2019

Human Action Recognition in Videos Using Transfer Learning

Kaiqiang Huang
Technological University Dublin

Sarah Jane Delany
Technological University Dublin, sarahjane.delany@tudublin.ie

Susan McKeever
Technological University Dublin, susan.mckeever@tudublin.ie

Human Action Recognition in Videos using Transfer Learning

Kaiqiang Huang, Sarah Jane Delany, Susan Mckeever

Technological University Dublin

Abstract

A variety of systems focus on detecting the actions and activities performed by humans, such as video surveillance and health monitoring systems. However, published labelled human action datasets for training supervised machine learning models are limited in number and expensive to produce. The use of transfer learning for the task of action recognition can help to address this issue by transferring or re-using the knowledge of existing trained models, in combination with minimal training data from the new target domain. Our focus in this paper is an investigation of video feature representations and machine learning algorithms for transfer learning for the task of action recognition in videos in a multi-class environment. Using four labelled datasets from the human action domain, we apply two SVM-based transfer-learning algorithms: adaptive support vector machine (A-SVM) and projective model transfer SVM (PMT-SVM). For feature representations, we compare the performance of two widely used video feature representations: space-time interest points (STIP) with Histograms of Oriented Gradients (HOG) and Histograms of Optical Flow (HOF), and improved dense trajectory (iDT) to explore which feature is more suitable for action recognition from videos using transfer learning. Our results show that A-SVM and PMT-SVM can help transfer action knowledge across multiple datasets with limited labelled training data; A-SVM outperforms PMT-SVM when the target dataset is derived from realistic non-lab environments; iDT has a greater ability to perform transfer learning in action recognition.

Keywords: Video Representation, Action Recognition, Transfer Learning

1 Introduction

Human action recognition refers to the research task of identifying an action performed by human captured in a video. The area has drawn substantial attention in the computer vision research community, with application in everyday scenarios, such as video surveillance [Wang, 2013], human robot interaction [Lemaignan et al., 2017] and video retrieval [Rossetto et al., 2015]. Various research efforts in human action recognition have been completed in the past decade [Laptev, 2005, Dollár et al., 2005, Willems et al., 2008, Klaser et al., 2008, Laptev et al., 2008, Wang et al., 2011, Wang and Schmid, 2013], with supervised machine learning as a common approach to developing classification models. However, as demand for human action recognition in real application systems increases, the volume of available labelled action training data is a challenge when attempting to develop generalised and robust models. It is difficult for instance to collect well-defined and labelled video datasets to support intelligent video surveillance applications without substantial manual efforts to create relevant training video snippets of specialised human action scenarios. Therefore, these traditional approaches of recognising actions via focused labelled training datasets do not scale well as the requirements for action recognition expand. To address the issue of acquiring labelled datasets, we focus on the application of transfer learning to the task of human action recognition in videos. Transfer learning is based on the premise of applying knowledge gained for one problem to a different but related problem. For our purpose, the learning embedded in an existing trained action recognition model (termed "source" model) is transferred or re-used in a new "target" domain that has overlapping learning needs, such as the need to recognise the same human action tasks but in different physical environments. Thus, the need to start from scratch with a new training dataset for each new target classification model may be eliminated. Some previous works have examined the use of

transfer learning in computer vision tasks, including object detection [Aytar and Zisserman, 2011], video concept detection [Yang et al., 2007, Duan et al., 2009], and action detection and recognition [Cao et al., 2010, Liu et al., 2011, Bian et al., 2012]. In general, these research works focus on representing image and video instances using local features, and encoding them by bag of visual words (BoVWs). For video representation, two commonly used methods of representing video features are space-time interest points (STIP) with Histograms of Oriented Gradients (HOG) and Histograms of Optical Flow (HOF) [Laptev, 2005, Laptev et al., 2008] and improved dense trajectory (iDT) [Wang and Schmid, 2013]. In this paper, we investigate three specific research questions: 1) Which video feature representation (STIP+HOG/HOF and iDT) perform best for the transfer of knowledge? 2) Can A-SVM and PMT-SVM outperform to non-transfer SVM for transfer learning? 3) What quantity of training examples are needed in the target domain to achieve optimal action recognition results? We address these questions using four labelled video datasets from the human action domain. A further contribution is testing of the above in a multi-class environment, rather than binary class as per previous work [Yang et al., 2007, Aytar and Zisserman, 2011].

