Spatio-temporal Attention Mechanisms for Activity Recognition

Srijan Das

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Mécanismes d'Attention Spatio-temporels pour la Reconnaissance d'Activité

Srijan DAS

INRIA Sophia Antipolis, STARS

Présentée en vue de l'obtention du grade de docteur en Informatique d'Université Côte d'Azur
Dirigée par : François BRÉMOND
Co-encadrée par : Monique THONNAT
Soutenue le 1 Octobre 2020

Devant le jury, composé de :
Président du jury : Matthieu CORD, Professeur, Sorbonne Université
Examinateur : Michael RYOO, Professeur Associé, Stony Brook Université, Google Brain
Rapporteurs :
Christian WOLF, HDR, INSA Lyon
Dima DAMEN, Professeur Associé, Université de Bristol UK

informatiques mathématiques
Inria

Mécanismes d'Attention Spatio-temporels pour la Reconnaissance d'Activité

Spatio-temporal Attention Mechanisms for Activity Recognition

Jury :

Président du jury :

Matthieu CORD - Professeur, Sorbonne Université, France

Rapporteurs :

Christian WOLF - HDR, INSA Lyon, France

Dima DAMEN - Professeur Associé, Université de Bristol, UK

Examinateurs :

Michael RYOO - Professeur Associé, Stony Brook Université, Google Brain, USA

François BRÉMOND - DR, INRIA Sophia Antipolis, France

Monique THONNAT - DR, INRIA Sophia Antipolis, France

MÉCANISMES D'ATTENTION SPATIO-TEMPORELS POUR LA RECONNAISSANCE D'ACTIVITÉ

Srijan Das
Directeur de thèse: François Brémond
Co-Directeur de thèse: Monique Thonnat
STARS, Inria Sophia Antipolis, France

RÉSUMÉ

Cette thèse vise la reconnaissance d'actions humaines dans des vidéos. La reconnaissance d'actions est une tâche difficile en vision par ordinateur posant de nombreux défis complexes. Avec l'émergence de l'apprentissage en profondeur et de très grandes bases de données provenant d'Internet, des améliorations substantielles ont été apportées à la reconnaissance de vidéos. Par exemple, des réseaux de convolution 3D de pointe comme I3D pré-entrainés sur d'énormes bases de données comme Kinetics ont réussi à améliorer substantiellement la reconnaissance d'actions de vidéos Internet. Mais, ces réseaux à noyaux rigides appliqués sur l'ensemble du volume espace-temps ne peuvent pas relever les défis présentés par les activités de la vie quotidienne (ADL). Nous sommes plus particulièrement intéressés par la reconnaissance vidéo pour les activités de la vie quotidienne ou ADL. Outre les défis des vidéos génériques, les ADL présentent - (i) des actions à grain fin avec des mouvements courts et subtils comme verser du grain ou verser de l'eau, (ii) des actions avec des modèles visuels similaires différant par des modèles de mouvement comme se frotter les mains ou applaudir, et enfin (iii) de longues actions complexes comme faire la cuisine. Afin de relever ces défis, nous avons apporté trois contributions principales. La première contribution comprend une stratégie de fusion multimodale pour prendre en compte les avantages des modalités multiples pour classer les actions. Cependant, la question demeure: comment combiner plusieurs modalités de bout en bout? Comment pouvons-nous utiliser les informations 3D pour guider les réseaux RVB de pointe actuels pour la classification des actions? À cette fin, notre deuxième contribution est un mécanisme d'attention axé sur la pose pour la classification des actions. Nous proposons trois variantes de mécanismes d'attention spatio-temporelle exploitant les modalités de pose RVB et 3D pour relever les défis susmentionnés (i) et (ii) pour des actions courtes. Notre troisième contribution principale est un modèle temporel combinant représentation temporelle et mécanisme d'attention. La représentation vidéo conservant des informations temporelles denses permet au modèle temporel de modéliser de longues actions complexes, ce qui est crucial pour les ADL. Nous avons évalué notre première contribution sur trois petites bases de données publiques: CAD-60, CAD-120 et MSRDailyActivity3D. Nous avons évalué nos deuxième et troisième contributions sur quatre bases de données publiques: une très base de données d'activité humaine: NTU-RGB + D 120, son sous-ensemble NTU-RGB + D 60, une base de données d'activité humaine difficile du monde réel: Toyota Smarthome et une base de données d'interaction personne-objet de petite dimension Northwestern UCLA. Nos expériences montrent que les méthodes proposées dans cette thèse surpassent les résultats de pointe.

Mots clés: reconnaissance d'action, analyse spatio-temporelle, mécanisme d'attention, actions longues et complexes.

SPATIO-TEMPORAL ATTENTION MECHANISMS FOR ACTIVITY RECOGNITION

by
Srijan Das
Supervisor: François Brémond
Co-Supervisor: Monique Thonnat
STARS, Inria Sophia Antipolis, France

ABSTRACT

This thesis targets recognition of human actions in videos. Action recognition is a complicated task in the field of computer vision due to its high complex challenges. With the emergence of deep learning and large scale datasets from internet sources, substantial improvements have been made in video understanding. For instance, state-of-the-art 3D convolutional networks like I3D pre-trained on huge datasets like Kinetics have successfully boosted the recognition of actions from internet videos. But, these networks with rigid kernels applied across the whole space-time volume cannot address the challenges exhibited by Activities of Daily Living (ADL).

We are particularly interested in discriminative video representation for ADL. Besides the challenges in generic videos, ADL exhibits - (i) fine-grained actions with short and subtle motion like *pouring grain* and *pouring water*, (ii) actions with similar visual patterns differing in motion patterns like *rubbing hands* and *clapping*, and finally (iii) long complex actions like *cooking*. In order to address these challenges, we have made contributions.

The first contribution includes - a multi-modal fusion strategy to take the benefits of multiple modalities into account for classifying actions. In an attempt to comply with the global optimization strategies for action classification,our second contribution consists in articulated pose driven attention mechanisms for action classification. We propose, three variants of spatio-temporal attention mechanisms exploiting RGB and 3D pose modalities to address the aforementioned challenges (i) and (ii) for short actions. Our third main contribution is a Temporal Model on top of our attention based model. The video representation retaining dense temporal information enables the temporal model to model long complex actions which is crucial for ADL.

We have evaluated our first contribution on three small-scale public datasets: **CAD60, CAD120** and **MSRDailyActivity3D**. On the other hand, we have evaluated our remaining contributions on four public datasets: a large scale human activity dataset: **NTU-RGB+D 120**, its subset **NTU-RGB+D 60**, a real-world challenging human activity dataset: **Toyota Smarthome** and a small scale human-object interaction dataset **Northwestern UCLA**. Our experiments show that the methods proposed in this thesis outperform the state-of-the-art results.

Keywords: action recognition, spatio-temporal, attention mechanism, long complex actions.

Dedicated to
Ma, Baba, Didi,
who always picked me up on time
and encouraged me to go on every adventure,
especially this one,
and those guardian angel in heaven,
my grandmother and uncle.

ACKNOWLEDGMENTS

First, I would like to express my deepest gratitude to my thesis supervisor, François Brémond, who has the attitude and the substance of a genius. He convincingly conveyed a spirit of adventure in regard to research, and an excitement in regard to mentoring. Without his guidance and persistent help this dissertation would not have been possible. The good advice, support and friendship of my co-supervisor, Monique Thonnat, has been invaluable on both an academic and a personal level, for which I am grateful.

I would like to acknowledge Université of Côte d'Azur for supporting my PhD thesis. Thanks to my thesis reviewers, Christian Wolf and Dima Damen, that they kindly agreed to review my PhD manuscript. Also, thanks to Matthieu Cord and Micheal S. Ryoo for serving as members to my thesis committee.

Special thanks to Michal Koperski, who was a deligent mentor during my initial days at INRIA. I won't forget to mention Rahul Pandey, whose technical help made my initial days at INRIA wonderful. I also owe a great debt of gratitude to Saurav Sharma for introducing me to the interesting world of Machine Learning. Thanks to Sambit Bakshi and Imon Mukherjee, whose initial guidance led me to take up academic research seriously.

"*Great things in research are never done by one person. They are done by a team of people.*" Thanks to all my colleagues from STARS team for technically, scientifically, and personally helping me in this journey. This includes all my collaborators and interns with whom I had a great time. Thanks for the stimulating discussions, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last four years.

I would like to thank Toyota Motors Europe, especially Gianpiero Francesca for these years of collaboration and support. I thank Marc Vesin and Francis Montagnac, the administrators of NEF computation cluster. Without their technical support in this era of performing exhaustive experimental analysis to support research claims, my thesis would have been incomplete.

"*Anything is possible when you have the right people there to support you.*" Thanks to all my friends for tolerating my idiosyncrasies and crazy habits. You might not know this, but you helped me find happiness in being the person that I really am. I would also like to extend my thanks to all my relatives, especially to my aunt, Sabya Da, and their family for standing beside my parents in my absence.

Last but not the least, I would like to thank my parents for supporting me unconditionally and always motivating me for higher studies. "*They sacrificed their sleep so that I could dream big.*" And the ending quotes for the person I idealize. "*A sister like you is such a wonderful gift. There have been at least a thousand times when I have thanked my lucky stars for you.*"

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Humans perform many high level tasks that involves complicated processing, which eventually takes place in our brain. The same is true for computers especially for any computer vision task. Human brains can easily perceive the happenings in their nearby environment. This is true even for a child brain. But computers are not good with complicated reasoning for perceiving the environment. Thus, the primary goal of computer vision is to improve computer abilities to interpret images and videos. Such abilities will play a key role for the future of artificial digital world which comprises intelligent machines like robots, automated cars, etc.

One of the ultimate goals of artificial intelligence research is to build a machine that can accurately understand humans' actions and intentions, so that it can better serve us. Imagine that a patient is undergoing a rehabilitation exercise at home, and his/her robot assistant is capable of recognizing the patient's actions, analyzing the correctness of the exercise, and preventing the patient from further injuries. Such an intelligent machine would be greatly beneficial as it saves the trips to visit the therapist, reduces the medical cost, and makes remote exercise into reality. Other important applications including visual surveillance, entertainment, and video retrieval also need to analyze human actions in videos. In the center of these applications are the computational algorithms that can understand human actions.

In this thesis, we are interested in Human Action Recognition. Action recognition can be defined as the ability to determine whether a given action occurs in the video stream. It is one of key components of the intelligent systems by the ability to interpret such information will ultimately bring computers closer to human skills.

This chapter introduces the problem of action recognition. In section 1.2, we present the key applications of action recognition. In section 1.3, we discuss the research challenges involved in this problem domain. We conclude the chapter with the list of contributions (section 1.4) and a layout of the thesis structure (section 1.5).

Figure 1.1: Large scale distribution of video data from different sources. These sources vary from videos from the internet sources/ captured using monitoring cameras to videos captured using wearable sensor or robots.

1.1 Problem statement

In this section, we define human action recognition problem which can be categorized into two mainstream tasks. The first task is action classification and the second is action detection.

Action Classification
Consider a set of videos $\mathbb{V}$ and a set of corresponding action labels $\mathbb{L}$. Each video $V \in \mathbb{V}$ contains one label $l_V \in \mathbb{L}$. Thus the goal of action classification is to predict the label l_V based on a video representation of video V. This statement could be extended for multiple action instances in a trimmed video, where $l_V \in \mathbb{L}$ is a set of labels.

Action Detection
Action detection is an extension of action classification problem where the objective is to predict all the labels for a given video, as well as starting and ending time of each predicted action.

In this thesis, we focus mainly on action classification on video clips ranging from few seconds to few minutes. We argue that for an effective and efficient real-world action recognition system, we must focus on designing high quality action classifiers. Thus, we focus on video representation for classifying short action clips. Note, for convenience and with a slight abuse of term, hereafter in this thesis we often refer to action recognition as the problem of action classification.

1.2 Applications

In this section, we present some applications based on video analysis. With increasing video data on the web and acquired videos for different scenarios as presented in fig. 1.1, learning strong video representation has become a crucial computer vision task. Modeling actions in a video is a key approach for any video analysis problem.

Video Retrieval
Nowadays, we can observe a fast growth of internet broadcasting services such as YouTube or Vimeo and social media services such as Facebook or Twitter. People can easily upload and share videos on the Internet. However, managing and retrieving videos according to video content is becoming a tremendous challenge as most search engines use the associated text data to manage video data. The text data, such as tags, titles, descriptions and keywords, can be incorrect, obscure, and irrelevant, making video retrieval unsuccessful. An alternative method is to analyze human actions in videos, as the majority of these videos contain some human actions.

Video Surveillance
Security issue is becoming more important in our daily life, and it is one of the most frequently discussed topics. Nowadays we can observe surveillance cameras almost at every corner of the city. The primary goal of surveillance cameras is to increase the security level and protect us from acts of violence, vandalism, terrorism, stealing. Thanks to action recognition systems, video surveillance cameras could be analyzed all the time. With the input of a network of cameras, a visual surveillance system powered by action recognition algorithms may increase the chances of capturing a criminal on a live video stream, and reduce the risk caused by criminal actions.

Human Computer Interaction
The evolution of Human Computer Interaction led us to many different devices which make communication easier, faster and more comfortable. With recent advancements in computer vision and camera sensors, we have reached a point where scenes from science-fiction movies like the one from "Interstellar" are not fiction, but reality. Modern capabilities of sensors like Kinect and Leap Motion powered with gesture or action recognition algorithms allow us to build interfaces where a user can operate computer without any need of holding any device. The computer is able to recognize actions or gestures done by a user and trigger appropriate actions. Such solutions have been already introduced to TV sets. In entertainment industry Xbox game console equipped with Kinect sensor is able to interpret whole body motion, leading to new levels of game experience, especially in sport

games. All recent advancements in Human Computer Interaction are possible thanks to development of gesture and action recognition algorithms.

Robotics

Robots ability of perceiving human actions plays a key role in many robotics applications. For instance, autonomous vehicles which can be considered as specific type of robot. The ability to observe and anticipate the situation of a road is important. Thus efficient interpretation of intention of other traffic participants will play a key role in future autonomous vehicles. The anticipation capabilities may concern vehicles, but also pedestrians. For instance, autonomous vehicles are required to asses and anticipate pedestrian intention to cross a road. This would let autonomous car avoid potentially dangerous situations.

Human-robot interaction is also popularly applied in home and industry environment. Imagine that a person is interacting with a robot and asking it to perform certain tasks, such as *passing a cup of water* or *performing an assembling task*. Such an interaction requires communications between robots and humans, and visual communication is one of the most efficient ways.

Healthcare

According to a recent report of the United Nations [15], the global population aged 60+ is projected to grow from 0.9 billion in 2015 to 1.4 billion in 2030. This demographic trend translates to the dramatic need for an increase of the workforce in healthcare. To relieve such workforce stress, activity detection system is becoming important to help monitor the health state of older patients and support the early detection of potential physical or mental disorders. For instance, monitoring patient eating habits allows doctors to track the state of a patient and react before serious health conditions arise. Thanks to such systems seniors can stay longer at home without the need of being hospitalized, which can greatly improve their comfort and quality of life.

1.3 Research challenges

Action recognition is a very challenging research problem and many unanswered questions keep this problem unsolved until this date. Over the last few years, static image classification has taken new strides. We are now closer to solving tasks like image classification, object detection and even semantic segmentation [16]. On the other hand, when it comes to video understanding we are still struggling to answer basic questions like: What are the right categories of actions? Do activities have well-defined spatial and temporal extent? How to model time in videos?

Figure 1.2: Example of some videos retrieved from internet sources (at left) and some ADL videos (at right). Note the meaningful background information for the web videos compared to the similar background for ADL videos.

The challenges in the field of action recognition begins from the most fundamental question - what are the right action categories? Unlike objects where categories are well defined, action categories defined by verbs are relatively few in number. Should we focus our analysis on action categories like *drinking* or more specific ones like, *drinking from cup*? Verbs like *drinking* and *walking* are unique on their own, but verbs like *take* and *put* are ambiguous unless nouns and even prepositions are included: *take pills*, *take off shoes*, *put something on table*, *put on shoes*.

The next ambiguity lies in annotating the actions where an action persists implicitly within another another. For instance, a person using telephone while walking. Do we have a single primary label for this action, i.e. *using telephone* or do we have both the labels *using telephone* and *walking*? Most of these scenarios especially, the action *walking* remains implicitly in many other actions. How do we model such actions?

Actions, when annotated in real-world scenarios naturally result in a Zipf's law type of imbalance across action categories. There will be many more examples of typical actions (*walking* or *sitting*) than memorable ones (*using telephone*), but this is how it should be! Recognition models need to operate on realistic "long tailed" action distributions rather than being scaffolded using artificially balanced datasets. But how do we handle such data imbalance in action recognition models?

Finally, the challenge of domain adaptation in the field of action recognition. While capturing data, three approaches have been attempted to achieve scalability: (i) action videos from the web, (ii) crowd-sourcing scripted actions, and (iii) long-term collections of natural interactions in homes. While the latter offers more realistic videos, many actions are collected in only a few environments and only from a single view-point. This leads to learned representations which do not generalise well [17]. Are the current action

recognition models able to combat the shifts like changes in environment, changes in illumination and changes in view points?

A significant problem in the past has been the absence of generic datasets for action recognition. Most of the major advances in recognition methods are often paired with the availability of annotated data. For instance, significant boosts in image recognition accuracy on the AlexNet and VGG architectures [18, 19] were possible thanks to ImageNet [20] and COCO [21] datasets. On the other hand, historical video datasets like UCF-101 [22] and HMDB-51 [23], Kinetics [24] are gathered from video web services (e.g. YouTube). Such datasets introduce data bias as they mainly contain actions concerning sports, outdoor actions and playing instruments. In addition, these actions have a significant inter-class variance (e.g. bike riding vs. sword exercising), which usually does not characterize daily living actions. Besides, most video clips only last a few seconds. Does these data really help in understanding a large variety of actions?

In this thesis, we present approaches for learning video representations to discriminate visually similar actions, specifically Activities of Daily Living (ADL). As discussed above, these boring videos [25] are not commonly available in internet sources. Some representative frames from web videos and ADL videos are illustrated in fig. 1.2. Moreover, very few action recognition methods have been dedicated for recognition of ADL w.r.t. methods developed to discriminate generic videos. Below, we discuss the challenges involved in learning representations for actions, with more focus on ADL because of their intrinsic complexities as illustrated in fig. 1.3.

How to handle time?
Unlike image classification, action recognition involves handling space as well as time dimension. Temporal extent of an action is ambiguous. Imagine the case of annotating action temporal boundaries. It is difficult to reach a consensus when annotating, because often it is difficult to agree on the start and end times of an action. For example, when does the action *drinking* start? Is it when the glass touches the mouth, or when the person start holding it? This kind of ambiguities introduces noise and biases in the training data [16]. Consequently, learning video representation based on different temporal aggregation strategies have been developed over the years. However, handling the temporal dimension in web videos through pooling mechanisms, sequential networks and temporal convolutions have ensured optimal solutions for dedicated end tasks respectively. These videos have strong motion and thus are discriminated with simple temporal aggregation techniques. However, strategies suitable to model temporal information for ADL are yet to be explored. ADL involve variation in appearance, motion patterns and view points which prevents learning an invariant or generic joint spatio-temporal patterns across the video.

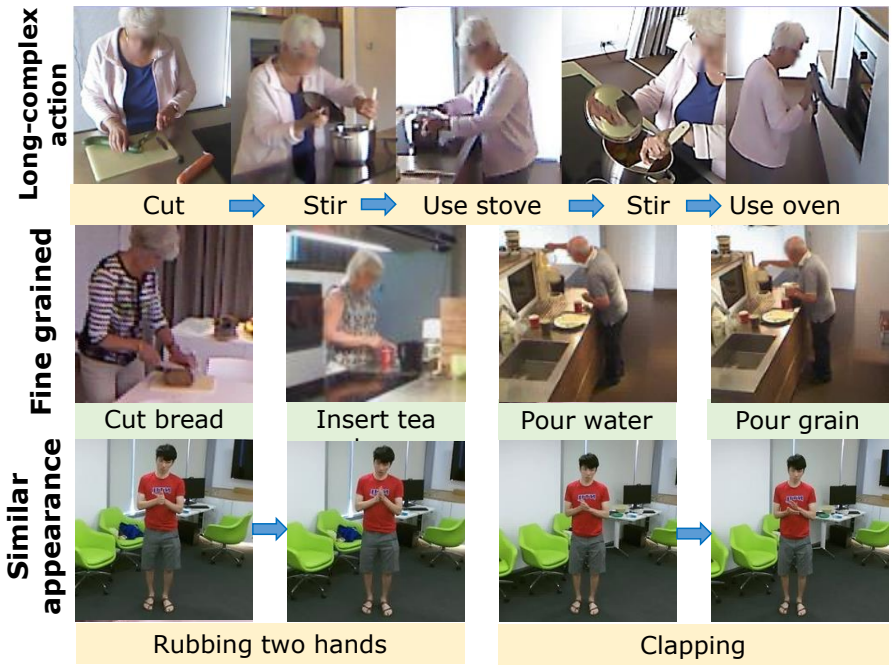


Figure 1.3: Illustration of challenges in ADL: Long-complex actions like *cooking* (top), fine-grained actions (middle), actions with similar visual pattern (below).

Modeling temporal information is already a challenge for short video clips for ambiguous temporal boundaries. Besides that, another important challenge that commonly persists in ADL is long-range temporal dependencies in actions like *making breakfast* or *cooking*. These actions are composed of several sub-actions occurring in a sequence. For example, *cooking* is composed of *cutting*, *stirring*, *using stove*, etc. These actions exhibit three properties: composition, temporal extent and temporal order. Here, we assume that an action concludes before another starts. However in ADL, concurrent actions occur quite often and require their understanding in parallel. Thus, modeling these sub-actions along with linking them over time to recognize a coarse action is a challenging task. The difficulty remains in learning the long-term relationship among the sub-actions in the videos, which the current state-of-the-art algorithms fail to incorporate.

How to learn representations for recognizing fine-grained actions?
Fine-grained actions can be defined in two ways depending on their spatial and temporal extent. With respect to space, similar actions with different object interactions like *drinking with a cup* or *drinking with a bottle* require concentrated focus of attention to encode the object information to discriminate the actions. With respect to time, actions like *cutting bread* and *typing keyboard* involve subtle motion for a very short period of time and thus require fine-grained understanding for a compact and discriminative video

representation. State-of-the-art methods processing the volume of a video cannot model the subtle variation in motion patterns and lack mechanism to encode fine spatial details. Thus, these methods fail to learn discriminative representation for fine-grained actions.

How to disambiguate similar appearance actions?
Actions from internet sources have high inter-class variation which enable the classifiers to discriminate them effectively. For instance, actions like *swimming* and *playing soccer* can be distinguished only by looking at their background which is the water and green grass respectively. Whereas in ADL, the contextual information does not help much because of similar indoor environment, low inter-class variation adds to the challenges. ADL exhibit similar visual patterns while differing in motion patterns like *rubbing hands* and *clapping* require fine and joint modeling of spatio-temporal information in videos. Most of the state-of-the art methods do not couple space and time together which implies that the temporal information is processed after processing the spatial information. As a result, discriminating the similar actions with subtle variation in motion patterns are not captured by the prior methods.

1.4 Contributions

Although the methods proposed in this thesis are applicable for generic video representation, as mentioned earlier, we are particularly focusing on ADL. Our contributions are motivated by the complex challenges involved in ADL. We argue that, addressing the ADL challenges should improve the overall classification of a wide diversity of actions. In order to address the challenges in ADL, we have made three key contributions. The first contribution includes a multi-modal fusion strategy to take benefits of multiple modalities into account for classifying actions. Our second contribution is a pose driven attention mechanism for action classification. Our third contribution is a Temporal Model combining temporal representation and attention mechanism. Below, we briefly discuss the contributions made in this thesis.

1.4.1 Multi-modal Fusion

Recognizing actions in videos involve understanding different cues. Cues computed from different modalities are complementary in their feature space. Thus, fusing them in a common feature space enables a classifier to learn even more discriminative features compared to their classification in individual feature space. Thus, to incorporate the effectiveness of each modality, we propose a new strategy to fuse RGB, optical flow and 3D poses for action recognition. This proposed methodology aims at handling different temporal extents

present in ADL, thus addresses the challenge of handling time in videos. We show that RGB cue modeling appearance variation, optical flow modeling short-term motion, and temporal evolution of 3D poses modeling long-term motion, when combined in a strategic manner improves the action recognition accuracy. The question remains how do we combine these cues?

Further to this, we invoke a similar action discrimination module to disambiguate the actions with similar appearance. The objective is to make use of the individual properties of each cue to disambiguate the actions with similar appearance.

1.4.2 Spatio-temporal Attention Mechanisms

Next, we propose a novel variants of spatio-temporal attention mechanisms to address the challenges in ADL. In this proposed methodology, instead of multi-modal representation, we aim at infusing other modalities into the RGB network through attention mechanism. We present three variants of spatio-temporal attention mechanisms on top of the state-of-the-art 3D ConvNet [3] (to be discussed in Related Work). These attention mechanisms address the challenge of classifying fine-grained actions and disambiguating similar appearance actions. Attention mechanisms provide a strong clue to focus on the pertinent Region of Interest (RoI) with respect to space and time in a spatio-temporal feature map. Below, we briefly describe the variants of our proposed pose driven attention mechanisms.

1. Human body Parts based spatial attention mechanism
In this work, we aim at focusing on the pertinent human body parts relevant for an action. The pose driven attention provides a set of attention weights for the spatio-temporal features for each human body part. We argue that such video representation will incorporate fine-grain details related to space for modeling an action.

2. Separable spatio-temporal attention mechanism
In order to generalize the above methodology, we aim at focusing on the RoI within the entire human body region. This relaxes the burden of training a large number of network parameters along with a compact video representation. Moreover, in this work we propose to provide spatial and temporal attention weights to focus on the RoI in space and to focus on the key frames for learning discriminative representation for an action. However, the question remain, how to learn spatio-temporal attention weights - separately or jointly? We explore such choices for attention and proposed our separable spatio-temporal attention mechanism.

3. Video-Pose Embedding
The above two variants however, do not consider the correspondences between the 3D poses and RGB cue. In this variant of Basic, we propose a spatial embedding to provide an accurate alignment between the modalities exploited to learn the video representation. We show that such self-supervising task not only improves the preciseness of attention weights but also learns effective video representation to discriminate ADL. The attention network in this model utilizes the graphical structure of 3D poses and jointly learns the spatio-temporal attention weights for a video.

1.4.3 Temporal Representation for ADL

In order to address the challenge of modeling long complex actions in ADL, we propose an end-to-end Temporal Model to incorporate long-range temporal dependencies in actions without losing subtle details. The temporal structure of an action is represented globally by different temporal granularities and locally by temporal segments. We also propose a two-level pose driven attention mechanism to take into account the relative importance of the segments and granularities.

1.4.4 Experimental Studies

We have evaluated our first contribution on three small-scale public datasets. We have evaluated our second and third contributions on four public datasets. We have conducted exhaustive ablation studies to show the effectiveness of each of our proposed methods. Finally, we discuss the limitations of our proposed methods and possible future research directions.

1.5 Thesis structure

- **Chapter 1** "**Introduction**" introduces the action recognition problem. We describe the problem motivation and provide key possible applications. Then we discuss main challenges in action recognition problem and summarize our contributions.

- **Chapter 2** "**Related Work**" introduces the state-of-the-art action recognition methods based on how they address the research challenges we have defined. We provide a brief study on the handcrafted approaches proposed in this problem domain. However, our literature survey mainly focuses on the recent deep learning approaches. Consequently, we present a detail description on how different modalities (RGB, optical flow and 3D poses) are processed in different network architectures to model

temporal information in videos. We present previous works on how different modalities are combined for the task of action recognition. We present a detailed study on different attention mechanisms based action recognition approaches relevant to our problem. Finally, we present a comparative study of the different public datasets in the action recognition domain. We also describe the datasets that have been used in this thesis for our experimental studies.

- **Chapter 3** "**Multi-modal Video Representation for ADL**" presents a multi-modal fusion strategy for effective action recognition. We present an entire action recognition framework - feature extraction from different relevant cues for learning discriminative features in ADL, multiple levels of feature fusion mechanism followed by a classifier learning the action labels. We also present a similar action discrimination module for the refinement of the learned action labels by the classifier.

- **Chapter 4** "**Attention Mechanisms for Visual Representation**" presents an attention mechanism based action recognition framework for the wide diversity of actions in ADL. We present three variants of pose driven attention mechanism - to focus on relevant human body parts for an action representation, to focus on the relevant RoI in a spatio-temporal feature map of a video, and to improve the former attention networks by providing accurate alignment of 3D poses and RGB cue.

- **Chapter 5** "**Temporal Representation for ADL**" presents for an accurate and effective temporal representation of videos for action recognition. We aim at modeling long complex actions in ADL through this Temporal Model. We present a new temporal representation of a video through different temporal segments with several temporal granularities. This temporal representation of video is accompanied by a two-level pose driven attention mechanism for obtaining even more discriminative video representation compared to the former.

- **Chapter 6** "**State-of-the art comparison**" presents a performance comparison with the state-of-the-art techniques. We evaluate our approaches on the relevant datasets. We also investigate advantages and limitations of our proposed methods.

- **Chapter 7** "**Conclusion & Future Work**" outlines our approaches and possible future perspectives of this thesis work. We provide discussion about advantages and limitations of our work. In this chapter we also suggest future directions, presenting short-term and long-term perspectives.

Chapter 2

Related Work

2.1 Introduction

In this chapter we review the methods for Action Recognition published in recent years. This literature study revolves round how action recognition has been approached in the recent years for generic videos, how they are limited for ADL. We broadly categorize our study based on the challenges discussed earlier. Firstly, we discuss how the state-of-the-art action recognition frameworks handle time for discriminative video representation. Within this category, we present the handcrafted and deep learning approaches in this domain based on their input modalities.

In this thesis, we mainly focus on RGB, optical flow and 3D poses obtained from RGB-D sensors due to their effectiveness over modalities like IR, depth map and so on. Secondly, we discuss how previous studies have made attempts to discriminate fine-grained and similar actions. In this category, we focus on multi-modal representation of videos leveraging the advantages of each modality. Finally, in an attempt to globally optimize all modalities in a single RGB network, we study the recent attention mechanism based approaches to address the challenges in ADL.

Since most of our literature survey covers a wide range of **Deep Learning** algorithms, we first discuss some concepts that have been used hereafter.

2.2 Background

- Deep Neural Network (DNN) - consists of an input layer, multiple hidden layers and an output layer. The input layer receives raw inputs and computes low level features like *lines, corners*. The hidden layers transform and combine the low-level features into high-level features depicting semantic concepts. Finally, the output layer uses the high-level features to predict results.

- Convolutional Neural Network (CNN) - consists of an input layer, multiple hidden layers and an output layer. The hidden layers of a CNN typically consist of a series of convolutional layers followed by pooling operations. In a convolutional layer, a kernel (also called filter) slides across the input feature map. At each location, the product between each element of the kernel and the input element is computed and summed up as the output in the current location. Besides convolutional layer, pooling operation also plays an important role in CNNs. Pooling operations reduce the size of feature maps by using some function to summarize sub-regions, such as taking the average or the maximum value. For more details about CNNs, please refer to [26].

- Recurrent Neural Network (RNN) - can be thought of as multiple copies of the same network, each passing a message to a successor. They are networks with loops in them, allowing information to persist. However, these networks are not capable of learning long-term dependencies because of gradient vanishing factor.

- Long Short term Memory (LSTM) - is a special kind of RNN which alleviates the effect of vanishing gradient issue in RNN. LSTM consists of a cell state and four gates. The cell state is a kind of conveyor belt. It runs straight down the entire chain, with only some minor linear interactions. Its very easy for information to just flow along it. Whereas gates are a way to optionally let information through. They are composed of sigmoid activated layer and a point wise multiplication operation. The four gates are - forget gate, input gate, self-recurrent gate and output gate. For more details on LSTM, refer to [27].

Below, we detail the relevant state-of-the-work categorized according to the challenges they address.

2.3 How to handle time?

In this section, we categorize the methods into handcrafted approaches and deep learning approaches. In each category, we discuss the previous methods based on individual modalities - (i) RGB and Optical Flow and (ii) 3D Poses.
An important question is why RGB, Optical Flow and 3D poses? We argue that RGB and 3D poses are the privileged modalities that provide salient features for discriminating ADL. RGB provides information based on appearances of the subject performing an action and optical flow provides information about short-term motion of an action. While 3D poses provide crucial geometric information related to an action which is robust to illumination changes, view changes. Their complementary nature motivates us to investigate further

to combine these modalities to address the challenges in ADL. Below, we detail the action recognition frameworks proposed leveraging these modalities individually.

2.3.1 Handcrafted Approaches

In this section, we discuss how the problem of video classification, especially action recognition has been approached prior to the era of deep learning.

Approaches based on RGB and Optical Flow
The key idea behind video analysis is to capture characteristic features from a local spatio-temporal representation of a video.
An image is a 2-dimensional data formulated by projecting a 3-D real-world scene, and it contains spatial configurations (e.g., shapes and appearances) of humans and objects. A video is a sequence of those 2-D images placed in chronological order. Therefore, a video input containing an execution of an action can be represented as a particular 3-D XYT space-time volume constructed by concatenating 2-D (XY) images along time (T).

Space-time approaches are those that recognize human activities by analyzing the space-time volumes of action videos. A typical space-time approach for human action recognition is as follows. Based on the training videos, the system constructs a model representing each action. When an unlabeled video is provided, the system constructs a 3-D space-time volume corresponding to the new video. The new 3-D volume is compared with each action model (i.e., template volume) to measure the similarity in shape and appearance between the two volumes. The system finally deduces that the new video corresponds to the action that has the highest similarity. This example can be viewed as a typical space-time methodology using the 3-D space-time volume representation and the template-matching algorithm for recognition.

Below, we detail some popular methods prior to the era of deep learning based on space-time approaches categorized into (A) space-time volumes, (B) space-time local features, and (C) space-time trajectories.
(A) Space-Time Volumes - The key idea of the recognition using space-time volumes is in the similarity measurement between two volumes. In order to compute correct similarities, a wide range of space-time volume representations have been developed.
Instead of concatenating entire images along time, some approaches only stack the foreground regions of a person (i.e., silhouettes) to track shape changes explicitly [28]. An approach to compare volumes in terms of their patches corresponding to the neighborhood has been proposed as well [29]. Ke et al. [30] used over-segmented volumes, automatically calculating a set of 3-D XYT volume segments that corresponds to a moving human. Rodriguez et al. [31] generated filters capturing characteristics of volumes by adopting the maximum average correlation height (MACH), in order to solve the problem of action

recognition.

(B) Space-Time Local Features - The approaches in this section use local representation extracted from 3D space-time volume to recognize actions. The motivation behind these approaches is the fact that a 3-D space-time volume essentially is a rigid 3-D object. This implies that if a system is able to extract appropriate features describing characteristics of each action's 3-D volume, the action can be recognized by solving an object-matching problem.
Laptev and Lindeberg [32] recognized human actions by extracting sparse spatio-temporal interest points from videos. They extended the previous local feature detectors [33] commonly used for object recognition, in order to detect interest points in a space-time volume. Dollar et al. [34] proposed a new spatio-temporal feature detector for the recognition of human (and animal) actions. Their detector is especially designed to extract space-time points with local periodic motions, obtaining a sparse distribution of interest points from a video. Bregonzio et al. [35] proposed an improved detector for extracting cuboid features, and presented a feature selection method. Rapantzikos et al. [36] extended the cuboid features to color and motion information as well, in contrast to previous features using intensities only (e.g., [32]; [34]).

(C) Space-Time Trajectories - Feature trajectories is an idea that arisen on top of local representation. Many authors [37, 38, 39, 10, 40] claimed that 2D spatial domain and temporal domain posses very different characteristics. Therefore, many authors [37, 38, 39, 10, 40] proposed methods where they track spatial Points of Interest (PoI) across time. The trajectory shape and descriptors computed based on volume around the trajectory points are used as video representation. Messing et al. [38] extracted feature trajectories by Harris3D interest points [32] with the KLT tracker [41]. Wang et al. [10, 40] proposed local descriptors around dense trajectories which are densely sampled PoI. These POIs were tracked using optical flow. The local descriptors around the dense trajectory points are HoG [42], HoF [43], MBH [44] features. The methods which leverage trajectory information showed impressive results in action recognition. As a results, these methods are used even today by blending the dense trajectories with deep features [45, 1]. However, these RGB based approaches are 2D spatial data and misses crucial depth information. As a result, these approaches do not address the challenge of view-invariance which is a characteristic property of ADL.

Approaches based on 3D poses

To focus on the view-invariant challenge, temporal evolution of 3D poses have been leveraged for skeleton based action recognition [46, 47, 48, 49, 50, 5, 51, 52, 53, 54]. 3D poses are robust to illumination changes, view changes and provide geometric information associated to the actions. These underlying properties of the 3D poses have motivated

the vision community to move steps forward in 3D human action analysis [5, 51, 52, 53]. Below, we briefly describe how the 3D poses are obtained from the depth sensors.

The most popular skeleton detection method that we use in this entire thesis, is using Kinect as discussed in [55]. In this algorithm the human poses are inferred in a two stage process: first by computing a depth map and then the body position. It begins with 100,000 depth images with known skeletons from a motion capture system and then computer graphics is used to render all sequences for 15 different body types. Thus a million training examples are produced which are used to learn a randomized decision forest for mapping the depth images to body parts. Then, the mean shift algorithm is used to robustly compute the modes of probability distributions to transform the body image into a skeleton.

After the introduction of affordable depth sensors, many approaches [46, 47, 48] have been proposed that use human skeletons for modeling actions. Note that, by skeletons we mean human 3D poses in the entire thesis and we use them interchangeably. Vemulapalli et al. [49] represented each skeleton using the relative 3D rotations between various body parts. Their skeletal representation becomes a point in a Riemannian manifold. Then, using this representation, they model human actions as curves in this manifold and perform classification in the Lie algebra. Wu et al. [50] have proposed a hierarchical dynamic framework that first extracts high level skeletal features and then uses the learned representation for estimating label probability to infer action classes.

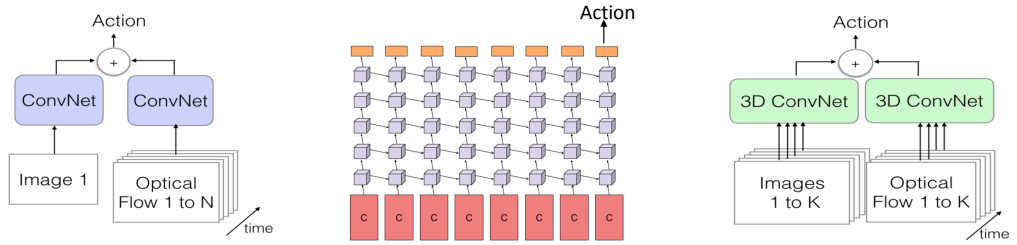
All these methods discussed till now are handcrafted approaches. These methods extract features representing the input spatio-temporal data and then fed to a classifier. The limitation lies in the lack of a global optimization strategy that could learn representations for classifying the actions directly. Hence below we discuss the deep learning approaches in the recent years for the task of action recognition.

2.3.2 Deep Learning Approaches

With the emergence of deep learning and the enormous success of image classification tasks, many authors [1, 56, 45] have proposed methods using Deep Neural Network (DNN) for video classification. Different architectures of deep learning networks have been proposed (2D CNN, 2D CNN + RNN, 3D CNN) as shown in fig. 2.1. In the following, we detail these approaches.

Approaches based on RGB and Optical Flow

In this section, we present the evolution of handling temporal information in videos from state-of-the-art methods. This evolution involves a transition from using 2D CNN + pooling techniques to 3D CNNs capable of taking spatio-temporal inputs for video representation.



(a) 2D CNN based approaches (b) 2D CNN + RNN based approaches (c) 3D CNN based approaches

Figure 2.1: Deep Learning based key action recognition approaches to model temporal information in videos. These approaches use (a) 2D CNN (left), (b) 2D CNN + RNN (middle), and (c) 3D CNN (right) to aggregate temporal information for action classification. The figures have been extracted from [1, 2, 3] respectively.

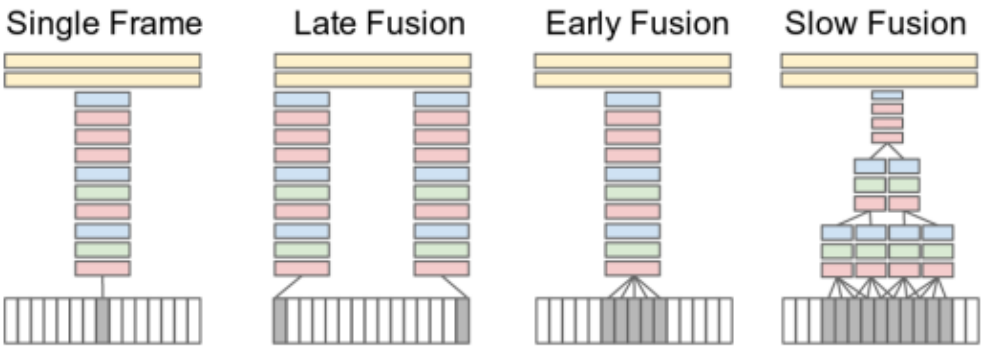


Figure 2.2: Explored approaches for fusing information over temporal dimension through the network in [2]. Red, green and blue boxes indicate convolutional, normalization and pooling layers respectively (Figure from [2]). In the Slow Fusion model, the depicted columns share parameters.

(A) 2D CNN based approaches - Karpathy et al. [2] extended the connectivity of a CNN in time domain to take advantage of local spatio-temporal information. They explored different strategies for the fusion of input data along different temporal dimension in CNNs as shown in fig. 2.2 - (i) a model based on single frame (ii) an early fusion model which combines information across an entire time window immediately at the pixel level. (iii) a late fusion model which places two separate single frame networks with shared parameters, and then merges the two streams via a fully connected layer. (iv) a slow fusion model which is a balanced mix between the two approaches that slowly fuses temporal information throughout the network such that higher layers get access to progressively more global information in both spatial and temporal dimensions. Finally, they [2] have proposed a multi-resolution CNN architecture for action recognition. The input frames are fed into two separate streams of processing: a context stream that models low-resolution image and a fovea stream that processes high-resolution center crop. This multi-resolution CNN architecture with slow fusion along temporal domain proved to be effective for action classification in sport videos with dissimilar background. Simonyan et al. [1] have proposed to model motion with CNN (temporal stream) trained on optical flow input, together with another CNN (spatial stream) trained on input still images. The spatial stream models the appearance whereas the temporal stream models the short-term motion. This approach was later extended by Feichtenhofer et al. [56], they replaced the late fusion of both the streams by fusing two nets at lower layers and perform joint training. Recently, the costing burden of optical flow computation is balanced by two in one stream network [57]. Such representations with no requirement of optical flow at test time results in better video representation compared to their fusion. Authors in [45, 58] have proposed to extract CNN features from the tracks of different human body parts (left hand, right hand, full body and upper body). The body parts are extracted using 2D pose information computed either from kinect sensors or pose estimation algorithms [59, 60]. Finally, the RGB and optical flow based CNN features are fused and classified using a linear SVM classifier to recognize fine-grained actions. All these approaches utilizing optical flow frames [1, 56, 45] can model short-term motion ($\leq$ 1 second). The two-stream CNNs [1, 56, 45] uses stacked optical flows computed in short time windows as inputs, and the order of the optical flows is fully discarded in the learning process. This is not sufficient for video classification, as many complex contents can be better identified by considering the temporal order of short-term actions [61]. Take "birthday" event as an example - it usually involves several sequential actions, such as "making a wish", "blowing out candles" and "eating cakes".

(B) 2D CNN + RNN based approaches - To address the above limitation, a lot of studies have proposed action recognition using Recurrent Neural Networks (RNNs) to better

model long-term temporal information. Inspired from image captioning, machine translation and video description tasks, authors in [62, 63, 64, 65, 66, 67, 61, 68, 69] used the concept of encoder + decoder to recognize actions. As shown in fig. 2.1(b), the key idea in these approaches include encoding the frame-level features using a 2D CNN (encoder) and finally perform a complex temporal pooling of the aforementioned features using a sequential networks like LSTM (decoder) before classifying the actions. These encoders are generally image classification networks pre-trained on ImageNet [20]. Besides, the key idea remains the same, variants of RNNs like stacked LSTMs, Gated Recurrent Unit (GRU), bi-directional LSTMs are used in [65, 66, 67, 68]. Similar to [1], authors in [61, 69] have proposed multi-stream fusion of RGB, optical flow or audio based features computed by an encoder. However, these sequential networks rely too much on strong human motion as in sport videos. Consequently these methods outperforms the prior methods for sport videos [70] but fails to discriminate actions in ADL [58]. These approaches do not address the key challenge of modeling subtle motion possessed by ADL.

