Amir Roshan Zamir, and Mubarak Shah. "UCF101: A dataset of 101 human actions classes from videos in the wild." arXiv preprint arXiv:1212.0402 (2012).
28. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.
29. Van Dyk, David A., and Xiao-Li Meng. "The art of data augmentation." Journal of Computational and Graphical Statistics 10, no. 1 (2001): 1-50.
30. Wang, Pichao, Wanqing Li, Chuankun Li, and Yonghong Hou. "Action recognition based on joint trajectory maps with convolutional neural networks." Knowledge-Based Systems 158 (2018): 43-53.
31. Wang, Pichao, Wanqing Li, Philip Ogunbona, Jun Wan, and Sergio Escalera. "RGB-D-based human motion recognition with deep learning: A survey." Computer Vision and Image Understanding 171 (2018): 118-139.
32. S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. Machine learning, 79(1-2):151–175, 2010.
33. Csurka, Gabriela. “A Comprehensive Survey on Domain Adaptation for Visual Applications.” Domain Adaptation in Computer Vision Applications (2017).
34. Wang, Mei and Weihong Deng. “Deep visual domain adaptation: A survey.” Neurocomputing 312 (2018): 135-153.
35. Tzeng, Eric et al. “Adversarial Discriminative Domain Adaptation.” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 2962-2971.
36. Ajakan, Hana et al. “Domain-Adversarial Neural Networks.” ArXiv abs/1412.4446 (2014): n. pag.
37. Goodfellow, Ian J., Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville and Yoshua Bengio. “Generative Adversarial Nets.” NIPS (2014).

38. , Xu, Tiantian, et al. "Dual many-to-one-encoder-based transfer learning for cross-dataset human action recognition." Image and Vision Computing 55 (2016): 127-137.
39. , S. Sadanand, J.J. Corso, Action bank: a high-level representation of activity in video, Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (May) (2012) 1234–1241. ISSN 10636919. http://dx.doi.org/10.1109/CVPR. 2012.6247806.
40. Tas, Yusuf, and Piotr Koniusz. "Cnn-based action recognition and supervised domain adaptation on 3d body skeletons via kernel feature maps." arXiv preprint arXiv:1806.09078 (2018).
41. Piotr Koniusz, Yusuf Tas, and Fatih Porikli. Domain adaptation by mixture of alignments of second- or higher-order scatter tensors. CVPR, 2017.
42. Zhang, Jianguang, et al. "Semi-supervised image-to-video adaptation for video action recognition." IEEE transactions on cybernetics 47.4 (2016): 960-973.
43. Hachiya, Hirotaka, Masashi Sugiyama, and Naonori Ueda. "Importance-weighted least-squares probabilistic classifier for covariate shift adaptation with application to human activity recognition." Neurocomputing 80 (2012): 93-101.