The rest part of this paper is structured as following. Related work is described in Section 2. In Section 3, our approach is explained in detail, including the dataset description, video representations, transfer learning algorithms and evaluation metrics. In Section 4, the designed experiments and the relevant results are demonstrated and explained. Finally, the conclusion and future work are drawn in Section 5.

2 Related Work

A key strand of related work is video feature representations for machine learning. Local video representations have been successfully applied to for human action recognition applications [Laptev et al., 2004, Laptev et al., 2008]. There are typically three steps to extract representations - interest points detection, descriptors extraction and aggregation. Interest points denote points in an image or video which are likely to give strong signal information, such as a corner of junction. Interest point detectors select space-time locations and scales from videos and distinguish the sparseness of the interest points. Then, the local descriptors encode appearances and motions around the selected interest points based on the measurement of space-time gradients and optical flow. For interest points detection, space-time interest point (STIP) was proposed by [Laptev, 2005], which is an extension of Harris corner detection [Harris and Stephens, 1988] by finding out significant variations in both spatial and temporal domains. Another commonly used detector is the Hessian 3D detector, which is also extended from 2D counterparts [Willems et al., 2008]. In relation to STIP and Hessian3D detectors, the cuboid detector was proposed to overcome the limitation of STIP that insufficient interest points may be detected to represent actions [Dollár et al., 2005]. To obtain the local descriptors from detected interest points, HOF was presented to encode pixels-level motions from optical flow fields through local patches as descriptors [Laptev et al., 2008]. HOG3D was proposed to describe motion features in human actions [Klaser et al., 2008], which is an extension of HOG used in human detection from images [Dalal and Triggs, 2005]. Another robust descriptor is motion boundary histogram (MBH) that compute gradients over optical flow fields. Notably, trajectory-based methods such as iDT track interest points to form trajectory-based 3D volumes, computing HOG/HOF and MBH as descriptors for each trajectory [Wang and Schmid, 2013]. In some supervised machine learning classifiers such as SVM, the size of the input vector should be fixed-length vectors. However, the number of local descriptors varies for each video. Therefore, BoVWs are used to aggregate the sets of local descriptors into fixed length vectors, which can be fed into supervised classifiers [Laptev et al., 2004, Dollár et al., 2005, Laptev et al., 2008, Klaser et al., 2008]. Generally, in BoVWs, each video is represented as a histogram of the frequency of the visual words appearing as the closest match to the local features in the codebook.

In traditional transfer learning approaches, a discriminative transfer learning method based on least squares SVM (LS-SVM) was proposed by [Tommasi et al., 2010], to learn object categories from few samples. Domain transfer SVM (DT-SVM) was introduced by [Duan et al., 2009] to minimise the data distribution mismatch between the target and source domain using the maximum mean discrepancy (MMD) criterion that measures the distribution similarity of two domains. Additionally, the transfer learning adaptive boosting (TrAdaBoost) is introduced by [Dai Wenyuan et al., 2007], that is the extension of adaptive boosting (AdaBoost) [Freund

Datasets		#Classes	#Clips	Derived from	Average Length(s)	SD.(s)
Weizmann [Blank et al., 2005]	D1	10	93	Lab	2.45	0.87
KTH [Laptev et al., 2004]	D2	6	600	Lab	19.33	5.92
HMDB51 [Kuehne et al., 2011]	D3	51	6766	Movies, YouTube, Web	3.17	2.27
UCF101 [Soomro et al., 2012]	D4	101	13320	YouTube	7.21	3.76

Table 1: The summary of action datasets

and Schapire, 1997], to reduce the weighted training error on the data with different distribution, and in the meantime preserving the properties of AdaBoost. Furthermore, the A-SVM transfer learning variant of SVM was originally introduced by [Yang et al., 2007]. It learns from the source model by regularising the distance between the learned model and source model, used in video concept detection application. The work of [Aytar and Zisserman, 2011] then focused on object detection, referencing the use of A-SVM, and proposing PMT-SVM, which can increase the amount of transfer without penalising margin maximisation. In this work, we will first examine two video representations that are STIP+HOG/HOF and iDT, used in conjunction with two classifier-based transfer learning approaches that are A-SVM and PMT-SVM. Both transfer learning approaches are presented for binary classification tasks only. We will compare their performance for the tasks of distinguishing multiple classes of human actions.