(C) 3D CNN based approaches - Due to this reason, Tran et al. [71] have proposed 3D (XYT) convolution to model the spatio-temporal patterns within an action. Note, for simplicity and slight abuse of notation, we will denote 2D + T (XYT) convolutions as 3D convolutions throughout the thesis. The 3D kernels provide tight coupling of space and time towards better action classification. The current studies on 3D ConvNets describe them as a good descriptor being generic, compact, simple and efficient [71]. 3D convolutional deep networks can model appearance and motion simultaneously. In 3D ConvNets, convolution and pooling operations are performed spatio-temporally while in 2D ConvNets they are performed only spatially. Recently, Carreira and Zisserman [3] fabricated a 3D CNN based fully convolutional network namely I3D for action classification. The design choice of I3D enables it to leverage pre-training from ImageNet [20]. This is done by inflating the 2D kernels to 3D kernels. Moreover, asymmetric operations are imposed along space and time, for example, initial layers apply $1 \times 3 \times 3$ convolutional operations compared to $3 \times 3 \times 3$ to handle the higher dimension along the spatial domain. I3D with 9 inception modules, multiple bottlenecks [72] to reduce parameter complexity, and pre-trained on ImageNet [20] and Kinetics [24], is well engineered for video classification problems. With the success of I3D holistic methods like slow-fast network [73] and MARS [7] have been fabricated for generic datasets like Kinetics [24] and UCF-101 [22]. Slow-fast network [73] adapted the concept of fovea and context stream from [2] using 3D CNN as visual backbone. With the slow pathway, this network captures spatial semantics of the image frames whereas with the fast pathway, this network captures motion at fine temporal resolution. The slow and fast pathways are implemented by operating the videos at low and high frame rate respectively. In order to optimize the network, the fast

pathway (with high frame rate) reduces the channel capacity.

Further, authors in [74] have proposed a balance between speed and accuracy by building an effective and efficient video classification system through systematic exploration of critical network design choices. They have proposed to replaced the expensive 3D convolutional operations by separable 3D convolutional operations ($1 \times 3 \times 3$ convolution followed by $3 \times 1 \times 1$ convolution) at the initial layers, suggesting that temporal representation learning on high-level "semantic" features is more useful. When combined with separable convolutions and feature gating inspired from [75], their system results in an effective video classification system that produces very competitive results on several action classification benchmarks. Similarly, Hara et al. [76] explored I3D with residual connections, namely 3D ResNet, to improve the action classification performance. Towards this direction of exploring 3D video networks, Tran et al. [77] empirically demonstrate that the amount of channel interactions plays an important role in the accuracy of 3D group convolutional networks. Their experimental analysis substantiates two main findings: (1) it is a good practice to factorize 3D convolutions by separating channel interactions and spatio-temporal interactions as this leads to improved accuracy and lower computational cost, and (2) 3D channel separated convolutions provide a form of regularization, yielding lower training accuracy but higher test accuracy compared to 3D convolutions. Recently, Feichtenhofer proposed a family of efficient video networks [78] by expanding a tiny image classification architecture along multiple network axes, in space, time, width and depth. The candidate axes are temporal duration, frame rate , spatial resolution, network width, bottleneck width, and depth. The resulting architecture is referred as X3D (Expand 3D) for expanding from the 2D space into 3D space-time domain. The 2D base architecture is driven by the MobileNet core concept of channel-wise1 separable convolutions, but is made tiny by having over 10$\times$ fewer multiply-add operations than mobile image models. The expansion then progressively increases the computation by expanding only one axis at a time, train and validate the resultant architecture, and select the axis that achieves the best computation/accuracy trade-off. The process is repeated until the architecture reaches a desired computational budget.

All the methods discussed above have two main limitations for ADL recognition. These networks [3, 74] (i) cannot capture the varieties in temporal extents of complex actions, and too short (about 99 frames) for long-range temporal modeling, and (ii) with similar spatio-temporal kernels applied across the whole space-time volume of a video, are too rigid to capture salient features for subtle spatio-temporal patterns for ADL.
To overcome (i), several strategies with different frame level sampling strategies have been exploited detailed below. For (ii), later we discuss the literature for attention mechanisms used on top of these 3D CNNs.

Approaches for long complex actions - To learn long-range temporal patterns, previous works have considered processing long videos into several video segments [79, 80, 81]. To learn video-wide representations, Temporal Relation Network (TRN) [79] learns relations between several video segments. A pairwise relation is learned using a composite function (which is a multilayer perceptron) among the segments. Each segment is represented by a sampled frame. The final classification is performed by capturing temporal relations at multiple time scales. However, TRN introduces noise by averaging the features at multiple time scales. Similarly in Temporal Segment Network (TSN) [80, 81], a video is divided into a fixed number of non-overlapping segments, and a frame is randomly sampled from each segment. Then a consensus function aggregates the information from the sampled frames. A similar segment based method has been proposed in [82] with self-attention to adaptively pool the frame-level softmax scores for each segment to obtain the video-level prediction. Note that we will discuss more about self-attention mechanisms later in this chapter. All these segment based methods [79, 80, 81] can encode the temporal evolution of the image features sparsely sampled from the whole videos. However, these sampled frames are disconnected which prevents the extraction of subtle motion patterns. So, these methods perform well on internet videos (videos with strong motion w.r.t human posture and background) and videos with distinctively high human motion (for [70, 22]), whereas they do not model the smooth local temporal structure for ADL.

Recently, Wang et al. [83] have proposed to represent videos as space-time region graphs. Their graph nodes are the object region proposals from different frames in a long video. These nodes are connected by two relations: (i) similarity relations capturing the long range dependencies between correlated objects and (ii) spatial-temporal relations capturing the interactions between nearby objects. But this representation succeed in modeling temporal footprint of 128 time steps (4-5 sec) at max. Therefore, Hussein et al. [4] have proposed a timeception layer to address complex actions with long-range temporal dependencies of up to 1024 time steps, jointly. Timeception layer takes as input the features from 3D visual backbone like I3D or even 2D visual backbone like ResNet-152. These features correspond to the time steps from the previous layer in the visual backbone. Note that unlike [79, 80], time steps here correspond to spatio-temporal features from each video segments and not a single frame representation. Then the Timeception layer splits the input features into several groups, and temporally convolves each group as hown in fig. 2.3. It further comprises of multi-scale temporal only convolutions to tolerate variety of temporal extents in complex actions. Timeception makes use of grouped convolutions and channel shuffling to learn cross-channel correlations efficiently rather than 1×1 spatial convolutions. Thus, the effectiveness of this layer substantiates the need of dense temporal information in videos especially for recognizing ADL with subtle motion. However, timeception with dedicated kernels for each time-scale does not take into

account the subtle temporal transitions in a video segment but focus on temporal transitions. Recently, Wu et al. [84] have proposed Long-term feature bank model to extract information over the entire span of a video to augment state-of-the-art video models that otherwise would only view short clips of 2-5 seconds. The idea is based on two concepts: (1) a long-term feature bank that intuitively acts as a 'memory' of what happened during the entire video - they compute this as Region of Interest (RoI) features from detections at regularly sampled time steps; and (2) a feature bank operator (FBO) that computes interactions between the short-term RoI features (describing what actors are doing now) and the long-term features. The interactions may be computed through an attentional mechanism, such as a non-local block [85], or by feature pooling and concatenation. This model focuses only on addressing the challenge of long temporal relationships within a video. But the model design with sampling few frames out of many future frames, does not take into account the fine temporal transitions.

Thus to sum up the studies mostly based on 3D CNNs for short and long actions, we point out that: (i) video based networks with millions of training parameters must be designed carefully to maintain an optimal trade-off between efficiency and accuracy, (ii) strategies like expanding 2D networks, pre-training the video networks on large diversified dataset like *Kinetics*, applying separable convolutions (for space and time), applying group convolutions (for channel interactions) are instrumental for effective action recognition, and finally (iii) extra processing with blocks like non-local and timeception or Long Feature Banks are crucial for modeling long-term relationships in videos. In table 2.1, we present a comparative study of the different popular video networks with their performance on Kinetics-400 dataset, GFLOPs, and the number of training parameters. We also indicate if any prior pre-training on ImageNet [20] is done. This comparison along with their performance on Kinetics is important to follow as these provide us an intuition of what video backbone we should opt for our classification task.

In addition, none of these RGB based approaches address the challenge of view invariant action recognition. This is due to the incapability of the current convolutional architectures to be view adaptive.

Approaches based on 3D Poses

With the emergence of deep learning, the evolution of 3D poses are exploited using sequential networks like LSTMs [27]. Figure 2.4 illustrates the configuration of body joints for a given 3D pose acquired with Kinect V2.

(A) RNN based approaches - Similar to CNN + RNN based approaches using RGB cues, 3D poses are fed to RNNs. Such methods take the temporal evolution of the 3D poses into account and thus discriminate actions even with similar appearance but dissimilar

Table 2.1: Comparative study of different video networks. We present their Top-1 accuracy on Kinetics, GFLOPs, and the number of training parameters. NL stands for NonLocal and R50 or R101 stands for 3D ResNet 50/101.

Model	pre-training ImageNet	Top-1 Accuracy (%)	GFLOPs	Param
I3D [3]	✓	71.1	108	12M
Two-stream I3D [3]	✓	75.7	216	25M
Two-stream S3D [74]	✓	77.2	143	23.1M
Nonlocal R50 [85]	✓	76.5	282	35.3M
Nonlocal R101 [85]	✓	77.7	359	54.3M
Two-stream I3D [3]	×	71.6	216	25M
CSN [77]	×	77.8	109	32.8M
SlowFast (R101) [73]	×	77.9	106	53.7M
SlowFast (R101+ NL) [73]	×	79.8	234	59.9M
X3D [78]	×	80.4	194.1	20.3M

motion. Differential RNN [86] added a new gating mechanism to the traditional LSTM to extract the derivatives of internal state (DoS). The derived DoS is fed to the LSTM gates to learn salient dynamic patterns in 3D skeleton data. HBRNN-L [87] have proposed a multilayer RNN framework for action recognition on a hierarchy of skeleton-based inputs. At the first layer, each sub-network received the inputs from one body part. On next layers, the combined hidden representation of previous layers are fed as inputs in a hierarchical combination of body parts. The work of [88] introduced an internal dropout mechanism applied to LSTM gates for stronger regularization in the RNN-based 3D action learning network. To further regularize the learning, a co-occurrence inducing norm was added to the network's cost function which enforced the learning to discover the groups of co-occurring and discriminative joints for better action recognition. Shahroudy et al. [5] have proposed part-aware LSTM which has part-based memory sub-cells dedicated to every human body part for action classification. Although LSTM networks are designed to explore the long-term temporal dependency problem, it is still difficult for LSTM to memorize the information of the entire sequence with many timesteps [89]. In addition, it is also difficult to construct deep LSTM to extract high-level features [90].

(B) CNN based approaches - Thus, authors in [91, 91, 92] have proposed another framework to represent 3D poses as pseudo image. This enables the framework to leverage the successful image classification CNNs for action classification. Ke et al. [6] transformed a skeleton sequence into three clips of gray-scale images as illustrated in fig. 2.5. Each clip consists of four images, which encode the spatial relationship between the joints by inserting reference joints into the arranged joint chains. They employed the pre-trained VGG19 model to extract image features and applied the temporal mean pooling to represent an

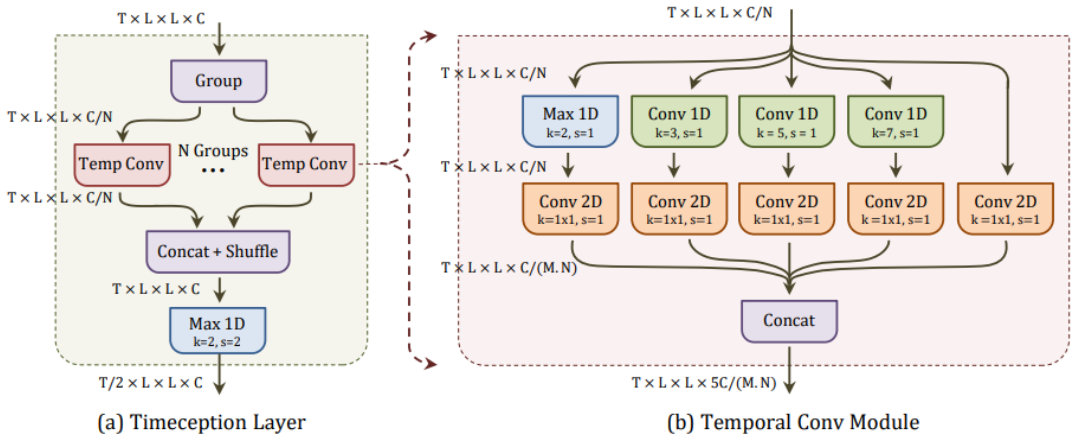


Figure 2.3: The core component of Timeception on left. At right, a zoom of temporal convolution operation applied in a Timeception layer. Figure extracted from [4].

action. Similarly, [91, 92] first transforms the skeletal time-series data into an appropriate pseudo-image representation. Then, they make use of the powerful image classifiers like VGG [19] to classify the actions.

(C) GCN based approaches - With advancements in CNN architectures, dedicated CNNs for graph based data, namely Graph Convolution Networks (GCNs), are introduced in [93]. On this note, 3D poses can be considered as a graph with joints as vertexes and bones as edges. In [94, 95], the key idea is to fed a GCN with a graph representation of a skeleton frame. A variant of GCN with its kernels convolving around the neighboring nodes computes a high-dimensional output. These per-frame output features are aggregated either by a conventional 2D CNN as discussed above (for [94]) or by a temporal convolutional network (for [95]). Such aggregation is followed by action classification. Yan et al. [96] first apply GCNs to model the skeleton data. They construct a spatial graph based on the natural connections of joints in the human body and add the temporal edges between corresponding joints in consecutive frames. A distance-based sampling function is proposed for constructing the graph convolutional layer, which is then employed as a basic module to build the final spatio-temporal graph convolutional network (ST-GCN). However, the graph employed in ST-GCN is heuristically predefined and represents only the physical structure of the human body. Thus it is not guaranteed to be optimal for the action recognition task. For example, the relationship between the two hands is important for recognizing classes such as *clapping* and *reading*. However, it is difficult for ST-GCN to capture the dependency between the two hands since they are located far away from each other in the predefined human-body-based graphs. To solve the above problems, a novel Adaptive Graph Convolutional network is proposed in [97]. It parameterizes two types of

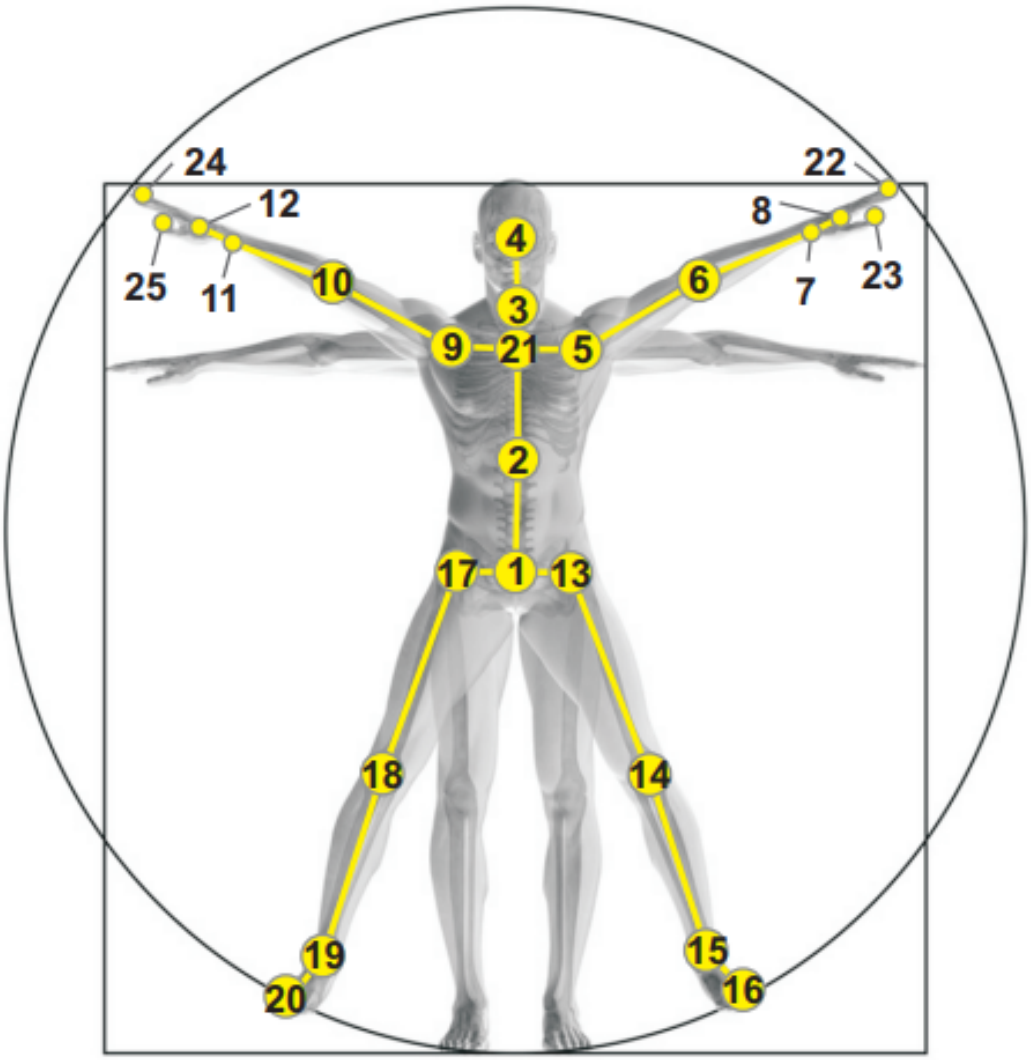


Figure 2.4: An example of 3D pose joint configuration extracted from the figure in [5]. The labels of the joints are: 1-base of the spine 2-middle of the spine 3-neck 4-head 5-left shoulder 6-left elbow 7-left wrist 8- left hand 9-right shoulder 10-right elbow 11-right wrist 12- right hand 13-left hip 14-left knee 15-left ankle 16-left foot 17- right hip 18-right knee 19-right ankle 20-right foot 21-spine 22- tip of the left hand 23-left thumb 24-tip of the right hand 25- right thumb

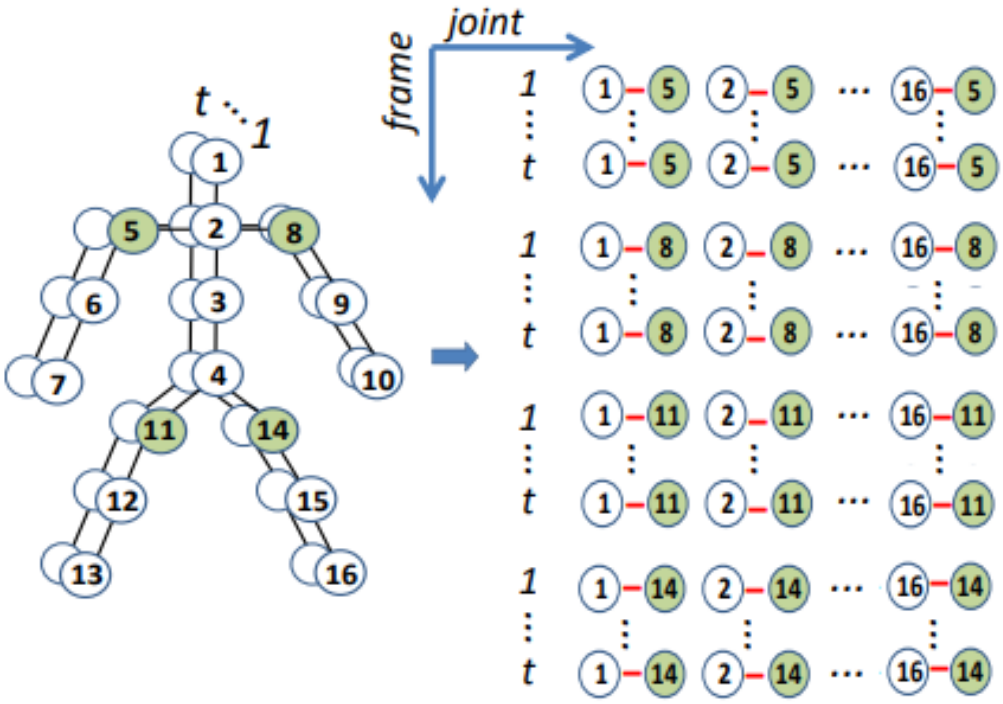


Figure 2.5: Clip Generation of a skeleton sequence in [6] (Figure from [6]). The skeleton joints of each frame are first arranged as a chain by concatenating the joints of each body part (i.e., 1-2-3-...-16). Four reference joints shown in green (i.e., left shoulder 5, right shoulder 8, left hip 11 and right hip 14) are then respectively used to compute relative positions of the other joints to incorporate different spatial relationships between the joints. Consequently, four 2D arrays are obtained by combining the relative positions of all the frames of the skeleton sequence. The relative position of each joint in the 2D arrays is described with cylindrical coordinates. The four 2D arrays corresponding to the same channel of the coordinates are transformed to four gray images and as a clip. Thus three clips are generated from the three channels of the cylindrical coordinates of the four 2D arrays.

graphs, the structure of which are trained and updated jointly with convolutional parameters of the model. One type is a global graph, which represents the common pattern for all the data. Another type is an individual graph, which represents the unique pattern for each data. Both of the two types of graphs are optimized individually for different layers, which can better fit the hierarchical structure of the model. This data-driven method increases the flexibility of the model for graph construction and brings more generality to adapt to various data samples.

Different from [95, 94], ST-GCN and AGCN perform graph convolutions across space as well as time using several partition strategies. Note that in spite of the differences in these skeleton based action recognition methods using GCNs, he underlying graph convolutional operation remains the same across all the methods. The output f_{out} of a GCN whose input is a graph $\mathcal{G}$ with adjacency matrix A, is computed by

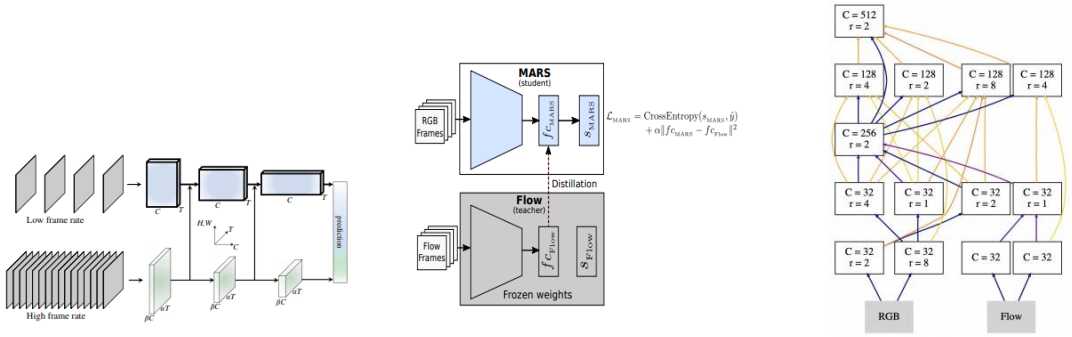
$$f_{out} = D^{-\frac{1}{2}}(A + I)D^{-\frac{1}{2}}\mathcal{G}W, \tag{2.1}$$

where W is the weight matrix and D is the diagonal degree matrix with $D_{ii} = \Sigma_j(A_{ij} + I_{ij})$ its diagonal elements. Compared to sequential networks and pseudo image based methods [96, 94, 95], graph-based methods make use of the spatial configuration of the human body joints and thus, are more effective.

Thus, over the recent years, with the introduction of several large-scale RGB-D datasets for action recognition, 3D skeleton action recognition have made remarkable improvements in action classification. Actions with strong human motion, similar appearance and dissimilar motion, and discriminating posture are now better classified with respect to RGB based methods. However, the skeleton based action recognition lacks in encoding the appearance information, especially actions with object interactions, which is also critical for ADL recognition. Thus, the intuitive direction of research is towards multi-modal video representation in order to leverage the pros of RGB and 3D poses.

2.4 Discriminating Fine-grained Actions & Similar Actions

Unlike the problem of object detection, the diversity of actions in generic videos varies a lot. Recognizing fine-grained actions and discriminating similar actions are the intrinsic challenges in ADL. Previous methods have attempted to address these challenges. One approach is based on combining the advantages of privileged modalities in order to make use of their complementary discriminative power as discussed below.



(a) Lateral connections in slow-fast network (b) Teacher-student network in MARS NAS in AssembleNet

Figure 2.6: Recent approaches for combining RGB and optical flow cues in Neural Networks. (a) Late fusion in two-stream models [3], (b) Teacher-student network in MARS [7], and (c) NAS in AssembleNet [8]. Figures extracted from [3, 7, 8] respectively.

2.4.1 Multi-modal Representation

We have already seen how two-stream [1, 56, 45] fusion that learns separate features from the optical flow and RGB modalities, outperforms single modality approaches. Recent studies have shown substantial improvements in using new strategies for fusing RGB and optical flow features. Some of these approaches are illustrated in fig. 2.6. (i) One approach is **Feature-level or late fusion** between the streams processing both the modalities. In [1], Simonyan et al. have used late fusion of scores obtained from RGB and optical flow streams. These streams are trained independently. This fusion configuration is quite popular due to its effectiveness and straightforward implementation. Further in [56], feature-level fusion between both the modalities have been proposed to learn joint representation for action classification. (ii) Another approach is learning a teacher-student network [7] that mimics the motion stream at inference time without actually computing them (optical flow). In this network, a teacher network - a motion stream is trained independently for the end task which is action classification. Then the RGB stream is trained for action classification along with mimicking the features learned by the motion stream. This is accomplished by a distillation loss that minimizes the euclidean distance between the features learned by them. The experiments shows that MARS at test time is more effective than the individual streams even by minimizing the test time considerably. (iii) The third approach is very popular nowadays, i.e. searching a neural architecture to fuse the RGB and optical flow modalities. In [8], Ryoo et al. have proposed a Neural Search Architecture (NAS) to combine the RGB and Optical flow stream. This search mechanisms answers many interesting questions like: how should we combine RGB and optical flow (by concatenation or summation)? At which layers these modalities should be combined?

The appearance and motion features are complementary and their fusion utilizes the correlation between features from both modalities. This makes them more discriminative in common feature space rather than their individual feature space. However, optical flow modality modeling short-term motion is less effective for ADL which exhibits actions with subtle motion. Moreover, with the introduction of 3D convolution [71], now some inherent optical flow information is extracted along with the appearance information. On the other hand, evolution of pose estimation algorithms as well as pose based approaches for action recognition have made huge improvement in the recent years. Especially for videos captured in monitoring views, to some extent skeleton based action recognition have mitigated the drawbacks of RGB based approaches like robustness to illumination and view. As a result, going towards multi-modal video representation for ADL not only using RGB and optical flow but also 3D poses is an obvious direction of research. Note that it is not straightforward to combine RGB and 3D poses as it is possible for that of RGB and optical flow. The reason is that both the modalities have so different characteristics in terms of representation and dimension. This limits them from blending at any stage in a network.

Below, we briefly study few approaches leveraging multiple modalities for action recognition. These approaches exclude the ones involving only RGB and optical flow (as already discussed above).

The use of different modalities via a Markov chaining is proposed in [98]. Zolfaghari et al. have proposed a chained multi-stream network in [98] exploiting pose, appearance and motion, fusing them in order to have a sequential refinement of action labels. But the drawback of such chaining models includes mutual dependence of the different cues used. Rahmani et al. [99] have proposed an approach that fuses depth image and skeleton data. Features from multi-modal information are concatenated before sending to a fully connected layer for classification. In [100], the RGB cue and the skeleton evolution are fused to make better use of both the spatial information and the temporal information. Shahroudy et al. [101] have proposed a new hierarchical bag-of-words feature fusion technique based on multi-view structured sparsity learning to fuse atomic features from RGB and skeletons for the task of action recognition. With the gains of the distillation loss for knowledge representation [102], Luo et al. [103] have proposed a graph distillation that incorporates rich privileged information from a large-scale multi-modal dataset in the source domain. Whereas in the target domain, they have limited data and a subset of the modalities during training, and only one modality during testing. Their graph distillation method learns a dynamic distillation across multiple modalities for action detection in multi-modal videos. Recently, Perez-Rua et al. [104], have proposed a novel multi-modal search space and exploration algorithm to solve the task of action recognition in an efficient yet effective manner. The proposed search space is constrained in such a way that

it allows convoluted architectures to take place while also containing the complexity of the problem to reasonable levels. However, these searching algorithms learns a network architecture based on a particular data distribution. Thus, these networks are susceptible to domain shifts and thus, cannot be generalized for a large diversity of actions.

The existing studies on action recognition show the diversity of approaches and information used. This gives us a hint of different cues for modeling the actions along with eliminating the noise introduced because of interference among the cues. Understanding the pose, appearance and motion of a subject performing an action is critical for action recognition. Thus, we focus on combining the pros of different cues with a learning strategy optimized for modeling ADL in chapter 3.

It is desirable to fuse multi-modal information into an integrated set of discriminative features. But, these modalities are heterogeneous and they must be processed by asymmetric networks to show their effectiveness. This limits their performance in simple multi-modal fusion strategy [101, 100, 103]. As a consequence, many pose driven attention mechanisms have been proposed to guide the RGB cues for action recognition.

2.4.2 Attention Mechanisms

Attention is, to some extent, motivated by how we pay visual attention to different regions of an image or words in one sentence. Human visual attention allows us to focus on a certain region with "high resolution" while perceiving the surrounding image in "low resolution", and then adjust the focal point or do the inference accordingly. This phenomenon is known as attention mechanism in artificial intelligence. In a nutshell, attention in deep learning can be broadly interpreted as a vector of importance weights: in order to predict or infer one element, such as a pixel in an image or a word in a sentence, we estimate using the attention vector how meaningful it is. Recently, two classes of attention have emerged, *hard* and *soft* attention.

Hard vs Soft attention -*Hard attention* is the principle of taking hard decisions while choosing parts of the input data. This selection reduces the task (object recognition) complexity as the Region of Interest (RoI) can be placed in the center of the fixation and irrelevant features of the visual environment outside the fixed region are naturally ignored. For instance, Mnih et al. [105] have proposed a visual hard-attention for image classification. They train RNN to select an appropriate local region to be focused on. The feature extraction from local region by a glimpse sensor (hard cropping of RoI and CNN feature extraction) is guided by an agent controller that receives an award for taking a correct decision. The parameters deciding where to look next are learned using Reinforcement Learning. Similar hard-attention mechanism has been used in multiple object recognition, object localization and saliency map generation as quoted in [106]. Another example

is [107], where hard-attention is used but for action detection. A set of parameters approximates which frame to observe next and when to emit an action prediction. Similar to [105], these parameters are learned from the hidden state vectors of an RNN across its time steps. All these methods [107] are stochastic algorithms that cannot be learned easily through gradient descent and backpropagation, preventing a global optimization of the network. Whereas, using attention models in the recent deep networks require a differentiable loss in order to train the network in an end-to-end manner by standard back-propagation.

On the other hand, *Soft attention* takes the entire input (image or video) and then soft-weights the RoI as per their relevance for the end task. For instance, Wang et al. [108] weighs each part of the RoI to compute normalized mask features, which is further combined with the original convolutional features to generate attention aware features. Jaderbaerg et al. [109] have proposed a differentiable module called Spatial Transformer Network (STN) that can be placed at any stage in a neural network. STN with its components -localization network, grid generator and sampler performs an affine transformation on the RoI of the input image. Another class of soft-attention is self-attention that has been shown to be very useful in machine reading, abstractive summarization, or image description generation [110, 111]. This architecture has been proposed in [110] for seq2seq tasks like language translation, to replace traditional recurrent models. The main idea of the original architecture is to compute self-attention by comparing a feature to all other features in the sequence. This is carried out efficiently by not using the original features directly. Instead, features are first mapped to a query (Q) and memory (key and value, K & V) embedding using linear projections, where typically the query and keys are lower dimensional. The output for the query is computed as an attention weighted sum of values V , with the attention weights obtained from the product of the query Q with keys K. In practice, the query here is the word being translated, and the keys and values are linear projections of the input sequence and the output sequence generated so far. Next, we detail, how these soft attention mechanisms have been applied for the task of action recognition.

Attention mechanisms for action recognition - Sharma et al. [112] have proposed a recurrent mechanism for action recognition from RGB data, which assigns weights to different parts of a convolutional features map along time. Instead of using RGB images, authors in [113] use 3D joints with spatio-temporal attention mechanism for action recognition. They have proposed an end-to-end network with three RNN networks, one for classification, one to selectively focus on the discriminative joints of the skeleton (spatial attention), and one for assigning weights to the key sequences (temporal attention). Baradel et al. [114] have proposed a similar technique as [54] replacing the input of classification RNN with patches around human hands. Their attention model soft assigns weights to

the RGB hand patches taking advantage of articulated pose. Later in [115], they have extended [114] by replacing the RNN based attention sub-networks with CNN ones. Most of these methods provide spatial attention on the input spatial features extracted from 2D ConvNets fed to the RNN and temporal attention on the output latent spatio-temporal features. Recently, Baradel et al. [106] have proposed a visual attention module that learns to predict glimpse sequences corresponding to the RoI in the image along with tracking them over time using a set of trackers, which are soft-assigned within an external memory. In short, their method includes selecting the glimpses from spatio-temporal features and soft-assigning them to multiple recurrent networks (called workers). However, there is no tight coupling between extracting the features and learning the attention, hence fails to globally optimize their proposed network.

With the success of 3D CNNs, recently several attention mechanisms have been proposed on top of the aforementioned 3D ConvNets to extract salient spatio-temporal patterns. For instance, Wang et al. [85] have proposed a non-local module on top of I3D which computes the attention of each pixel as a weighted sum of the features of all pixels in the space-time volume. This non-local module is widely used with other action recognition networks like 3D ResNet [76], slow-fast [73] and GCN [83]. This module though effective for the classification of actions in internet videos relies too much on the appearance of the actions, i.e., pixel position within the space-time volume. As a consequence, it fails to disambiguate ADL with similar motion.

Attention mechanisms have been a very popular research topic recently.However, its application for recognition of ADL has not been explored much. In this thesis, we make an attempt to explore this research direction. In table 2.2, we present a comparative study of the attention based methods we have described above.

2.5 Datasets

In this section, we provide a literature survey of the public datasets available for the task of action classification. We provide an analysis of the various datasets, some of their intrinsic challenges and a description of the datasets we use for validating our proposed algorithms in this thesis. Below, in this section, we analyze the different datasets to answer an important question - **How far are we from recognizing actions in the wild?**

To deploy action-recognition algorithms in real-world applications, a validation on videos replicating real-world challenges is crucial. To well compare the challenges of currently-available datasets, we identify in the following a set of indicators on how well each of these datasets addresses the main real-world challenges.

- **Context**: The context is the background information of the video. Some action

Methods	P	RGB	SA	TA	Mechanism
visual attention [112]	×	✓	✓	×	One 2D CNN for extracting image features + One LSTM classification network that also computes the spatial attention weights for the next time step.
STA [113]	✓	×	✓	✓	Three RNNs: One for classification, one for computing spatial attention weights to choose the discriminative joints, and the last one for computing temporal attention weights to select key frames.
STA hands [114]	✓	✓	✓	✓	One 2D CNN for extracting hand features + Three RNNs. One for classification, another for spatial attention weights to choose the relevant hands, and the last one for computing temporal attention weights to select key frames.
improved STA hands [115]	✓	✓	✓	✓	One 2D CNN for extracting hand features + One RNN + Two CNNs. RNN for classification, one CNN for spatial attention weights to choose the relevant hands, and another CNN for computing temporal attention weights to select key frames.
Glimpse Cloud [106]	○	✓	✓	✓	One inflated 3D CNN for extracting RGB features + $C+1$ RNNs. One RNN + STN for spatial attention C RNNs for distributed tracking and recognition, and external memory assigns soft-weight to different workers.
NonLocal [85]	×	✓	✓	✓	A Layer used on top of any existing 3D CNNs. A self-attention mechanism to compute weights for spatio-temporal feature map.

Table 2.2: Comparative study of different attention mechanisms for action recognition. We indicate the modalities used by these methods: 3D poses (P) and RGB. SA and TA indicates spatial and temporal attention repectively.

Table 2.3: Comparative study highlighting the challenges in real-world setting datasets. Along with indicating the defined challenges, we also present the view type, scene information (indoor/outdoor) and the type of videos (web based, ADL, kitchen and so on) in the datasets. Here, we denote MSRDailyActivity3D [9] dataset by MSR ADL.

Dataset	Context	Duration variation	CV challenge	Composite actions	View Type	Spont. acting	Camera framing	Fine-grained actions	Type
ACTEV/VIRAT [116]	free	Medium	Yes	No	Monitoring	Medium	Low	No	Surveillance
SVW [117]	biased	Low	No	No	Shooting	High	High	No	Sport
HMDB [23]	biased	Low	No	No	Shooting	Medium	High	No	Youtube
Kinetics [3]	biased	Low	No	No	Shooting	Medium	High	No	Youtube
AVA [118]	biased	Low	No	No	Shooting	Medium	High	No	Movies
EPIC-KITCHENS [119]	free	High	No	Yes	Egocentric	High	High	Yes	Kitchen
Something^2 [120]	free	Low	No	No	Shooting	Low	High	Yes	Object int.
MPII Cooking2 [121]	free	High	Yes	Yes	Monitoring	Medium	Medium	Yes	Cooking
DAHLIA [122]	free	High	Yes	No	Monitoring	Medium	Medium	No	Kitchen
CAD-60 [123]	free	Low	No	No	Shooting	Low	High	No	ADL
CAD-120 [13]	free	Low	No	No	Shooting	Low	High	No	ADL
MSR ADL [9]	free	Low	No	No	Shooting	Low	High	No	ADL
NUCLA [124]	free	Low	Yes	No	Shooting	Low	High	No	Object int.
NTU-60 [5]	free	Low	Yes	No	Monitoring	Low	High	No	ADL
NTU-120 [125]	free	Low	Yes	No	Monitoring	Low	High	No	ADL
Charades [25]	free	Low	Yes	Yes	Shooting	Low	High	Yes	ADL
Smarthome	free	High	Yes	Yes	Monitoring	High	Low	Yes	ADL

datasets feature a rich variety of contextual information (context biased). In some cases, the contextual information is so rich that it is sufficient on its own to recognize activities. For instance, in UCF and kinetics, processing the part of the frames around the human is often sufficient to recognize the activities. On the other hand, in datasets recorded in environments with similar backgrounds (context free), the contextual information is lower and thus cannot be used on its own for action recognition. This is true, for instance, for datasets recorded indoor such as Smarthome and NTU RGB+D [5].

- **Spontaneous acting**: This denotes whether the subjects tend to overstate movements following a guided script (low spontaneous acting). Subjects acting freely a loose script tend to perform activities spontaneously in a natural way (high spontaneous acting).

- **Camera framing**: When videos are recorded by a cameraman, subjects mostly appear in the middle of the image and facing the camera (high camera framing). On the other hand, when videos are recorded automatically by a monitoring system using fixed cameras, subjects can be often occluded or partially outside the field of view (low camera framing).

- **Cross-view challenge** (CV challenge): In real-world applications, a scene may be recorded from multiple angles, and from some of these angles the subject of the action, and/or the object used to perform it, might be occluded. Action recognition algorithms should be robust to multi-view scenarios. We therefore assign a high multi-view mark to datasets recorded with multiple cameras at the same time; low multi-view mark to datasets recorded by a single camera.

- **Duration variation**: The duration of activities may vary greatly both inter-class and intra-class. A high variation of duration is more challenging and more representative of the real-world. We assign high duration variation to datasets in which the length of video samples varies by more than 1 minute both intra-class and inter-class; low duration variation otherwise.

- **Composite actions**: Some complex actions can be split into sub-actions (eg., cooking is composed of *cutting*, *stirring*, *using stove*, etc.). Recognizing both coarse and fine-grained actions is often needed. This indicator simply states whether the dataset contains composite activities.

- **Fine-grained actions**: Recognizing both coarse and fine-grained activities is often needed for real-world applications. For example, *drinking* is a coarse action with fine-grained details of the object involved in it, say *can, cup,* or *bottle*.

Table 2.3 shows the comparison of the publicly available real-world action datasets based on the above indicators.

ADL are usually carried out indoor. NTU-RGB+D-120 [125], an extended version of NTU-RGB+D-60 [5], is the largest dataset for ADL, comprising more than 114K samples with multi-view settings. However, NTU-RGB+D-120 was recorded in laboratory rooms and the actions are performed by students with strict guidance. This results in guided actions and actors facing the cameras. EPIC-KITCHENS [119] contains only egocentric videos, showing mostly the hands of the person, which are very different from third view videos. MPII Cooking 2 [121] is an ADL dataset recorded for cooking recipes in an equipped kitchen. The dataset has 8 camera views, with composite action labels. This dataset focuses on one cooking place, thus limiting the environment diversity and the number of action classes. Charades [25] was recorded by hundreds of people in their own home with very fine-grained action labels. However, self-recorded actions are very short (10 seconds/action), often not natural, and always performed in front of the camera. Hence, current ADL datasets address only partially the challenges of real-world scenarios. Whereas, Toyota Smarthome: a dataset recorded in a semi-controlled environment and real-world settings, provides a wide diversity of ADL. Here we summarize the key characteristics of Smarthome: (1) The dataset was recorded in a real apartment using 7 Kinect sensors [126] monitoring 3 scenes: dining room, living room and kitchen (2) Subjects were recorded for an entire day, during which they performed spontaneous daily actions without any script. (3) Action duration ranges from a couple of seconds to a few minutes. (4) Because the camera positions were fixed, the subject resolution varies considerably between videos. (5) Sub-action labels are available for composite activities such as *cooking*, *make coffee*, etc. The annotations of Smarthome dataset include different labels assigned to the same action performed using different objects (eg., *drink from cup*, *drink from can*, and *drink from bottle*). Thus, Toyota Smarthome posses numerous real-world challenges that the other public datasets are missing mostly due to their strict script.

Now, we quantify the comparison of datasets providing their statistics. To date, there are more than 50 human action recognition datasets. Although each one of them has unique, beneficial characteristics for the evaluation of action recognition algorithms, they have also limitations as discussed above. Table 2.4 lists the most popular public ADL datasets to our knowledge with their key features. In terms of dataset size (i.e., number of video samples and action classes). NTU RGB+D-120 is the largest dataset with 114K videos and Toyota Smarthome is the third largest dataset with 4.2M frames at 20 *fps*.

From the above research analysis and quantitative study, we point out that each dataset has its own intrinsic challenges. But, based on our motivation which is one more step towards real-word action recognition, we make choices for selecting evaluation datasets based on both dataset complexity as well as its size for our experiments. For our first

Table 2.4: Comparison of different daily living action datasets for action recognition. The datasets are ordered according to year of their publication.

Dataset Name	#Subjects	#Action Class	#Videos	#Viewpoint	Modalities	#Year
CAD-60 [123]	4	12	60	1	RGB+D+Skeleton	2011
RGBD-HuDaAct [127]	30	13	1189	1	RGB+D	2011
MSRDailyActivity3D[9]	10	16	320	1	RGB+D+Skeleton	2012
Act4[128]	24	14	6844	4	RGB+D	2012
CAD-120 [13]	4	10+10	120	1	RGB+D+Skeleton	2013
DML-SmartAction[129]	16	12	932	2	RGB+D	2013
NUCLA[124]	10	10	1475	3	RGB+D+Skeleton	2014
Office Activity[130]	10	20	1180	3	RGB+D	2014
UWA3D Multiview II[131]	10	30	1075	5	RGB+D+Skeleton	2015
NTU RGB+D-60 [5]	40	60	56880	80	RGB+D+IR+Skeleton	2016
NTU RGB+D-120 [125]	106	120	114480	155	RGB+D+IR+Skeleton	2019
Toyota Smarthome [132]	18	31	16129	7	RGB+D+Skeleton	2019

contribution which is mostly scalable for small-scale dataset, we use three public datasets - **CAD-60, CAD-120** and **MSRDailyActivity3D**. For our contribution based on attention mechanisms, we validate our methods on four public datasets - **NTU RGB+D-60, NTU RGB+D-120, Toyota Smarthome** and **Northwestern UCLA**. Below, we briefly describe these datasets and their evaluation protocols which are used for validating the proposed methods in this thesis.

2.5.1 CAD-60

This dataset [123] contains the RGB frames, depth sequences and skeleton. The data was captured by Microsoft Kinect sensor. The data set consists of 12 actions performed by 4 subjects. The actions are performed in 5 different environments: office, kitchen, bedroom, bathroom, and living room. All together data-set contains 60 videos and some sample frames are illustrated in fig 2.7. The challenge in this dataset includes extremely small number of training samples. For evaluation, we follow one-subject out validation protocol as mentioned in [123].

2.5.2 CAD-120

This dataset [13] contains the RGB frames, depth sequences and skeleton. All together there are 120 videos available. Actions are performed by 4 different subjects performing 10 high-level activities. Each high-level action was performed three times with different objects. The challenges in this dataset includes - (i) the activities vary from subject to subject significantly in terms of length; (ii) low inter-class variation with similar actions like *stacking* and *unstacking objects*. Some sample frames from this dataset are illustrated in fig. 2.8. For evaluation, we follow one-subject out validation protocol as mentioned in [133].