3 Our Approach

The focus of our work in this paper is to compare several video representations and SVM-variant transfer learning algorithms for the tasks of recognising human actions in videos for multiple classes. We use a supervised learning approach to test transfer learning, using a variety of labelled human action datasets. Our approach is as follows: firstly, we train a source model with a full labelled human action dataset. We then test this model with another (target) human action dataset with the same classes but different video characteristics. This determines the extent to which knowledge from the source model can be transferred to the target task and forms our baseline. We then enhance this baseline source model by retraining with the inclusion of target dataset instances, so see what proportion of target dataset instances are needed to achieve improved recognition accuracy. Next, we explain how we carry out this approach to compare multiple datasets, two feature representations and multiple classifiers.

3.1 Datasets

We collated four public labelled human action datasets in order to conduct experiments. Each dataset contains videos, where each video contains one human action example (such as jumping), with one label per video. The summary of the four datasets is described in Table 1, showing their original research publishers, video source (lab versus social media) and video duration. Note that **D1** and **D2** were collected in controlled lab environments and settings with static backgrounds and less camera motion. In contrast, **D3** and **D4** were collected from realistic video sources and movies with dynamic background and camera motion. Due to the small number of samples in dataset **D1**, we will only use this dataset as a source domain, never as target domain which requires higher numbers of video instances for training and testing splits. Actions in videos used different terminology across the datasets. When identifying the overlapping actions across the datasets, we relabelled the same actions to have the same label value for all datasets, shown in Table 2. For example, the class of clapping was originally called "hand clapping" in **D2**, and the class of biking was originally called "ride" in **D3**. For **D4**, we combined the classes of boxingPunchingBag and boxingSpeedBag into the single class of boxing.

3.2 Video Representations

In order to train a human action recognition model, we need to select suitable feature representations for the input training and test video samples. We select two feature representations for comparison: STIP+HOG/HOF and iDT, both encoding with the standard BoVWs approach [Laptev et al., 2004]. In STIP, rather than performing scale selection as in [Laptev, 2005], we simply extract features at multiple levels of spatio-temporal scales that are spatial scales σ and temporal scales τ in order to reduce computational cost [Laptev et al., 2008]. In

Task	Source	#Clips (class %)		Target	#Clips (class %)		Common Classes (#CC)
		STIP+HOG/HOF	iDT		STIP+HOG/HOF	iDT	
1	D1	38 (26% 26% 48%)		D2	300 (33.3% 33.3% 33.3%)		Running,Walking,Waving (3)
2	D1	65 (42% 15% 15% 28%)		D3	400 (25% 25% 25% 25%)		Jumping,Running,Walking,Waving (4)
3	D1	37 (73% 27%)		D4	246 (50% 50%)		Jumping,Walking (2)
4	D2	400 (25% 25% 25% 25%)		D3	400 (25% 25% 25% 25%)		Clapping,Running,Walking,Waving (4)
5	D3	400 (25% 25% 25% 25%)		D2	400 (25% 25% 25% 25%)		
6	D2	200 (50% 50%)		D4	420 (71% 29%)		Boxing,Walking (2)
7	D4	420 (71% 29%)		D2	200 (50% 50%)		
8	D3	700 (14.3% per class)	688 (PushUps:13% Others:14.5%)	D4	931 (14% 16% 15% 13% 17% 11% 14%)		Biking,Diving GolfSwing,Jumping Punch,PushUps,Walking (7)
9	D4	931 (14% 16% 15% 13% 17% 11% 14%)		D3	700 (14.3% per class)	688 (PushUps:13% Others:14.5%)	

Table 2: Transfer learning tasks for action recognition, showing source and target datasets, including number of video instances and class breakdown

addition, HOG and HOF are generated on a 3D patch in the neighbourhood of each detected STIP. The size of each 3D patch is related to the corresponding scales, as the spatial and temporal path size refer to $2k_s\sigma$ and $2k_t\tau$. Each patch is divided into a grid with $n_x * n_y * n_t$ spatio-temporal blocks, and 4-bin HOG and 5-bin HOF are computed from all blocks as descriptors. Next, in iDT, dense trajectories are extracted from multiple spatial scales in videos, and the trajectory-aligned descriptors are computed to describe the corresponding trajectory. For example, an object, such as hand, is firstly identified by few interest points, and then the movement of object is tracked to form a trajectory based on optical flow field, such as hand clapping. Unlike the sparse sampling used in STIP, dense sampling is applied into iDT in order to involve every single pixel in videos, where can consider much more information from videos. In addition to computing HOG and HOF descriptors, iDT employs MBH which encodes relative motions between pixels by computing the derivatives of horizontal and vertical directions of the optical flows. To the end, each video is represented by the sets of vectors.

3.3 Action Recognition with Transfer Learning

For our classifiers, we choose SVM as this is commonly used in video recognition and has been the focus of transfer learning variations [Yang et al., 2007, Aytar and Zisserman, 2011]. We use the basic (non-transfer) SVM classifier as the baseline approach. We use two transfer learning variations of the SVM algorithm, A-SVM and PMT-SVM [Yang et al., 2007, Aytar and Zisserman, 2011], in order to explore which will perform best for our work. A-SVM and PMT-SVM perform transfer learning in action recognition via improving the process of transferring knowledge from source to target by minimising the distance between source model w^s and the learned model w. In addition, unlike transfer by minimising the difference between w^s and w in A-SVM, PMT-SVM attempts to minimise the projection of target model w learned from target domain onto the separating hyperplane orthogonal to source model w^s, which points out the amount of transfer can be increased without reducing margin maximisation. Full details of the implementation can be found online [1].

According to the originators [Yang et al., 2007, Aytar and Zisserman, 2011], both A-SVM and PMT-SVM were proposed for the task of binary classification, as is associated with SVM. Our task requires multi-class classification to distinguish between multiple possible actions as shown in Table 2. We extend our setup to multi-classification using a one-vs-all (OVA) strategy. OVA involves training a single classifier for each of the action classes, with the samples of a class as positive samples and all other samples as negatives. For classification, the test video sample is input to all of the trained classifiers, and the final label is derived from the corresponding classifier yielding the highest confidence score for a positive label.

3.4 Evaluation Metrics

Selecting an appropriate metric to evaluate classifiers depends upon the purpose of the research task. The metrics of mean average precision is widely used for evaluation of detection-based tasks, such as video concept detection and image object detection [Yang et al., 2007, Duan et al., 2009, Marszałek et al., 2009, Aytar and Zisserman, 2011]. However, in our work, we wish to measure the extent or level of accuracy with which a trained model has transferred knowledge from a source domain to target domain for the same set of classes.

[1] https://www.robots.ox.ac.uk/~vgg/software/tabularasa/

Therefore, we suggest recall as a suitable metric to evaluate the classifiers in our experiments. Recall is the number of correct predictions of class, divided by the number of results that have been predicted of class. We first calculate recall per class, then calculate average recall over all classes.

4 Experiments & Results

In this part, the aforementioned SVM-based algorithms and feature representations are evaluated in a variety of source-target configurations of our datasets for transfer learning. As part of this, the quantity of target training data needed to optimise the source model is reported. All experiments are implemented in MATLAB.

4.1 Experimental Setup

We conduct a variety of transfer learning experiments on four human action datasets, as shown by the dataset pairing configurations in Table 2. As shown, nine configurations are defined for performing action recognition with transfer learning using a pairing of source and target datasets. In addition, we conduct each of defined transfer learning tasks by five times to obtain the reasonable results. We extract the features of STIP+HOG/HOF and iDT for all videos. For each target dataset, we reserve a consistent balanced 30% block of the video instances for testing, with the remaining 70% available as training.