2.5.3 MSRDailyActivity3D

This dataset [9] consists of 16 actions such as: *drink, eat, read book, call cellphone, write on a paper, use laptop, use vacuum cleaner, cheer up, sit still, toss paper, play game, lie down on sofa, walk, play guitar, stand up, sit down*. Each action is performed by 10 subjects, and each subject performs each action in standing and sitting position, what adds an additional intra-class variation. In total, the dataset contains 320 videos recorded with 640×480 pixels spatial resolution. RGB frames, depth-map and skeleton are available for all videos. Some sample frames from this dataset are illustrated in fig. 2.9. For evaluation, we follow one-subject out validation protocol as mentioned in [133].

2.5.4 NTU RGB+D-60

NTU RGB+D-60, hereafter NTU-60, is acquired with a Kinect v2 camera and consists of 56880 video samples with 60 action classes. These actions are divided into three major groups: 40 daily actions (*drinking, eating, reading*, etc.), 9 health-related actions (*sneezing, staggering, falling down*, etc.), and 11 mutual actions (*punching, kicking, hugging*, etc.). Sample frames from this dataset are provided in fig. 2.10

The actions were performed by 40 subjects and recorded from 80 viewpoints. For each frame, the dataset provides RGB, depth and a 25-joint skeleton of each subject in the frame. For evaluation, we follow the two protocols proposed in [5]: cross-subject (CS) and cross-view (CV).

2.5.5 NTU RGB+D-120

NTU RGB+D-120, hereafter NTU-120, is a super-set of NTU-60 adding a lot of new similar actions. NTU-120 dataset contains 114k video clips of 106 distinct subjects performing 120 actions in a laboratory environment with 155 camera views. These 12 actions are divided into three major groups, including 82 daily actions (*eating, writing, sitting down, moving objects*, etc), 12 health-related actions (*blowing nose, vomiting, staggering, falling down*, etc), and 26 mutual actions (*handshaking, pushing, hitting, hugging*, etc). Sample frames from this dataset are illustrated in fig. 2.11. Compared to the preliminary version of the dataset, i.e. NTU-60, the characteristics of the newly added actions are:

- Fine-grained hand/finger motions
- Fine-grained object related individual actions
- Object-related mutual actions
- Different actions with similar posture patterns but with different motion speeds

- Different actions with similar body motions but with different objects involved
- Different actions with similar objects involved but with different body motions

For evaluation, we follow a cross-subject (CS_1) protocol and a cross-setting (CS_2) protocol proposed in [125]. The large amount of variation in subjects, views, and backgrounds makes it possible to have more sensible cross-subject and cross-setup evaluations for various 3D-based action analysis methods.

2.5.6 Toyota-Smarthome

Toyota Smarthome, hereafter Smarthome, has been recorded in an apartment equipped with 7 Kinect v1 cameras. It contains **31 daily living actions** and **18 subjects**. The subjects, senior people in the age range 60-80 years old, were aware of the recording but they were unaware of the purpose of the study. Each subject was recorded for 8 hours in one day starting from the morning until the afternoon. To ensure unbiased actions, no script was provided to the subjects. The obtained videos were analyzed and 31 different actions were annotated. Sample frames from each action are illustrated in fig. 2.14. The videos were clipped per action, resulting in a total of **16,115 video samples**. The dataset has a resolution of 640x480 and offers 3 modalities: RGB + Depth + 3D skeleton. The 3D skeleton joints were extracted from RGB using LCR-Net [134]. For privacy-preserving reasons, the face of the subjects is blurred using tinyface detection method [135].

Challenges. The dataset encompasses the challenges of recognizing natural and diverse actions. First, as subjects did not follow a script but rather performed typical daily activities, the number of samples for different activities is imbalanced (fig. 2.12). Second, the subject resolution varies considerably between videos and sometimes subjects are occluded. Third, the dataset consists of a rich variety of actions with different levels of complexity. Sub-activity labels are available for composite actions such as *cooking*, *make coffee*, etc. Fourth, the same action is assigned different labels when performed using different objects (for instance, *drink from cup, can, or bottle*). Finally, the duration of actions varies significantly: from a couple of seconds (for instance, sit down) to a few minutes (for instance, *read book* or *clean dishes*). All these challenges make the recognition of actions in Smarthome a difficult task. Figure 2.13 gives a visual overview of the dataset. For evaluation on this dataset, we follow cross-subject (CS) and cross-view (CV_2) protocols proposed in [132].

2.5.7 Northwestern-UCLA Multiview activity 3D Dataset

Northwestern-UCLA Multiview activity 3D Dataset, hereafter NUCLA, is acquired simultaneously by three Kinect v1 cameras. The dataset consists of 1194 video samples with

10 action classes. The activities were performed by 10 subjects, and recorded from three viewpoints as shown in fig. 2.15. We performed experiments on N-UCLA using the cross-view (CV) protocol proposed in [124]: we trained our model on samples from two camera views and tested on the samples from the remaining view. For instance, the notation $V^{3}_{1,2}$ indicates that we trained on samples from view 1 and 2, and tested on samples from view 3.

2.6 Conclusion

Thus in our literature survey we have observed that an effective action recognition algorithm relies on the effectiveness of its visual backbone. In this era, 3D CNNs are an obvious choice for action classification but with additional functionalities to address the challenges in ADL. On the other hand, multi-modal representation is also an essential ingredient for high performing action recognition framework, especially to address the view-invariance challenge. Hence, pose driven attention as proposed in [113, 115, 106] to leverage the 3D poses is an interesting research direction. Such mechanisms enables the network to learn spatio-temporal attention weights for action classification. But the questions that remains are: How to learn attention weights from poses and combine them with spatio-temporal feature maps? How to learn spatio-temporal attention weights jointly?

Moreover, in this chapter, we explore different datasets publicly available for evaluating our proposed action recognition frameworks. Different datasets have their own intrinsic challenges. Based on the challenges that we aim to address in this thesis, we filtered out seven public datasets for evaluating our frameworks.

Figure 2.7: A glimpse of the action classes in CAD60.

Figure 2.8: A glimpse of the action classes in CAD120.

Figure 2.9: A glimpse of the action classes in MSRDailyActivity3D.

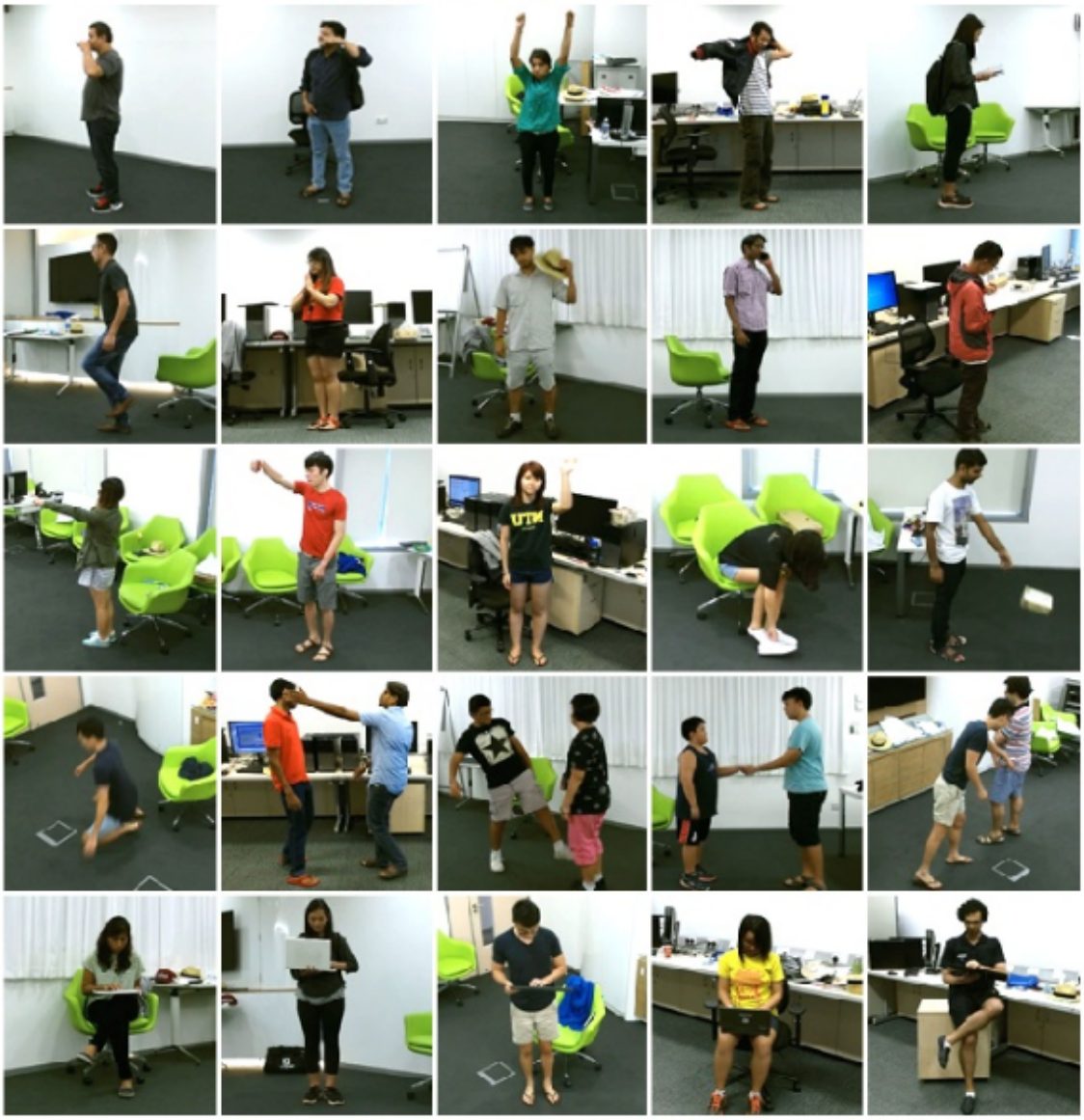


Figure 2.10: A glimpse of the action classes in NTU-60. First four rows show the variety in human subjects and camera views. Fifth row depicts the intra-class variation of the performances.

Figure 2.11: A glimpse of the action classes in NTU-120. First four rows show the variety in human subjects and camera views. Fifth row depicts the intra-class variation of the performances.

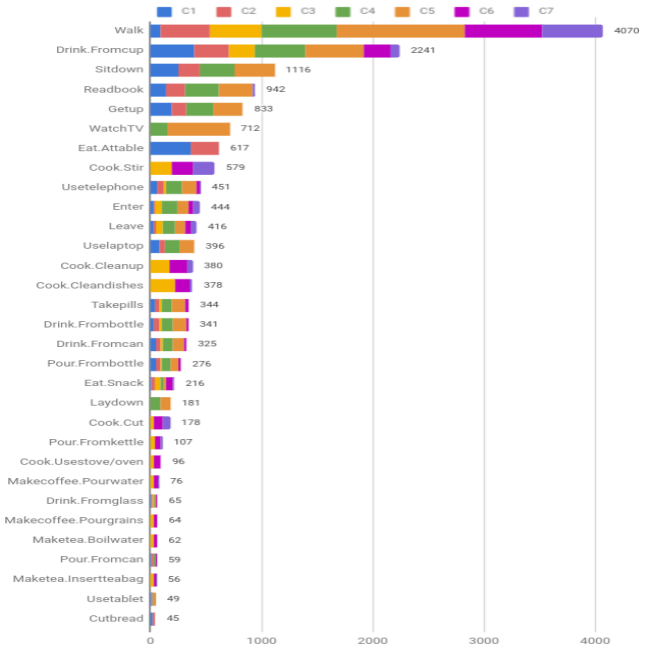


Figure 2.12: Number of video clips per action in Smarthome and the relative distribution across the different camera views. C1 to C7 represent 7 camera views. All the action classes have multiple camera views, ranging from 2 to 7.

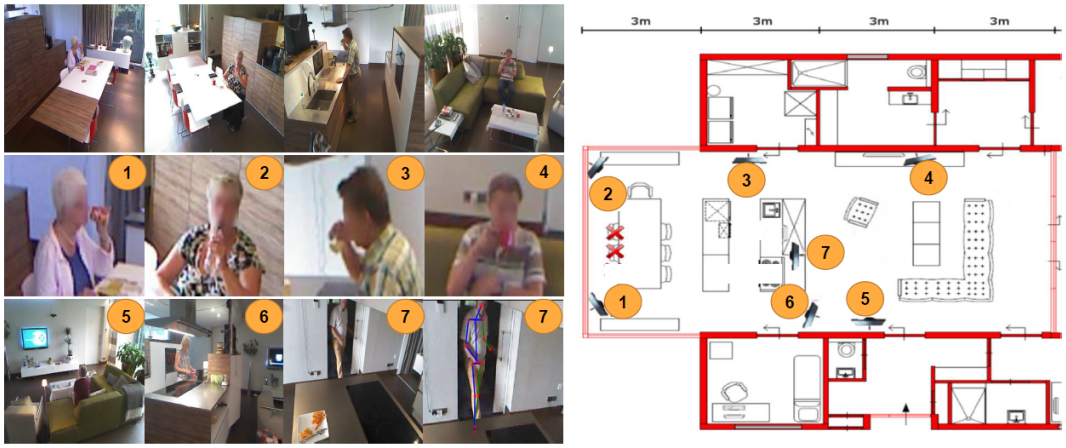


Figure 2.13: Sample frames from Smarthome dataset: 1-7 label at the right top corner respectively correspond to camera view 1, 2, 3, 4, 5, 6 and 7 as marked in the plan of the apartment on the right. Image from camera view (1) *Drink from can*, (2) *Drink from bottle*, (3) *Drink form glass* and (4) *Drink from cup* are all fine grained activities with a coarse label *drink*. Image from camera view (5) *Watch TV* and (6) *Insert tea bag* show activities with low camera framing and occlusion. Image with camera view (7) *Enter* illustrates the RGB image and the provided 3D skeleton.

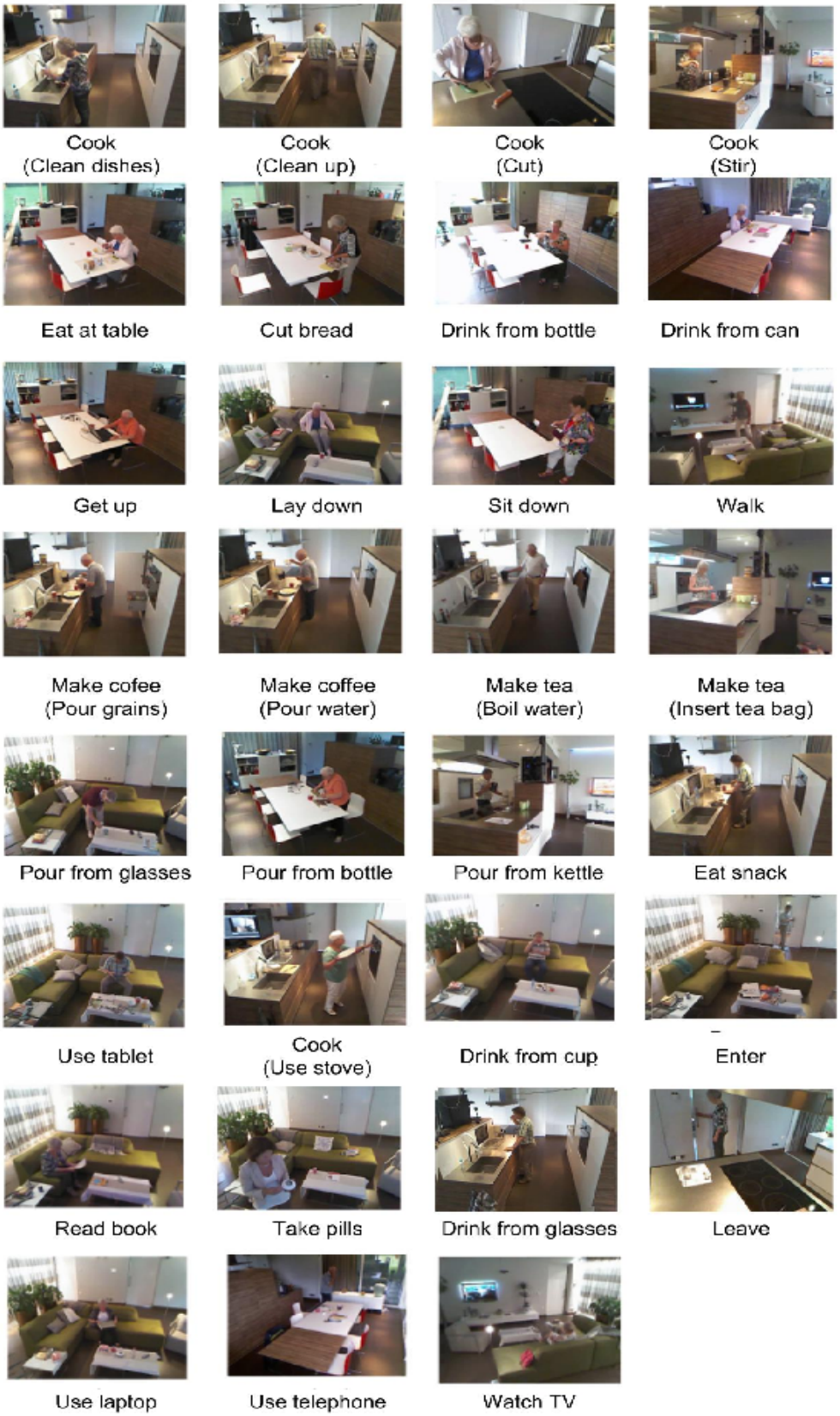


Figure 2.14: A glimpse of the 31 action classes in Smarthome.

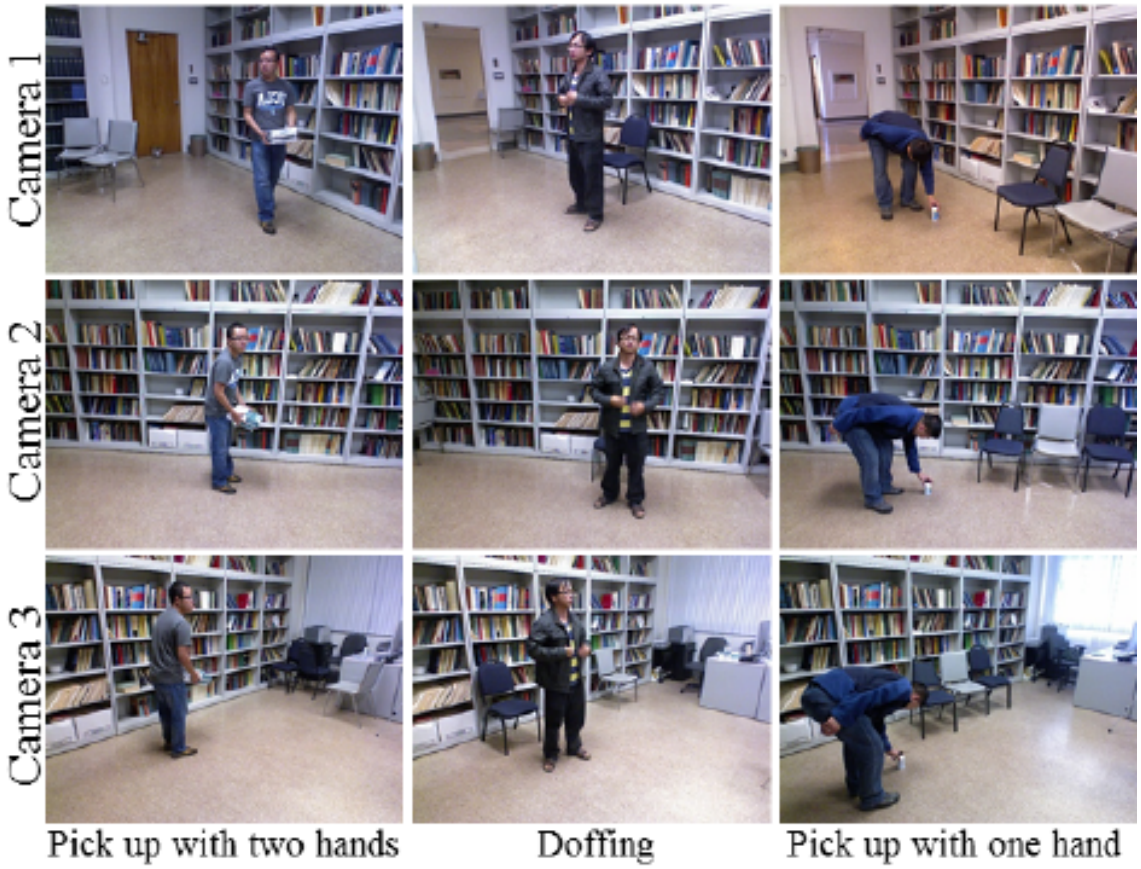


Figure 2.15: A glimpse of the action classes in North-western UCLA multi-view action dataset. The samples are taken from three actions captured across three different views.

Chapter 3

Multi-modal Video Representation for ADL

3.1 Introduction

As explained in the previous chapters, the major challenge in video analytics involves discriminative representation of temporal information in videos. We argue that multi-modal cues provide rich features to deal with the challenging scenarios in ADL. Consequently in this chapter, we propose a new architecture aiming to be effective and efficient for ADL recognition from RGB-D videos. The work presented in this chapter has been published as a full conference paper in AVSS 2018 [58] and as a special session industrial paper in MMM 2019 [136].

Over time, with the development of technology, features used for action recognition have taken new strides from computing simple SIFT features to dense trajectory [10] based features. Further, the emergence of deep learning, inspired the authors in [1] to use CNN features for modeling the appearance of actions in video sequences. On the other hand, introduction of cheap kinect sensors motivated the researchers to use 3 dimensional information of human poses to exploit the human skeleton geometry [52, 5]. Notably, approaches ranging from the handcrafted approaches using local features to approaches using DNN architectures on RGB images (CNNs) and 3D poses (RNNs) attempted to handle the temporal dimension in videos. Henceforth in this chapter, we describe our approach that leverages the advantages of using handcrafted features along with features from deep networks.

Compared to object detection, action recognition involves encoding object information involved in the action, pose information of the subject performing the action and their motion. Time is an important factor in this problem domain. Besides, the diversity of actions in ADL makes the problem of action recognition complex. This problem can be solved by

using different visual cues as in [1, 98] where each cue is responsible for modeling actions of specific categories. Current approaches using multi-modal video representation fail to achieve high performance rate and consistency in modeling the ADL. One of the primary reason behind their limitation [1, 56, 98, 103] is that these methods mostly rely on combination of appearance (RGB) and short-motion (optical flow) based features. Combination of these aforementioned cues performs effectively on videos from internet sources where the appearance and motion describing an action are prominent as well as distinct from one another. Clear visualization of motion in the absence of real-world challenges like occlusion or view-point changes makes the previous state-of-the-art methods in this domain, effective on datasets like UCF-101 [22] or HMDB-51 [23]. But what about scenarios where actions have subtle motion, occlusions, and poses similarities? Such scenarios are real challenges for ADL recognition.
In this chapter, we propose an answer to the following questions:

1. Which visual cue is effective for which action?
2. How these visual cues should be combined in order to mitigate the disadvantages of each cue?
3. How to disambiguate similar actions?

Consequently, we propose **a novel two-level fusion strategy** to combine the features in a common feature space to appropriately model the actions. We also address the challenge of recognizing similar actions in daily living activities by proposing a mechanism for **similar action discrimination**.

In the following, we present a broad view of the importance of each modalities for recognizing different action types in section 3.2, present our proposed architecture leveraging multiple modalities in section 3.3, present the experimental analysis on three public datasets in section 3.4, and finally conclude in section 3.5.

3.2 Feature Relevance depending on Action types

ADL consists of high variation of actions categories ranging from actions with similar poses like *stacking and unstacking objects, rubbing two hands and clapping,* actions with low motion like *typing keyboard, relaxing on couch,* and actions having temporal evolution of body dynamics like *walking, falling down* and so on. For optimizing action recognition it is important to establish a proper relationship between the nature of features and action categories to be modeled.
For ADL, features corresponding to mainly three types of visual cues are widely used in the literature, say

- **appearance** modeling the spatial layout varying with time in the action videos.

- **short-term motion** which is often computed through optical flow for instantaneous motion or based on short-term tracklets as in dense trajectories [10, 40].

- **pose based motion** obtained from modeling the temporal evolution of 3D articulated poses.

In fig. 3.1, we show a comparison of action recognition accuracy for some actions (*Drinking, Gaming, writing, playing guitar, talking on phone, writing on board* and some *random* actions) using short-term and pose based motion. For short-term motion, we use dense trajectories [10]. We ignore the HoG features in order to neglect appearance and have a fair comparison with pose based motion features from LSTM. In spite of both features modeling the motion, fig. 3.1 shows the complementary nature of both the features and their relevance with temporal dynamics of the subject performing action. On the other hand in table 3.1, we show the importance of appearance based features for action recognition. While computing Dense Trajectories, we compute the local features around the interest points for each image frame. More the number of interest points denote high motion within the neighboring frames. Thus, we average number of interest points of some actions to describe the motion of the actions. The 3^{rd} column in table 3.1 shows the difference in classification accuracy using appearance and short-term motion features (where $D = Accuracy$(Appearance)-$Accuracy$(Motion)).

Now, the question is how to combine the features to take advantage from each visual cue? The possible solutions are (i) early fusion of features where classification takes place in a common feature space, and (ii) late fusion of classifier scores where independent classifiers are trained on the features separately.
Early fusion is preferred when all the features characterize the actions because the correlation between them materialize in a precise level. If not, it is better to compute late fusion in order to balance the feature weights at the latest stage. So, we propose a two level fusion strategy to combine the relevant features at the most appropriate level depending on action categories.

3.3 Proposed Architecture for Action Recognition

In this section, we describe feature extraction of different cues followed by a two level fusion strategy and then, we explain how to disambiguate similar actions in ADL. Fig. 3.5 shows the overall architecture for the testing phase.

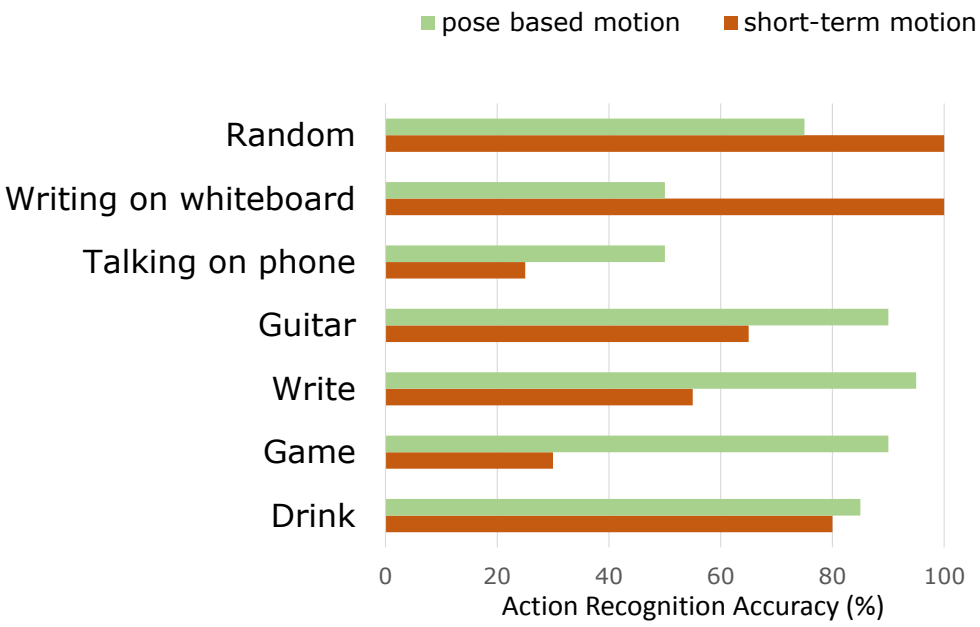


Figure 3.1: Comparison of action recognition accuracy on MSRDailyActivity3D [9] using short-term and pose based motion. Short-term motion is modeled by dense trajectories [10] and pose based motion is modeled by LSTM [11].

Table 3.1: Comparison of action recognition on CAD-120 [13] and MSRDailyActivity3D [9] based on appearance and motion. The table shows average number of detected features using Dense Trajectories [10] taken from [14]. Third Column represent the action classification accuracy improvement with appearance. This column shows that the appearance dominates the classification of action with subtle motion.

Action	Number of features	D
Relaxing on couch	1346	+100 %
Working on computer	1356	+50%
Still	1510	+75%
Talking on couch	2060	+50%
Drinking water	3079	-50%
Cooking (chopping)	4448	0%
Cooking (Stirring)	4961	0%
Brushing teeth	5527	-25%

3.3.1 Feature Extraction

In this section, we describe the different feature extractors we use for processing the multiple cues. Different cues are processed through different architectures designed for video-level classification.

Appearance Extraction - In [45], the authors have used the concept of two streams on the different parts of the human body for recognizing fine-grained actions. The human body parts are extracted from their skeleton joint information. As actions in ADL also involve fine-grained motion patterns, inspired from [45] we extract deep features from different body regions of the human to represent their appearance. The main objective behind using these features is to model the static appearances along with encoding the object information carried while performing the actions. We also employ a feature selection technique to select the best image region involved in the training data distribution.

In appearance based feature extractor, CNN features from the left hand, right hand, upper body, full body and full images from each frame (cropped using their 2D joint information) are extracted to represent each body region for the classification task as illustrated in fig. 3.2. One approach is to use the 2D joint information extracted from the 3D joint information (from depth map). This projection of 3D joint information to 2D joint information has been used and detailed in the next chapter. In this chapter, we extract the 2D joints from RGB using pose estimation algorithm Convolutional Pose Machine [60]. These body joint information are used to extract the crops of human body parts.

Our experimental studies show that the aforementioned body region representation leads to a lot of redundancy. Sometimes, wrong patches extracted due to side view actions mislead the classifier converging to a wrong action. Thus, we propose a technique to select the best representation of the appearance feature by focusing on the body region with the most discriminative information. The patch representation for a given image-region i is convolutional network $f_{CNN}()$ with parameters θ_{CNN}, taking as input a crop taken from image I_t at the position of the part patch i:

$$z_t^i = f_{CNN}(crop(I_t, patch_i); \theta_{CNN}) \qquad i = \{1, ..., 5\} \tag{3.1}$$

We use pre-trained Resnet-152 for $f_{CNN}()$ to extract the deep features from the last fully connected layer which yields 2048 values described as our frame descriptors z_t^i for each part i. The next step is to compute a video descriptor. This video descriptor should be fixed in length irrespective of the number of frames in the video. Thus, we perform a temporal aggregation of the frame descriptors to compute the video descriptor. We consider min and max aggregation by computing minimum and maximum values for each descriptor

dimension j over T video frames

$$m_j = \min_{1 \leq t \leq T} z_t^i(j), \qquad M_j = \max_{1 \leq t \leq T} z_t^i(j) \tag{3.2}$$

The motivation behind using the extremes instead of average pooling is to capture the salient information in the temporal information. Note that this pooling mechanism might be effective in laboratory setting but not in real-world scenarios where extremes could be sensitive to the noises. The static video descriptor for part i is defined by the concatenation of time-aggregated k dimensional frame descriptors as

$$v_{stat}^i = [m_1, ..., m_k, M_1, ..., M_k]^T \tag{3.3}$$

For ResNet-152 as in our case, k is 2048. To capture temporal evolution of per-frame descriptors, we also consider temporal differences of the form $\Delta f_t^i = f_{t+\Delta t}^i - f_t^i$ for $\Delta t = 4$ frames. Similar to 3.3.1 we compute minimum Δm_j and maximum ΔM_j aggregations of Δz_t^i and concatenate them into the dynamic video descriptor

$$v_{dyn}^i = [\Delta m_1, ..., \Delta m_k, \Delta M_1, ..., \Delta M_k]^T \tag{3.4}$$

Finally, video descriptors for appearance for all parts and different aggregation schemes are normalized and concatenated into the CNN feature vector. The normalization is performed by dividing video descriptors by the average $L2 - norm$ of the z_t^i from the training set.

Note: Through out this chapter we use the term validation set for a portion of training data (around 20%) which is used to tune the learning parameters. The feature selection is done by feeding these CNN features z_i to a linear SVM classifying separately each patch i. These SVMs compute classification scores on a validation set separately for each patch i. We select the patch i of image-region with the best classification score on the validation set. This allows us to select the best body region for characterizing the appearance feature. As per our observation, these selected appearance features not only represent the best static appearances but also have the best combinational power with the motion based information.

Short-term Motion Extraction - For modeling short-term motion, we use improved dense trajectories toolbox provided in [40]. As explained in chapter 2, the PoI are tracked through out the video and described by the local features around them. These local features include HoG [42], HoF [43] and MBH [44] features. However, these frame-level features require a video-level representation. Such video-level representations must have fixed dimension to promote the use of efficient linear classifiers like SVM. Thus, a fisher

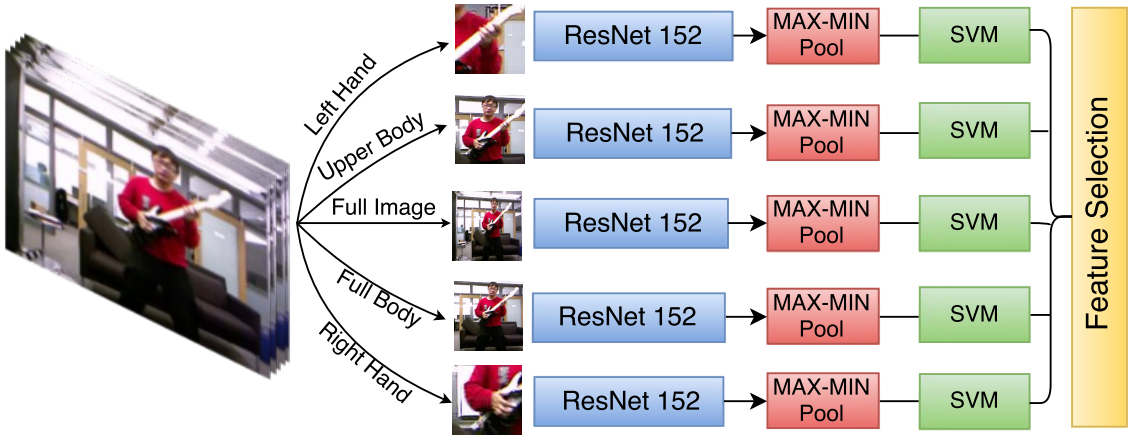


Figure 3.2: Each image frames are divided into five parts from their pose information which are input to ResNet-152 followed by max-min pooling. The classification from the SVM determines the part to be selected.

vector encoding is used to compute a video descriptor from the frame-level local features. Fisher vector representation of a video is obtained using standard Mixture of Gaussians (MoG) model as described in [137].

Pose based Motion Extraction - The main focus of the previous methods [5, 54, 53] includes using the RNNs to discover the dynamics and patterns for 3D human action recognition. The sequential nature of the 3D skeleton joints over the time makes the RNN learn the discriminative dynamics of the body. In pose based motion extractor, we fed transformed body pose information (to be described) to a 3-layer stacked LSTM so as to model the temporal information as shown in fig. 3.4. The main reason for stacking LSTM is to allow for greater model complexity, to perform hierarchical processing on large temporal tasks and naturally capture the structure of sequences. A pre-processing step (transformation os skeleton) is performed following [5] to normalize the 3D skeleton in camera coordinate system as illustrated in fig. 3.3. The 3D skeleton joint is translated to the $hip - center$ followed by a rotation of the X axis parallel to the 3D vector from "right hip" to the "left hip", and Y axis towards the 3D vector from "spine base" to "spine". At the end, we scale all the 3D joints based on the distance between "spine base" and "spine" joints. Thus the transformed 3D skeleton v_t at time frame t which is represented as $[x_{r,t}, y_{r,t}, z_{r,t}]$ for $r \in$ joints (J) and (x, y, z) being the spatial location of r^{th} joint is input to the LSTM at time stamp t. We normalize the time steps in videos by padding with zeros. This is done to keep fixed time steps in LSTM to process a video sample.

Traditionally, authors in [63, 52, 5] solve action recognition problem as a many to one sequence classification problem. The loss is generally computed at the last time step of

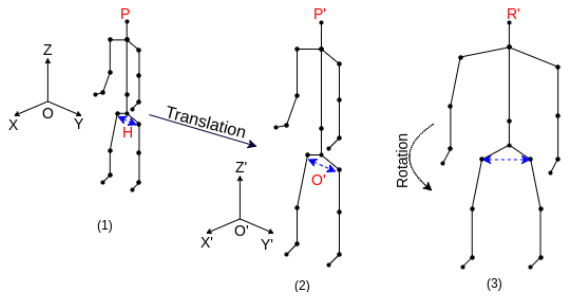


Figure 3.3: A set of pre-processing step on 3D articulated poses for transforming the skeletons to a normalized coordinate plane.

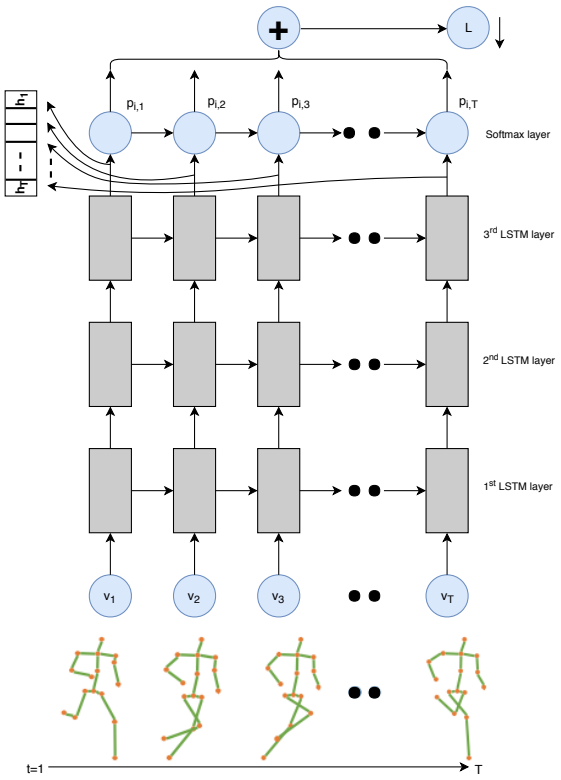


Figure 3.4: Three-layer stacked LSTM with $t = T$ time steps. The skeleton joint coordinates v_t are input at each time step. L is the loss computed over time and h is the latent vector from last layer LSTM.

the video which is back-propagated through time. In this work, we compare the LSTM cell output with the true label of the video at each time step. In this way we get time-step number of sources to correct errors in the model (via backpropagation) rather than just one at the end for each video. Thus the cost function of the LSTM for videos is computed by averaging the loss at each frame as follows

$$L = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T} y_i \log(p_{it}) \tag{3.5}$$

where L is categorical cross-entropy computed for N video samples in a mini-batch over T time-steps, y_i is the sample label, and $p_{it} \in (0,1) : \Sigma_t = 1 \forall i,t$ is the prediction for a video. This loss L is back-propagated through time. Here, the LSTM treats each temporal sequence independently as a sample, whose prediction is again determined by the current and previous gate states. This method provides better performance compared to the minimization of the loss at the last time step only due to better feedback back-propagated through time. So, we extract the latent vectors from every time step t of the last layered LSTM as shown in fig. 3.4.

For convenience of understanding, hereon we denote the appearance based features, short-term motion and pose based motion features by F_1, F_2 and F_3 respectively.

3.3.2 Two-level Fusion Strategy

In order to fuse the features extracted from different cues, we propose a two-level fusion strategy to take advantage of each cue. The first level of fusion (early) combines the features in a balanced way to address actions which are characterized by most of the features. The second level of fusion (late) puts more emphasis on selection of features which are characterizing specific actions in a prominent manner.

For early fusion, we concatenate appearance (F_1) and short-term motion (F_2) leading to $F_x = [F_1, F_2]$ because they are highly correlated. For late fusion, we put more importance on pose based motion because this feature is very complementary to the previous ones. Temporal information from poses is not discriminative for all the actions, so fusing temporal information at an early stage adds noise to the classifier. For actions like *relaxing on couch*, *talking on phone*, *writing on whiteboard* and so on temporal information may not be important. Thus encoding the pose based motion to a feature space along with appearance and short-term motion leads to common feature space where the actions are not discriminative. Such a strategy of early feature fusion of all the cues introduce high bias to the models leading to under-fitting of the training data distribution. We provide more details about this observation in our experimental section. Consequently, we propose to fuse the pose based motion (F_3) features using a late fusion strategy where the fusion

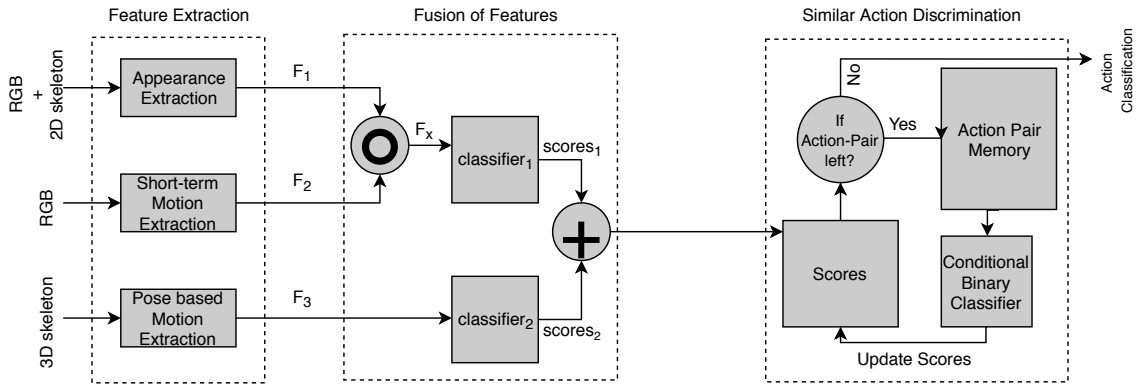


Figure 3.5: Big picture of the architecture proposed to combine the features with two-level fusion strategy for the testing phase. The action-pair memory module keeps track of action pairs with high similarities. Such action pairs are forwarded to binary classifier to disambiguate the similar actions.

focuses on the individual strength of modalities.

In the two-level fusion strategy, the fused representation of appearance and motion of a video F_x and the pose based motion representation of a video F_3 is input to two linear SVM classifiers. Classifiers clf_1 and clf_2 learn the mapping $\mathbb{X} \rightarrow \mathbb{Y}$, where $F_x \in \mathbb{X}$ for clf_1, $F_3 \in \mathbb{X}$ for clf_2 and $y \in \mathbb{Y}$ is a class label. For a given SVM parameter θ, the algorithm performs a parameter search on a large number of SVM parameter combinations to obtain the optimal value θ^*. So, θ_1^* and θ_2^* are the optimal SVM parameter of clf_1 and clf_2 respectively. The second level of fusion is performed on the test set by fusing the classification scores of the respective classifiers. For this, we introduce a fusion parameter α to balance the visual cues; α ranging between [0,1]. Let $scores_1 = P(y|F_x, \theta_1^*)$ and $scores_2 = P(y|F_3, \theta_2^*)$ be the classification scores computed by clf_1 and clf_2 respectively (see fig. 3.5). Then the second level of fusion is performed by computing the action classification score s.

$$s = \alpha P(y|F_x, \theta_1^*) + (1 - \alpha) P(y|F_3, \theta_2^*) \tag{3.6}$$

A small value of α means that the pose based temporal information (F_3) is the dominant visual cue. Thanks to the fusion strategy, an optimized pool of features is extracted to feed the classifiers dedicated to the different action categories.