Due to the limited number of videos in dataset **D1**, it is not used as a target dataset, as target datasets require training/test splits. Table 1 describes the details of each dataset, such as the selected categories, features, and the number of training and testing examples. In addition, dataset **D4** with iDT feature representation is not involved in experiments because of the expensive computation cost and huge data storage requirement. Dataset **D4** was collected from realistic environments, so it could contain much redundant information in videos, such as clutter background and other objects, which can highly increase the feature size. Also, the number of clips and average duration are nearly more than another real-environment dataset **D3** by one third and twice, respectively (see in Table 1 and Table 2). Consequently, we have four configurations that include iDT (Task 1, 2, 4 and 5). For each experiment, our baseline is the source model trained solely on the source dataset, tested against the reserved test proportion of the target dataset. Non-transfer SVM, A-SVM and PMT-SVM models are then trained with increasing proportions of training samples from the target domain, and tested on the same test set of the target dataset. Note that for non-transfer SVM, re-training is performed by adding combining the videos from the source dataset with the relevant proportion of training video from target domain, and retraining the model from scratch. In contrast, for the transfer methods used in A-SVM and PMT-SVM, the source model is enhanced with the target training samples without full retraining.

Parameters: STIP+HOG/HOF was implemented using published framework code[2]. We apply the default parameters, which are spatial scales $\sigma^2 = 4, 8, 16, 32, 64, 128, 256, 512$,temporal $\tau^2 = 2, 4$, the spatial and temporal patch size factor $k_s = 9, k_t = 4$, and $n_x, n_y = 3$ and $n_t = 2$ as each grid size of blocks. Based on the detected STIPs, HOG and HOF are computed as 72-dimension and 90-dimension vectors, respectively, resulting in a 162-dimension vector to represent an interest point in the video with concatenation.

Also, iDT is implemented using published source codes [3]. We select the default parameters, which are spatial cells $n_\sigma = 2$, temporal cells $n_\tau = 3$, trajectory length $L = 15$, stride for dense sampling feature points $W = 5$ and the neighbourhood size $N = 32$ for computing descriptors. Therefore, 8-bin HOG and 9-bin HOF are obtained in $2 * 2 * 3$ spatial-temporal blocks, resulting in 96-dimension vector and 108-dimension vector, respectively. Then, 192-dimension vector is computed as MBH and 30-dimension vector is computed to describe trajectory shape according to its length. In total, a 426-dimension vector is generated to represent a tracked trajectory in the video.

[2] https://www.di.ens.fr/~laptev/download/stip-2.0-linux.zip

[3] http://lear.inrialpes.fr/~wang/improved_trajectories

Percentage (training samples in target)			0%	5%	10%	15%	20%	25%	30%	35%	40%
Task (Source - Target)	Feature	Classifier									
1 (D1 - D2) (Lab - Lab)	STIP+HOG/HOF	Non-SVM	0.31	0.4	0.46	0.58	0.62	0.73	0.79	0.85	0.86
		A-SVM		0.5	0.64	0.79	0.9	0.94	0.96	0.98	0.96
		PMT-SVM		0.51	0.66	0.8	0.91	0.94	0.96	0.98	0.96
	iDT	Non-SVM	0.5	0.9	0.92	0.93	0.94	0.95	0.96	0.97	0.97
		A-SVM		0.93	0.97	0.97	0.98	0.98	0.99	0.98	0.99
		PMT-SVM		0.94	0.97	0.97	0.98	0.98	0.99	0.98	0.99
2 (D1 - D3) (Lab - Real)	STIP+HOG/HOF	Non-SVM	0.28	0.31	0.33	0.35	0.34	0.35	0.39	0.4	0.39
		A-SVM		0.32	0.34	0.37	0.39	0.35	0.37	0.42	0.42
		PMT-SVM		0.35	0.35	0.36	0.4	0.37	0.39	0.41	0.41
	iDT	Non-SVM	0.2	0.37	0.43	0.44	0.45	0.44	0.5	0.53	0.54
		A-SVM		0.34	0.39	0.44	0.46	0.46	0.5	0.55	0.54
		PMT-SVM		0.33	0.39	0.45	0.45	0.47	0.5	0.54	0.54
3 (D1 - D4) (Lab - Real)	STIP+HOG/HOF	Non-SVM	0.39	0.65	0.78	0.85	0.83	0.86	0.84	0.84	0.86
		A-SVM		0.64	0.67	0.74	0.76	0.79	0.81	0.78	0.82
		PMT-SVM		0.65	0.67	0.73	0.76	0.78	0.8	0.78	0.82
4 (D2 - D3) (Lab-Real)	STIP+HOG/HOF	Non-SVM	0.18	0.34	0.34	0.45	0.45	0.45	0.46	0.51	0.5
		A-SVM		0.36	0.35	0.42	0.42	0.47	0.45	0.49	0.5
		PMT-SVM		0.36	0.37	0.43	0.42	0.43	0.44	0.49	0.49
	iDT	Non-SVM	0.24	0.34	0.42	0.44	0.48	0.5	0.51	0.51	0.54
		A-SVM		0.32	0.4	0.38	0.42	0.48	0.48	0.5	0.51
		PMT-SVM		0.28	0.38	0.39	0.44	0.48	0.48	0.5	0.49
5 (D3 - D2) (Real - Lab)	STIP+HOG/HOF	Non-SVM	0.21	0.44	0.57	0.66	0.76	0.79	0.84	0.84	0.88
		A-SVM		0.41	0.67	0.75	0.86	0.87	0.94	0.94	0.93
		PMT-SVM		0.45	0.67	0.76	0.85	0.88	0.94	0.93	0.93
	iDT	Non-SVM	0.28	0.84	0.94	0.96	0.97	0.98	0.97	0.98	0.98
		A-SVM		0.89	0.95	0.97	0.98	0.99	0.98	0.99	0.99
		PMT-SVM		0.9	0.95	0.97	0.98	0.99	0.99	0.99	1
6 (D2 - D4) (Lab - Real)	STIP+HOG/HOF	Non-SVM	0.43	0.65	0.73	0.76	0.81	0.79	0.82	0.82	0.87
		A-SVM		0.6	0.72	0.77	0.78	0.77	0.8	0.81	0.85
		PMT-SVM		0.59	0.72	0.78	0.77	0.78	0.8	0.81	0.86
7 (D4 - D2) (Real - Lab)	STIP+HOG/HOF	Non-SVM	0.37	0.69	0.85	0.95	0.92	0.93	0.95	0.98	0.97
		A-SVM		0.66	0.83	0.95	0.92	0.95	0.96	0.99	0.98
		PMT-SVM		0.67	0.84	0.95	0.92	0.96	0.97	0.99	0.98
8 (D3 - D4) (Real - Real)	STIP+HOG/HOF	Non-SVM	0.09	0.32	0.38	0.47	0.47	0.48	0.52	0.56	0.57
		A-SVM		0.35	0.42	0.52	0.51	0.56	0.59	0.62	0.63
		PMT-SVM		0.37	0.42	0.52	0.51	0.57	0.59	0.61	0.63
9 (D4 - D3) (Real - Real)	STIP+HOG/HOF	Non-SVM	0.18	0.34	0.4	0.46	0.49	0.53	0.58	0.58	0.58
		A-SVM		0.36	0.46	0.51	0.54	0.57	0.59	0.6	0.6
		PMT-SVM		0.39	0.5	0.56	0.56	0.59	0.62	0.63	0.63

Table 3: Average Recalls with different classifiers and video representations for all tasks

To evaluate the model performance, we use a standard BoVWs approach to represent each video instance, aggregating all extracted features from videos into a fixed-length vector to be fed into classifiers. At first, we set the number of visual words to 4000, which is an empirically promising result for different action datasets [Wang et al., 2009]. Then, local space-time features are quantised into a codebook by clustering, and a video is represented as the fixed-length frequency histogram according to this codebook [Wang et al., 2011]. To reduce the computation cost, as similarly done by [Wang et al., 2011], we cluster a subset of 10% randomly selected training features using k-means in order to generate the codebook. We then assign each remaining feature to the closest match based on this codebook. In the training process, we set the same hyper-parameters to all classifiers based on this work [Aytar and Zisserman, 2011].

4.2 Results

We present the results as per the focus of our investigation: a comparison between non-transfer SVM and SVM-based transfer learning algorithms, and secondly, a comparison of the two video feature representation approaches.