3.3.3 Similar Action Discrimination

ADL datasets have similarity in action-pairs like *stacking, unstacking objects; cleaning objects, taking food* and so on. Thus, a classifier trained with generic features mis-classifies

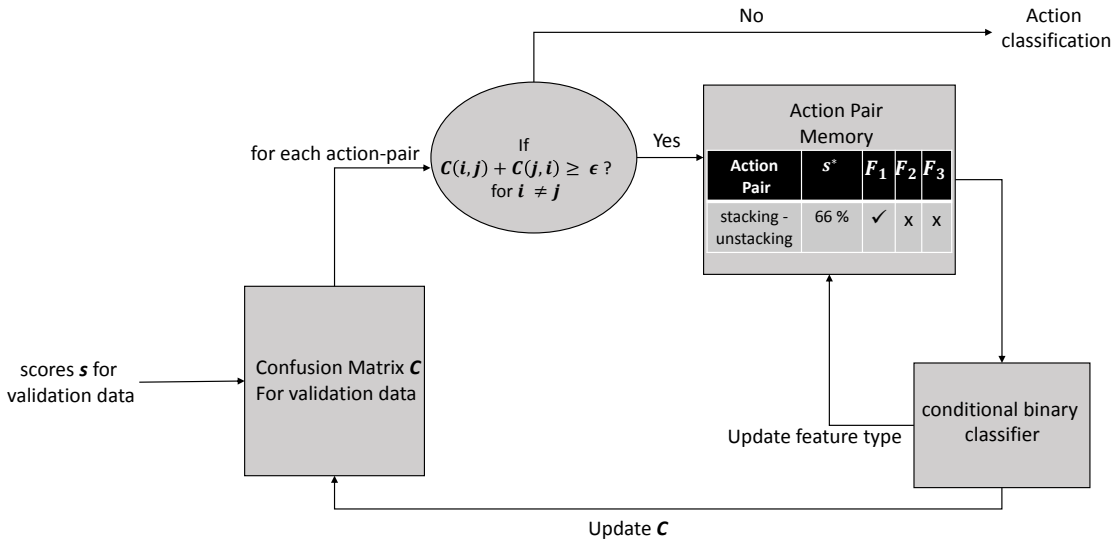


Figure 3.6: A fine picture of Similar Action Discrimination Module in the training phase. We have illustrated an example of the information stored in an action-pair memory module.

the similar action types in the absence of finer analysis. Thus, there are samples in the training data distribution which are vulnerable to mis-classification due to the lack of prominent discriminative features. These samples require a stable classifier to discriminate them from the other action categories. So, we propose a mechanism for similar action discrimination consisting of a memory module and a binary classifier. The objective is to disambiguate similar actions by exploiting their predicted scores from the fusion phase.

Training phase - The algorithm checks for the confused pair of actions in the fused scores of the validation set as illustrated in fig. 3.6. Let C be the confusion matrix of the actions classified in the validation set and a_r represents the action r, then the algorithm checks the false positives in C. If $C(i,j) + C(j,i) \geq \varepsilon$ with $i \neq j$, then action a_i and a_j are prone to mis-classification. The action pair memory module depicted in fig. 3.6 keeps a track of these action pairs in descending order of mis-classification score in the validation step. The last level of classifier is a binary classifier that classifies the similar action pair (a_i, a_j).

Thus, classifiers are dedicated to a small set of ambiguous actions which are very similar to each other. Handling ambiguities through binary classifier also includes a selection of features. As these actions may have similar motion, pose or temporal dynamics, different feature combination strategies are exploited to classify the two ambiguous actions.

Thus, the action-pair memory module also keeps track of which features to use or fuse for disambiguating the similar actions in the validation set. The feature or combination of features with maximum classification accuracy in the validation set is recorded in the action-pair memory module. In the training phase, the action-pair memory module learns the similar action pairs along with the feature type required to disambiguate them by a greedy approach from the validation set.

Testing Phase - The classification scores are generated from the fusion phase (scores from the late fusion). The video samples with predicted labels if present in the action pair module, are classified by a conditional binary classifier. This dedicated binary classifier uses the features recorded in the action-pair memory module. The final classification score is updated from the classification score of the binary classifier and the same process is repeated until all the actions susceptible to mis-classification as per the action-pair memory module undergoes binary classification. This finite looping of discriminating similar actions in a binary classifier is bounded by the number of action-pairs recorded in the action-pair memory module in terms of time complexity. This strategy of noiseless classification through a conditional binary classifier results in discriminating similar actions which is a common challenge in ADL.

3.4 Experiments

We validate our proposed action recognition approach on 3 public datasets - CAD-60, CAD-120, and MSRDailyActivity3D. We also validate this proposed method on NTU-60 in order to compare with other end-to-end methods fabricated for large scale datasets. An illustration of similar action-pair is shown in fig. 3.7 for better understanding of the challenges in ADL. In this section, we first present the implementation details of our proposed method, hyper-parameter setting, followed by a set of ablation studies. This ablation studies include qualitative, quantitative results and the effectiveness of proposed Similar Action Discrimination module.

3.4.1 Implementation Details

Feature Extraction - For *appearance extraction*, we use 2D convolutional features (from ResNet-152 pre-trained on ImageNet) for different body regions. We compute 2D poses for human body crop extraction using Convolutional Pose Machines [60]. In the case of availability of large training database, we also use 3D convolutional features from I3D [3] network. For *pose based motion extraction*, we build a 3 layered stacked LSTM framework on the platform of keras toolbox [138] with TensorFlow [139]. The number of neurons for each LSTM layer is set to 64, 64, 128, 512 for CAD-60, CAD-120, MSRDailyActivity3D and

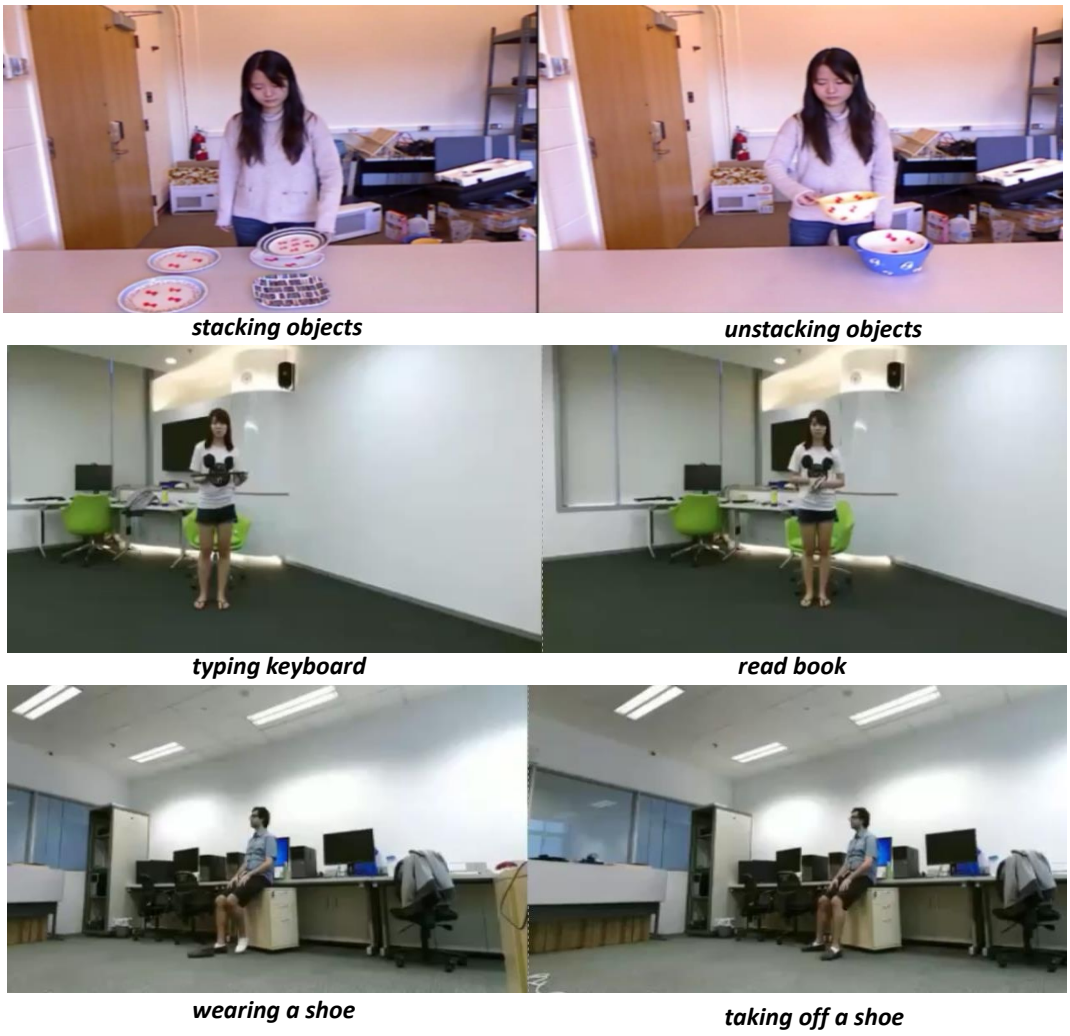


Figure 3.7: Examples of action-pair with high degrees of similarity from CAD-120 and NTU60 datasets.

NTU-60 dataset. Dropout [140] is used with a probability of 0.5 to eliminate the problem of over-fitting. The concept of Gradient clipping [141] is used by restricting the norm of the gradient to not to exceed 1 in order to avoid the gradient explosion problem. Adam optimizer initialized with learning rate 0.005 is used to train the network.

Fusion of Features - For $classfier_1$ and $classifier_2$, we use scikit-learn [142] implementation of SVM.

Similar Action Discrimination - This stage to disambiguate similar actions is implemented in *Python* with a scikit-learn [142] implementation of SVM for the binary classifier.

3.4.2 Hyper-parameter setting

Parameter α responsible for score fusion of classifiers clf_1 and clf_2 is trained in the feature fusion phase. This is done by globally searching the best value of α ranging between [0,1] for which the validation data yields maximum action classification accuracy in the training phase. This trained α is used for testing.

Similarly, parameter ε used for selecting confused action-pairs is handcrafted. Its value depends on the action categories present in the training samples. The value of ε is set manually in function of the confusion matrix during training of the second level fusion stage. Higher values are set for data distribution where confused action pairs have more likelihood. For instance, the value of ε ranges from 0.1 for NTU-60 to 0.44 for CAD-120.

3.4.3 Qualitative Results

In this section, we perform a qualitative evaluation of our two-level fusion strategy by visualizing the high dimensional data using t-SNE tool [12]. t-Distributed Stochastic Neighbor Embedding (t-SNE) is a technique for dimensionality reduction that is particularly well suited for the visualization of high-dimensional data. In fig. 3.8, we visualize the actions *drink* and *sitdown* using short-term motion, appearance, and their combination. These actions are mis-classified when the person performs the action *drinking* while *sitting* on a sofa. From the figure, it is clear that the action groups are visually better discriminated using combination of the cues. This depicts the effectiveness of using common feature space for appearance and short-term motion.

3.4.4 Quantitative Results

In this section, we report the action classification scores of the individual features along with their combination. Table 3.2 reports the action classification accuracy on three datasets CAD-60, CAD-120 and MSRDailyActivity3D using appearance, short-term and

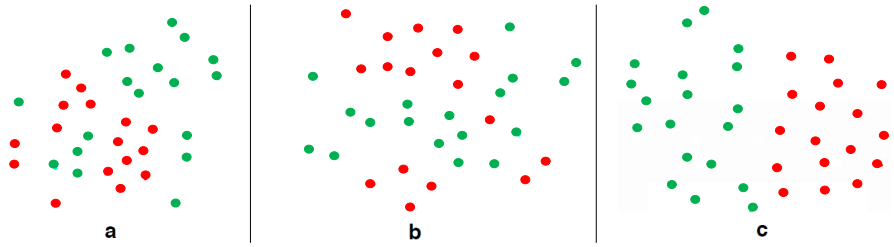


Figure 3.8: t-SNE [12] representation of *drink* (in red) and *sitdown* (in $blue$) action using (a)short-term motion only (1^{st} column), (b)appearance only (2^{nd} column) and (c)both appearance and short-term motion (3^{rd} column) where the actions are more discriminated as compared to their individual feature space.

Table 3.2: Ablation study on how each feature performs individually and with different combination techniques for action classification on CAD-60, CAD-120 and MSRDailyActivity3D. The performance is evaluated in terms of action classification accuracy (in %). In early fusion, we fused all the features with $l2-$ normalization and proposed fusion is our two-level fusion strategy. $MSR3D$ signifies MSRDailyActivity3D, F_1 is appearance, F_2 is short-term motion and F_3 is pose based motion.

Dataset	F_1 (2D-CNN)	F_2 (IDT)	F_3 (LSTM)	$F_1 + F_2$	$F_1 + F_2 + F_3$ Early Fusion	Proposed Fusion
CAD-60	**89.70**	72.05	67.64	95.58	70.58	**98.53**
CAD-120	72.58	**79.84**	63.70	83.06	63.70	**87.90**
MSR3D	80.93	81.87	**91.56**	90	91.56	**97.81**

pose based motion. The performance obtained using different features are very data-dependent. For example, we get better results on MSRDailyActivity3D using pose based motion, CAD-120 using short-term motion and CAD-60 using appearance features. Table 3.2 shows the importance of using the two-level fusion scheme which takes into account the distinguishable characteristics of all features by performing a late fusion of appearance, short-term motion with pose based motion. This is shown by comparing our fusion strategy with naive early fusion of all features. Our proposed fusion outperforms the former as depicted in table 3.2.

3.4.5 Effect of using the mechanism of Similar Action Discrimination

This section presents an ablation study on the similar action discrimination mechanism and how the action-pair module works. In table 3.3, we show the confused actions with their corresponding mis-classification rate in CAD-120 for every cross-actor split (mentioned in section 2.5). The action-pair module keeps a track of the confused actions which

Table 3.3: Action-pair memory content for different splits in CAD-120 ($left$). Each split signifies cross-actor setup for classification evaluation. The second column represents the action pairs confused among each other with their summation of mis-classification accuracy in third column (with validation set). The threshold for this dataset is set to 0.4.

split	Action Pairs	$C(i,j)+$ C(j,i)
1	*cleaning object* and *taking food*	0.44
1	*stacking and unstacking objects*	0.67
2	*cleaning object* and *taking food*	0.66
2	*stacking* and *unstacking objects*	0.66
3	*stacking* and *unstacking objects*	0.55
4	*cleaning object* and *taking food*	0.55
4	*stacking* and *unstacking objects*	0.44

Table 3.4: Improvement in action classification accuracy on using conditional binary classifier for all the datasets used. $MSR3D$ signifies MSRDailyActivity3D.

Dataset	Acc. before binary classifier	Acc. after binary classifier
CAD-60	98.52 %	98.52 %
CAD-120	87.90%	94.40 %
MSR3D	97.81%	97.81 %
NTU-60	84.95 %	87.09 %

are classified separately by a binary classifier which is also a linear SVM. For CAD-120, IDT+FV (short-term motion along with appearance because of presence of the HOG) discriminates the confused action pairs with 100 % accuracy. The drawback of this module includes its dependency on the distribution of the validation set. This drawback is depicted in table 3.3 where the cross-validation fails to capture confused action pairs like *cleaning objects and taking food* (in 3^{rd} row, left). Table 3.4 reports the action classification accuracy on all the datasets used before and after applying the action-pair module. This module does not have any effect on CAD-60 and MSRDailyActivity3D on which the actions are already classified with remarkable accuracy.

3.5 Runtime Analysis

The fully automated architecture has been trained on two GTX 1080 Ti GPUs (each for extracting RGB based video descriptors from CNN network and training LSTM on skeleton sequences) and a single CPU (for extracting IDT features with fisher vector encoding) in parallel. IDT being computationally expensive (with a processing speed of less than 4 fps) decides the computational time involved in the feature extraction process. The proposed

architecture including the fusion strategy along with the action-pair module only takes as additional cost 10 ms time delay for a forward pass of an image frame on a single CPU.

3.6 Conclusion

In this chapter, we have proposed a framework for action recognition mixing a high level fusion strategy and machine learning techniques. The proposed hybrid architecture is fully automated enabling the hyper-paramaters except ε to learn themselves. We justify the use of this two-level fusion mechanism by qualitative and quantitative analysis. We also propose an action-pair memory module to disambiguate similar actions. Our proposed action recognition architecture datasets is effective for small scale datasets like CAD-60, CAD-120 and MSRDailyActivity3D. However, for scaling up such large dataset like NTU-60 is not optimal. This is because the deep features extracted are optimized independently and finally used as feature extractors rather than taking the full advantage of DNN methods, which is global optimization.

We emphasize the fact that the existing features are quite capable of distinguishing the ADL if combined in a strategic way. The quality of recognition rate achieved in this work ranging from 87 % to 98% is satisfactory. But, what about addressing a large diversity of actions? That would require a model training over a large distribution of data in an optimal manner. Thus, incorporating the feature extraction process along with fusion mechanisms and similar action discrimination should be approached in a single network. We also need to make sure that such network leverages the pros of different modalities. With this aim, a possible direction of research is attention mechanism using privileged modalities.

Chapter 4

Attention Mechanisms for Visual Representation

4.1 Introduction

As discussed in the previous chapter, scaling up the action recognition algorithms to address a wide diversity of actions is the need of the hour. One possible solution is going towards end-to-end video convolutional networks. But can these networks address the challenges in ADL?
In this chapter, we present an action recognition framework in section 4.2, present our proposed frameworks P-I3D in section 4.3, Separable STA in section 4.4, and VPN in section 4.5. We present our experimental analysis on four public datasets in section 4.6, and finally conclude in section 4.7.

The parts of this work have been published in conference venues - WACV 2019 [143], ICCV 2019 [132] and ECCV 2020 [144].

In this chapter, we discuss how the current video convolutional networks like C3D [71], I3D [3] are fabricated for generic videos. These networks with the same kernel operation over the whole spatio-temporal video cannot handle the challenges in ADL. Although, recent studies show [71, 73, 145] that 3D convolutional operations can better model the temporal information than other recurrent operations popular in this domain. But, challenges like recognizing fine-grained actions and disambiguating similar actions require functionalities beyond fixed kernel operations. Thus, in this thesis, we focus on improving the 3D convolutional networks with additional functionalities that can address the challenges in ADL. These functionalities include attention mechanism.

Previous attention mechanisms for action recognition are based on RNNs as the classification network [112, 115]. However, the effectiveness of the 3D convolutional networks inspire us to use them as classification network. The question remains, how can

we invoke attention mechanisms in 3D ConvNets? Few studies [85, 146, 4] have shown improvements over traditional 3D ConvNets as video backbone as discussed in chapter 2. However, these algorithms largely address the concerns in generic videos retrieved from internet sources.

On the other hand, from the last chapter, we concluded the pros of using different modalities for action recognition. Each modality with complementary features discriminates the actions when combined in a strategic manner. With the above studies on 3D ConvNet, we argue that such networks with spatio-temporal kernels operating over the video volume models the appearance and short-term motion simultaneously. Thus, in our work, we rely on 3D ConvNets for modeling appearance and short-term motion without actually using optical flow. Moreover, optical flow in ADL which mostly contain subtle motion do not contribute much for discriminating action representation.

But what about poses? We have presented in the last chapter that 3D poses do provide the view-invariance property whereas lacks appearance information. Moreover, fusing poses which are processed in dissimilar networks compared to RGB or optical modalities, is challenging. One of the reason appears to be over-fitting as illustrated in [147]. Multi-modal networks fusing lately the modalities, generally have higher train accuracy and lower validation accuracy. One may suspect that the over-fitting is caused by the increased number of parameters in the multi-modal networks. Thus, we focus on **using poses to provide information regarding the Region of Interests (RoIs) in a video**. Consequently, we propose algorithms based on pose driven attention mechanisms that provide pertinent attention weights to guide the RGB classification network which is 3D ConvNets in our case.

In this chapter, we present three novel pose driven attention mechanisms for discriminative visual representation of actions. Each proposed method complements the former by improving their underlying drawbacks. Each method includes primarily a video backbone, an attention network and a classifier. The attention network is driven by 3D articulated poses that provide attention weights to the RGB cue processed by the video backbone. In summary the methods described in this chapter include -

- An action recognition framework where the attention network assigns soft-weights to the different human body parts relevant for an action.

- An action recognition framework which generalizes over a wide variety of actions by invoking a focusing of attention on the RoIs in a spatio-temporal feature map. Here, we introduce the concept of dissociating the spatial and temporal attention weights.

- An action recognition framework improved by providing an accurate and tight embedding between the RGB images and their corresponding mis-aligned 3D poses. We

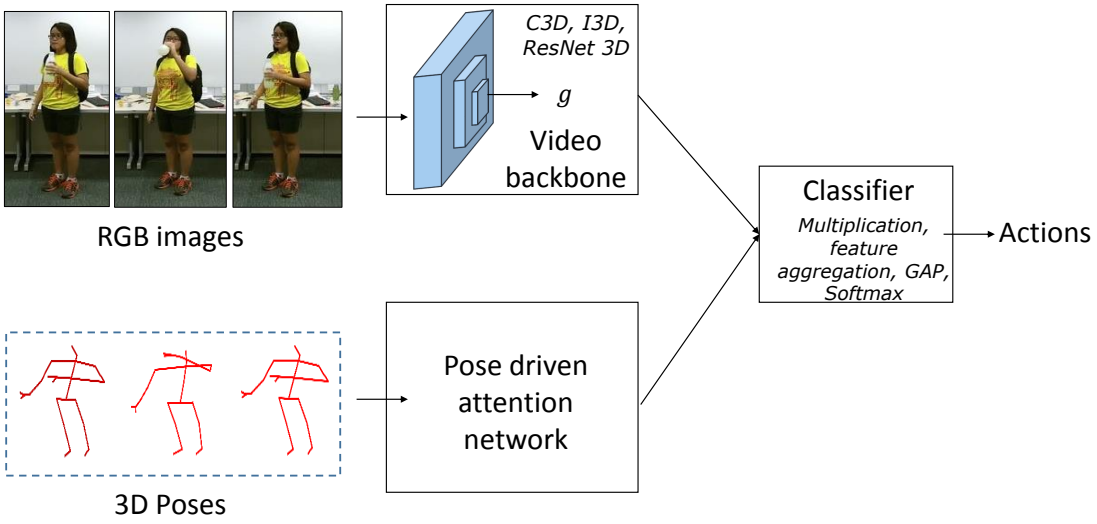


Figure 4.1: Schema of our Action Recognition Framework with input as RGB snippets and 3D poses. It consists of a video backbone for spatio-temporal video representation, a pose driven attention network, finally a classifier module which combines the attention weights and finally classifies the actions.

also improve the functionalities of the attention network by processing the 3D articulated poses through GCNs considering their graphical structure and finally, providing a joint spatio-temporal attention weights.

Below we describe our Action Recognition Framework that is used throughout out this chapter.

4.2 Action Recognition Framework

Our action recognition framework consists of a video backbone, an attention network, and a classifier as shown in fig. 4.1.

Video Backbone - This is a sub-network in the framework that processes the video input. The video backbone processes the input stack of images to compute a spatio-temporal feature map. Practically, this video backbone is a 3D ConvNet $f()$ with parameters θ. The input to this video backbone are successive crops of human body along the video. In case of multiple persons in the scene, we extract an image crop encapsulating all of them. The cropping operation $crop()$ is explained in details in the next proposed framework. The input dimension of the RGB modality is $T \times M \times N \times C$ where, T is the number of input

stack of images, $M \times N$ represents the spatial resolution of each image and $C = 3$ represents the three channels in RGB. Starting from the input of T human-cropped frames from a video V, the spatio-temporal representation g is the feature map extracted from an intermediate layer of the 3D ConvNet like I3D [3]. The feature map g is thus, given by

$$g = f(crop(V); \theta) \tag{4.1}$$

The resulting dimension of g is $t_c \times m \times n \times c$, where t_c time, $m \times n$ the spatial resolution and c channels are all spatio-temporally squeezed by the operations performed in the 3D ConvNet $f()$. Thus, the pose driven attention mechanism provides attention weights to the video representation g.

Attention Network - The pose driven attention network contains first, a sub-network called pose backbone to process the 3D poses and then subsequent operations to learn the attention weights for the end-task. The end-task in this case is classifying the actions. The attention network is initialized with uniform weights and then with the cross-entropy loss optimizing the model output, learns relevant attention weights. Note that the only way of evaluating such mechanisms is by evaluating the performance of the classification task since no ground-truth for the attention weights are available.

Classifier - This module performs the linear combination of the attention weights and the spatio-temporal feature maps. In order to compute the linear combination, the attention weights must match the dimension of the spatio-temporal feature map g. Thus, the attention weights are duplicated to match the desired dimension. The resultant feature map is called modulated feature map. This module includes all kinds of feature combinations of the spatio-temporal feature maps. These combinations are detailed in the proposed frameworks. This module also consists of a Global Average Pooling (GAP) over the modulated spatio-temporal feature map. This operation squeezes the feature map into a low dimension feature vector. This feature vector retains only the saliency in the former feature map.

Finally, the classifier consists of a bottleneck or a $1 \times 1 \times 1$ convolutional operation with number of filters equal to the number of classes. Then, the resultant feature vector is flattened and soft-max activated with a dense layer to assign the action classes.

4.3 Spatial attention (P-I3D)

Previous studies [45, 114, 148] have shown significant improvement in discriminating similar actions in ADL by focusing on human body parts. Capturing appearance information from human body parts could ensure modeling of high-level object information. Individual human body parts compared to the whole scene as input to the state-of-the-art

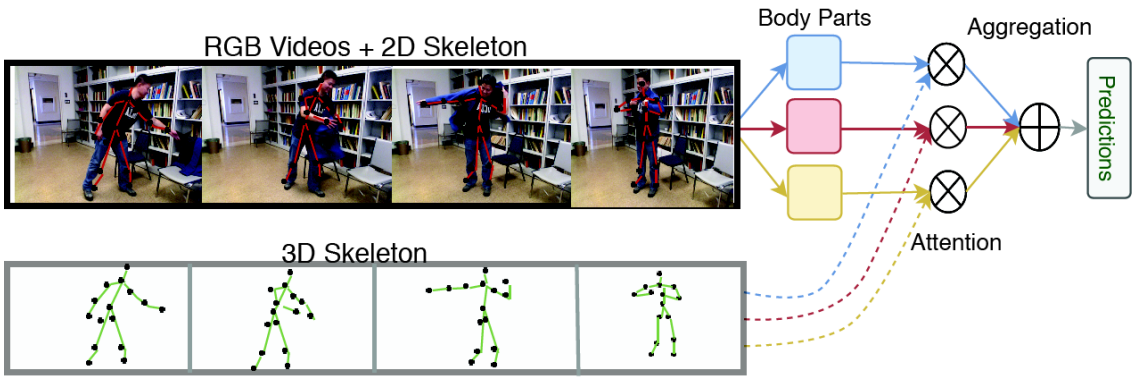


Figure 4.2: Schema of our proposed framework for an action "donning". The 3D pose information determines the attention weights to be given to the spatio-temporal features extracted from the RGB videos corresponding to three relevant body parts of the person performing the action.

action recognition networks could actually facilitate them to better capture fine-grained details of the objects interacted with while performing an action. Therefore, we propose a weighted aggregation of human body parts to train an end-to-end action classification framework. Weighting the body parts for action classification is a mutually recursive problem. Body part selection for action classification depends on action and vice-versa. So, we propose a pose driven spatial attention mechanism to weight the body parts for action classification. We call it spatial attention because the weighting of human body parts can be considered as an operation performed in the spatial scale.

Fig. 4.2, shows a schema of our proposed framework. The action "*donning*" is recognizable by looking at the motion of the object grasped by the hands (which is the *jacket*). Spatio-temporal features extracted from these body parts could be sufficient to model the action. On one hand, for actions like "*jumping*", "*running*", and so on, simple aggregation (summation or concatenation) of the representation from the human body parts models the actions better than using the human body part representations individually. On the other hand, for actions like "*drinking*" and "*making a phone call*" simple aggregation of the human body parts diminishes the distinctness of the spatio-temporal features for action classification because of providing equal weightage to relevant and irrelevant body parts. So, we propose an RNN attention mechanism to provide appropriate weights to the relevant human body parts involved in the action. Such attention mechanism further improves the action classification.

We propose and end-to-end 3D convolutional network with soft attention mechanism for action classification. We exploit the 3D articulated poses of the actor performing action to determine which part of the body can best model an action category. Fig. 4.3 shows the

overall architecture, which consists of three I3D [3] sub-networks for extracting spatio-temporal features from human body parts (left hand, right hand and full body) and RNN attention sub-network to assign different degrees of importance to the body parts. The input to the network is the RGB video with the sequence of corresponding 3D joints. The challenge in this task includes identifying the appropriate feature space where the spatio-temporal features from the tracks of human body parts are required to be aggregated. Another challenge includes the joint training of the video backbone and the attention network to weight the relevant body parts.
In the following, we discuss the body part representation, RNN attention from the articulated human poses and joint training these sub-networks together to model the actions.

4.3.1 Body Part representation

Different body parts have different degrees of importance for a particular human action. Fine-grained human action recognition can be performed by extracting cues from RGB streams. We employ a cropping operation to extract the tracks of human body parts, for instance - full body, left hand and right hand from the pixel coordinates detected by the middleware.

For illustration, we present the cropping operation of full body in fig. 4.4. Consider the 3D poses $P_t = (P_{t,1}, P_{t,2}, \cdot P_{t,J})$ for $P_{t,j} \in \Re^3$ at time t. These 3D poses with coordinates (x', y', z') are first transformed to 2D poses (x, y) in the camera coordinate system using -

$$x = \frac{P_{camera}.x'}{P_{camera}.z'}; \qquad y = \frac{P_{camera}.y'}{P_{camera}.z'} \tag{4.2}$$

where P_{camera} is camera to world matrix specific for a type of sensors. Thus, now we have $P_t^{2D} = (P_{t,1}^{2D}, P_{t,2}^{2D}, \cdot P_{t,J}^{2D})$ for $P_{t,j}^{2D} \in \Re^2$ at time t. Now in the 2D-plane, we compute the characteristic bounding box coordinates P_t^{max} and P_t^{min} from P_t^{2D} for full body crops using

$$P_t^{max} = max_y(min_x(P_t^{2D})); \qquad P_t^{min} = min_y(max_x(P_t^{2D})) \tag{4.3}$$

where max_x represents the max operation along x-axis and so on. Thus, the top corner coordinate $P_t^{max} - \lambda$ and bottom corner coordinate $P_t^{min} + \lambda$ are used to extract the full-body crop. Note that λ is an excess pixel factor added to the image crops.

The hands and the full body are of higher relevance to the actions performed in ADL. So unlike [45], we only crop these three body parts instead of five body parts. But in practice, our framework is not restricted to these three body parts and can be extended to K body parts depending upon its application type. We aggregate the spatio-temporal representation of the human body parts in order to leverage the representation of the relevant body parts for an action. Before aggregating, the part based sub-networks depicted

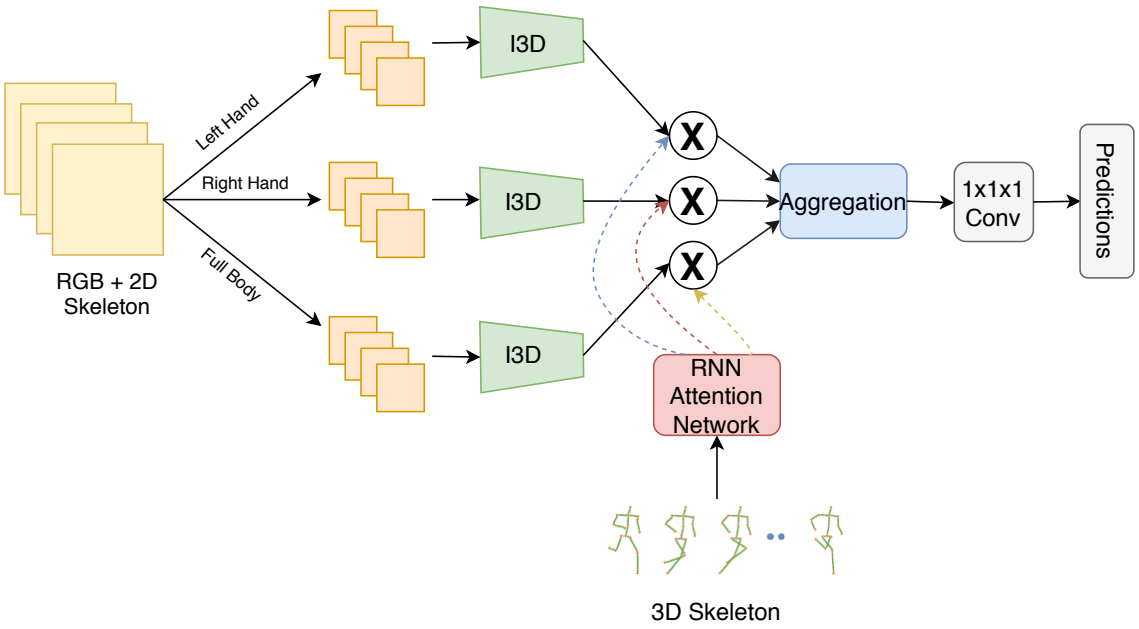


Figure 4.3: End to End action classification network (P-I3D). The input to the network is RGB videos with 3D skeletons. Actor body regions like left hand, full body and right hand are extracted from their corresponding 2D pose information. RNN based attention network takes 3D skeleton input (trained on action classification) to provide spatial attention on the spatio-temporal features from I3D (extracted from global average pooling layer after all inception blocks).

in fig. 4.3, are pre-trained using the human body parts for the task of classification. This leads to a generation of high-level spatio-temporal features representing each human body part.

The body part representation is obtained from a video backbone $f()$ explained in the previous section. Taking as input a stack of cropped images from a video V_i, the body part representation g_i of the body part i is computed by spatio-temporal convolutional network $f()$, with parameters θ using equation 4.1. Thus, the body part representation is formulated by

$$g_i = f(crop(V_i); \theta) \qquad i = \{1, ..., K\} \tag{4.4}$$

4.3.2 RNN Attention Network

The action of a person can be described by a series of articulated human poses represented by the 3D coordinates of joints. We use the temporal evolution of human skeletons to model the attention to be given to different body parts. The 3D skeleton from depth-map

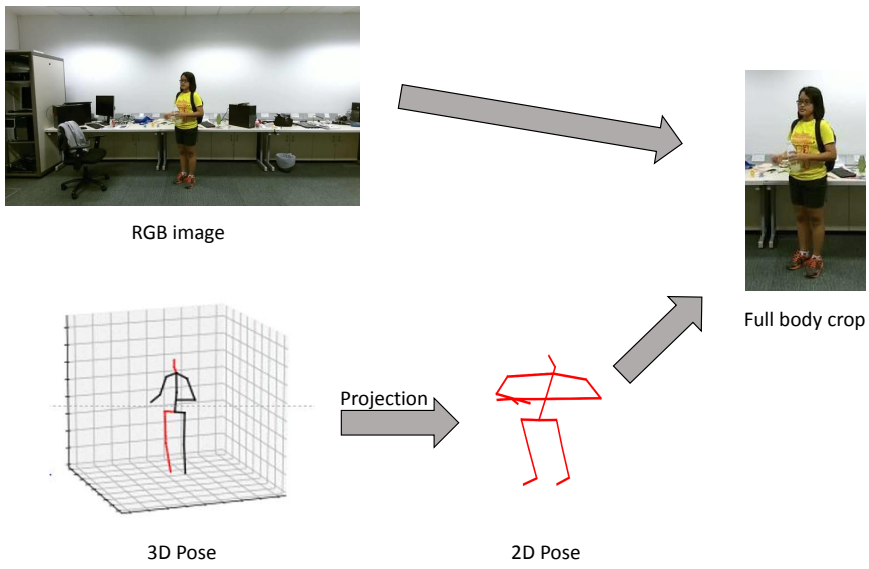


Figure 4.4: Illustration of extraction of a full body crop from its 3D poses.

captured by kinect sensor is exploited to pre-train a pose backbone for action classification to learn the temporal dynamics of skeleton joints for different action classes. This pre-training of the pose backbone is required to extract latent features with spatio-temporal structure for soft weighting the human body parts involved in an action. An obvious choice of this pose backbone is an RNN which models the temporal evolution of the skeletons joints for action discrimination.

Consequently, the pose backbone consists of three LSTM layers as used in the state-of-the-art with $P_t = (P_{t,1}, ..., P_{t,J})$ with $P_{t,j} \in \Re^3$ and J the number of joints as input. Note that poses are stacked along t_p temporal dimension. h^s is the concatenated hidden state vector of all the t_p time steps from the last LSTM layer. This LSTM which is used as a pose backbone computes high-level information describing the variance of poses in a video. Thus, the next step is to place few learnable parameters to compute the attention weights from this pose based feature vector h^s. Therefore, a dense fully connected layer is added on top of the LSTM with $tanh$ activation to obtain the scores s. These scores s learn the importance of different body parts which is formulated as

$$s = W_s tanh(W_h h^s + b_s) + b_{us} \tag{4.5}$$

where W_s, W_h are the learnable parameters, b_s, b_{us} are the bias. Similar to [113, 114], our proposed attention network learns the attention weights from the output of LSTM cell

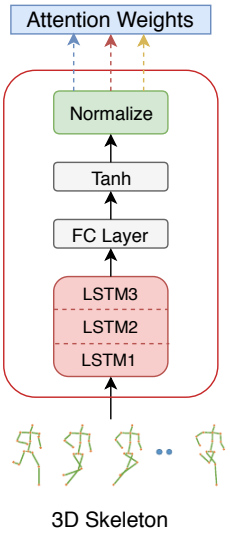


Figure 4.5: A detailed picture of RNN attention model which takes 3D skeleton poses input and computes weight attention on the spatio-temporal features from different body region of the actor.

states at each time step. The novelty of our spatial attention network lies in obtaining the attention scores from the latent spatio-temporal information of the whole video. The objective of such video based attention mechanism is to soft weight the spatio-temporal body parts representation. The global transition of the 3D poses in the whole video could only determine the importance of the relevant human body parts. The next step involves learning the relevant attention weights from the scores s. Finally, the obtained scores are normalized using a `softmax` layer to obtain the attention weights. For the k^{th} body part, the activation as the part selection gate is computed as

$$\alpha_k = \frac{exp(s_k)}{\Sigma_{i=1}^{K} exp(s_i)} \tag{4.6}$$

The RNN attention network provides weights to the different body parts representation. The part based feature maps are aggregated to obtain a fixed representation for an action video. Now, the remaining question is to choose the appropriate feature space in the 3D ConvNet, say I3D to aggregate the body part features. We chose the feature aggregation space at the penultimate layers in the 3D ConvNet. Consequently, the spatio-temporal features from the last layer of I3D are used for aggregating the body parts because these features are spatio-temporally rich and distinct with respect to action categories. The aggregation of these body part features lead to the formation of distinguishable spatio-temporal features F as

$$F = \Sigma_{k=1}^{K} inflate(\alpha_k) \circ g_k \tag{4.7}$$

where g_k is the k^{th} 4-D body part representation, $\circ$ is the hadamard product and the $inflate(.)$ operation duplicates the attention weights to match the dimension of the feature map g. For aggregation, we also explore assigning attention at different levels of spatio-temporal feature space in I3D with both summation and concatenation operations for aggregation, discussed later in ablation studies. The former tends to squash feature dynamics by pooling strong feature activations in one body part with average or low activations in other body part. Whereas the concatenation operation leads to formation of high dimensional features before classification. This leads to a drop of classification performance due to increase in training parameters in the penultimate layer of the framework.

4.3.3 Joint training the sub-networks

Joint training the I3D sub-networks consisting of several inception blocks and RNN attention network is a challenge due to the vanishing gradient problem and different backpropagation strategy (BPTT in case of LSTM). Thus we pre-train all the sub-networks separately and joint train them freezing the RNN layers to backpropagate. This strategy along with the formulated cross entropy loss discussed below enables the network to assign weights to the body parts, thus modeling the actions.

Regularized Objective Function - We formulate the objective function of the end-to-end network with a regularized cross-entropy loss and K being the number of body parts as,

$$L = L_c + \lambda_1 \sum_{k=1}^{K} (1 - \alpha_k)^2 + \lambda_2 ||W_{uv}||_2 \tag{4.8}$$

$$L_c = \sum_{i=1}^{C} y_i \log \hat{y}_i \tag{4.9}$$

where $\mathbf{y} = (y_1, ..., y_C)$ represents the ground-truth labels. y_i=1 if it belongs to i^{th} class and y_j=0 for $j \neq i$. $\hat{y_i}$ denotes the probability of the sample belonging to class i, where $\hat{y}_i = p(C_i|X)$. λ_1 and λ_2 are the regularization parameters.

The first regularization item forces the model to pay attention at each human parts. This is because the model is prone to ignoring some body parts completely though they have valuable contribution in modeling the actions. So, we impose a penalty as $\alpha_k \approx 1$ encouraging the model to pay more balanced attentions to different tracks of human parts. The second regularization item is to reduce over-fitting of the networks. W_{uv} denotes the weight matrix in connecting the layer u and v. In practice, this regularization is applied in the last layer of the framework.

The optimization is difficult due to the mutual influence of the I3D sub-networks and the

pose based RNN attention network. The methodology of separate pre-training of the pose based sub-networks ensures faster convergence of the networks. The training procedure is described in algorithm 1.

Algorithm 1 Joint Training of the RNN attention network with body part I3D sub-networks

Input: RGB video, 3D joint coordinates, model training parameters $N1$, $N2$ (e.g., $N1$ = 10, $N2$ = 25).

1: Initialize I3D sub-networks with model weights trained on IMAGENET and Kinetics.
//**Pre-train I3D sub-networks.**
2: Fine-tune I3D network with RGB data from different body parts individually.
//**Pre-train Pose backbone which is a 3-layer stacked LSTM.**
3: Train the three layered stacked LSTM network for action classification taking as input 3D skeleton of actors in video frames.
//**Initialize other attention network parameters.**
4: Add a Fully connected layer $tanh$ and a `softmax` layer on top of stacked LSTM and initialize the attention scores with equal values and the remaining network parameters using Gaussian.
5: Jointly train the Pose based RNN network with part-wise I3D network for $N1$ iterations to obtain the attention scores.
//**Jointly Train the Whole Network**
6: Fine-tune the whole network by fixing the learned Pose backbone for further $N2$ iterations.

Output: the learned network.

The drawback of this proposed framework includes generalizing it over a wide variety of actions. (1) The body part representation involving a 3D ConvNet for each body parts is expensive in terms of the number of parameters. So, choosing an optimal number of body part for a data distribution is trivial. (2) The next issue with the proposed framework is the absence of temporal attention mechanism and completely relying on the temporal operations of the 3D kernels of I3D. An interesting question can be - Is temporal attention important? Previous methods like [115, 106] show that temporal attention significantly improves the model by providing relatively higher degree of importance to the key frames in an action. Thus, the next proposed framework Separable STA aims at alleviating these two underlying drawbacks.

4.4 Separable Spatio-Temporal Attention (Separable STA)

To address the limitations in the previous framework, we propose a novel attention mechanism on top of currently high-performing spatio-temporal convolutional networks [3]. Firstly, we aim at generalizing our framework by eliminating the requirement of body part based representation. Thus, we propose to have a focus of attention on those pixels in

the spatial domain with high relevance. Secondly, we aim at incorporating the concept of temporal attention to the RGB cue by exploiting the 3D poses.

Inspired by [149], our framework uses both spatial and temporal attention mechanisms. We dissociate the spatial and temporal attention mechanisms (instead of coupling them). Coupling spatial and temporal attention is difficult for spatio-temporal 3D ConvNet features as the spatial attention should focus on the important parts of the image, and the temporal attention should focus on the pertinent segments of the video. As these processes are different, our idea is to dissociate them. In our architecture, two sub-networks independently regress the attention weights, based on 3D human skeletons inputs. The proposed attention mechanism aims at addressing a wide diversity of action eliminating its dependence on the restricted human body parts representation. On one hand, actions with human-object interaction require spatial attention to encode the information on the object involved in the action. On the other hand, actions with temporal dynamics such as *sitting* or *standing up* require temporal attention to focus on the key frames that characterize the motion.

Similar to our Spatial attention, we use pose driven attention mechanism on top of the 3D ConvNets [3]. The spatial and temporal saliency of human activities can be extracted from the time series representation of pose dynamics, which are described by the 3D joint coordinates of the human body.

4.4.1 Spatio-temporal representation of a video

The input of our model is successive crops of human body along the video and their 3D pose information. Note that these crops are the full human body crops. In using, these full body crops, we loose the spatial resolution compared to the concept of using different human body parts as in P-I3D. But, data augmentation through random cropping within the full body crops should tackle this issue. We focus on the pertinent regions of the spatio-temporal representation from 3D ConvNet, which is a 4-dimensional feature map. Starting from the input of T human-cropped frames from a video V, the spatio-temporal representation g is the feature map extracted from an intermediate layer of a video backbone. The sampling of these T frames from the whole video is detailed in the implementation details. Similar to our P-I3D, here the video backbone is a 3D ConvNet - I3D [3] $f(.)$, with parameter θ using equation 4.1. We define two separate network branches, one for spatial and one for temporal attention (see fig. 4.6). These branches apply the corresponding attention mechanism to the input feature map g and output the modulated feature maps g_s (for spatial attention) and g_t (for temporal attention). g_s and g_t are processed by a Global Average Pooling (GAP) layer and then concatenated. Finally, the prediction is computed from the concatenated feature map via a $1 \times 1 \times 1$ convolutional

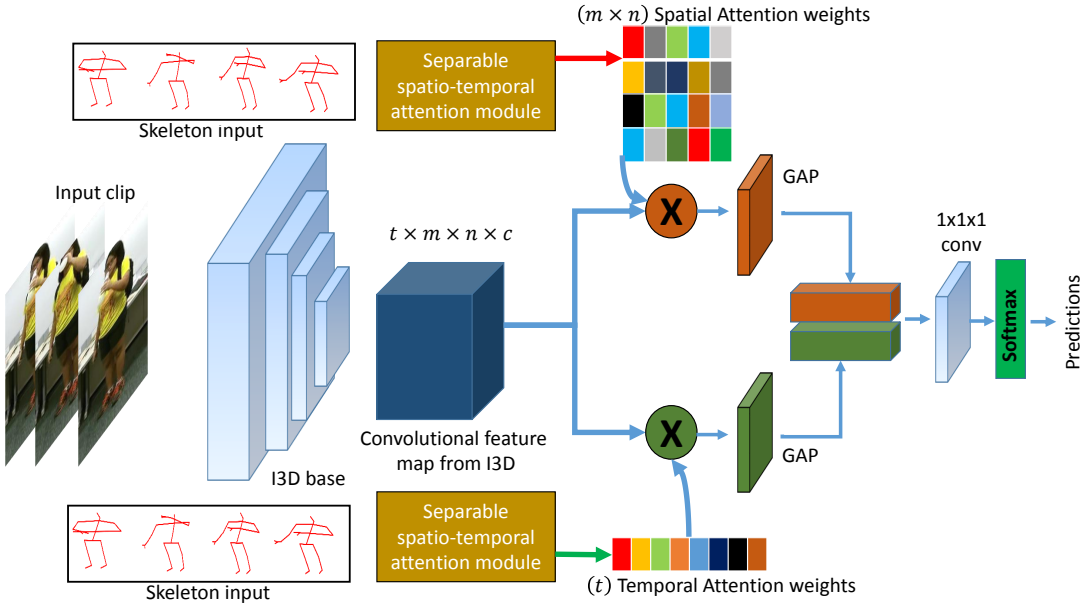


Figure 4.6: Proposed end-to-end separable spatio-temporal attention framework. The input of the network is human body tracks of RGB videos and their 3D poses. The two separate branches are dedicated for spatial and temporal attention individually, finally both the branches are combined to classify the activities. Dimension c for channels has been suppressed in the feature map for better visualization.

operation followed by a `softmax` activation function.

4.4.2 Separable attention network

In this section, we elaborate our pose driven separable attention network shown in fig. 4.7. In this attention network, we learn two distinct attention sets, one for spatial and one temporal weights. These weights are linearly multiplied with the feature map g, to output the modulated feature maps g_s and g_t.

We use 3D skeleton poses to compute the spatio-temporal attention weights. The inputs to the attention network are the feature vectors computed by a Pose backbone on the 3D poses. Similar to the previous framework, the aforementioned Pose backbone is a 3 layered stacked LSTM pre-trained on 3D joint coordinates for activity classification. The input is a full set of J joints per skeleton where the joint coordinates are in the form $P_t = (P_{t,1}, ..., P_{t,J})$ for $P_{t,j} \in \Re^3$ at time step t. Similar to P-I3D, poses are stacked along t_p temporal dimension. The attention network consists of two separated fully connected layers with $tanh$ squashing followed by fully connected layers that compute the spatial and temporal attention scores s_1 and s_2, respectively (see fig. 4.7). The scores s_1 and s_2 express the importance of the elements of the convolutional feature map g along space

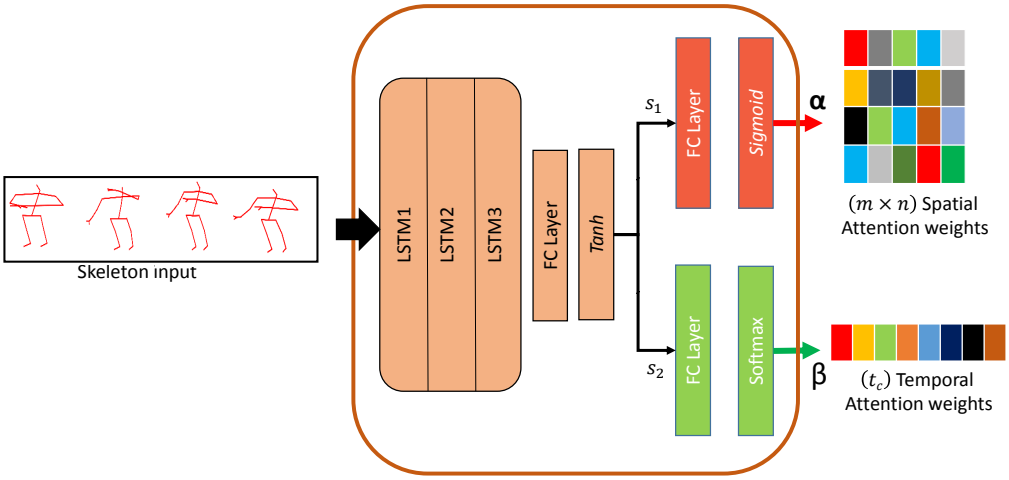


Figure 4.7: A detailed picture of pose driven RNN attention network which takes 3D pose input and computes $m \times n$ spatial and t_c temporal attention weights for the $t_c \times m \times n \times c$ spatio-temporal features from I3D.

and time. These scores s_r (i.e., s_1 and s_2 for $r = 1, 2$) can be formulated as:

$$s_r = W_{s_r} tanh(W_{h_r} h_r^* + b_{h_r}) + b_{s_r} \tag{4.10}$$

where W_{s_r}, W_{h_r} are learnable parameters and b_{s_r}, b_{h_r} are the biases. h_r^* is the concatenated hidden state vector of all the time steps from the Pose backbone.
The attention weights for spatial ($\alpha = \{\alpha_1, \alpha_2, ..., \alpha_{m \times n}\}$) and temporal ($\beta = \{\beta_1, \beta_2, ..., \beta_{t_c}\}$) domain are computed from the scores s_1 and s_2 as:

$$\alpha_k = \frac{exp(s_{1,k})}{exp(s_{1,k}) + 1}; \qquad \beta_k = \frac{exp(s_{2,k})}{\Sigma_{i=1}^{t_c} exp(s_{2,i})} \tag{4.11}$$

where $s_1 = \{s_{1,1}, s_{1,2}, ..., s_{1,m \times n}\}$ and $s_2 = \{s_{2,1}, s_{2,2}, ..., s_{2,t_c}\}$ is obtained from equation 4.10. Normalizing the high number of $m \times n$ spatial attention weights with `softmax` leads to extremely low values, which can hamper their effect. To avoid this, we use sigmoid activation as in [113]. These attention weights play the role of soft selection for $m \times n$ spatial elements of the convolutional feature map g.
Finally, the modulated feature maps with spatial and temporal attention (g_s & g_t) are computed as

$$g_s = inflate(\alpha) \circ g; \qquad g_t = inflate(\beta) \circ g \tag{4.12}$$

where $\circ$ is the hadamard product and the $inflate(.)$ operation duplicates the attention

weights to match the dimension of the feature map g. The attention model is joint-trained with the 3D ConvNet.

4.4.3 Training jointly the attention network and 3D ConvNet

Unlike the existing attention networks for activity classification [113, 114], jointly training the separable spatio-temporal attention network and the 3D ConvNet is relatively straightforward as depicted in algorithm 2. The training phase involves fine-tuning the 3D ConvNet without the attention branches for activity classification. Then, the attention network is jointly trained with the pre-trained 3D ConvNet. The training procedure is described in algorithm 2. This ensures faster convergence as demonstrated in [106]. The 3D ConvNet along with the attention network is trained end-to-end with a regularized cross-entropy loss L formulated as

$$L = L_C + \lambda_1 \sum_{j=1}^{m \times n} ||\alpha_j||_2 + \lambda_2 \sum_{j=1}^{t_c} (1 - \beta_j)^2 \tag{4.13}$$

where L_C is the cross-entropy loss for C activity labels. λ_1 and λ_2 are the regularization parameters. The first regularization term is used to regularize the learned spatial attention weights α with the l_2 norm to avoid their explosion. The second regularization term forces the model to pay attention to all the segments in the feature map as it is prone to ignore some segments in the temporal dimension although they contribute in modeling activities. Hence, we impose a penalty $\beta_j \approx 1$. We impose different regularization constraint for learning spatial and temporal attention weights. This is because of different activations for learning spatial and temporal attention weights. `Sigmoid` activation for learning spatial attention weights transforms each scalar component of score s_1 in the range [0,1] whereas `Softmax` activation for learning temporal attention weights normalizes the scores s_2 to 1.

In this framework, we improve the efficiency of the our framework by reducing the number of training parameters by K times, where K is the number of human body parts. Separable STA generalizes the spatial attention framework by attending over the RoIs in the convolutional feature map. Moreover, Separable STA also introduces temporal attention driven by poses for features generated from 3D ConvNets.

The limitations that remains - (i) there is no accurate correspondence between the 3D poses and the RGB cues in the process of computing the attention weights [114, 115, 106, 143, 132]; (ii) the attention sub-networks [114, 115, 106, 143, 132] neglect the topology of the human body while computing the attention weights; (iii) the attention weights in [143, 132] provide identical spatial attention along the video. As a result, action pairs with similar appearance like *jumping* and *hopping on one foot* are mis-classified.

Therefore, next we propose a framework based on Video-Pose Network (VPN) to miti-

Algorithm 2 Joint Training of the Separable attention network with video backbone I3D network

Input: RGB video, 3D joint coordinates, model training parameters $N1$ (e.g., $N1 = 10$).

1: Initialize I3D network with model weights trained on IMAGENET and Kinetics.
//**Pre-train I3D network.**
2: Fine-tune I3D network with RGB data from full body body crops.
//**Pre-train Pose backbone which is a 3-layer stacked LSTM.**
3: Train the three layered stacked LSTM network for action classification taking as input 3D skeleton of actors in video frames.
//**Initialize other attention network parameters.**
4: Add a Fully connected layer with $tanh$ activation. This is followed by two branches. One with Fully connected layer to learn s_1 and another with Fully connected layer to learn s_2. Normalise s_1 using a sigmoid activation and s_2 with a softmax activation. Initialize the attention scores with equal values and the remaining network parameters using Gaussian.
//**Jointly Train the Whole Network**
5: Jointly train the separable attention network with the I3D network for $N1$ iterations to obtain the attention scores.

Output: the learned network.

gate the drawbacks of the previous proposed framework and most importantly generalize the use of RGB and 3D pose modalities in an appropriate way.

4.5 Video Pose Network (VPN)

As we have discussed, previous attempts have been made to utilize 3D poses to weight the discriminative parts of a RGB feature map [115, 106, 114, 143, 132]. But these methods have improved the action recognition performance but they do not take into account the alignment of the RGB cues and the corresponding 3D poses. Therefore, we propose a spatial embedding to project the visual features and the 3D poses in the same referential. Before describing this framework, we answer two intuitive questions below.

First, why is spatial embedding important? - The pose driven attention networks can be perceived as guiding networks to help the RGB cues focus on the salient information for action classification. For these guiding networks, it is important to have an accurate correspondence between the poses and RGB data. So, the objective of the spatial embedding is to find correspondences between the 3D human joints and the image regions representing these joints as illustrated in fig 4.8. This task of finding correlation between both modalities can (i) provide informative pose aware feedback to the RGB cues, and (ii) improve the functionalities of the guiding network.

Second, why not performing temporal embedding? - We argue that the need of em-

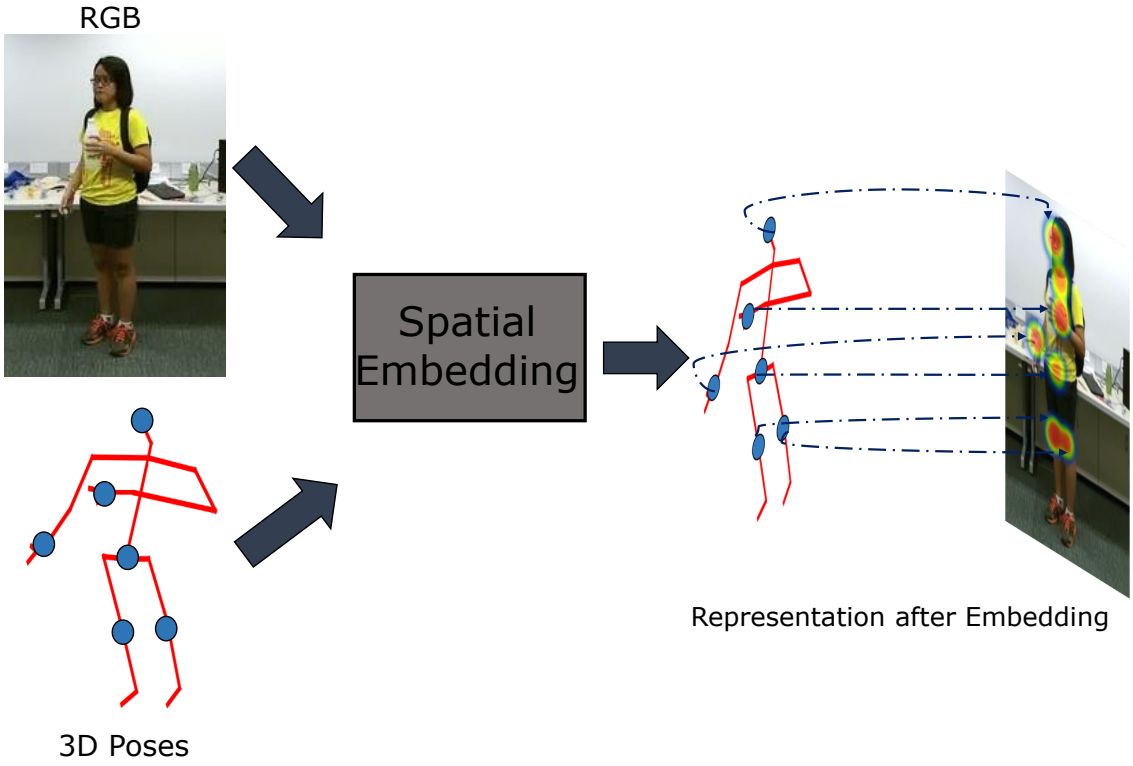


Figure 4.8: Illustration of spatial embedding. Input is a RGB image and its corresponding 3D poses defined in the 3D camera referential. For convenience, we only show 6 relevant human joints. The embedding enforces the human joints to represent the relevant regions in the image.

bedding is to provide proper alignment between the modalities. Across time, the 3D poses are already aligned assuming that there is a 3D pose for every images. However, even if the number of 3D poses does not correspond to the number of image frames (as in [115, 106, 114, 143, 132]), the fact that variance in poses for few consecutive frames is negligible, especially for ADL, implies temporal embedding is not needed.

We propose a recognition framework based on a Video-Pose Network, **VPN** to recognize a large variety of human actions. VPN consists of a spatial embedding and an attention network. VPN exhibits the following characteristics: (i) a spatial embedding learns an accurate video-pose embedding to enforce the relationships between the visual content and 3D poses, (ii) an attention network learns the attention weights with a tight spatio-temporal coupling for better modulating the RGB feature map, (iii) the attention network takes the spatial layout of the human body into account by processing the 3D poses through Graph Convolutional Networks (GCNs).
This framework is end-to-end trainable and our proposed VPN can be used as a layer on top of any 3D ConvNets.

Our objective is to design an accurate spatial embedding of poses and visual content to

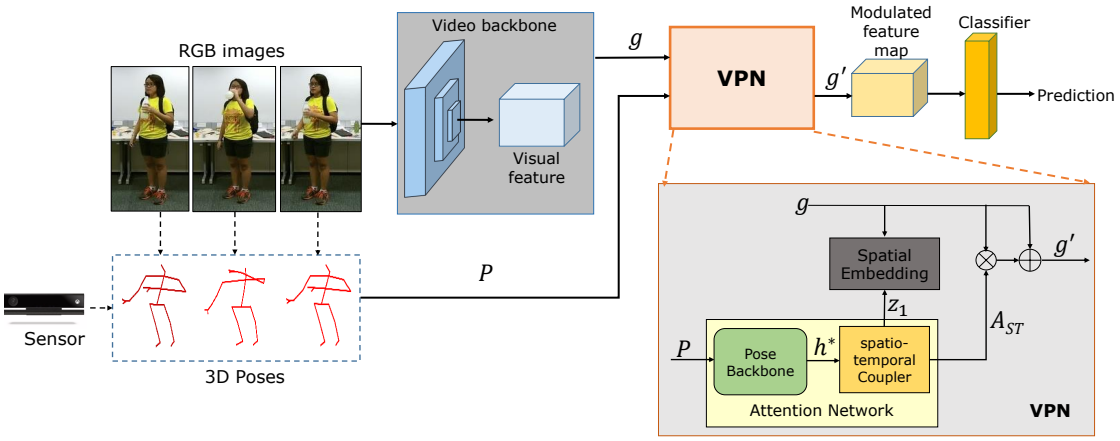


Figure 4.9: **Proposed Action Recognition Framework**: Our framework takes as input RGB images with their corresponding 3D poses. The RGB images are processed by a video backbone which generates a spatio-temporal feature map g. The proposed **VPN** takes as input the feature map g and the 3D poses P. VPN consists of two components: an attention network and a spatial embedding. The attention network consists of a Pose Backbone and a Spatio-temporal Coupler. VPN computes a modulated feature map g'. This modulated feature map g' is then used for classification.

better extract the discriminative spatio-temporal patterns. As shown in fig. 4.9, the input of our proposed recognition model are the RGB images and their 3D poses. The 3D poses are either extracted from depth sensor or from RGB using LCRNet [134]. The framework based on Video-Pose Network **VPN** takes as input the visual feature map and the 3D poses. Below, we discuss the action recognition framework in details.

4.5.1 Video Representation

As in Separable STA, taking as input a stack of human cropped images from a video clip, the spatio-temporal representation g is computed by a 3D convolutional network $f(.)$ (the video backbone in fig. 4.9) with parameters θ using equation 4.1. Then, the feature map g and the corresponding poses P are processed by the proposed network.

4.5.2 VPN components

VPN can be thought as a layer which can be placed on top of any 3D convolutional backbone. VPN takes as input a 3D feature map (g) and its corresponding 3D poses (P) to perform two functionalities. First, to provide an accurate alignment of the human joints with the feature map g. Second, to compute a modulated feature map (g') which is further classified for action recognition. The modulated feature map (g') is weighted along space and time as per its relevance. VPN exploits the highly informative 3D pose information

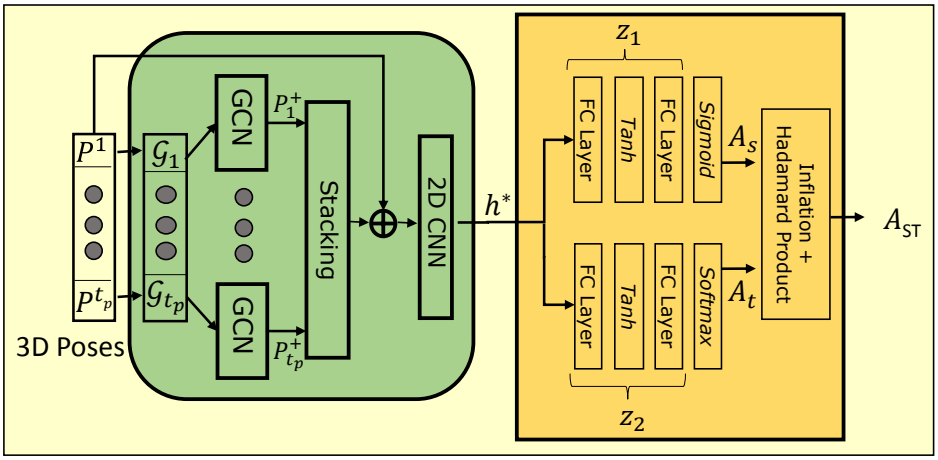


Figure 4.10: We present a zoom of the attention Network with: (A) a GCN Pose Backbone, and (B) a spatio-temporal Coupler to generate spatio-temporal attention weights A_{ST}

to transform the visual feature map g through spatial embedding and finally, compute the attention weights. This network has two major components as shown in fig 4.9: (I) an attention network and (II) a spatial embedding. Though the intrinsic parameters of the attention network and the spatial embedding learn in parallel, we present these two components in the following order for better understanding.

4.5.2.1 Attention Network

The attention network consists of a Pose Backbone and a spatio-temporal Coupler as shown in fig. 4.10. Such a framework for pose driven attention network is unique compared to the other state-of-the-art methods using poses and RGB. The proposed attention network unlike [115, 106], P-I3D and Separable STA takes into account the human spatial configuration and it also learns coupled spatio-temporal attention weights for the visual feature map g.

Pose Backbone - The input poses along the video are processed in a Pose Backbone. The pose based input of VPN are the 3D human joint coordinates $P \in \mathbb{R}^{3\times J\times t_p}$ stacked along t_p temporal dimension, where J is the number of skeleton joints. The Pose Backbone processes these 3D poses to compute pose features h^* which are used further in the attention network for computing the spatio-temporal attention weights. They carry meaningful information in a compact way, so the proposed attention network can efficiently focus on salient action parts.

For the Pose Backbone, we use Graph Convolutional Networks (**GCNs**) to learn the spatial relationships between the 3D human joints to provide attention weights to the visual feature map g. We aim at exploiting the graphical structure of the 3D poses. In fig. 4.10, we illustrate our GCN pose backbone (marked (A)). For each pose input $P_t \in \mathbb{R}^{3\times J}$ with J joints, we first construct a graph $\mathcal{G}_t(P_t, E)$ where E is the $J \times J$ weighted adjacency matrix:

$$e_{ij} = \begin{cases} 0, & \text{if } i = j \\ \alpha, & \text{if joint i and joint j are connected} \\ \beta, & \text{if joint i and joint j are disconnected} \end{cases}$$

These connections defined in the adjacency matrix are obtained from the skeleton anatomy which is specific depending on the pose generation algorithms. For instance, if the poses are obtained from Kinect sensor, the anatomy differs from the one computed using algorithms like LCRNet [134]. Each graph $\mathcal{G}_t$ at time t is processed by a GCN to compute feature P_t^+:

$$P_t^+ = D^{-\frac{1}{2}}(E + I)D^{-\frac{1}{2}}\mathcal{G}_t W_t, \tag{4.14}$$

where W_t is the weight matrix and D is the diagonal degree matrix with $D_{ii} = \Sigma_j(E_{ij}+I_{ij})$ its diagonal elements. For all $t = 1, 2, ..., t_p$, the GCN output features P_t^+ are stacked along time, resulting in a 3D tensor $[P_1^+, P_2^+, ..., P_{t_p}^+]$.

Finally, the 3D pose tensor is combined with the original pose input by a residual connection followed by a set of convolutional operations. Now, the GCN pose backbone provides salient features h^* because of its use of the graphical structure of the 3D joints.

Spatio-temporal Coupler - The attention network in VPN learns the spatio-temporal attention weights from the output of Pose Backbone in two steps as shown in fig. 4.10 (B). In the first step, the spatial and temporal attention weights (A_S and A_T) are classically trained as in [113] to get the most important body part and key frames for an action. The output feature h^* of Pose Backbone follows two separate non-linear mapping functions to compute the spatial and temporal attention weights. These spatial A_S and temporal A_T weights are defined as

$$A_S = \sigma(z_1); \qquad A_T = softmax(z_2) \tag{4.15}$$

where $z_r = W_{z_r} tanh(W_{h_r} h^* + b_{h_r}) + b_{z_r}$ (for $r = 1, 2$) with sub-scripted W and b, the corresponding weights and biases are the latent spatial and temporal attention vectors. The dissociated attention weights A_S and A_T having dimension $m \times n$ and t_c respectively, can undergo a linear mapping to obtain spatially and temporally modulated feature maps. The resultant model is equivalent to the separable STA framework. In contrast, now we

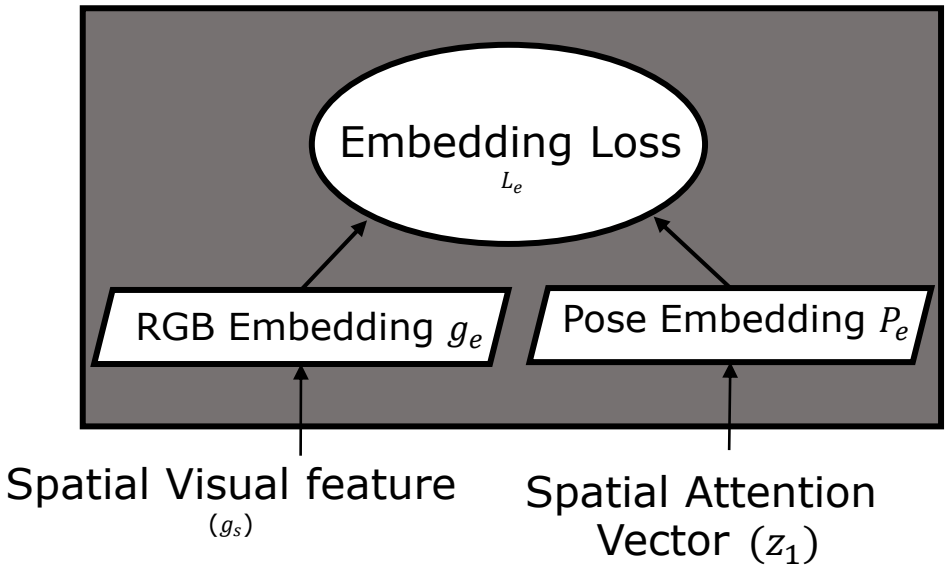


Figure 4.11: The spatial Embedding computing loss L_e. This back-propagates through the visual cue and the pose backbone.

propose to further perform a coupling of the spatial and temporal attention weights. Thus in the second step, joint spatio-temporal attention weights are computed by performing a Hadamard product on the spatial and temporal attention weights. In order to perform this matrix multiplication, the spatial and temporal attention weights are inflated by duplicating the same attention weights in temporal and spatial dimension respectively. Hence, the $m \times n \times t_c$ dimensional spatio-temporal attention weights A_{ST} are obtained by

$$A_{ST} = inflate(A_S) \circ inflate(A_T) \tag{4.16}$$

This two-step attention learning process enables the attention network to compute spatio-temporal attention weights in which the spatial saliency varies with time. The obtained attention weights are crucial to disambiguate actions with similar appearance as they may have dissimilar motion over time.

Finally, the spatio-temporal attention weights A_{ST} are linearly multiplied with the input video feature map g, followed by a residual connection with the original feature map g to output the modulated feature map g'. The residual connection enables the network to retain the properties of the original visual features.

4.5.2.2 Spatial Embedding of RGB and Pose

The objective of the embedding model is to provide tight correspondences between both pose and RGB modalities used in VPN. The previous two attention mechanisms (P-I3D and Separable STA) attempt to provide the attention weights on the RGB feature map using 3D pose information without projecting them into the same 3D referential. The mapping with the pose is only done by cropping the person within the input RGB images. The spatial attention computed through the 3D joint coordinates does not correspond to the part of the image (no pixel to pixel correspondence), although it is crucial for recognizing fine-grained actions. To correlate both modalities, an embedding technique inspired from image captioning task [150, 151] is used to build an accurate RGB-Pose embedding in order to enable the poses to represent the visual content of the actions (see fig. 4.11).

We assume that a low dimensional embedding exists for the global spatial representation of video feature map $g_s = \Sigma_{i=1}^{t_c} g(i,:,:,:)$ (a D_v-dimensional vector) and its corresponding pose based latent spatial attention vector z_1 (a D_p-dimensional vector). The mapping function can be derived from this embedding by

$$g_e = T_v g_s \quad and \quad P_e = T_p z_1, \tag{4.17}$$

where $T_v \in R^{D_e \times D_v}$ and $T_p \in R^{D_e \times D_p}$ are the transformation matrices that project the video content and the 3D poses into the common D_e dimensional embedding space. This mapping function is applied on the global spatial representation of the visual feature map and the pose based features in order to attain the aforementioned objective of the spatial embedding.

To measure the correspondence between the video content and the 3D poses, we compute the distance between their mappings in the embedding space. Thus, we define an embedding loss as a hyper-sphere feature metric space

$$L_e = ||\widehat{T_v g_s} - \widehat{T_p z_1}||_2^2 \qquad s.t. \;\; ||T_v||_2 = ||T_p||_2 = 1 \tag{4.18}$$

$\widehat{T_v g_s} = \frac{T_v g_s}{||T_v g_s||_2}$ and $\widehat{T_p z_1} = \frac{T_p z_1}{||T_p z_1||_2}$ are the feature representations projected to the unit hyper-sphere. The norm constraint $||T_v||_2 = 1$ & $||T_p||_2 = 1$ simply prevents the trivial solution $\hat{T_v} = \hat{T_p} = 0$. In equation 4.18, $\widehat{T_v g_s} = \frac{T_v g_s}{||T_v g_s||_2} = \frac{g_e}{||g_e||_2}$ and $\widehat{T_p z_1} = \frac{T_p z_1}{||T_p z_1||_2} = \frac{P_e}{||P_e||_2}$ are the feature representations projected to the unit hypersphere. Here, we compute the norm $||g_e||_2$ and $||P_e||_2$ using

$$||g_e||_2 = \sqrt{\Sigma_i g_{e_i}^2 + \varepsilon} \qquad \& \qquad ||P_e||_2 = \sqrt{\Sigma_i P_{e_i}^2 + \varepsilon} \tag{4.19}$$

where ε is a small positive value to prevent dividing by zero.

The embedding loss L_e along with the global classification loss provide a linear transformation on the RGB feature map that preserves the low-rank structure for the action representation and introduces a maximally separated features for different actions. Now, the kernels at the visual backbone are updated with a gradient proportional to $(g_e - P_e)$, which in turn transforms the visual feature map to learn pose aware characteristics. Consequently, we strengthen the correspondences between video and poses by minimizing the embedding loss. This embedding ensures that the pose information to be used for computing the spatial attention weights aligns with the content of the video.

Note that the embedding loss also provides feedback to the pose based latent spatial attention vectors (z_1), which in turn transfers knowledge from the 2D image space to pose 3D referential. This allows the attention network to provide better and meaningful spatial attention weights (A_s) compared to the attention network without the embedding. We will quantify this observation in the experiments.

4.5.3 Training jointly the 3D ConvNet and VPN

VPN can be trained as a layer on top of any 3D ConvNet. The 3D ConvNet can be pretrained for the action classification task for faster convergence. Finally, VPN is plugged into the 3D ConvNet for an end-to-end training as presented in algorithm 3 with a regularized loss L formulated as

$$L = \lambda_1 L_C + (1 - \lambda_1) L_e + \lambda_2 L_a \tag{4.20}$$

Here, L_C is the cross-entropy loss, L_e is the embedding loss; the trade-off between these two losses is captured by linear fusion with a positive parameter λ_1; L_a is the attention regularizer with λ_2 weighting factor. The attention regularizer consists of the spatial and temporal attention weight regularizer and is formulated as

$$L_a = \sum_{j=1}^{m \times n} ||A_s(j)||_2 + \sum_{j=1}^{t_c} (1 - A_{t_c}(j))^2 \tag{4.21}$$

This additional regularization term L_a ensures that the attention weights are not biased to provide extremely high values to the parts of the spatio-temporal feature map with more relevance and completely neglecting the other parts. The different regularization constraint for learning the spatial and temporal attention weights follow the same strategy as in Separable STA.

Hence, in this chapter we present three novel action recognition frameworks based on pose driven attention mechanism. Our action recognition framework has evolved from using pose driven attention mechanism for selecting human body parts to a general mech-

Algorithm 3 Joint Training of VPN with video backbone I3D network

Input: RGB video, 3D joint coordinates, model training parameters $N1$ (e.g., $N1 = 10$).

1: Initialize I3D network with model weights trained on IMAGENET and Kinetics.
//Pre-train I3D network.
2: Fine-tune I3D network with RGB data from full body body crops.
//Incorporate the Pose backbone which is a stack of t_p GCNs.
//Initialize the attention network parameters.
3: Add a Fully connected layer with $tanh$ activation. This is followed by two branches. One with Fully connected layer to learn z_1 and another with Fully connected layer to learn z_2. Normalise z_1 using a sigmoid activation and z_2 with a softmax activation to yield A_S and A_T. Then, perform an inflation operation followed by a hadamard product of the attention weights to compute the final spatio-temporal attention weights A_{ST}. Initialize the attention scores with equal values and the remaining network parameters using Gaussian.
//Jointly Train the Whole Network
4: Jointly train VPN with the I3D network for $N1$ iterations to obtain the attention scores.

Output: the learned network.

anism within full human body representation. We incorporated spatial and temporal attention mechanisms by providing a tight coupling between them. Finally, in VPN we also provide an accurate embedding between the 3D poses and the RGB cue to enforce them in common semantic space.

4.6 Experiments

In this section, we validate the effectiveness of our proposed action recognition frameworks on public datasets. First, we discuss the implementation details of each framework for conducting the experiments.

4.6.1 Implementation details

Training - For all the frameworks, we initialize the **video backbone** - I3D from the Kinetics-400 [24] + ImageNet [20] classification models. Data augmentation and training procedure for training the I3D on tracks of human body follow [3]. For fine-tuning the video backbone independently, we use SGD optimizer with an initial learning rate of 0.01. We use a regularization with a weighting factor of 0.001 between the penultimate layer and the classification layer in I3D. The video backbone takes $T = 64$ video frames as input. These 64 frames are sampled from the whole video clip with the starting frame randomly selected within the first half of the video. All the subsequent frames are sampled with a stride of 2. In case, the size of the video is less than 64, we loop around the video, i.e.

continue stacking up the frames from the beginning of the video. Feature map g in the video representation stage of the frameworks, is extracted from the output of `GAP` layer (in case of P-I3D) or output of `Mixed_5c` (in case if Separable STA and VPN). Hence, the dimension of the feature map g which is modulated is $t_c = 7$, $m \times n = 1 \times 1$ and $c = 1024$ for P-I3D and $t_c = 8$, $m \times n = 7 \times 7$ and $c = 1024$ for Separable STA and VPN.

Now, next is incorporating the **attention network** with the video backbone. The **pose backbone** takes as input a sequence of t_p 3D poses uniformly sampled from each clip. The **pose backbone** is three layer stacked LSTM in Spatial attention and Separable STA framework while it is a single layer t_p temporally stacked GCNs in VPN framework. Hyper-parameter $t_p = 20, 20, 30$ and 5 for NTU-60, NTU-120, Smarthome and N-UCLA respectively.

For pose backbone in P-I3D and Separable-STA, each LSTM layer consists of 512, 512, 128 and 128 LSTM units (neurons) for NTU-60, NTU-120, Smarthome and NUCLA respectively. The LSTMs have t_p time steps each of which yields an output of dimension equal to the number of neurons. The output vector from each time step is concatenated to compute h^*.

For pose backbone in VPN, we use t_p number of GCNs, each processing a pose from the sequence. The weighting parameters α and β for computing the adjacency matrix of the pose based graph are set to 5 and 2 respectively. GCN projects the input joint coordinates to a $64 - dimensional$ space. The output of the GCN is passed to a set of convolutional operations (see fig. 4.10(I)(A)) which consists of three 2D convolutional layers each is followed by a Batch Normalization layer and a ReLU layer to compute h^*. The output channels of the convolutional layers are 64, 64 and 128.

Hyper-parameter settings - For **P-I3D**, we perform all the experiments with $k = 3$, i.e. three human body parts. We also set λ_1 to 0.00001 for NTU-60, NTU-120, and Smarthome and 0.0001 for N-UCLA datasets respectively, and λ_2 to 0.001 for all the datasets. For **Separable STA**, we set λ_1 & λ_2 to 0.00001 for all the datasets. For **VPN**, the trade off (λ_1) and regularizer (λ_2) parameters are set to 0.8 and 0.00001 respectively for all the datasets.

For training the entire global network for the task of classification, a global-average pooling layer followed by a dropout [140] of 0.3 and a `softmax` layer are added at the end of the network for class prediction. The network is trained with a 4-GPU machine where each GPU has 4 video clips in a mini-batch. We sample 10% of initial training set as a validation set, for hyper-parameters optimization and for early stopping. Our network is trained for 30 epochs in total, with Adam optimizer [152] having initial learning rate of 0.01 and decay rate of 0.1 after every 10 epochs.

Inference. For the classification at test time, we perform fully convolutional inference in spatial space as in [85] for Separable STA and VPN. For P-I3D, we take the body crops depending on K at training time. The classification is performed for all stack of 64 frames in a video to cover the temporal space as well. The final classification is obtained by max-pooling the `softmax` scores.

We present a summary of the implementation details in table 4.1.

Table 4.1: Summary of implementation details for P-I3D, Separable STA and VPN

Components	**P-I3D**	**Separable STA**	**VPN**
Video backbone & input dimension	I3D $64 \times 224 \times 224 \times 3$	I3D $64 \times 224 \times 224 \times 3$	I3D $64 \times 224 \times 224 \times 3$
Extraction Layer of g & output dimension	GAP $7 \times 1 \times 1 \times 1024$	Mixed_5c $8 \times 7 \times 7 \times 1024$	Mixed_5c $8 \times 7 \times 7 \times 1024$
Pose backbone & input dimension	3-layer LSTM $t_p \times (3 * J)$	3-layer LSTM $t_p \times (3 * J)$	GCN $t_p \times (3 * J)$
hyper-parameters	$\lambda_1 \in [0.0001, 0.00001]$ $\lambda_2 = 0.001$	$\lambda_1 = 0.00001$ $\lambda_1 = 0.00001$	$\lambda_1 = 0.8$ $\lambda_1 = 0.00001$

4.6.2 Ablation study of Spatial attention (P-I3D)

In this section, we show the effectiveness of P-I3D by performing ablation studies on NTU-60 and NUCLA datasets. Table 4.2 and 4.3 show the performance of different image patches based on tracks of human body parts. The statistics show a considerable improvement in the classification accuracy on focusing at the individual body parts rather than using the whole images and thus including the unnecessary background information. In table 4.2, we also quantitatively analyze the best position in the I3D [3] network to aggregate the latent spatio-temporal features from the different human body parts. By sum_r, we mean the aggregation of the spatio-temporal features after $(9 - r)$ inception blocks pre-trained on individual body parts in I3D and then using r inception blocks to further extract meaningful information from the aggregated features. Our observation depicts that aggregation at the last inception block without the need of further inception blocks best models the action implying that aggregation of high-level rich features trained on individual body parts does not need further 3D convolutional operations to extract distinguishable spatio-temporal features. For aggregation, we explore the use of summation (sum_r) and concatenation ($concat\_r$) operator at the end of I3D network (since concatenation at earlier layers is not feasible because of curse of dimensionality). Experimental results (in table 4.2 and 4.3) show the effectiveness of summation operation of spatio-

temporal features unlike the usual concatenation operation of spatial features as in [114]. In addition, table 4.3 also shows the effectiveness of using NTU-60 pre-training for NUCLA. This corroborates the fact that how much important is the step of pre-training of such deep attention network in order to be effective on small datasets. Finally, we also present the action classification accuracy using our proposed attention mechanism given that the body part features are aggregated by a summation operation. We denote this final result by *sum + attention* in table 4.2 and 4.3. A significant improvement in the action classification accuracy on incorporating the attention mechanisms shows its efficacy for learning discriminative representation for ADL.

Table 4.2: Ablation study on NTU RGB+D dataset with Cross-Subject (CS) and Cross-View (CV) protocol. The values denote action classification accuracy (in %)

Methods	CS	CV	Avg
Full image	70.93	80.53	75.73
Left hand	84.31	84.75	84.53
Right hand	82.94	81.83	82.38
Full body	85.47	87.26	86.36
sum_2	89.30	92.02	90.66
sum_1	90.39	92.19	91.29
sum_0	90.8	92.5	91.65
concat_0	89.05	92.07	90.56
sum+attention	93	95.4	94.2

Table 4.3: Ablation study on Northwestern-UCLA Multi-view Action 3D with Cross-View $V^3_{1,2}$ protocol. The values denote action classification accuracy (in %)

Methods	$V^3_{1,2}$	$V^3_{1,2}$ (NTU pre-trained)
Full image	83.95	87.93
Left hand	77.37	80.60
Right hand	78.50	80.38
Full body	85.99	88.79
sum_0	86.80	91.37
concat_0	86.63	90.30
sum+attention	87.50	93.10

Effectiveness of the Proposed Attention Model - In this section, we define the notion of successful and unsuccessful attention scores for a video by looking at the body part based classification accuracy for the corresponding action category. For instance, if an action **X**

is classified with a higher average action classification accuracy using a human body part P_1 compared to other human parts, then the average attention score of body part P_1 for action X should be higher than the other parts. If the aforementioned condition is not satisfied, we call it an unsuccessful attention.

In fig. 4.12, we illustrate the attention scores for some representative action categories. In fig. 4.13, we illustrate the corresponding average classification accuracy of each human body part, their aggregation and proposed attention framework for the same action categories illustrated in fig. 4.12. The body part with highest classification accuracy is correctly assigned with attention weights resulting in improved classification accuracy of our proposed *sum + attention* network. However, the other body parts may not receive meaningful attention scores. The activity regularizer dynamically focus on all the body parts which in turn may overlap with one another.

In fig. 4.14, we illustrate the unsuccessful attention scores for some representative action categories. In fig. 4.15, we show the effect of unsuccessful attention scores on our P-I3D framework. Failure in computing the right spatial attention on human body parts does not affect the framework and performs similar to the aggregation framework (with no attention) for actions like "*drinking water*" and "*brushing teeth*". This is because of the dominance of all the human body parts involved in the action. For action like "*touching head*", unsuccessful attention delivered to the appearance based spatio-temporal features degrades the performance of the whole network.

In fig. 4.16, we show some sample visualization of the human body parts with their respective attention scores. The actions are *drinking*, *kicking*, *brushing hair* where the left hand, full body and again the left hand respectively seem to be relevant for classifying the action. The corresponding attention scores show the correctness of the attention mechanism.

4.6.3 Ablation study of Separable STA

In this section, we perform an ablation study on Separable STA to show the superiority of this framework compared to its baselines. Here, we find answer to this question: Why this attention network: Separable STA offers dissociated attention weights?

Table 4.4 evaluates other strategies to implement the proposed attention mechanism. Among the strategies we include the implementation of single attention mechanisms (spatial or temporal) and all the different ways to combine them. The strategies included in the study are: I3D base network with (1) no attention (No Att); (2) only $m \times n$ dimensional spatial attention (SA); (3) only t dimensional temporal attention (TA); (4) temporal attention applied after SA (SA+TA); (5) spatial attention applied after TA (TA+SA); and with (6) $m \times n \times t$ spatio-temporal attention coupled together from pose driven model (joint STA). For the implementation of SA+TA and TA+SA, we adopt the joint training

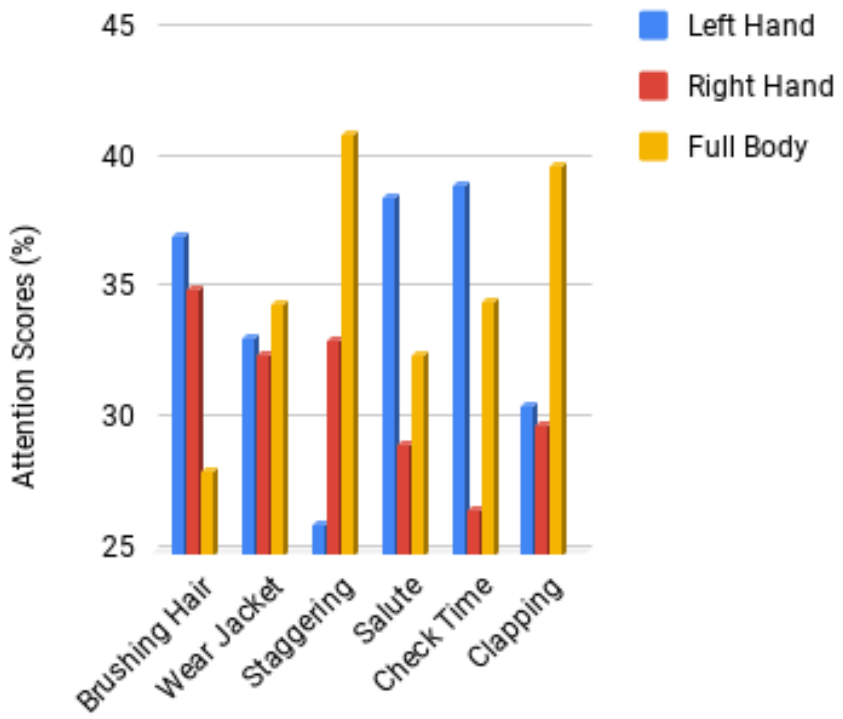


Figure 4.12: Examples of successful attention scores on some action categories.