Table 3 shows the average recall for baseline and transfer learning approaches, showing the impact on classification accuracy of increasing amounts of training samples from target domain. We denote on the table whether the video datasets are captured from lab or real-life environments, to help determine if this has any bearing on our results. Note that 0% refers to no target training data included, with classifiers trained from the source data only, and tested on the consistent target testing set. We are interested in minimising the need for target data, so we are particularly interested in the impact of small amounts of training samples. We con-

sider up to 40% training samples to be used from the target domain. In Task 1 and 5 (target dataset is from lab environments), iDT performs substantially better than STIP+HOG/HOF at level of less than 20% target training samples involved. In addition, in Task 2 and 4 (target dataset is from a non-lab environments), our two video representations support similar accuracies. We conclude that in our experiments, iDT is a better feature representation for action recognition with transfer learning when the target dataset is collected from a lab environment, with clean backgrounds and less camera motion. However, it is still a challenging task to perform transfer learning for videos captured in realistic environments with much noisy information. As dense sampling is used in iDT, we suggest that other video representations using dense sampling may improve transfer learning in action recognition as well, but this requires further investigation.

Looking at the classifiers, A-SVM and PMT-SVM perform slightly better than non-transfer SVM in Task 1 and 5, but they show no obvious advantage over non transfer SVM when the target dataset is from a real environment (Task 3, 4, 6). A-SVM performs better than PMT-SVM for such realistic target datasets. This can be explained by PMT-SVM being relatively sensitive to bad training samples, such as more actors and frequent camera motions [Aytar and Zisserman, 2011]. In a nutshell, SVM-based transfer learning algorithms assist in transferring knowledge for human action detection when the target dataset is from lab environments. However, the level of improvement is not substantial or in some cases worse than baseline SVM when the target datasets are from real-life non-lab environment. In the configuration of read-to-real (Task 8 and 9), as baseline (0%), the results are very poor as there is much other information, not just contain action instances when the datasets are derived from realistic environment. However, in this specific configuration, by adding more target training samples, the results gradually increased, and A-SVM and PMT-SVM moderately outperform over non-transfer SVM. Based on this observation, we suggest that SVM-based transfer learning classifiers can provide benefit with transfer classifier-based knowledge when both domains are derived from realistic environments.

5 Conclusion

In this paper, we have conducted an empirical comparison of a variety of SVM based classifiers and video feature representations for the task of human action recognition in videos using transfer learning, in a multi-class setting. We have used multiple datasets in an effort to produce more generalisable results, and to observe the impact of video environments on transfer learning. In our comparison, we used non-transfer SVM and SVM-based transfer learning algorithms (A-SVM and PMT-SVM), and two commonly used video feature representation approaches (STIP+HOG/HOF and iDT). In the action recognition task, we observed that the video representations based on dense sampling (iDT) supported better model knowledge transfer than the sparse representations. In addition, we note that the ability of classifier-based transfer learning algorithms in assisting transfer classifier-based with small amounts of labelled training samples from the target domain was limited to lab-environment datasets as target. Transferring knowledge to real-environment datasets is still challenging when using limited target data.

As future work, we aim to use larger datasets on transfer learning as our datasets were limited in number of classes per video. Given the promising results using iDT, we will examine other video representations based on dense sampling to determine if there are variations on dense sampling that may give improved action recognition accuracies using transfer learning. Additionally, other classifier-based transfer learning algorithms also will be considered to study and compare, such as DT-SVM [Duan et al., 2009], LS-SVM [Tommasi et al., 2010] and TrAdaBoost [Dai Wenyuan et al., 2007].

Acknowledgments

This project is funded under the Fiosraigh Scholarship of Technological University Dublin.

References

[Aytar and Zisserman, 2011] Aytar, Y. and Zisserman, A. (2011). Tabula rasa: Model transfer for object category detection. In *ICCV*, pages 2252–2259. IEEE.

[Bian et al., 2012] Bian, W., Tao, D., and Rui, Y. (2012). Cross-domain human action recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 42(2):298–307.

[Blank et al., 2005] Blank, M., Gorelick, L., Shechtman, E., Irani, M., and Basri, R. (2005). Actions as space-time shapes. In *Proc IEEE Int Conf Comput Vis*, pages 1395–1402.