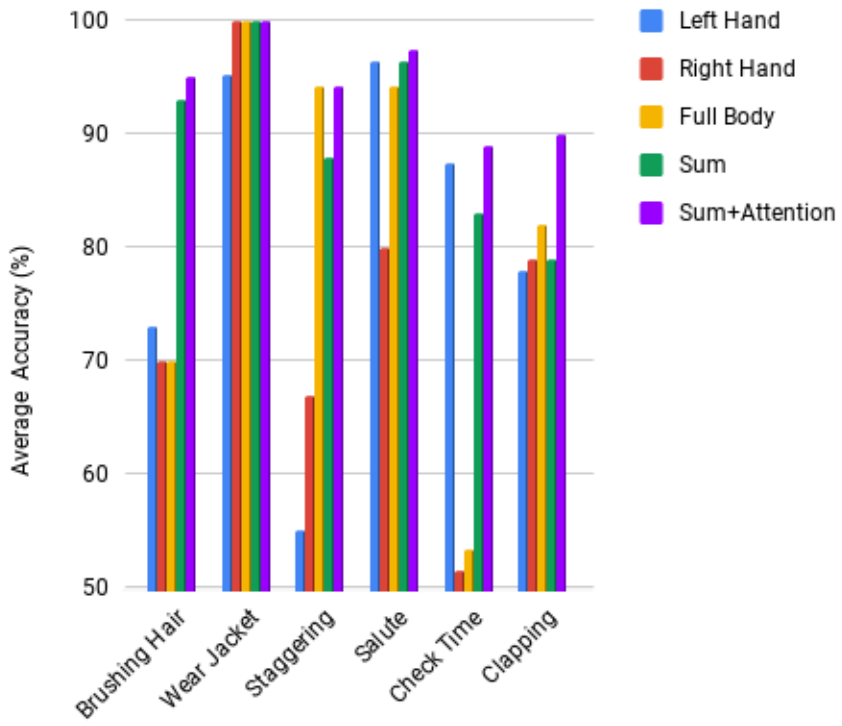


Figure 4.13: Examples of average classification accuracy on individual body parts, their aggregation and our proposed attention network on action categories presented in fig. 4.12

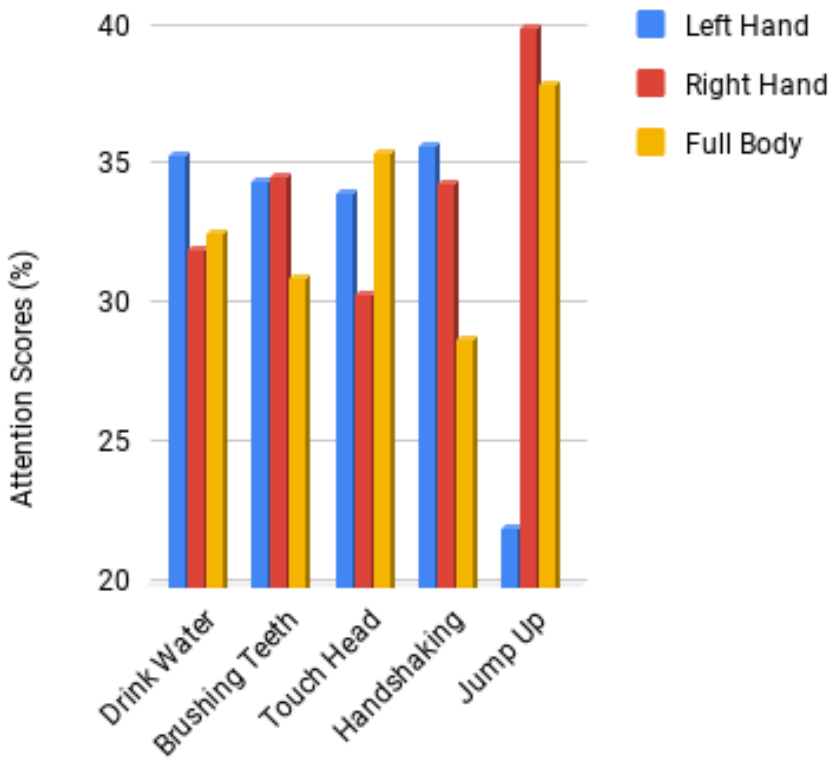


Figure 4.14: Examples of unsuccessful attention scores on some action categories.

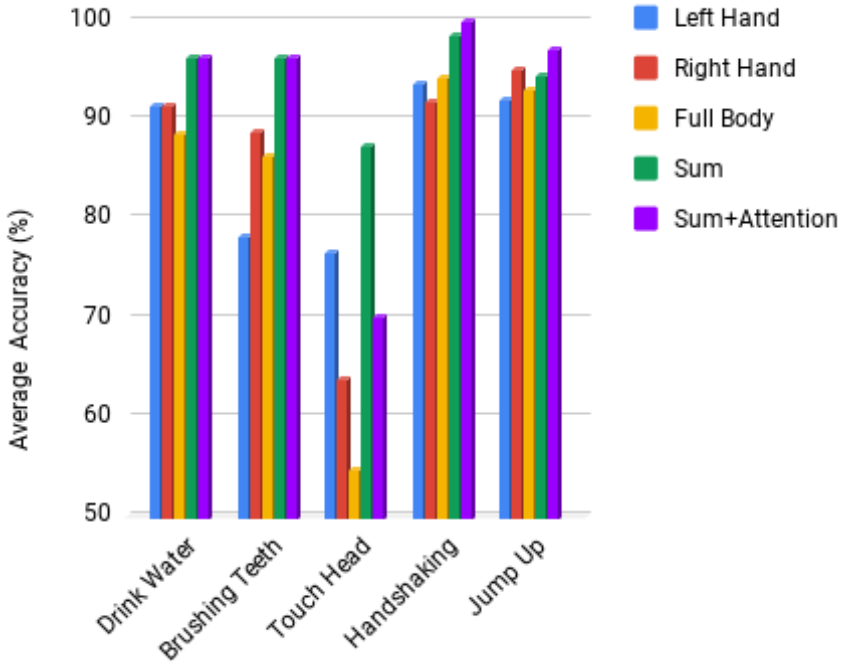


Figure 4.15: Examples of average classification accuracy on individual body parts, their aggregation and our proposed attention network on action categories presented in fig. 4.14

mechanism proposed in [113]. Our proposed separable STA outperforms all other strategies by a significant margin. It is interesting to note that coupling spatial and temporal attention as in joint STA for 3D ConvNets decreases the classification accuracy. The reason for this can be seen from the classification accuracy achieved by SA and TA separately on the different datasets. In Smarthome and NUCLA, spatial attention is much more effective than temporal attention because several activities of both datasets involve interactions with objects. On the other hand, NTU contains activities with substantial motion (such as *kicking*, *punching*) and human-object interaction. Therefore, both spatial and temporal attention contribute to improve the classification accuracy. However, the possibility for the second attention to significantly modify the I3D feature maps is limited once the first attention has modified it. For this reason, we believe that dissociating both attention mechanisms is more effective than coupling them in series.

Comparison of Separable STA with baseline I3D - Figure 4.17 compares I3D base network with or without **separable STA**. The comparison is based on the per-class accuracy improvement on Smarthome and NTU-CS (cross-subject protocol). For Smarthome, the **spatial attention** alone contributes to a large improvement due to the ability to recognize fine-grained activities involving interactions with objects, such as *Pour.fromkettle*

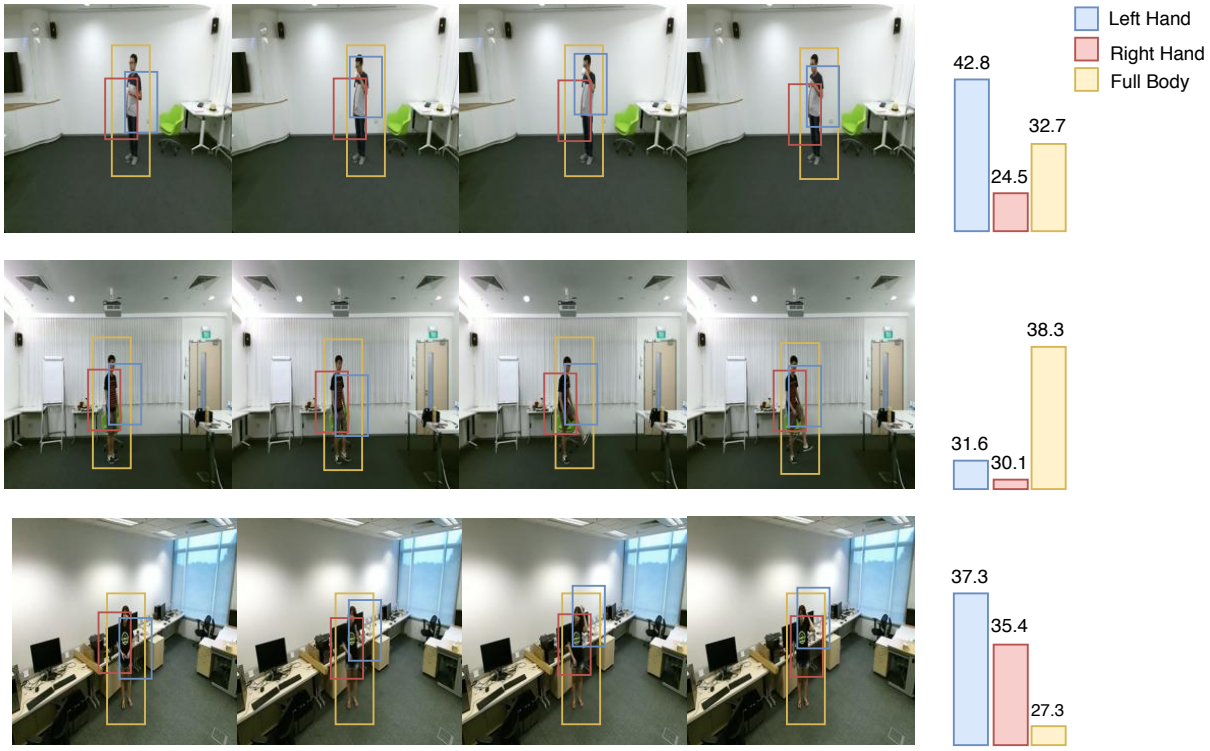


Figure 4.16: Example of video sequences with their respective attention scores. The action categories presented are drinking water with left hand (1st row), kicking (2nd row) and brushing hair with left hand (last row).

Table 4.4: Action classification accuracy (in %) on NTU, NUCLA and Smarthome datasets to show the effectiveness of our proposed separable spatio-temporal attention mechanism (separable STA) in comparison to other strategies. No Att indicates no attention.

Datasets	No Att	SA	TA	SA+TA	TA+SA	Joint STA	Separable STA
NTU-CS	85.47	90.46	90.76	89.07	90.01	90.30	**92.20**
NTU-CV	87.26	93.69	91.25	92.39	92.60	92.48	**94.61**
NUCLA	85.47	90.09	79.31	74.57	74.35	87.93	**92.46**
Smarthome-CS	72.09	73.13	70.30	71.25	70.40	71.68	**75.31**
Smarthome-CV_1	56.61	60.27	43	41.94	40.93	55.71	**61.06**
Smarthome-CV_2	61.58	66.36	57.03	58.31	56.61	61.95	**68.25**

(+21.4%) for CS and *Uselaptop* (+13.4%), *Eat.snack* (42.8%) for CV. The **temporal attention** improves the classification of activities with low and high motion. Examples of this are static activities such as *WatchTV* (+8.8%) for CS and *Readbook* (+9.6%) for CV; and dynamic activities such as *sitdown* (+22.2%). For NTU-CS, the largest accuracy gains are observed for *brushing hair* (+28.2%), *taking off a shoe* (+23.3%) and *cross hands in front* (+20.6%). These are activities in which the distinctive features are localized in space and time. Even for those classes for which our separable STA performs worse than I3D alone, the accuracy drop is very limited.

But, what we ignore with Separable STA is handling noisy poses, which is crucial for such pose-driven attention mechanisms. Recognizing actions with subtle motion like *cutting bread*, *using stove* under constraints of the subject being occluded or not in the middle of the frame, still remains a challenge. This is where VPN provides a spatial embedding to handle the mis-alignment of noisy 3D poses.

4.6.4 Ablation Study of VPN components

In this section, we perform an ablation study to show the effectiveness of the VPN components. The presence of ADL challenges like fine-grained and similar appearance activities is in higher magnitude in NTU-120 and Smarthome datasets. So, we perform all our ablation studies on these two datasets. Our VPN based framework includes two novel components, the spatial embedding and the attention network. Both of them are critical for good performance on ADL recognition. We show the importance of the attention network and the spatial embedding of VPN in table 4.5. We also show the effectiveness of the spatial embedding with different instantiation of the attention network in table 4.6.

How effective is VPN? In order to answer this point, we show the action classification accuracy with baseline I3D (l_1) which is the video backbone and then incorporate the VPN components: the attention network (l_2) and the spatial embedding (l_3) one-by-one in table 4.5. The attention network (l_2) improves significantly the classification of the actions (relatively upto 10.9% on NTU-120 and 11.9% on Smarthome) by providing spatio-

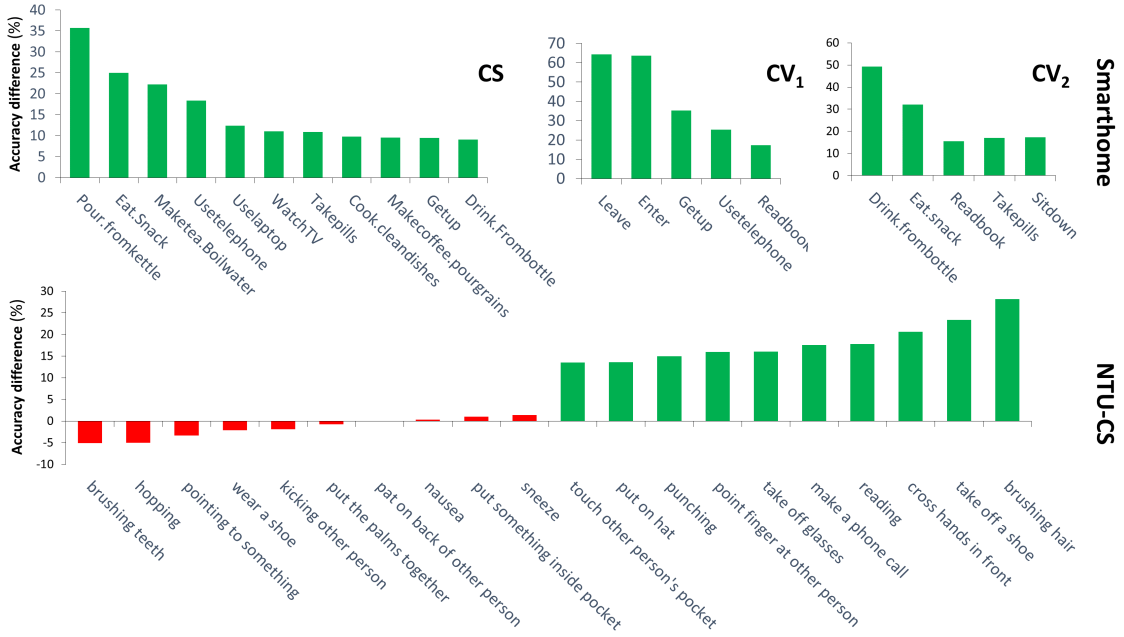


Figure 4.17: Per-class accuracy improvement on Smarthome and NTU-CS when using separable STA in addition to I3D. For Smarthome, we present the top 11, top 5 and top 5 classes for CS, CV_1 and CV_2 respectively. For NTU-CS, we present the 10 best and 10 worst classes.

Table 4.5: Ablation study to show the effectiveness of each VPN component.

VPN components	NTU-120 CS_1	NTU-120 CS_2	Smarthome CS	Smarthome CV_2
l_1: visual backbone	77.0	80.1	53.4	45.1
l_2: l_1 + attention network	85.4	86.9	56.4	50.5
l_3: l_2 + spatial embedding	**86.3**	**87.8**	**60.8**	**53.5**

Table 4.6: Performance of VPN with different choices of Attention Network.

Model	Pose Backbone	Coupler	NTU-120 CS_1	NTU-120 CS_2	Smarthome CS	Smarthome CV_2
l_4: VPN	LSTM	×	84.7	83.6	57.1	50.6
l_5: VPN	GCN	×	85.6	86.8	60.1	53.1
l_6: VPN	LSTM	✓	85.3	84.1	57.6	51.5
l_7: VPN	GCN	✓	**86.3**	**87.8**	**60.8**	**53.5**

Table 4.7: Performance of VPN with different embedding losses l_e.

Loss	NTU-120 CS_1	NTU-120 CS_2	Smarthome CS	Smarthome CV_2
KL-divergence $D_{KL}(f_e\|\|P_e)$	85.5	87.1	57.2	50.9
KL-divergence $D_{KL}(P_e\|\|f_e)$	85.6	86.9	57.0	51.1
Bi-directional KL-divergence	86.1	87.2	57.2	51.7
Normalized Euclidean loss	**86.3**	**87.8**	**60.8**	**53.5**

temporal saliency to the I3D feature maps. With the spatial embedding (l_3), the action classification further improves (relatively upto 1% on NTU-120 and 7.8% on Smarthome).

Diagnosis of the attention network - In table 4.6, we further illustrate the importance of each component in the attention network, i.e. the Pose Backbone and the spatio-temporal coupler. We have designed a baseline attention network with LSTM as pose backbone following our previous frameworks in P-I3D and Separable STA. We compare the LSTM pose backbone in l_4 and l_6 with our proposed GCN instantiation in l_5 and l_7. The attention network without a spatio-temporal coupler provides dissociated spatial and temporal attention weights in l_4 and l_5 in contrast to our proposed coupler in l_6 and l_7. Firstly, we observe that the GCN pose backbone makes use of the human joint topology, thus improves the classification accuracy in all scenarios with or without the coupler. Consequently, actions like *Snapping Finger* (+24.5%) and *Apply cream on face* (+23.9%) improves significantly with GCN instantiation (l_6) compared to LSTM (l_7). Secondly, we observe that the spatio-temporal coupler provides fine spatial attention weights for the most important frames in a video, which enables the model to disambiguate actions with similar appearance but dissimilar motion. Consequently, the coupler (l_7) improves the classification accuracy up to 1.1% both on NTU-120 and Smarthome w.r.t. dissociating the attention weights (l_5). For instance, with dissociation of the attention weights, *rubbing two hands* was confused with *clapping* and *flicking hair* was confused with *putting on headphone*. With VPN, the coupler improves the classification accuracy of actions *rubbing two hands* and *flicking hair* by 25% and 19.6% respectively.

Table 4.8: Impact of Spatial Embedding on Spatial Attention. Note that all the models use spatial attention mechanism.

Model	Pose Backbone	Spatial Embedding	Temporal Attention	NTU-120 CS_1	NTU-120 CS_2	Smarthome CS	Smarthome CV_2
VPN	LSTM	×	×	81.7	81.2	45.5	50.0
VPN	LSTM	✓	×	82.7	82.0	56.5	52.6
VPN	GCN	×	×	82.6	84.3	49.1	51.7
VPN	GCN	✓	×	**83.1**	**85.3**	**58.4**	**53.1**
VPN	GCN	✓	✓	**86.3**	**87.8**	**60.8**	**53.5**

Which loss is better for learning the spatial embedding? In this ablation study (Table 4.7), we compare different losses for projecting the 3D poses and RGB cues in a common semantic space. First, we compare the KL-divergence losses [153, 102] ($D_{KL}(g_e||P_e)$ and $D_{KL}(P_e||g_e)$) from P_e to g_e and vice-versa. The KL-divergence losses $D_{KL}(g_e||P_e)$ and $D_{KL}(P_e||g_e)$ for n samples are computed by

$$D_{KL}(g_e||P_e) = \sum_{i=1}^{n} f_e^i log(\frac{g_e^i}{P_e^i}) \tag{4.22}$$

$$D_{KL}(P_e||g_e) = \sum_{i=1}^{n} P_e^i log(\frac{P_e^i}{g_e^i}) \tag{4.23}$$

where f_e^i and P_e^i are visual and pose embedding of the i^{th} input sample. Then, we compare a bi-directional KL-divergence loss [154, 155, 156] ($D_{KL}(f_e||P_e)$ + $D_{KL}(P_e||f_e)$) to our normalized euclidean loss. We observe that (i) the loss using $D_{KL}(f_e||P_e)$ and $D_{KL}(P_e||f_e)$ deteriorates the action classification accuracy as the feedback is in one direction either towards RGB or poses, implying two-way feedback for the visual features and the attention network is necessary, (ii) our normalized euclidean loss outperforms the bi-directional KL divergence loss, exhibit its superiority.

Impact of Embedding on Spatial attention - In table 4.8, we show the impact of spatial embedding on the attention network providing spatial attention only. We perform the experiments with different choice of Pose Backbone, i.e. LSTM as in P-I3D or Separable STA and our proposed GCN. The spatial embedding provides a tight correspondence between the RGB data and poses. As a result, it boosts the classification accuracy in all the experiments. It is worth noting that the improvement is significant for Smarthome as it contains many fine-grained actions with videos captured by fixed cameras in an unconstrained Field of View for the person performing the action. Thus, enforcing the embedding loss enhances the spatial precision during inference. As a result, the classification accuracy of fine-grained actions like *pouring water* (+77.7%), *pouring grains* (+76.1%) for making

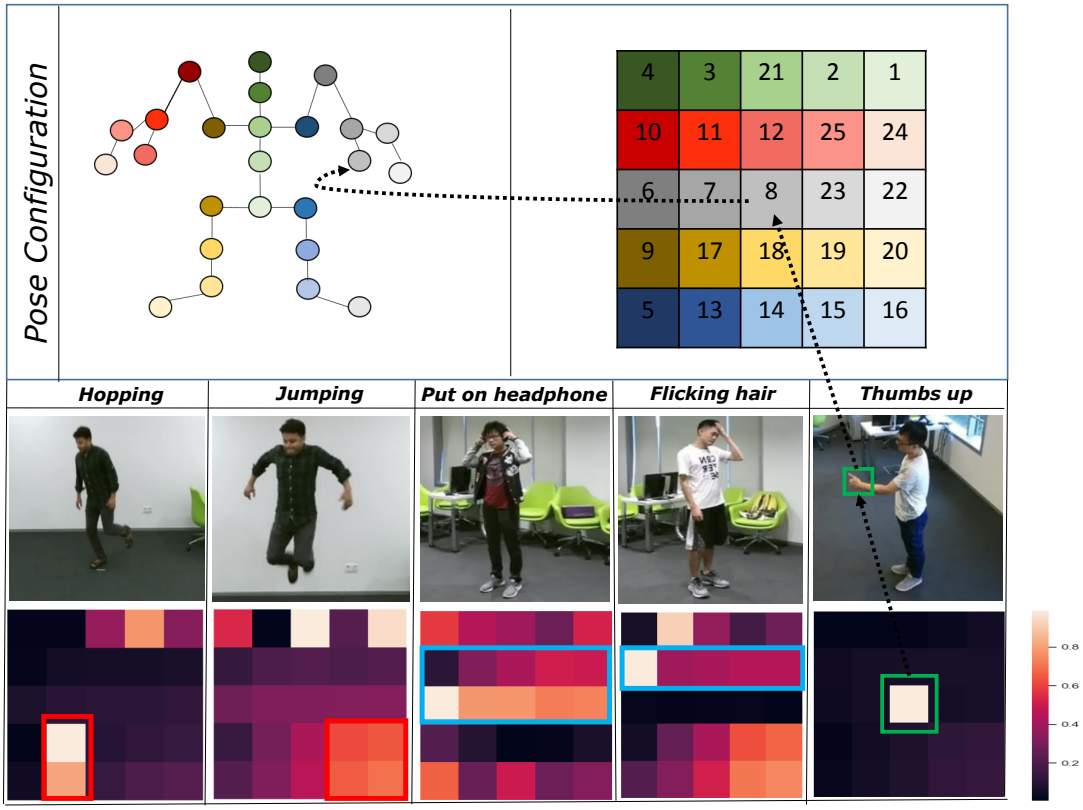


Figure 4.18: The heatmaps of the activations of the 3D joint coordinates (output of GCN) in the attention network of VPN. The area in the colored bounding boxes shows that different joints are activated for similar actions.

coffee, *cutting bread* (+50%), *pouring from kettle* (+42.8%) and *inserting teabag* (+35%) improves VPN with GCN pose backbone compared to its counterpart without embedding. Finally, we also show the impact of using temporal attention which significantly improves the action classification accuracy on all the datasets.

Qualitative Analysis of VPN Fig. 4.18 visualizes the activation of the human joints at the output of pose backbone (with GCNs) in VPN. The figure depicts the activations of the 3D joints. They are presented in a sequence of the human body topological order (follow first row of fig. 4.18) for convenient visualization. VPN is able to disambiguate actions with similar appearance like *hopping* and *jumping* due to high order activation at relevant joints of the human legs. The discriminative leg joints with high activation have been marked with a red bounding box in fig. 4.18 (third row). Similarly, for actions like *put on headphone* with two hands and *flicking hair* with one hand, the blue bounding boxes demonstrate high activation of both the hand joints for the former action as compared to high activation of a single hand joints for the latter. For a very fine-grained action like

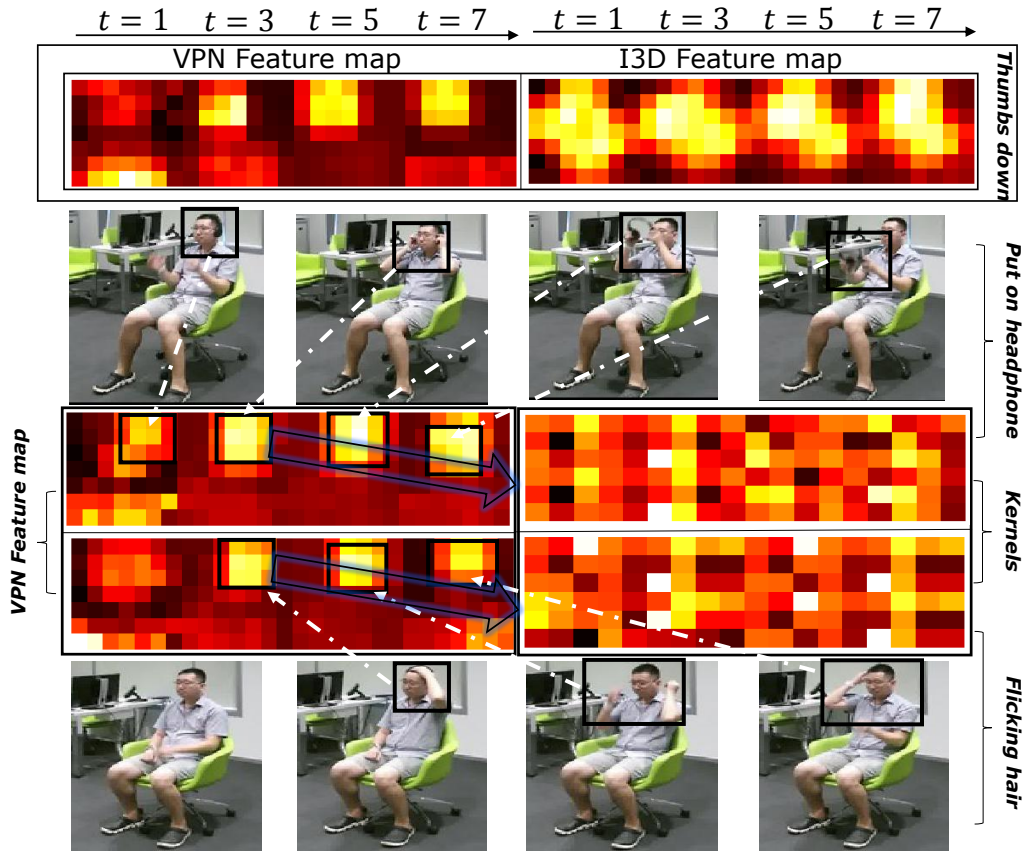


Figure 4.19: Heatmaps of visual feature maps & corresponding activated kernels for different time stamps. These heatmaps show that VPN has better discriminative power than I3D.

thumbs up, the thumb joint is highly activated as compared to the other joints. This shows that the GCN pose backbone in VPN is a crucial ingredient for better action recognition.

In fig. 4.19, we compare the heatmap of the VPN and I3D feature maps for different time stamps. We observe the sharpness in the VPN feature maps compared to that of I3D for *thumbs down* action which is localized over a small space. For similar actions like *put on headphone* and *flicking hair*, along with salience precision of the VPN feature map, the activations of their corresponding receptive fields show the discriminative power of VPN.

Illustration to show the impact of VPN components - In fig. 4.20, we illustrate a set of graphs showing the top-5 improvement of action classification accuracy using different components of VPN compared to I3D baseline. As discussed in the ablation study above, each component in VPN is critical for good performance on ADL recognition.

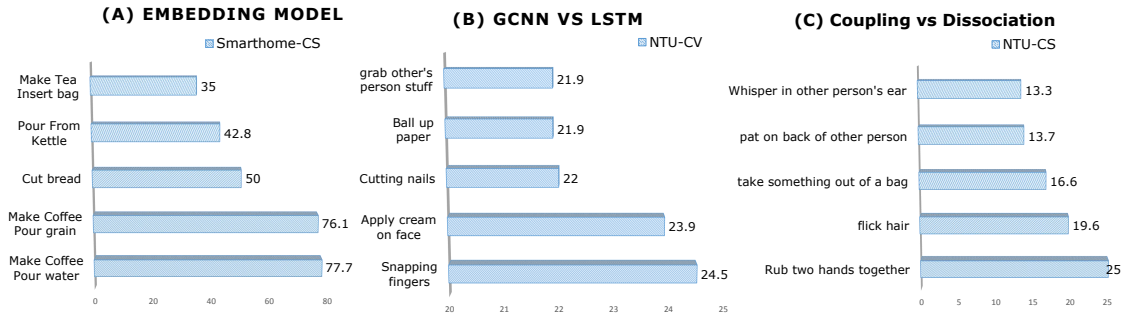


Figure 4.20: Graphs illustrating the superiority of each component of VPN compared to their counterparts (without the respective components). We present the Top-5 per class improvement for (A) VPN with embedding vs without embedding (only Spatial Attention), (B) VPN with GCN vs LSTM Pose Backbone, and (C) attention in VPN with vs without spatio-temporal coupler.

- The spatial embedding provides an accurate alignment of the RGB images and the 3D poses. As a result, the recognition performance of the fine-grained actions improves compared to its counterpart without embedding (see fig. 4.20 (A)).

- The LSTM pose backbone as used in P-I3D and Separable STA firstly fails globally optimize the attention network. Whereas, the GCN pose backbone of the attention network, not only provides a strategy to globally optimize the recognition model but also takes the human joint configuration into account for computing the attention weights. This further boosts the action classification performance (see fig. 4.20 (B)).

- The spatio-temporal coupler of the attention network provides discriminative spatio-temporal attention weights which enables the recognition model to better disambiguate the actions with similar appearance (see fig. 4.20 (C)). For instance, *rub two hands together* is confused with *clapping* and *flicking hair* is confused with *taking on headphones* when classified with baseline I3D. Now, with VPN the mis-classification between these action pairs drop significantly.

Comparison of VPN with baseline I3D - In fig. 4.21, we illustrate the performance of VPN w.r.t. I3D baseline for the dynamicity of an action along the videos. This dynamicity is computed by averaging the Procrustes distance [157] between subsequent 3D poses along the videos. If the average distance is large, it means the poses change a lot in an action. VPN significantly improves for actions with subtle motion like *hush* (+52.7%), *staple book* (+40.7%) and *reading* (+36.2%) which indicates the efficacy of VPN for fine-grained actions. Note that these actions *hush, staple book* and *reading* possessing subtle motion falls in the range between [0, 0.25] of action dynamicity. The degradation of the VPN performance for high action dynamicity is negligible (-0.8%).

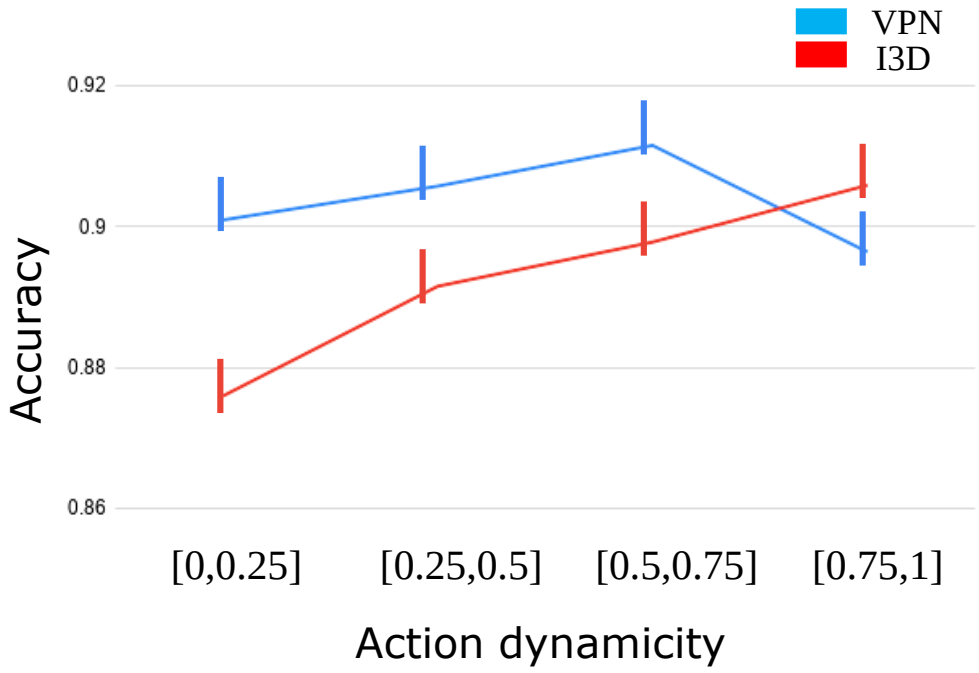


Figure 4.21: We compare our model against baseline I3D across action dynamicity. Our model significantly improves for most actions.

In fig. 4.22, we show the t-SNE plots of the feature spaces produced by I3D and VPN for some selected actions with similar appearance. It clearly shows the discriminative power of VPN for actions with similar appearance which is a frequent challenge in ADL.

We provide some visual results in fig. 4.23 where VPN outperforms I3D baseline. These are the sample examples where VPN is able to distinguish between the action pairs that are often confused with I3D. Hence, the concept of guiding networks utilizing poses improve the representation learned for the visually similar actions. We also provide the confusion matrix for action classification on NTU RGB+D 120 and Toyota Smarthome using VPN. In fig 4.24, we present the confusion matrix of VPN on NTU RGB+D (on right) and a zoom of it around the red bounding box (on left). We also present the corresponding zoom of the confusion matrix of I3D. We are particularly interested in the mis-classifications performed by VPN and thus, we zoom into the region with relatively low classification accuracy. We observe that actions like *staple book* and *taking something out of bag* were confused with *cutting papers* and *put something into a bag* respectively when classified with I3D. However, with VPN these actions with similar motion are now better discriminated, improving their classification accuracy by approximately 42% and 27% respectively.

Similarly, in fig. 4.25, we present the confusion matrix of VPN on Toyota Smarthome dataset. In fig. 4.26, we show the poses for some images belonging to action videos

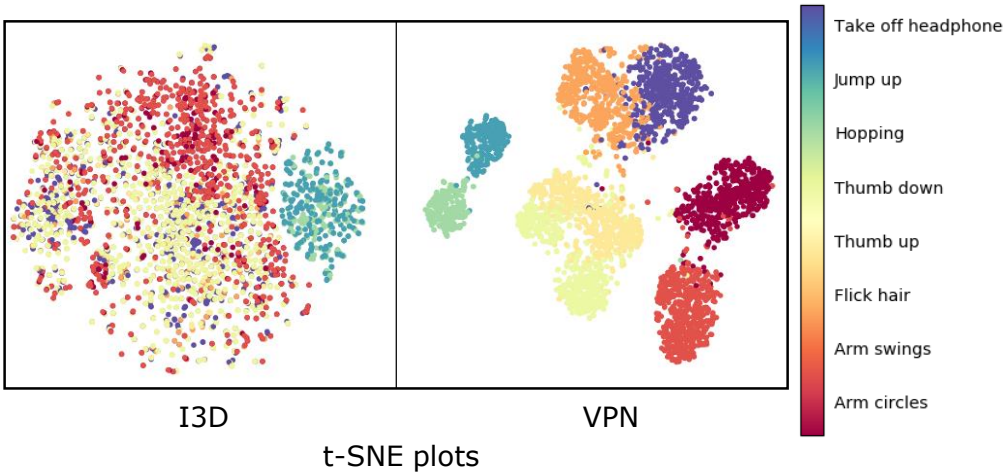


Figure 4.22: t-SNE plots of feature spaces produced by I3D and VPN for similar appearance actions.

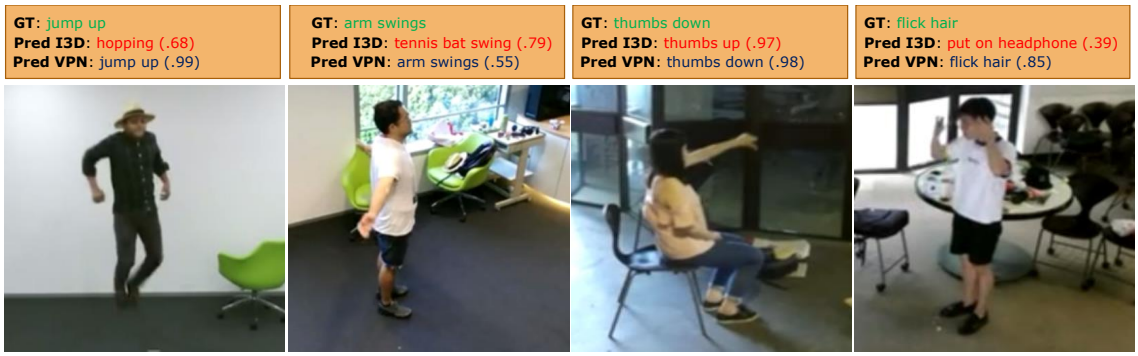


Figure 4.23: Visual results from NTU RGB+D 120 where VPN outperforms I3D.

mis-classified by I3D. Thanks to the high quality 3D poses for these videos, now VPN can correctly classify these actions taking the human topology of the 3D poses into account. However, we notice that actions like *Drink from glass* are not recognized due to extremely low number of training samples. We further notice that actions like *using tablet* are recognized with low accuracy of 13% and largely confused with *using laptop*. However, I3D completely mis-classifies the action *using tablet*. In fig. 4.25, we also show (with red bounding boxes) the set of actions that are still highly mis-classified. What are these actions? These are the fine-grained actions - the same action performed with different objects like *drinking* with a *can, bottle* or *cup*. A possible reason for this is that the 3D ConvNets compromises with the spatial dimension of the input while handling the temporal dimension in the same network.

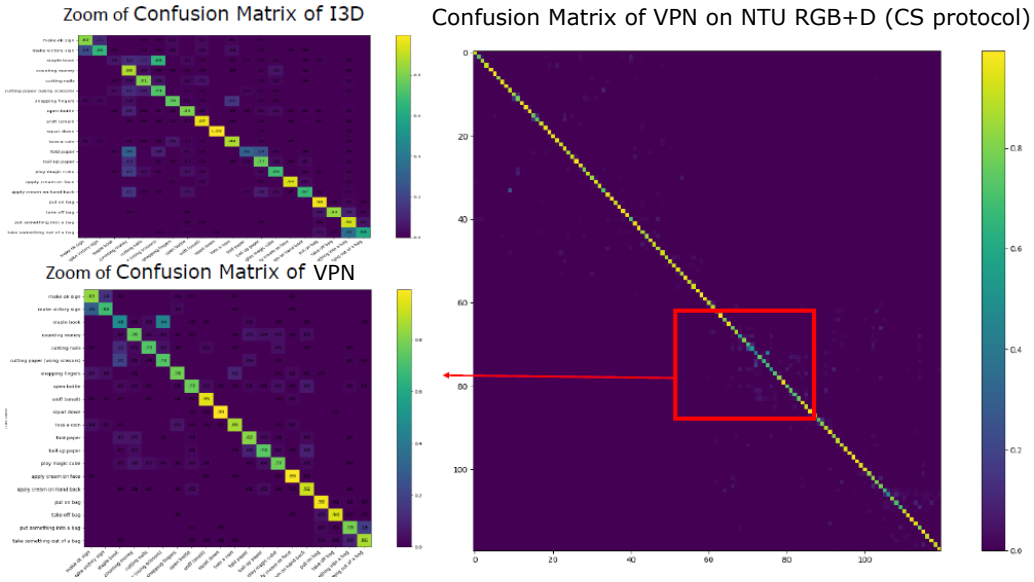


Figure 4.24: Confusion matrix of VPN on NTU RGB+D (CS Protocol) on the right. Zoom of the red bounding box on the left along with the corresponding confusion matrix of I3D. The intend to show this confusion matrix is to state the fact that challenges still remain in datasets with laboratory settings in spite of achieving higher accuracies.

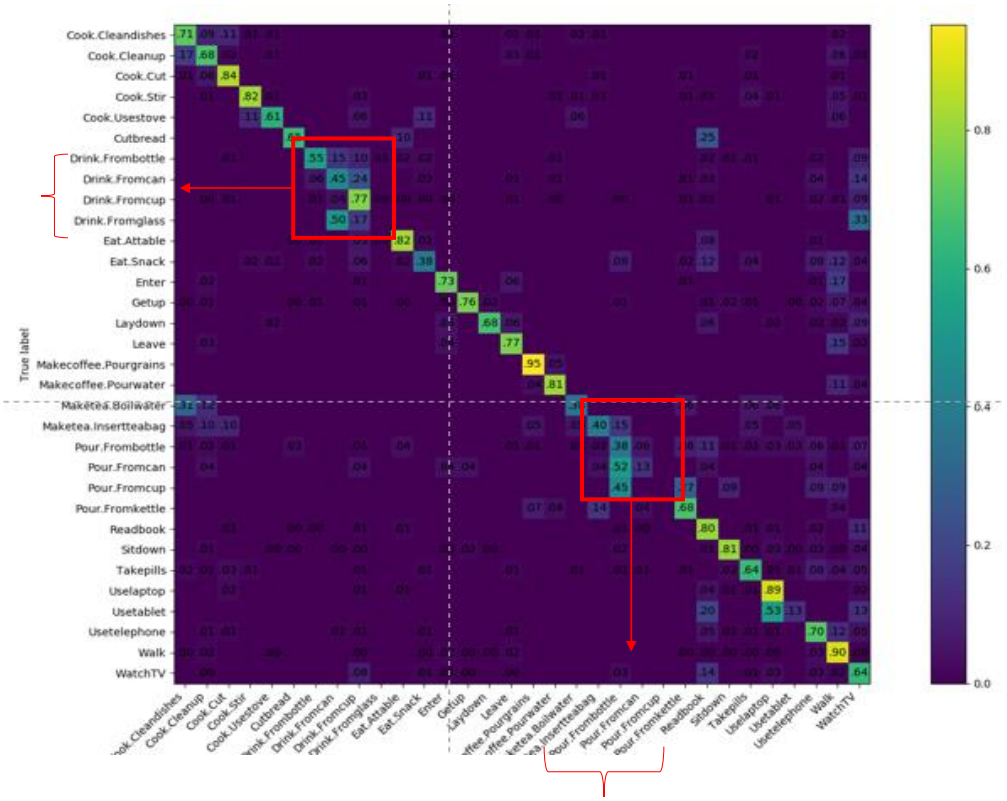


Figure 4.25: Confusion matrix of VPN on Toyota Smarthome (CS protocol). Red bounding boxes in the figure shows the set of fine-grained actions mis-classified among themselves.

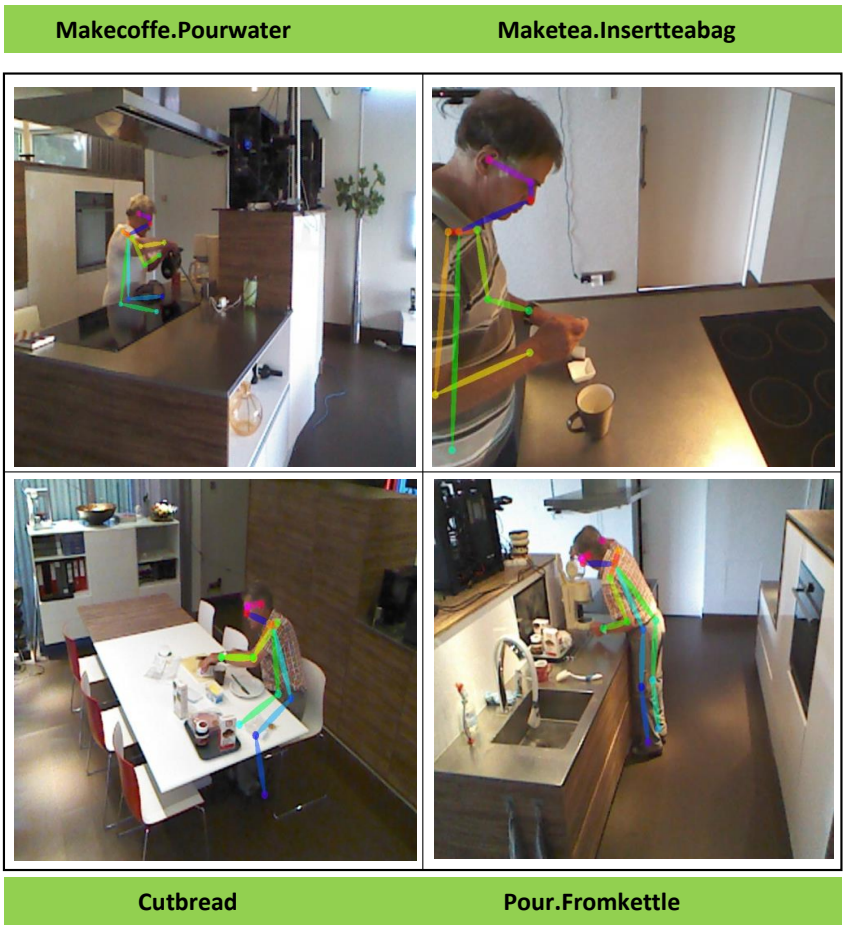


Figure 4.26: Illustration of poses for activities mis-classified with I3D but correctly classified with VPN.

4.6.5 Discussion

In this section, we discuss our assumptions and analysis made for the experiments performed.

Why do we sample different number of 3D poses and RGB frames? We have performed all our experiments with $T \neq t_p$. And still, we utilize these t_p number of poses to compute temporal attention weights for T number of RGB frames. One may argue that why these number of samples for both modalities are not the same? We believe that our temporal attention mechanism takes place with a latent representation of the stack of RGB images. And the attention weights are also computed from a latent representation h^* of the 3D poses. Both these latent representations depict the actions in temporal segments. The difference in number of samples for both the modalities is due to the different networks processing them. Therefore, we strategize to learn temporal attention weights in a latent space where the actions are represented with similar number of temporal segments for both the modalities.

What are the problems that still remain? As shown through the confusion matrix of VPN on Smarthome dataset, the action classification accuracy is still low for certain fine-grained actions. Actions like *drinking* when performed with different object should be able to be distinguished easily just by focusing on the objects. But often the video backbones like I3D or C3D fails to have a clear notion of the objects. We have often observed that our proposed models are prone to mis-classify action where the person is eating snack keeping a book nearby in the scene. The model often classifies it to the action *read book* completely ignoring the posture. We believe that the current state-of-the-art fails to model the associativity of the different objects in the scene with the human performing an action. Inspite of these 3D ConvNets which are pre-trained on ImageNets and then inflated for the task of video classification does not handle the spatial information effectively. It is because of a trade-off between handling space and time. These models incorporating upto 128 frames in a video, compromises with the spatial dimension by restricting the input frame size to 224×224.

What if the 3D poses are noisy? Since our attention mechanisms are pose driven. It depends on the quality of the 3D poses. But in datasets like Smarthome, the 3D poses are not of high quality. The reason is low camera framing and instances of high occlusions. Whatsoever, our VPN tackles such situations of noisy poses through its spatial embedding. But, does it improve the quality of noisy poses? How much noise can be tolerated by these pose driven attention networks to compute meaningful attention weights? These are some interesting analysis that could affect the performance of our proposed frameworks. We plan to do these analysis in the near future.

4.6.6 Runtime

In this section, we provide details regarding run-time for training and testing our proposed frameworks. Below all the statistics have been provided while training and testing our models in 4 GTX 1080 Ti GPUs.

P-I3D - Pre-training the part-wise I3D network on the Smarthome dataset with CS setting takes 12 hours. Pre-training the Stacked pose based LSTM takes 1 hour. Pre-training the RNN Network for developing attention for the human body parts take 15 hours and further fine-tuning takes 4 hours. At test time, a single forward pass of an image frame over the full model takes 547ms on a single GPU.

Separable STA - Training separable STA framework end-to-end takes 5h on Smarthome in CS settings. Pre-training the I3D base network with RGB human crops and stacked LSTM with 3D poses takes 21h and 2h respectively. At test time, a single forward pass for a video takes 338ms on 4 GPUs.

VPN - Training VPN framework end-to-end takes 6h on Smarthome in CS settings. Pre-training the I3D base network with RGB human crops takes 21h. At test time, a single forward pass for a video takes 378ms on 4 GPUs.

Our codes for P-I3D have been open-sourced at https://github.com/srijandas07/P-I3D. The codes for Separable STA have been successfully deployed to Toyota Motors Europe. Details of this work is provided in our project page https://project.inria.fr/toyotasmarthome/. We plan to open-source the codes for VPN in future.

4.7 Conclusion

In this chapter, we present three major contributions for effective and efficient visual representation. We present three pose driven attention mechanisms to improve the functionalities of RGB network like I3D, namely **P-I3D**, **Separable STA** and **VPN**. We enriched our attention network from - (i) spatial attention to dissociated spatio-temporal attention and finally from dissociated to coupled attention mechanism; (ii) poses being processed considering only their temporal evolution to now exploiting their graphical structure in VPN; (iii) simple straightforward exploitation of poses to compute attention weights to take the mis-alignment of 3D poses with the image frames into account.

Our ablation studies on each framework show that the components proposed in our attention networks accomplish their purpose and encourage the classifier to provide discriminative video representation. In this chapter, we address the challenges of similar action discrimination and recognition of fine-grained actions through our attention network. We

also select the state-of-the-art video based CNN to better handle the temporal information in ADL videos. But, these video based CNN like I3D are typically fabricated for short videos. Are we still addressing the complexities involved in handling long composite actions? So, the next step is going towards temporal representation of long and complex actions.

Chapter 5

Temporal Representation for ADL

5.1 Introduction

In the previous chapter, we have seen how attention mechanisms can yield better video representations for recognizing ADL. However, our proposed frameworks are limited to short action videos and we aim at understanding representations for long and complex actions in videos. Therefore, in this chapter, we focus on temporal representation of videos. While handling temporal information in videos, we evolved from using short-term motion using Dense Trajectories [40] to RNNs [27] and finally to 3D ConvNets [71]. But, still these networks are fabricated for modeling temporal information limited to 128 frames. What about videos where action ranges for several minutes? What about actions like variants of *cooking* that are composed of several sub-actions. In fig. 5.1 , we show an example of a person *cooking* which comprises of sub-actions like *cutting, stirring, using stove, stirring* and *using oven*. Recognizing such complex actions require fine-grained understanding of all the event with precision followed by developing a relationship among them.

Figure 5.1: An illustration of a long complex action. A person cooking which comprises sub-actions like *cutting, stirring, using stove, stirring* and *using oven*.

As explained, the 3D ConvNets have been tailored to capture the short-term dynamics

of full 2D+T volume of a video. The use of attention mechanisms on top of these 3D ConvNets somewhat alleviates the challenge of recognizing (i) fine-grained actions and (ii) similar action discrimination. However, these models fail to capture long-range temporal information of actions. Thus, these challenges (i.e., (i) and (ii)) often described as low inter-class variation are aggravated in the presence of long & complex actions. Hence, challenges pertaining to low inter-class variation require temporal attention for discriminating them correctly. Such low inter-class variation is often caused by either similar motion with subtle variation such as *taking out something from pocket/putting something inside pocket*, or complex long-term relationship such as *stirring while cooking/using stove while cooking*.

Also, actions with similar motion tend to have discriminative spatio-temporal features over a small time scale. For instance, wearing and taking off a shoe can be distinguished by taking into account whether or not the shoe is separated from the human body in the first few frames. In order to solve the aforementioned challenge, we need to process the videos at multiple time scales to capture specific subtle motion. Thus, our objective is to capture spatio-temporal relations at multiple time scales and link them over time to disambiguate such temporally complex actions.

As described in chapter 2, there are several studies dealing with the challenge of modeling long temporal relationships in videos. Popular methods like TSN [80, 81] and TRN [79] make use of uniformly sampled image frames from the whole video. These methods show convincing results on internet videos having strong motion varying in long temporal space. Whereas for ADL, subtle motion varies for a very short time-interval which limits these methods on ADL. For ADL, dense temporal information is required to be processed so as to not miss the subtle motion involved in fine-grained actions. This objective inspires us to use the concept of temporal granularity, i.e. temporal windows of different lengths to process the same image frames. Thus, the subtle motion can be represented from the spatio-temporal feature output of 3D ConvNets.
The next challenge remains modeling of long temporal relationship. RNNs have been used in many such problems [158] to link features over time. Attention mechanisms like non-local networks have addressed this underlying challenge in 3D ConvNet (long-temporal relationships), ranges to 128 time steps at max. This non-local network computes the relative distance (using Gaussian embedding) among all its pixels in a spatio-temporal cube. However, this operation computing the affinity between the features does not go beyond the spatio-temporal cube, thus does not account for long-term temporal relations. Thus, these non-local blocks are mostly limited to short internet videos. Recently, Timeception [4] has been developed on top of 3D ConvNets for the purpose of accounting dense temporal information and modeling them for long complex actions. But, this module with dedicated kernels fails to capture the variations of motion involved in ADL and in the wild.

Thus, we propose to invoke a two-level attention mechanism to better provide a temporal representation of a video.

Consequently, in this chapter, we propose a Temporal Model to have a focus of attention on the spatio-temporal features of the relevant time scale. This is effectuated by splitting the videos into uniform temporal segments at different time scale (namely granularity). This is followed by a two-level attention mechanism to manage 1) relative importance of each segment for a given granularity and to manage 2) the various granularities (see fig 5.2).

The Temporal Model which comprises the classification network and the attention network, is trained end-to-end for recognizing actions. We make two hypotheses: the input video clip contains a single class label, and the articulated poses are available as in the previous chapter. Why do we need the poses? Based on our past experiences and concluded statistics, 3D poses with its robust and invariant characterization provide crucial hints for learning spatio-temporal attention weights. Based on previous studies regarding temporal evolution of 3D poses [5, 113], we believe that 3D poses can certainly guide the RGB cue to enhance the temporal representation of a video. This is supported by the effectiveness of 3D poses for the task of detecting actions in an untrimmed videos [159]. Thus, similar to the previous chapter, we utilize the 3D poses as input to our attention network which in turn provides temporal attention weights.

To summarize, in this chapter we present:

- an end-to-end Temporal Model to address the recognition of temporally complex actions. This is done by
  - splitting a video into several temporal segments at different levels of temporal granularity.
  - employing a two-level pose driven attention mechanism. First to manage the relative importance of the temporal segments within a video for a given granularity. Second to manage the relative importance of the various temporal granularities.
- An extensive ablation study to corroborate the effectiveness of our proposed Temporal Model. Besides, we propose a Global Model to have a generic and complete approach for action recognition.

In this chapter, we present a Temporal Model in section 5.2, a Global Model for action recognition in section 5.3 and we present our experimental analysis on four public datasets in section 5.4. Finally, we conclude in section 5.5. The work presented in this chapter has been published as a full conference paper in WACV 2020 [160].

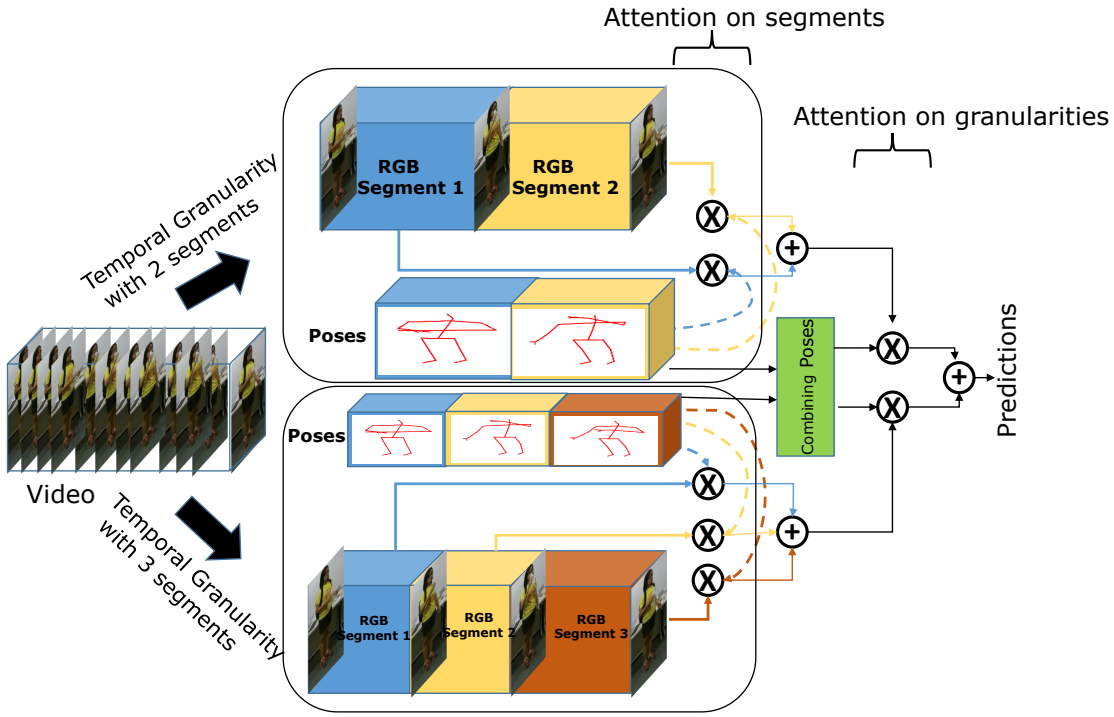


Figure 5.2: Framework of the proposed approach in a nutshell for two temporal granularities. The articulated poses soft-weight the temporal segments and the temporal granularities using a two-level attention mechanism.

5.2 Temporal Model

In this section, we present the Temporal Model for learning and recognizing actions that exhibit complex temporal relationships. This approach involves three stages (see fig. 5.3) to classify the actions. *Stage A* consists in splitting the video into several temporal segments at different levels of temporal granularity (see subsection 5.2.1). *Stage B* classifies the temporal segments of each granularity. It has a Recurrent 3D Convolutional Neural Network ($R-3DCNN$) and an attention mechanism ($TS-att$) so that the different temporal segments are tightly coupled in an optimized manner (see subsection 5.2.2). *Stage C* performs the fusion of the different temporal granularities to classify the action videos (see subsection 5.2.3).

5.2.1 Temporal Segment Representation

In the first stage (*stage A*), our goal is to split the video into several partitions. However, determining the number of such partitions is a difficult task and depends on the content of the action. Thus, for a coarse-to-fine video analysis, a hierarchy of temporal segments

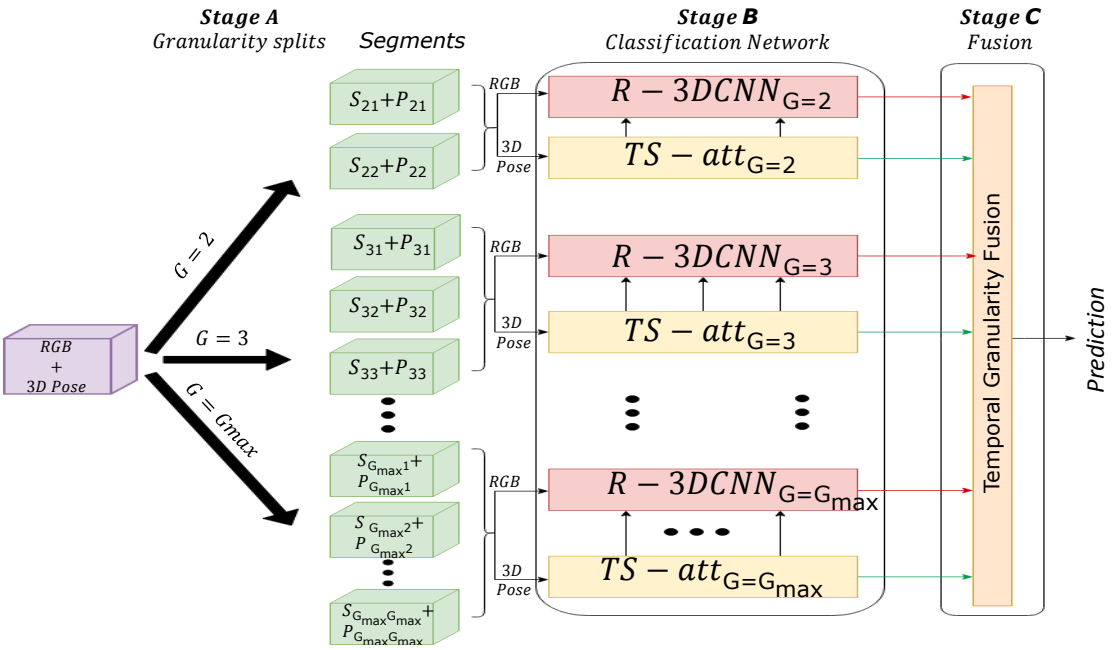


Figure 5.3: Proposed Model with three *stages*. *Stage A* splits the video into different segments at different granularities. *stage B* is the classification network composed of Recurrent 3D CNN ($R-3DCNN$) and an attention mechanism. *Stage C* performs a fusion of the temporal granularities for predicting the action scores.

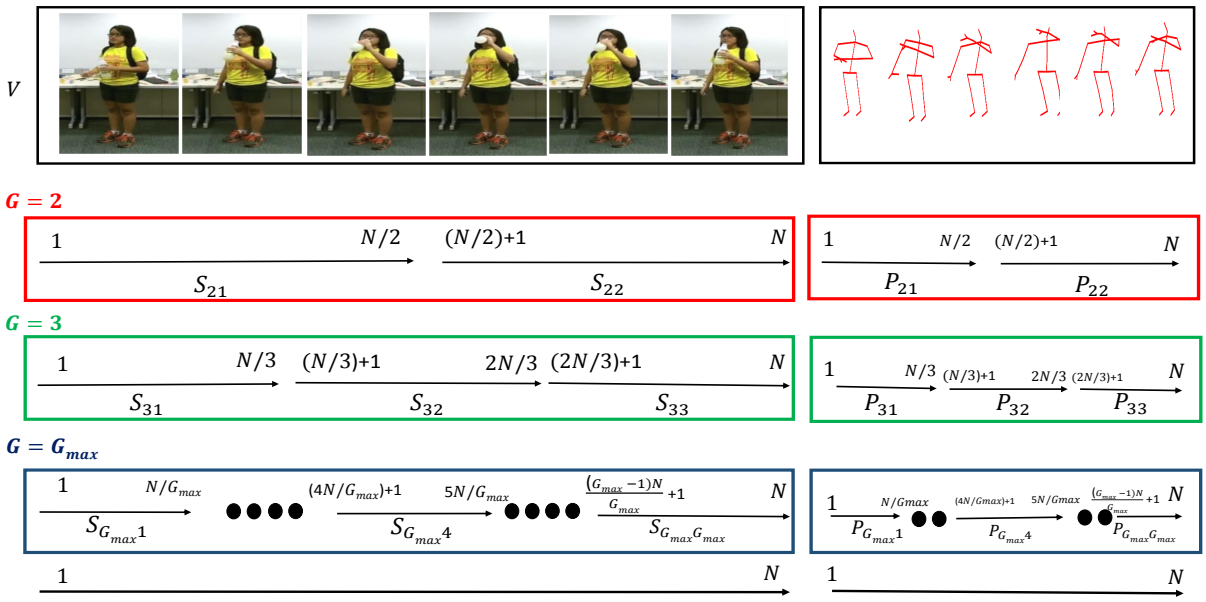


Figure 5.4: A *drinking* video (from NTU-RGB+D [5]) with RGB frames (at left) and 3D poses (at right) is represented with coarse to fine granularities. G representing granularity ranges from 2 to $G_{max}(\leq N)$.

is built. For a given level in the hierarchy (or granularity), the video is divided into non-overlapping segments of equal length.

Formally, given a video V (RGB+Pose) at granularity G, we divide it into G temporal segments. The video with N frames is processed at different levels of granularity $\{2, 3, ..., G_{max} \mid G_{max} \leq N\}$. Thus at granularity G, each temporal segment $S_{Gi} \mid i = \{1, 2, .., G\}$ is a stack of RGB images and $P_{Gi} \mid i = \{1, 2, .., G\}$ is a stack of 3D poses. See an example with a drinking video from NTU-RGB+D [5] in fig. 5.4.

Note that $G = 1$ represents the whole video and is not input to the proposed Temporal Model.

5.2.2 Classification Network

Stage B follows several steps to process the temporal segments for each granularity as described below (see fig. 5.5).

5.2.2.1 Recurrent 3D Convolutional Neural Network

A. Processing the Temporal Segments - The first step (step 1) computes the local features for each temporal segment S_{Gi}. These features are computed by a video backbone which

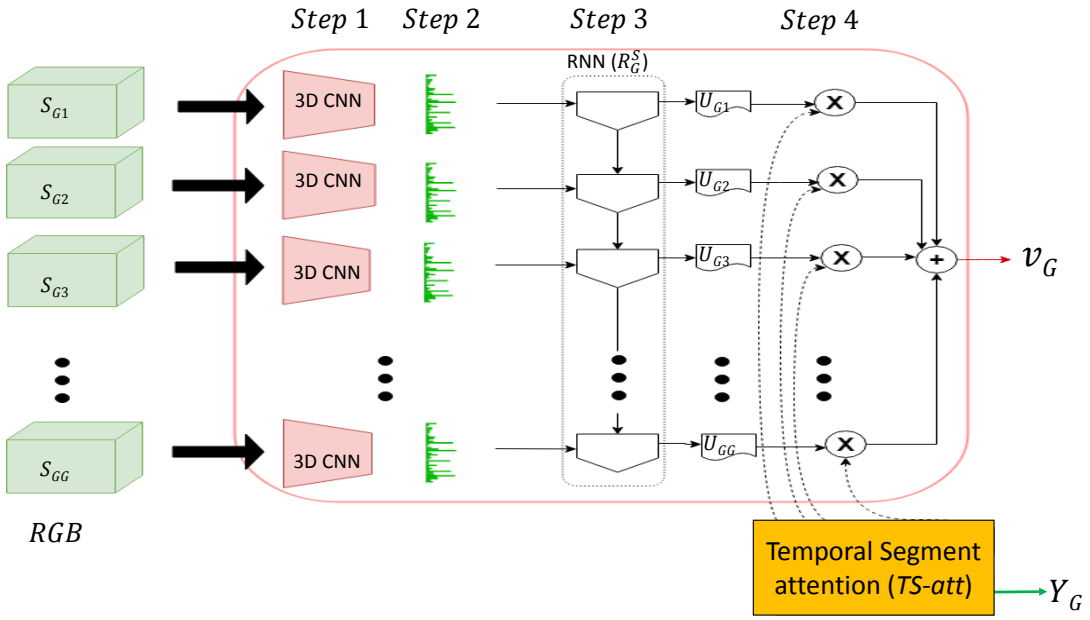


Figure 5.5: A zoom of the **classification network (stage B)** for a given granularity G. The inputs to the RNN R_G^S are the $flattened$ 3D convolutional features of the temporal segments S_{Gi}. Temporal segment attention soft-weighs the temporal segments.

is a 3D CNN, called $f(.)$. The spatio-temporal representation $ST(V,G)$ is given by:

$$\begin{aligned} ST(V,G) &= ST(\{S_{G1}, S_{G2}, ...S_{GG}\}) \\ &= [f(S_{G1};\theta_w), f(S_{G2};\theta_w), .., f(S_{GG};\theta_w)] \end{aligned}$$

The output of the video backbone $f(.)$ with parameters θ_w is a 4-dimensional convolutional feature map. This ST representation is obtained at each level of temporal granularities. In step 2, these convolutional features for each segment S_{Gi} are resized to a single dimensional tensor by a $flatten(.)$ operation.

B. Combining the Temporal Segments - Step 3 is the global sequential processing of the video at a granularity (G) by the combination of all its temporal segments S_{Gi}. For each granularity G, the aforementioned combination is performed by a recurrent network R_G^S which models the long-term dependencies among the dense temporal segments. Thus, $R-3DCNN$ in fig. 5.5 is the recurrent network R_G^S with 3D CNN $f(.)$ as a backbone. The input of R_G^S with parameters θ_G^S, is the succession of flattened feature maps $f(S_{Gi})$. The output U_{Gi} at each time step i of the recurrent network R_G^S is given by:

$$U_{Gi} = R_G^S(flatten(f(S_{Gi})); \theta_G^S) \tag{5.1}$$

Step 4 of the classification network combines the output of step 3 with soft-weights provided by a temporal attention mechanism, which is described below.

5.2.2.2 Attention on Temporal Segments

For a video, some of the segments may contain discriminative information while the others provide contextual information. We argue that poses (3D joint coordinates) are clear indicators to select the prominent sub-sequences in a video as proposed in [113, 114]. This is because of their capability to understand the human body dynamics which is an important aspect in daily living actions.

For a granularity G, the temporal segment attention ($TS-att$) includes two parts (see fig. 5.6). First, the 3D poses of the temporal segments P_{Gi} are processed by an RNN $R^p_{G,i}$ (with parameters $\theta^p_{G,i}$). Then, the output set of the first RNNs are processed by another RNN R^p_G (with parameters θ^p_G) to combine all the temporal segments into G weights corresponding to the importance of the temporal segments. The soft attention $\alpha_{G,j}$ for j^{th} segment of a given granularity G is predicted by learning the mapping:

$$\alpha_{G,j} = \frac{exp(R^p_G(R^p_{G,j}(P_{Gj};\theta^p_{G,j});\theta^p_G))}{\sum_{i=1}^{G} exp(R^p_G(R^p_{G,i}(P_{Gi};\theta^p_{G,i});\theta^p_G))} \tag{5.2}$$

Thus the final output v_G of the classification network is a result of adaptive pooling of U_{Gi}, given by:

$$v_G = \sum_{j=1}^{G} inflate(\alpha_{G,j}) \circ U_{Gj} \tag{5.3}$$

where the $inflate(.)$ operation duplicates the attention weights to match the dimension of U_{Gj} and $\circ$ is the hadamard product. Note that the last time step output feature Y_G of the pose based RNN R^p_G is forwarded to the next step for computing temporal granularity attention.

For each granularity G, $(G+1)$ recurrent networks are required for $G \geq 2$, which may look expensive but at the same time they operate on **lightweight 3D pose information**. Thus, for G_{max} granularities, number of required recurrent networks η are

$$\begin{aligned} \eta &= 3 + 4 + ... + (G_{max} + 1) \\ &= (1 + 2 + 3 + 4 + ... + G_{max}) - 2 \\ &= \frac{G_{max}(G_{max}+1)}{2} - 2 \end{aligned} \tag{5.4}$$

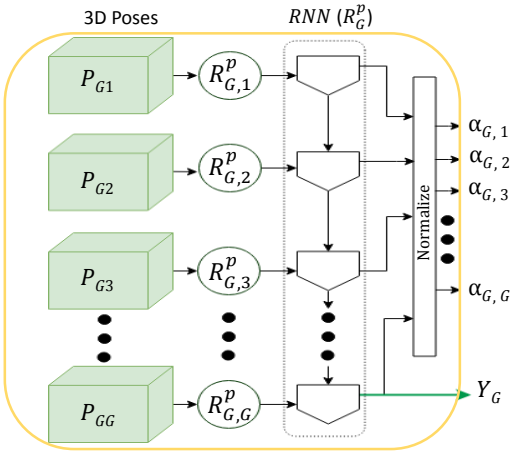


Figure 5.6: Temporal Segment attention ($TS-att$) from 3D poses for a given granularity G. P_{Gi} being input to the RNN $R^p_{G,i}$ followed by their combination using RNN g^p_G to assign soft-weights $\alpha_{G,j}$. Note that the output features of the last time step Y_G is forwarded to the next step for temporal granularity attention.

Again, for each recurrent network with $n-$dimensional input and $m-$dimensional output

$$numberofparameters = 4(nm + n^2 + n) \tag{5.5}$$

This equation independent of the number of time steps of the recurrent network. In our case, n varies from 150 (in case of Kinect sensor) to 39 (in case of 3D poses extracted from RGB). And m depends on the number of hidden neurons in the recurrent network. Thus, with a low dimensional n and m, our pose driven attention network are computationally very efficient. Moreover, the choice of granularities may not be sequential like $G = 2, 3, 4$ but non-sequential like $G = 2, 5, 7$. This depends on an expert's knowledge who decides this hyper-parameter based on the length of the videos in the training distribution.

5.2.3 Fusion of different temporal granularities

Clipping videos into shorter segments may not be an optimal solution for capturing the subtle motion of an action. So, we propose a temporal granularity attention ($TG-att$) to find the extent of fine temporal segments required to recognize an action. In *stage C* of fig. 5.3, the temporal segment attention ($TS-att$) described above is extended to soft-weight the output features of the classification network ($R-3DCNN$) for each granularity (see fig. 5.7). The last time step output features of the pose based RNN R^p_G which are forwarded in the last step, are now concatenated to form a feature vector Y. So, $Y = [Y_2, Y_3, .., Y_{G_{max}}]$ where $Y_G = R^p_G(R^p_{G,j}(P_{Gj}; \theta^p_{G,j}); \theta^p_G)$ for $j \in [1, G_{max}]$ and

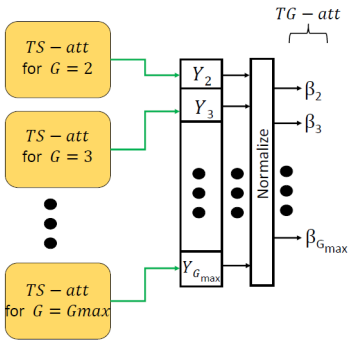


Figure 5.7: Attention for temporal granularity ($TG - att$) to globally focus on the video representation v_G for a given granularity. The model extended from fig 5.6 soft-weights the video representations for G_{max} granularities of the video.

$G \in [2, G_{max}]$. The attention weight β_G for G^{th} granularity is computed by

$$\beta_G = \frac{exp(Y_G)}{\sum_{i=2}^{G_{max}} exp(Y_i)} \tag{5.6}$$

This attention weight is used for focusing on the pertinent temporal granularities. Finally, the prediction for C classes is the weighted summation of the scores at all the granularities followed by a softmax operation:

$$prediction = softmax(\sum_{G=2}^{G_{max}} inflate(\beta_G) \circ v_G) \tag{5.7}$$

where the $inflate(.)$ operation duplicates the attention weights β_G to match the dimension of v_G and $\circ$ is the hadamard product.

We use categorical cross-entropy loss (L_c) to train the whole network. Note that we do not use any attention regularizer in this framework as it does not improve the performance of our Model. The reason is because the Temporal Model is not trained jointly unlike the previous frameworks with the video backbone which allows the optimization to take place without any constraints.

5.3 Global Model for Action Recognition

Above, we have described our **Temporal Model** where temporal segments with different granularities ($G_{max} \geq 2$) are dynamically fused to classify actions. We call a video backbone used for the whole video sequence without any temporal decomposition (i.e. $G = 1$), as the **Basic Model**. The Basic Model is simply the video backbone of the Temporal Model

Algorithm 4 Training of Temporal Model

Input: RGB video, 3D joint coordinates, model training parameters.

1: Initialize I3D network with model weights trained on IMAGENET and Kinetics.
//**Pre-train I3D network.**
2: Fine-tune I3D network with RGB data from full body crops.
3: Extract the I3D features $ST(V, G)$ at each granularity G for all the temporal segments.
//**Initialize the Temporal Model - classification network and the attention network parameters.**
4: Classification network R_S^G is a one layer GRU at granularity G. Attention network constitutes $R_{G,j}^p$ and R_G^p at a granularity G which are also single layer GRUs.
5: Output of R_G^P at j^{th} time step is normalized with softmax to compute attention weight $\alpha_{G,j}$ for j^{th} temporal segment and granularity G.
6: Output of R_G^P for all $G = 1 : G_{max}$ are combined and add a FC layer. The output of this FC layer is normalized with a softmax activation to compute attention weights β_G for granularity G. Initialize the attention scores with equal values and the remaining network parameters using Gaussian.
//**Train the Whole Network**
7: Jointly train classification network with the attention network with cross-entropy loss L_c to classify the actions.

Output: the learned network.

to classify the actions. Note that the Basic Model though termed Basic can also be a complex architecture, for instance when the video backbone is any of our previous attention framework. With a slight abuse of term, the model is termed Basic to denote the non temporal decomposition of the videos while processing them, unlike the Temporal Model.

As stated in [161], temporally segmenting videos can destroy the local structure of some short actions. So, we define a **Global Model** for action classification by performing a late fusion of the proposed Temporal Model and the Basic Model. This is done by performing dot product operation of the model scores at logit level. We do not perform soft weighting of the temporal segment with $G = 1$ (i.e., the Basic Model) to classify the actions. The reason is the presence of asymmetric operations in the sub-networks (RNN and 3D CNN) with $G = 1$ and with $G > 1$ which makes the proposed attention model difficult to train. The training algorithm of the Temporal Model is presented in algorithm 4. The algorithm involves first pre-training the video backbone for the task of action classification. Then, the Temporal Model is trained along with the 3D poses as input to the attention network. Upon convergence, the attention network learns to provide relative importance to the temporal granularities and temporal segments relevant to an action.

5.4 Experiments

In this chapter, we perform all our ablation studies on NTU-60 and NUCLA datasets. However, we also action classification results for NTU-60, NTU-120, Smarthome and NUCLA datasets when the Temporal Model uses our proposed attention based video backbones (P-I3D, Separable STA and VPN).

5.4.1 Implementation details

Network architectures - For the 3D CNN $f(.)$, we use I3D [3] pre-trained on ImageNet [162] and kinetics [3]. Shareable parameters are used to extract spatio-temporal representations of each temporal segment. Spatio-temporal features are extracted from the *Global Average Pooling layer* of I3D. Recurrent networks R^S_G modeling global temporal structure are Gated Recurrent Networks (GRUs) with single hidden layer of size 512. All the recurrent networks for $R^p_{G,j}$ and R^p_G are also GRUs with a hidden state of size 150. We use 3D pose information from depth based middle-ware [55] or obtain the 3D poses from RGB using LCRNet [134].

Training - First, we initialize the **video backbone** - I3D from the Kinetics-400 [24] + ImageNet [20] classification models. Data augmentation and training procedure for training the I3D on tracks of human body follow [3]. For fine-tuning the video backbone independently, we use SGD optimizer with an initial learning rate of 0.01. We use a regularization with a weighting factor of 0.001 between the penultimate layer and the classification layer in I3D. The feature vector $ST(V, G)$ at granularity G is obtained from the output of `GAP` (Global Average Pooling) layer.

Then the classification network is trained using the Adam Optimizer [152] with an initial learning rate of 0.0005. We use mini-batches of size 32 on 4 GPUs. Straightforward categorical cross-entropy with no regularization constraints on the attention weights has been used to train the network end-to-end. For training the pose driven attention network, similar to [5], we uniformly sample the pose segments into sub-sequences of respectively 5 and 4 frames for NTU and N-UCLA. We use the 3D CNN (I3D) trained on NTU as a pre-trained model and fine-tuned it on N-UCLA.

Hyper-parameter settings - The hyper-parameter G_{max} is the most sensitive choice in our Temporal Model. We have tested different values of G_{max}: 2,3, and 4. In the ablation study, we show that the choice of taking up to 4 granularities is meaningful for the short actions described above.

5.4.2 Ablation study for Temporal Attention

In this section, we show the effectiveness of our proposed two-level attention mechanism on NTU (CS and CV) and NUCLA datasets. We provide two ablation studies to evaluate the benefit of the **(A) temporal segment attention (**$TS-att$**), (B) temporal granularity attention (**$TG-att$**)** compared to baseline I3D.

(A) Fig. 5.8 is a plot of action classification accuracy w.r.t. the number of granularities. The dotted and solid lines represent the classification results without/ with ($TS-att$) respectively. The relatively higher accuracy scores of the solid line for $G>1$ as compared to the dashed line indicates the effectiveness of the proposed first level $TS-att$ attention. Fig. 5.8 also shows that, as we introduce the Temporal Model for $G>1$, the classification performance improves as compared to the performance of baseline I3D network ($G=1$) for NTU-60. This implies that the temporal decomposition in the Temporal Model improves the classification of temporally complex action videos (examples provided at the end of this section). As we go for finer granularities from $G=2$ to 4, the action classification accuracy goes down, say from 89.7% to 87.4% for NTU-CS with $TS-att$. This is due to the short duration of actions present in the database mentioned above such as *clapping* (-7.2%), *taking out something from pocket* (-6.4%) which lack temporal structure. It is interesting to note that the classification performance degrades for N-UCLA, when processed in segments. This is due to the small duration of the actions whose temporal structures are hampered while decomposing them into segments. However, we observe that actions like *pick up something with one hand or two hands* are now classified correctly when processed with Temporal Model rather than the Basic Model. Thus, the visual features learned in the Temporal Model are complementary to that of the Basic model.

(B) Table 5.1 shows the improvement of the classification score with the combination of granularities ($G=2,3,4$). In table 5.1, our Temporal Model without temporal granularity attention indicates simple summation of the features from each granularity. Whereas our Temporal Model with temporal granularity attention performs weighted summation to combine the granularity based features. We observe that the accuracy of the Temporal Model from the Basic model improves by 5.2% on NTU, even without employing temporal granularity attention ($TG-att$). $TG-att$ attention further improves the action classification score by 0.85% on NTU dataset. Table 5.1 also shows the importance of fusing together the Basic and Temporal Model into a Global Model. There are some actions which are correctly recognized by the Basic Model but mis-classified by the Temporal Model such as *punching* (-13.4%) and *throwing* (-8.4%). Temporal decomposition of these actions with very few key frames, does not improve their recognition. So, thanks to the

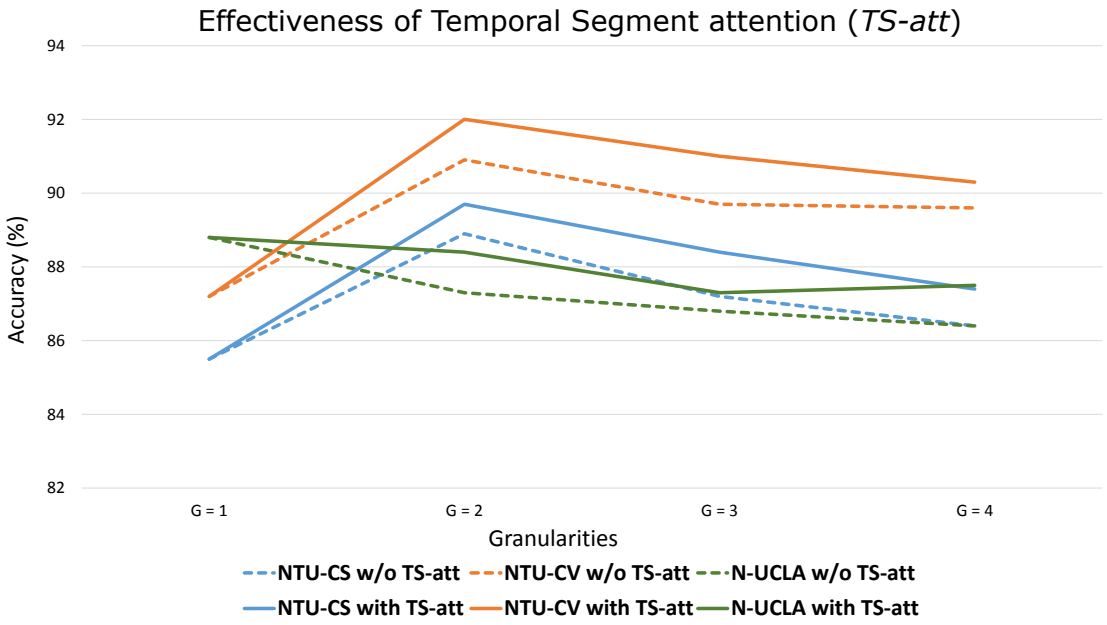


Figure 5.8: A plot of Accuracy in % (vertical axis) vs number of granularities G (horizontal axis) to show the effectiveness of the temporal segment attention ($TS-att$) on NTU-RGB (CS & CV) and N-UCLA($V^3_{1,2}$). Note that the accuracy for $G = 1$ is on the I3D base network.

late fusion of the aforementioned Models, we manage to recover the correct recognition of these actions. Thus, the Global Model improves the action classification performance by approximately 2% as compared to the Temporal Model over all the datasets.

Comparison of Global Model with Basic Model - To analyze the gain obtained by the Temporal Model, we study the difference in classification accuracy between the Basic Model and the Global Model for the 20 best classes in fig. 5.9. Our Global Model improves **53 out of 60** action classes. The most significant improvements concern actions with repetitive cycles like *brushing teeth* (+17.1%), *handwaving* (+16.9%), and *use a fan*

Table 5.1: Ablation study to show the effectiveness of the temporal granularity attention ($TG-att$) and the Global Model compared to the Basic and Temporal Models on NTU-RGB (CS & CV) and N-UCLA($V^3_{1,2}$). Acc. denotes action classification accuracy.

Model	G	$TG-att$ Acc. (%)	NTU-CS Acc. (%)	NTU-CV Acc. (%)	N-UCLA
Basic	1	×	85.5	87.2	88.8
Temporal	2,3,4	×	89.9	91.9	88.2
Temporal	2,3,4	✓	90.6	92.8	89.5
Global	1,2,3,4	✓	**92.5**	**94.0**	**91.0**

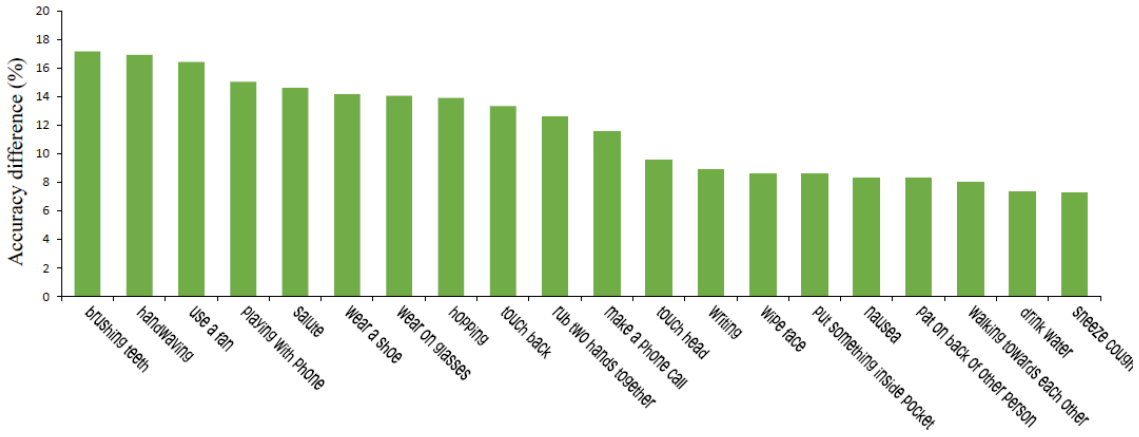


Figure 5.9: Accuracy difference per action label of the 20 best classes for NTU dataset between the Temporal Model and the Global Model. The base network is I3D and the results are averaged over the CS and CV protocols.

(+16.4%). These actions have long-term temporal structure (the repetition of actions) which our proposed Temporal Model successfully deciphers. The Basic Model fails when it has to distinguish between action pairs with similar poses and subtle motion, such as *wearing and taking off a shoe* and *wearing and taking off glasses*. On the contrary, the temporal decomposition of these actions into segments enables the classifier to discriminate between similar pairs, and thus improves the recognition of *wearing a shoe* (+14.1%) and *wearing glasses* (+14.0%). For these actions, the temporal segments contain very specific and discriminative parts which enables the classifier to discriminate the similar ones. See fig. 5.10 in which our proposed Global Model outperforms the Basic Model (I3D). For action *taking on a shoe*, the first temporal segment S_{21} for granularity $G = 2$ discriminates it from *taking off a shoe*. Similarly, for action *put something inside pocket*, the second temporal segment S_{32} for temporal granularity $G = 3$ enables the classifier to recognize the action correctly. Actions like *cross hands in front* (-4.0%) and *punching* (-3.1%) are the two major worst classes. The Global Model has difficulties recovering these actions because the Temporal Model may add noise to the recognition score acquired by the Basic Model during their fusion. However, these drops in performance are not as significant as the improvements.

In table 5.2, we show the action classification accuracy of Temporal Model on four public datasets with video backbones proposed in the last chapter. As discussed how our proposed spatio-temporal attention mechanisms improve the action representation for short-term videos in the last chapter. So, we incorporate those as video backbones or Basic Model in this proposed framework. Consequently, we improve the action classification performance significantly on all the datasets compared to I3D as baseline video backbone.

Table 5.2: Action classification accuracy (in %) on 4 public datasets using different video backbone in Temporal Model. We also provide the corresponding accuracy on the Global Models. We denote Smarthome dataset by SH.