[Cao et al., 2010] Cao, L., Liu, Z., and Huang, T. S. (2010). Cross-dataset action detection. In *CVPR*, pages 1998–2005. IEEE.

[Dai Wenyuan et al., 2007] Dai Wenyuan, Y. Q., Guirong, X., et al. (2007). Boosting for transfer learning. In *Proceedings of the 24th International Conference on Machine Learning, Corvallis, USA*, pages 193–200.

[Dalal and Triggs, 2005] Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *CVPR*, volume 1, pages 886–893. IEEE Computer Society.

[Dollár et al., 2005] Dollár, P., Rabaud, V., Cottrell, G., and Belongie, S. (2005). Behavior recognition via sparse spatio-temporal features. VS-PETS Beijing, China.

[Duan et al., 2009] Duan, L., Tsang, I. W., Xu, D., and Maybank, S. J. (2009). Domain transfer svm for video concept detection. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1375–1381. IEEE.

[Freund and Schapire, 1997] Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Alvey vision conference*, volume 15, pages 10–5244. Citeseer.

[Klaser et al., 2008] Klaser, A., Marszałek, M., and Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. BMVA.

[Kuehne et al., 2011] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. (2011). Hmdb: a large video database for human motion recognition. In *Proc IEEE Int Conf Comput Vis*, pages 2556–2563. IEEE.

[Laptev, 2005] Laptev, I. (2005). On space-time interest points. *IJCV*, 64(2-3):107–123.

[Laptev et al., 2004] Laptev, I., Caputo, B., et al. (2004). Recognizing human actions: a local svm approach. pages 32–36. IEEE.

[Laptev et al., 2008] Laptev, I., Marszałek, M., Schmid, C., and Rozenfeld, B. (2008). Learning realistic human actions from movies. In *CVPR*, pages 1–8. IEEE Computer Society.

[Lemaignan et al., 2017] Lemaignan, S., Warnier, M., Sisbot, E. A., Clodic, A., and Alami, R. (2017). Artificial cognition for social human–robot interaction: An implementation. *Artificial Intelligence*, 247:45–69.

[Liu et al., 2011] Liu, J., Shah, M., Kuipers, B., and Savarese, S. (2011). Cross-view action recognition via view knowledge transfer. In *CVPR 2011*, pages 3209–3216. IEEE.

[Marszałek et al., 2009] Marszałek, M., Laptev, I., and Schmid, C. (2009). Actions in context. In *CVPR*, pages 2929–2936. IEEE Computer Society.

[Rossetto et al., 2015] Rossetto, L., Giangreco, I., Schuldt, H., Dupont, S., Seddati, O., Sezgin, M., and Sahillioğlu, Y. (2015). Imotion-a content-based video retrieval engine. In *MMM*, pages 255–260. Springer.

[Soomro et al., 2012] Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

[Tommasi et al., 2010] Tommasi, T., Orabona, F., and Caputo, B. (2010). Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3081–3088. IEEE.

[Wang et al., 2011] Wang, H., Kläser, A., Schmid, C., and Cheng-Lin, L. (2011). Action recognition by dense trajectories. In *CVPR 2011-IEEE Conference on Computer Vision & Pattern Recognition*, pages 3169–3176. IEEE.

[Wang and Schmid, 2013] Wang, H. and Schmid, C. (2013). Action recognition with improved trajectories. In *Proc ICCV*, pages 3551–3558.

[Wang et al., 2009] Wang, H., Ullah, M. M., Klaser, A., Laptev, I., and Schmid, C. (2009). Evaluation of local spatio-temporal features for action recognition. In *BMVC 2009-British Machine Vision Conference*, pages 124–1. BMVA.

[Wang, 2013] Wang, X. (2013). Intelligent multi-camera video surveillance: A review. *Pattern recognition letters*, 34(1):3–19.

[Willems et al., 2008] Willems, G., Tuytelaars, T., and Van Gool, L. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. In *European conference on computer vision*, pages 650–663. Springer.

[Yang et al., 2007] Yang, J., Yan, R., and Hauptmann, A. G. (2007). Cross-domain video concept detection using adaptive svms. In *Proceedings of the 15th ACM international conference on Multimedia*, pages 188–197. ACM.