Methods	NTU-60 CS	NTU-60 CV	NTU-120 CS_1	NTU-120 CS_2	SH CS	SH CV	NUCLA $V^3_{1,2}$
Temporal Model (I3D base)	90.6	92.8	85.2	86.5	55.4	52.8	89.5
Temporal Model (P-I3D base)	92.4	95.2	-	-	-	-	88.6
Temporal Model (STA base)	91.1	93.0	85.7	87.1	56.6	49.5	90.8
Temporal Model (VPN base)	92.5	96.2	85.5	87.1	60.4	49.9	91.4
Global Model (I3D base)	92.5	94.0	85.9	87.3	58.5	54.3	91.0
Global Model (P-I3D base)	93.9	96.1	-	-	-	-	93.5
Global Model (STA base)	93.6	95.4	87.0	88.1	60.0	54.6	92.8
Global Model (VPN base)	**94.1**	**96.5**	**87.5**	**89.0**	**62.6**	**55.5**	**93.7**

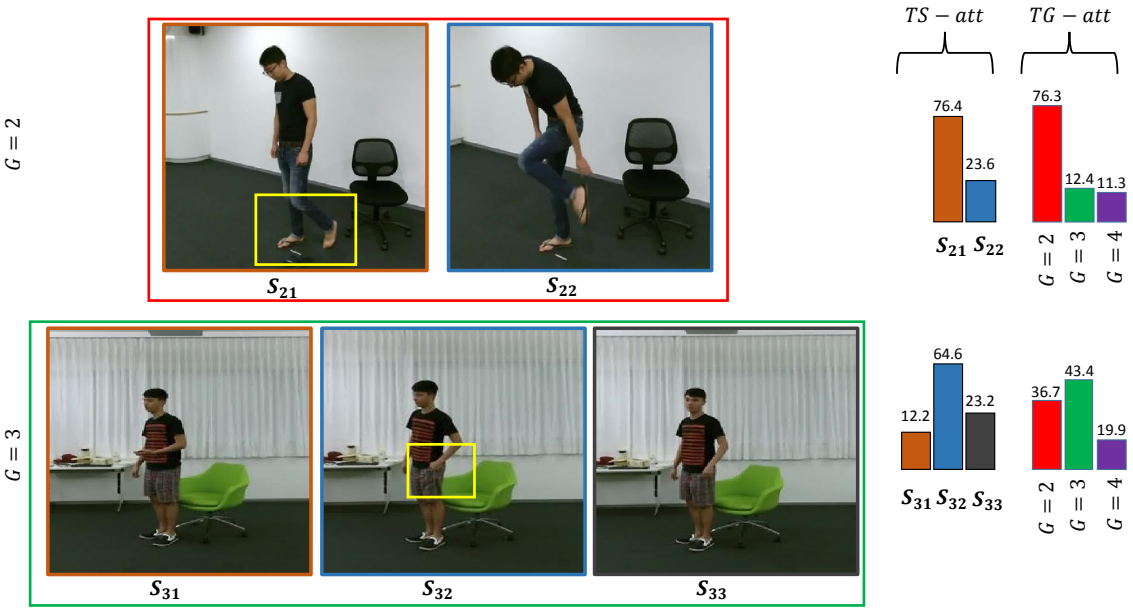


Figure 5.10: Examples of videos at left (*taking on a shoe* and *put something inside pocket*) and attention weights of temporal segments and granularities at right. Our proposed Global Model classifies these action videos correctly but Basic Model (I3D) does not. The distinctive context or gesture in the pertinent temporal segment is highlighted with yellow box.

5.4.3 Discussion

In this subsection, we discuss the limitations and possible future research directions for temporal representation of videos. As we have presented that our Temporal Model is sensitive to the choice of the hyper-parameter G_{max}, which is the number of granularities. The choice of this hyper-parameter is based on expert's knowledge regarding the distribution of the training data along temporal domain. Thus, an arbitrary choice of this hyper-parameter G_{max} can affect the performance of the whole model. We also believe that temporal representation of videos could be improved by processing the temporal segments within a granularity through Temporal Convolutional Networks (TCNs). Duration of actions may vary in time depending on several factors. These TCNs with dilation are capable of handling temporal variance within an action. Finally, utilizing our proposed Temporal Model for the task of action detection could be a possible future extension of this work.

5.4.4 Runtime

Training the Temporal Model end-to-end takes 3 hours with a single job spread over 4 GTX 1080 Ti GPUs. Pre-training the Basic Model on the NTU dataset takes 15 hours. 3D CNN (I3D) features are extracted in parallel over 16 GPUs for 4 granularities and thus varying the granularity does not affect the run time of the model. At test time, RGB preprocessing takes one second (loading Full-HD video and extracting 3D CNN features). The Temporal Model with granularity G_{max} = 4, takes 1.1 ms including the prediction from the Basic Model on a single GPU. The temporal attention module is very efficient because it works only on the 3D pose joints. Classification can thus be done close to real-time.
Our codes for Temporal Model have been open-sourced at https://github.com/srijandas07/temporal_model.

5.5 Conclusion

In this chapter, we have presented a new temporal representation of videos into temporal granularities. Each temporal granularity is again represented by temporal segments. A video representation constitutes a weighted combination of the granularities and the segments. The weighted combination is achieved by a pose driven attention mechanism. A two-level attention mechanism to soft-weight the relevant temporal segments within a granularity and to weight the relevant temporal granularities, yields a discriminative video representation. This temporal representation of videos addresses the underlying challenges of ADL which are long and often complex.

The future work with the Temporal Model includes adapting the Model for the task of action detection. We believe that these Models can be exploited for fast and efficient computation of temporal proposals of actions within a video.

Chapter 6

State-of-the-art comparison

6.1 Introduction

In this chapter, we provide the comparison of the proposed methods with the state-of-the-art. As discussed in chapter 2, we perform our experiments on both small-scale and large-scale public datasets. For our first contribution (detailed in chapter 3), we present the state-of-the-art results on three small-scale datasets CAD-60, CAD-120 and MSRDailyActivity3D. As this contribution, namely multi-modal video representation is difficult to implement on large datasets due to exhaustive resource requirement, we perform minor changes in the optimization strategy to adapt it for large scale datasets like NTU-60. We discuss about these changes in the optimization while presenting the state-of-the-art results on NTU-60.

We present this chapter in two folds, firstly presenting the state-of-the-art comparison for the three small scale datasets for multi-modal fusion strategy. Then, we present the state-of-the-art results on the relatively large-sale datasets, namely NTU-60, NTU-120, Smarthome and NUCLA for all our proposed attention mechanisms.

6.2 Comparison of Multi-modal Method with the state-of-the-art

In this section, we compare action classification accuracy of our proposed Multi-modal Method with the state-of-the-art on three public datasets. We present these comparisons, datasets-wise in Table 6.1, 6.2 and 6.3.

In chapter 3, we have shown how our proposed fusion strategy and similar action discrimination module are effective on the small-scale ADL datasets. Now, we show that our proposed Multi-modal method is superior than other state-of-the-art methods. In

Method	Year	Data	Accuracy [%]
STIP [163]	2014	RGB + Depth	62.50
Object Affordance [164]	2013	RGB + Poses	71.40
HON4D [165]	2013	RGB + Poses	72.70
Actionlet Ensemble [46]	2012	RGB + Poses	74.70
Dynamic Skeletons [166]	2015	RGB + Poses	84.10
MSLF [137]	2016	RGB + OF	80.36
P-CNN Fusion [58]	2017	RGB + OF	95.60
Multi-modal Method	2019	RGB + OF + Pose	**98.52**

Table 6.1: Comparison of our Multi-modal video representation for different modalities with the state-of-the-art methods on CAD-60. The state-of-the-art methods are indicated by their year of publication. The different modalities include RGB, Optical Flow (OF), 3D Poses, and Depth.

our state-of-the-art comparison, we indicate the data modalities used by different previous methods. For CAD-60, we outperform the state-of-the-art by 2.9% approximately as shown in table 6.1. The previous state-of-the-art method [58] though very close to our classification accuracy does not make use of the 3D poses. However, we observe that this method which does not take the temporal evolution of 3D poses into account for classifying actions is limited to learn discriminative representations for human-object interactions. We see later how this method [58] fails to obtain high accuracy on other datasets like MSRDailyActivity3D (as shown in Table 6.3).

We also observe in our experimental analysis on CAD-60 that the use of optical flow through dense trajectories improves the classification of actions with low amount of motion like *relaxing on couch*, *working on computer* and *staying still*. Similarly, actions like *brushing teeth* and *drinking water* are correctly classified with pose information due to these action having discriminative motion ranging over time. Finally, most of these actions with object interaction are classified with high accuracy only with the use of RGB modality. As a consequence, the fusion of all these modalities results in a high classification accuracy on CAD-60 dataset.

CAD-120 is a challenging dataset with the presence of temporally opposite actions like *stacking* and *unstacking objects*. Thus, the inter-class variation in this dataset is very low which makes this dataset highly challenging in addition to low number of training samples. However, our proposed Multi-modal method outperforms all the state-of-the-art methods by a margin of 4.7% as shown in Table 6.2. Lin et al. [167] in RSVM + LCNN models the spatio-temporal layout of CNN features, in addition it arbitrarily divides each action into a fixed number of segments (which is 4). We argue that such a temporal decomposition of videos may not be optimal for all the actions. We also show this phe-

Method	Year	Data	Accuracy [%]
P-CNN Fusion [58]	2017	RGB + OF	71.0
Salient Proto-Objects [168]	2014	RGB	78.20
SVM + CNN [167]	2016	RGB	78.30
TDD [169]	2015	RGB + OF	80.38
STS [170]	2013	RGB + Poses	84.20
Object Affordance [164]	2013	RGB + Poses	84.70
MSLF [137]	2016	RGB + OF	85.48
R-HCRF [171]	2016	RGB + Poses	89.80
RSVM + LCNN [167]	2016	RGB	90.10
Multi-modal Method	2019	RGB + OF + Poses	**94.40**

Table 6.2: Comparison of Multi-modal video representation for different modalities with the state-of-the-art methods on CAD-120 dataset. The state-of-the-art methods are indicated by their year of publication. The different modalities include RGB, Optical Flow (OF), 3D Poses, and Depth.

nomenon in Chapter 5 with our Temporal Model which inspired us to propose a Global Model ensuring completeness of the framework. Also, we observe that the significant improvement in our proposed Multi-modal method is due to the similar action discrimination module which successfully disambiguate similar actions like *stacking* and *unstacking objects* or *cleaning object* and *taking food*.

We present the state-of-the-art comparison on MSRDailyAcitivity3D in Table 6.3. This dataset consists of challenging scenarios with the same action performed in two ways like in sitting and standing position. This results in high intra-class variation in this dataset. We observed in chapter 3 that the 3D poses with considerably long duration of videos prove to be useful for discriminating actions correctly. The 3D poses when fed to simple LSTM accounts for 91.6% of action classification accuracy alone compared to 90% accuracy with combined RGB and optical flow as shown in 3.2. Thus, we outperform the state-of-the-art results on MSRDailyActivity3D by 0.3% only. The previous state-of-the-art method [172] although with a performance close to our method without the use of optical flow, fails on large datasets like NTU-60 (see Table 6.4). Also, note the lower classification accuracy of P-CNN Fusion [58] on this dataset compared to our Multi-modal video method. As discussed earlier, although this method attains high classification accuracy on dataset like CAD-60, is not consistent in terms of performance. This is due to its lack of modeling pose based temporal relationships.

For a better comparison, we also present the results of our Multi-modal method on NTU-60 dataset in the next section. In addition, next we present the state-of-the-art results of our proposed attention mechanism based frameworks on four public datasets (to be discussed one-by-one).

Method	Year	Data	Accuracy [%]
NBNN [173]	2013	RGB + Poses	70.00
HON4D [165]	2013	RGB + Poses	80.00
STIP + skeleton [163]	2014	RGB + Poses	80.00
SSFF [174]	2014	RGB + Poses	81.90
DSCF [175]	2013	RGB + Poses	83.60
P-CNN Fusion [58]	2017	RGB + OF	84.40
Actionlet Ensemble [46]	2012	RGB + Poses	85.80
RGGP + fusion [176]	2013	RGB + Poses	85.60
MSLF [137]	2016	RGB + OF	85.95
Super Normal [177]	2014	RGB + Poses	86.26
BHIM [178]	2015	RGB + Depth	86.88
DCSF + joint [175]	2013	RGB + Poses	88.20
Dynamic Skeletons [166]	2015	RGB + Poses	95.0
Range Sample [179]	2015	RGB + Poses	95.6
DSSCA-SSLM [172]	2018	RGB + Poses	97.50
Multi-modal Method	2019	RGB + OF + Pose	**97.81**

Table 6.3: Comparison of Multi-modal video representation for different modalities with the state-of-the-art methods on MSRDailyActivity3D dataset. The state-of-the-art methods are indicated by their year of publication. The different modalities include RGB, Optical Flow (OF), 3D Poses, and Depth.

6.3 NTU-RGB+D-60

NTU-RGB+D is a large dataset and is suitable for using deeper models. In this section, we present the state-of-the-art results on NTU-60 dataset in Table 6.4. We present the results in sections, categorized based on the input data modalities of the methods (Pose, RGB, Pose + RGB). We also indicate the year of publication of the previous methods or the chapter that presents the proposed method in this thesis. We also indicate whether the methods use any kind of attention mechanism.

Firstly, we present the action classification accuracy of our Multi-modal method on Cross-Subject protocol. This method achieves state-of-the-art performance even with using a 2D CNN based appearance model (ResNet-152) compared to 3D inflated ResNet-50 in case of Glimpse clouds [106]. In order to scale the Multi-modal method for large-scale dataset like NTU-60, we used SGD optimizer with hinge loss instead of using linear classifiers like SVM. This enables our optimization to occur in batch-wise iteration instead of taking all the training samples together in a single iteration as in case of SVM. With the aforementioned change, we could train a fusion based model on NTU-60, however with the lack of global optimization strategy, it is evidently not the optimal solution. We also use I3D [3] to model the appearance instead of using 2D CNN (2D ResNet-152) and

Table 6.4: Comparison between the proposed methods and the state-of-the-art methods using different modalities (3D poses and RGB) on NTU-60 dataset. The dataset is evaluated in terms of action classification accuracy (in %) on Cross-Subject (CS) and Cross-View (CV) protocols. Att indicates attention mechanism. ∘ denotes the poses are used only at training time. * indicates that this method has been re-implemented for this dataset.

Methods	Year	Pose	RGB	Att	CS	CV
Lie Group [180]	2014	✓	×	×	50.1	52.8
Skeleton Quads [181]	2014	✓	×	×	38.6	41.4
HBRNN-L [148]	2014	✓	×	×	59.1	64.0
Dynamic Skeletons [166]	2015	✓	×	×	60.2	65.2
Deep LSTM [5]	2016	✓	×	×	60.7	67.3
p-LSTM [5]	2016	✓	×	×	62.9	70.3
ST-LSTM [54]	2016	✓	×	×	69.2	77.7
STA-LSTM [113]	2017	✓	×	✓	73.2	81.2
Ensemble TS-LSTM [182]	2017	✓	×	×	74.6	81.3
GCA-LSTM [183]	2017	✓	×	✓	74.4	82.8
JTM [184]	2018	✓	×	×	76.3	81.1
VA-LSTM [53]	2017	✓	×	×	79.4	87.6
view-invariant [159]	2017	✓	×	×	80.0	87.2
AGC-LSTM [89]	2019	✓	×	✓	89.2	95.0
DGNN [95]	2019	✓	×	×	89.9	96.1
C3D [71]	2015	×	✓	×	63.5	70.3
ResNet50 + LSTM [106]	2019	×	✓	×	71.3	80.2
I3D* [3]	2017	×	✓	×	85.4	87.2
Action Machine [185]	2018	×	✓	×	94.3	97.2
DSSCA-SSLM [172]	2018	✓	✓	×	74.9	-
MTLN [98]	2017	✓	✓	×	79.6	84.8
STA-Hands [114]	2017	✓	✓	✓	82.5	88.6
altered STA-Hands [115]	2018	✓	✓	✓	84.8	90.6
Glimpse Cloud [106]	2018	∘	✓	✓	86.6	93.2
PEM [186]	2018	✓	✓	×	91.7	95.2
RRNX3D101+MS-AAGCN [187]	2019	✓	✓	×	**96.1**	**99.0**
Multi-modal method	2019	✓	✓	×	87.0	-
Multi-modal method (with I3D)	2019	✓	✓	×	92.2	-
P-I3D	2019	✓	✓	✓	93.0	95.4
Separable STA	2019	✓	✓	✓	92.2	94.6
VPN	2020	✓	✓	✓	93.5	96.2
Temporal Model (I3D base)	2020	✓	✓	✓	90.6	92.8
Temporal Model (P-I3D base)	2020	✓	✓	✓	92.4	95.2
Temporal Model (STA base)	2020	✓	✓	✓	91.1	93.0
Temporal Model (VPN base)	2020	✓	✓	✓	92.5	95.5
Global Model (I3D base)	2020	✓	✓	✓	92.5	94.0
Global Model (P-I3D base)	2020	✓	✓	✓	93.9	96.1
Global Model (STA base)	2020	✓	✓	✓	93.6	95.4
Global Model (VPN base)	2020	✓	✓	✓	94.1	96.5

report **92.2**% accuracy (illustrated by - *with I3D*). This performance boosting is because I3D can model better appearance and motion (90.4%) for large available data compared to 2D CNN architectures.

Next, we observe in Table 6.4 that our **P-I3D** framework is able to extract discriminative spatio-temporal features by efficiently weighing the relevant body parts needed for modeling an action. Its effectiveness is corroborated by the increase in action classification accuracy compared to its I3D baseline [3]. P-I3D outperforms the state-of-the-art results on NTU-60. On the contrary, Separable STA under-performs on NTU-60 compared to P-I3D. But it is to be noted that, Separable STA with a single I3D video backbone compared to K $(= 3)$ I3D video backbones in P-I3D, generalizes over the actions. This generalization of Separable STA is evident from the action classification results on Smarthome dataset which comprises of occlusion scenarios and so on.
Note that in table 6.4, Glimpse Cloud [106] uses pose information only for learning but performs significantly worse for cross-subject protocol on NTU dataset. We also argue that PEM [186], whose results are close to those obtained by P-I3D and Separable STA, uses saliency maps of pose estimation. However, these saliency maps can be noisy in case of occlusions, which occur often in Smarthome as well as in most real-world scenarios. On the contrary, our attention mechanism computes attention weights from poses, and the classification ultimately relies on the appearance cue. Our attention mechanisms significantly improve the results especially on NTU-60, by focusing on people interaction and human-object interaction.

Next, we compare VPN to the state-of-the-art on NTU-60. VPN outperforms all the previous methods including our P-I3D and Separable STA. In table 6.4, for input modality RGB+Poses, VPN improves the P-I3D by up to 0.8% on NTU-60 even by using one-third parameters compared to P-I3D. The state-of-the-art using Poses only [95] yields classification accuracy near to VPN for cross-view protocol (with 0.1% difference) due to their robustness to view changes. However, the lack of appearance information restricts these methods [95, 89] to disambiguate actions with similar visual appearance, thus resulting in lower accuracy for cross-subject protocol.

Next, our Temporal Model and Global Model are compared with previous methods. For Temporal and Global Model, we present the results with different video backbones. First with I3D base and then with our proposed P-I3D, Separable STA and VPN as Basic Model. In Table 6.4, we see that although the Temporal Model alone under-performs the other video backbones proposed in Chapter 4, they carry complementary information. This is evident when the Temporal Model is combined with the Basic Model - namely, the Global Model. We call them Global Model (I3D base) or (P-I3D base) and so on - with the base network in parentheses. With the Global Model, we recover the recognition of those actions that are under-represented in the Temporal Model. Such an under-representation

is due to temporal decomposition of extremely short actions (ranging for few seconds). P-I3D [143] with 42M trainable parameters as compared to simple I3D's 12M trainable parameters outperforms the state-of-the-art results on NTU (95% average over CS and CV) datasets when used as a backbone of the Temporal Model. The Global Model with P-I3D as base network has 80M trainable parameters and improves action, with similar motion like *wearing glasses* (+2.5%) and *taking off glasses* (+2.1%) compared to the Basic Model (P-I3D). Whereas, Temporal Model with VPN backbone with parameters comparable to Separable STA and one-third times lower than P-I3D results in superior classification accuracy. Thus, more effective the video backbone, more effective is the Temporal Model.

Finally, we note that methods like Action machine [185] and RNX3D101+MS-AAGCN [187] outperform our models on both the protocols. This is because of their usage of deeper 3D CNN (with 3D ResNet50 and ResNeXt101 respectively) compared to our InceptionV1 configuration of I3D.

6.4 NTU-RGB+D-120

NTU-120, an extension of NTU-60 dataset introduces the challenge of more similar actions compared to its former version. The similar state-of-the-art methods applied on NTU-60 do not perform effectively on this extended dataset as seen in Table 6.5. Similar to NTU-60, in this section, we present a comparison of all our proposed attention based methods with the state-of-the-art methods for multiple modalities (RGB, Pose) on NTU-120 in Table 6.5.

Although both Separable STA and VPN use attention mechanisms and both the methods outperform the state-of-the-art methods on this dataset. But it is worth noting that VPN improves further the classification of actions (by 4.7%) with similar appearance as compared to Separable STA. For example, actions like *clapping* (+44.3%) and *flicking hair* (+19.1%) are now discriminated with better accuracy. In addition, the superior performance of VPN in cross-view protocol for both NTU-120 implies that it provides better view-adaptive characterization compared to all the prior methods.

We also notice that the Temporal Model improves the classification accuracy when used with I3D or Separable STA as video backbone. Whereas VPN degrades the classification accuracy. But the complementary nature of this model shows its efficacy while evaluating for the Global Model. We clearly see in Table 6.5 that the Global Model with VPN video backbone outperforms all the state-of-the-art results on NTU-120 dataset. This also shows that VPN with its spatial embedding already classifies videos with challenging actions, especially the fine-grained ones. As a result, the Temporal Model with VPN as backbone does not significantly improve the action classification performance.

Table 6.5: Comparison between the proposed methods with state-of-the-art methods using different modalities (3D poses and RGB) on NTU-120 dataset. The dataset is evaluated in terms of action classification accuracy (in %) on Cross-Subject (CS_1) and Cross-Setup (CS_2) protocols. Att indicates attention mechanism. * indicates that this method has been re-implemented for this dataset.

Methods	Year	Pose	RGB	Att	CS_1	CS_2
p-LSTM [5]	2016	✓	×	×	25.5	26.3
Dynamic Skeletons [166]	2015	✓	×	×	50.8	54.7
ST-LSTM [54]	2016	✓	×	✓	55.7	57.9
Internal Feature Fusion [188]	2016	✓	×	✓	58.2	60.9
view-invariant (single stream) [159]	2017	✓	×	×	60.3	63.2
GCA-LSTM [183]	2017	✓	×	✓	61.2	63.3
Multi-Task CNN [189]	2018	✓	×	×	62.2	61.8
PEM (single stream) [186]	2018	✓	×	✓	64.6	66.9
2s-AGCN* [97]	2019	✓	×	✓	82.9	84.9
MS-G3D [190]	2020	✓	×	✓	86.9	88.4
Two-streams [1]	2014	×	✓	×	58.5	54.8
I3D* [3]	2017	×	✓	×	77.0	80.1
Two-streams + ST-LSTM [125]	2019	✓	✓	×	61.2	63.1
P-I3D	2019	✓	✓	✓	-	-
Separable STA	2019	✓	✓	✓	83.8	82.5
VPN	2020	✓	✓	✓	86.3	87.8
Temporal Model (I3D base)	2020	✓	✓	✓	85.2	86.5
Temporal Model (P-I3D base)	2020	✓	✓	✓	-	-
Temporal Model (STA base)	2020	✓	✓	✓	85.7	87.1
Temporal Model (VPN base)	2020	✓	✓	✓	85.5	87.1
Global Model (I3D base)	2020	✓	✓	✓	85.9	87.3
Global Model (P-I3D base)	2020	✓	✓	✓	-	-
Global Model (STA base)	2020	✓	✓	✓	87.0	88.1
Global Model (VPN base)	2020	✓	✓	✓	**87.5**	**89.0**

6.5 Toyota Smarthome dataset

Smarthome consists of very diverse videos of activities performed with or without interactions with objects. Existing state-of-the-art methods fail to address all the challenges posed by Smarthome (see Table 6.6).

The dense trajectories (DT) [10] obtain competitive results for actions with relatively high motion. However, dense trajectories are local motion based features and thus fail to model actions with fine-grained details and to incorporate view-invariance in recognizing activities. LSTM, fed with informative 3D joints, models the coarse activities based on body dynamics of the subject performing the activity, but fails to discriminate fine-grained activities due to the lack of object encoding.

Recent inflated convolutions [3] have shown significant improvement compared to RNNs. The non-local behavior of non-local block on top of I3D [85], along space-time in Smarthome is not view-invariant because its attention mechanism relies on appearance. On the contrary, our proposed attention mechanisms are guided by 3D pose information, which is view-invariant. The significant improvement of our attention mechanisms (P-I3D, Separable STA and VPN) on cross-view protocols shows its view-invariant property compared to existing methods. In fig. 6.1 we provide some visual examples in which Separable STA outperforms I3D (without attention). We also observe that, VPN significantly improves the state-of-the-art results on Smarthome. This is largely due to better understanding of the fine-grained actions like *cut bread*, *cooking.stirring* and so on, by VPN. We also note that the Temporal Model in this dataset sometimes outperforms the video backbone itself, for instance in CS protocol using Temporal Model with P-I3D and Separable STA as video backbones. This shows the importance of modeling temporal information in this dataset. As a consequence, the Global Model, especially VPN, shows significant improvement compared to all our results on Smarthome. Our results also substantiates the fact that how important is this pose driven attention mechanism for real-world action recognition. Even today when we are dealing with noisy 3D poses obtained from pose estimation algorithms [134] in the wild, our attention mechanisms recognize the actions. It is to be noted that AssembleNet++ [191] outperforms our models due to the additional use of optical flow as well as object cues. Moreover, it is a NAS architecture which leads to low generalization power of the framework.

We illustrate in fig. 6.2, the top-5 per-class classification improvement with VPN compared to baseline I3D [3] and to Separable STA [132], utilizing 3D poses. The significant accuracy improvements for actions with subtle motion like *hush* (+52.7%), *staple book* (+40.7%) and *reading* (+36.2%) as depicted in fig. 6.2 (a) illustrate the efficacy of VPN for fine-grained actions. It is worth noting that VPN improves further the classification of actions possessing similar appearance as compared to separable STA in fig. 6.2 (b). For

Figure 6.1: Separable STA correctly discriminates the activities with fine-grained details. The model without attention (I3D) is mislead by imposter objects (displayed in red boxes) in the image whereas our proposed separable STA manages to focus on the objects of interest (displayed in green boxes).

Table 6.6: Comparison between the proposed methods with state-of-the-art methods using different modalities (3D poses and RGB) on Smarthome dataset. The dataset is evaluated in terms of **mean average action classification accuracy** (in %) on Cross-Subject (CS) and Cross-View (CV) protocols. Att indicates attention mechanism.

Methods	Year	Pose	RGB	Att	CS	CV
DT [10]	2011	×	✓	×	41.9	23.7
LSTM [192]	2016	✓	×	×	42.5	17.2
I3D [3]	2017	×	✓	×	53.4	45.1
I3D+NL [85]	2018	×	✓	✓	53.6	43.9
AssembleNet++ [191]	2020	×	✓	✓	**63.6**	-
P-I3D	2019	✓	✓	✓	54.0	48.7
Separable STA	2019	✓	✓	✓	54.2	50.3
VPN	2020	✓	✓	✓	60.8	53.5
Temporal Model (I3D base)	2020	✓	✓	✓	55.4	52.8
Temporal Model (P-I3D base)	2020	✓	✓	✓	-	-
Temporal Model (STA base)	2020	✓	✓	✓	56.6	49.5
Temporal Model (VPN base)	2020	✓	✓	✓	60.4	49.9
Global Model (I3D base)	2020	✓	✓	✓	58.5	54.3
Global Model (P-I3D base)	2020	✓	✓	✓	-	-
Global Model (STA base)	2020	✓	✓	✓	60.0	54.6
Global Model (VPN base)	2020	✓	✓	✓	62.6	**55.5**

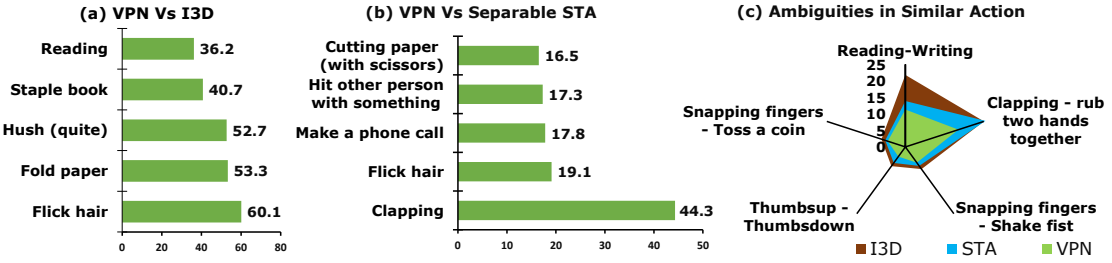


Figure 6.2: Graphs illustrating the superiority of VPN compared to the state-of-the-art methods in terms of accuracy (in %). We present the Top-5 per class improvement for VPN over (a) I3D baseline and (b) Separable STA. In (c), we present a radar for the average mis-classification score of few action-pairs: lower scores indicate lesser ambiguities between the action-pairs.

Table 6.7: Hyper-parameter specifications for various state-of-the-art methods evaluated on Smarthome.

Methods	**Hyper-parameter**	*CS*	CV_1	CV_2
LRCN	# Neurons	256	128	128
	Gradient clipping	1	1	1
	Dropout	0.5	0.6	0.5
LSTM	# Neurons	512	128	128
	Gradient clipping	1	1	1
	Dropout	0.5	0.6	0.5
I3D	Kernel Regularization	L_2 (0.01)	L_2 (0.01)	L_2 (0.01)
	Actitvity Regularization	L_1 (0.01)	L_1 (0.01)	L_1 (0.01)
	Dropout	0.2	0.5	0.5
I3D+NL	# NL blocks	1	1	1
	Kernel Regularization	L_2 (0.01)	L_2 (0.01)	L_2 (0.01)
	Activity Regularization	L_1 (0.01)	L_1 (0.01)	L_1 (0.01)
	Dropout	0.2	0.5	0.5

example, actions like *clapping* (+44.3%) and *flicking hair* (+19.1%) are now discriminated with better accuracy. Further, in fig. 6.2 (c) we present a radar for the average mis-classification score of few action-pairs. The smaller area under the curve for VPN compared to I3D baseline and Separable STA shows that it is able to better disambiguate the action-pairs even with low inter-class variation.

In table 6.7, we provide an overview of our hyper-parameter selection for the state-of-the-art methods bench-marked on Smarthome. This is to enable reproducibility of the results reported in this thesis since this dataset has been bechmarked by us. Note that the kernel and activity regularizers for I3D and I3D+NL are applied in the `softmax` layer. For I3D+NL, we experimented with various numbers of NL blocks at early and late stages. We obtained the highest accuracy with 1 NL block at the last stage. All hyper-parameter optimisations are performed on the validation set of Toyota Smarthome dataset.

6.6 Northwestern - UCLA Multi-view dataset

Northwestern- UCLA multi-view dataset is a human-object interaction dataset. Compared to the other datasets (NTU and Smarthome), it is a relatively small dataset. This introduces the challenge of training our attention mechanism based framework with millions of trainable parameters. An important requirement of our attention mechanisms is the availability of a large number of training samples, which is an issue in NUCLA.

We mitigate the issue of small-scale training samples by pre-training our video backbones with NTU-60 dataset. This improves the performance of our model by a large margin. An ablation study in chapter 4 (table 4.3) corroborates the aforementioned observation. Note that, we pre-train our video backbone with NTU-60 dataset and not pre-train the whole framework with the attention network. This is due to the diversity of actions present in both the datasets. Pre-training the whole framework would result in biased attention scores based on the maximum likelihood of the data distribution.

The pattern of the results on NUCLA in table 6.8 is similar to that of NTU-60 dataset. The classification accuracy improves for I3D baseline when combined with our attention mechanism. It is important to note that similar to NTU-60, here on NUCLA Separable STA under-performs w.r.t. P-I3D. Consistently, VPN outperforms all the proposed video backbones as well as the state-of-the-art results. HPM + TM [193] which achieves classification accuracy close to our attention models utilizes the depth-map information. The depth map is useful in this dataset because of multiple camera setups. Consequently, some videos are captured with the actor performing the action sideways, where the depth information is more crucial compared to the appearance under such scenario of occlusions.

We also note that our proposed Temporal Model and consequently Global Model does not significantly improve the action classification accuracy on NUCLA. This is mainly because of the lack of importance of temporal information in this human-object interaction dataset. These videos with very short time duration hardly encode any temporal order or dependencies.

Thus, spatial and temporal attention mechanisms in videos should be handled in a strategic manner as one or the other domain could be more discriminative while classifying actions in videos.

6.7 Conclusion

In this chapter, we have evaluated our proposed algorithms based on multi-modal fusion and attention mechanisms on public datasets. Firstly, we evaluate our Multi-modal Fusion strategy with similar action discrimination module on three public datasets CAD60, CAD120 and MSRDailyActivity3D. Our results outperform the state-of-the-art results on all

Table 6.8: Comparison between our proposed methods with state-of-the-art using different modalities (3D poses, Depth and RGB) on NUCLA dataset. The dataset is evaluated in terms of action classification accuracy (in %) on Cross-View ($V^3_{1,2}$) protocols. Att indicates attention mechanism. $\overline{Pose}$ indicates its usage only in the training phase. * indicates that this method has been re-implemented for this dataset.

Methods	Year	Data	Att	$V^3_{1,2}$
AOG [124]	2014	Depth	×	45.2
DVV [194]	2012	Depth	×	58.5
CVP [195]	2013	Depth	×	60.6
HPM+TM [193]	2016	Depth	×	91.9
Lie group [180]	2014	Pose	×	74.2
HBRNN-L [148]	2014	Pose	×	78.5
view-invariant [159]	2017	Pose	×	86.1
Ensemble TS-LSTM [182]	2017	Pose	×	89.2
SGN [196]	2020	Pose	×	92.5
Hankelets [197]	2012	RGB	×	45.2
nCTE [198]	2014	RGB	×	68.6
NKTM [199]	2015	RGB	×	75.8
I3D* [3]	2017	RGB	×	86.0
Action Machine [185]	2018	RGB	×	92.3
Glimpse Cloud [106]	2018	RGB+ $\overline{Pose}$	✓	90.1
P-I3D	2019	RGB + Pose	✓	93.1
Separable STA	2019	RGB + Pose	✓	92.4
VPN	2020	RGB + Pose	✓	93.5
Temporal Model (I3D base)	2020	RGB + Pose	✓	89.5
Temporal Model (P-I3D base)	2020	RGB + Pose	✓	88.6
Temporal Model (STA base)	2020	RGB + Pose	✓	90.8
Temporal Model (VPN base)	2020	RGB + Pose	✓	91.4
Global Model (I3D base)	2020	RGB + Pose	✓	91.0
Global Model (P-I3D base)	2020	RGB + Pose	✓	93.5
Global Model (STA base)	2020	RGB + Pose	✓	92.8
Global Model (VPN base)	2020	RGB + Pose	✓	**93.7**

the three small-scale public datasets. This fusion strategy makes use of all the modalities in a strategic manner so as to take into account their pros for the task of classifying actions. Although this method performs substantially well on small scale datasets, not scalable enough for large datasets. This Multi-modal method relies too much on the distribution of validation data which makes it a weak classifier for large scale dataset. A method adapting global optimization to optimize all the cues is a future research direction.

Then we evaluated our proposed attention mechanisms on four public datasets and compared them with the state-of-the-art methods. We observe that our proposed P-I3D achieves state-of-the-art results on NTU-60 and NUCLA. Even P-I3D outperforms our proposed Separable STA but with three times more trainable parameters. Simultaneously, Separable STA is the attempt to generalize the objective of P-I3D, specifically to attain spatio-temporal attention in set of images without pre-defined constraints of human body parts. The results of Separable STA on Smarthome dataset validates our approach to generalize over a challenging scenario where occlusion and temporal ordering plays a vital role in classifying actions.

But these methods P-I3D and Separable STA do not take into account the mis-alignment of 3D poses with the image frames. These 3D poses are in turn exploited for computing attention weights. Our proposed VPN based framework introduces the concept of Spatial embedding which enforces the 3D poses and the corresponding image frames to be projected into a common feature space. This enables the aforementioned attention models to perform exceptionally for all the datasets we validate. In addition, we also make use of the graphical structure of the 3D poses through GCNs in VPN. Cumulatively, this model achieves the state-of-the-art results which includes outperforming our proposed P-I3D and Separable STA on all the four datasets.

Finally we have evaluated our temporal representation of videos through our Temporal Model on all the datasets. The results on Temporal Model show a similar trend on NTU-60 and NUCLA, pertaining to lower classification accuracy compared to their video backbones. But this model extracts complementary information owing to the temporal representation of the videos and hence when combined with the Basic Model (only the video backbone probabilistic scores) - achieves the state-of-the-art results. We have validated the Temporal representation of videos with I3D backbone and also with our proposed P-I3D, Separable STA and VPN as video backbones. We also conclude that effective video backbones result in more effective action classification.

Chapter 7

Conclusion and Future Work

In this thesis we have proposed and evaluated several methods for action recognition in videos. Our experiments demonstrated that we have outperformed the state-of-the-art methods on seven public datasets. We conclude our work pointing out key contributions (section 7.1) and their limitations (section 7.2). Finally, we discuss short and long-term perspectives of our work (section 7.3).

7.1 Key Contributions

- **Multi-modal Video Representation** - We proposed a video representation utilizing multi-modal features from several cues. Current state-of-the-art mainly focuses on combining RGB and optical Flow, whereas we have proposed an effective way of combining RGB, optical flow and 3D Poses taking pros from each of them. The novelty in this work includes the two-level fusion mechanism dedicated for the specific modalities, firstly an early feature fusion of RGB and optical flow and secondly, a score level fusion of the aforementioned combined cues and 3D poses. We show that this is one of the most effective fusion mechanism to discriminate videos pertaining to different actions, utilizing the discriminative features from each modality. Finally, the challenge to disambiguate similar actions is addressed by proposing a similar action discriminator module. Based on the distribution of data in the validation set, this module invokes binary classifiers to disambiguate similar actions.

- **Spatio-temporal attention mechanisms** - We proposed three variants of attention mechanism for action recognition in short videos. Inspired from the effectiveness of our Multi-modal representation of videos, we have proposed a pose driven attention mechanism which makes use of the temporal evolution of 3D poses (human joints) to (i) weight the pertinent human body parts, (ii) weight the RoI in an image frame,

and (iii) weight the key frames in a video, relevant for classifying an action in videos. We proposed our Spatial attention network (P-I3D) to accomplish (i) and Separable STA, another attention network to accomplish (ii) and (iii) in a dissociated manner. Both these variants of attention network provide attention weights based on pose backbone (which are generally LSTMs) pre-trained for action classification on 3D poses. This pre-training trick enables the LSTMs to better understand the evolution of poses.

Then, we proposed a Pose driven attention mechanism for classifying actions, to provide spatial and temporal attention weights coupled together. We proposed a module named VPN, which makes use of the graphical structure of human anatomy through GCNs (instead of LSTMs), and provides spatio-temporal attention weights to the RGB cue. Apart from these steps to mitigate the limitations of the aforementioned frameworks, we also handled the problem often overlooked in vision exploiting multiple modalities. That is, the mis-alignment of 3D poses to the image frames. We proposed a spatial embedding module to enforce the 3D poses and the image-level features in a common semantic space. This improves the efficacy of our proposed attention network while enabling the framework to better discriminate similar and fine-grained actions in the video space compared to the prior methods.

- **Temporal representation of videos** - Our earlier proposed methods are all designed for short action videos. The presence of long and complex actions in the daily lives inspired us to dig deeper towards temporal representation of videos. We have proposed a Temporal Model, which divides a video at different temporal granularities. At a temporal granularity, the video is divided into several temporal segments, which correspond to a partition of the video. Our Temporal Model first processes these temporal segments through 3D CNN and then each granularity is represented by a linear combination of their constituent segments. Finally, a video is represented by the combination of all the granularities through a recurrent function. We improve this global video-level representation by providing a pose driven two-level attention mechanism - first to soft-weight the different temporal segments for each granularity and then to soft-weight the granularities. We show that this Temporal Model computes complementary features w.r.t. a model without temporal decomposition. To ensure completeness of our framework, we proposed a Global Model - a straightforward fusion of Temporal Model and a Basic Model (the model without temporal decomposition). This Global Model with our attention mechanism as video backbones outperform state-of-the-art results on four public datasets.

7.2 Limitations

The methods in this thesis have still some limitations. Some of these limitation can be extended and solved in the near future. While others are still open research questions. In this section, we present the limitations of our approach to solve the task of action recognition.

- **Multi-modal Video Representation** - Although we show a significant improvement in the classification accuracy of three public datasets compared to the state-of-the-art methods, it is not an optimal solution for generic datasets. Firstly, the deep features extracted are optimized independently and finally used as feature extractors rather than taking the full advantage of Deep Neural Network methods, which is global optimization. The next limitation of this fusion strategy is that this method falls short, as the model designer needs to choose empirically which intermediate features to consider for fusion. Evaluating all of the possibilities by hand would be extremely intensive or simply intractable. Indeed, the more modalities and the deeper they are, the more complicated it is to choose a mixture. This is all the more true when enabling nested combinations of multi-modal features. It is in fact a large combinatorial problem.

  Finally, the similar action discriminator module which is responsible to disambiguate similar actions, highly relies on the likelihood of the validation data. If the sampling of the validation data fails to capture the critical data samples that are not often separable in a common feature space, the aforementioned module fails to invoke a binary classifier for similar actions at inference time. Another concern is employing too many binary classifiers for similar action pair is too costly in terms of space and time for a model complexity.

- **Spatio-temporal attention mechanisms** - In our proposed methods for classifying action in short videos, we use pose driven attention mechanism. We compute the attention weights for the RoI in an image and also the key frames in a video. This is how our dependency on the 3D poses to compute rational attention weights which are never evaluated (due to no Ground-truth). We are aware of the fact that pose estimation algorithms though deployed in the wild are not effective in complex scenarios especially in case of occlusions. Visually, we have seen that the 3D poses in Smarthome dataset (extracted using LCRNet algorithm) are quite noisy. Thus, in such scenarios bad quality 3D poses surely hamper the computation of attention weights.

  The next question is where to apply the attention weights? Initially in P-I3D, we conducted few experiments to conclude that applying attention weights at the last

layer of a video backbone is more effective than applying it in the earlier layers. However, we believe that attention weights should be applied from the very initial convolutional layers, instead of applying them only at the end where the features represent a squeezed part of the video. Then, why attention does not seems to work when applied in the earlier layers? It is because of the max pooling operation in the CNNs which diminishes the semantic information of the video if high attention weights are applied to a certain region in the initial layers. Anyways, we still believe that applying attention in the initial layers or replacing conventional layers with attentional layers will be more effective than conventional way of applying attention weights.

- **Temporal representation of videos** - In this method, we explore multiple granularities in a video. However, this number of granularities is a hyper-parameter and is a choice of an expert depending on the dataset complexity. We observed that this hyper-parameter G_{max} is sensitive to the duration of the videos and also on the nature of actions. For instance, a *cooking* action composed of several sub-actions is a complex action in nature compared to an action like *drinking*. Thus, a wrong choice of the parameter number of granularities can hamper the performance of the action. Moreover, at a granularity, we divide the video into equally spaced non-overlapping partitions that we call temporal segments. This linear partitioning mechanism is a brute-force mechanism of dividing a video. We also believe that such a mechanism is not optimal as this might damage the temporal structure or continuity of an action.

7.3 Future Work

7.3.1 Short-term Perspectives

- **Generating 3D poses and classifying them for an action** - We aim at exploring more the effects of different algorithms to generate 3D poses for action recognition. For instance, LCRNet [134] computes frame-wise 3D poses whereas VideoPose3D [200] uses temporal convolutions to compute 3D poses in a video. These 3D poses differ in terms of their quality under occlusions and also in terms of smoothness. However, both these algorithms have their pros, one for generating high-quality smooth 3D poses for complex situations and another for generating precise 3D poses reflecting the fine-grained motion in the videos. Thus, the required quality of 3D poses depend on the type of action to be recognized. And so, generating these 3D poses and classifying them in an end-to-end manner is an interesting direction of research. This strategy should enable the model to generate 3D poses appropriate for classifying the actions.

- **Improving the Temporal Model** - For temporal representation of videos, we have incorporated our proposed spatio-temporal attention based video backbones for effectiveness and completeness of our framework. Thus, we use two attention networks for these frameworks, one for the video backbone and another for the Temporal Model. In order to have a global optimization, and better learning of attention weights compared to current dissociated learning mechanism, we could incorporate both the attention networks into one. This single attention network will serve both the functionalities of providing attention weights to the video backbone as well as for the temporal representation of videos.

  Another minor replacement can be done with processing the 3D poses in GCNs followed by TCNs instead of GRUs. Our literature survey and experiments with VPN shows a clear improvement of using the graphical structure of the human spatial configuration while computing attention weights through GCNs.

- **Extending Global Model for Action Detection** - As mentioned in the beginning of the thesis, we aim at finally detecting action in complex scenarios. The first step is action classification in clipped videos. Now, when we have a working framework for action recognition, i.e. the Global Model which could work effectively for short as well as long actions, extending it for untrimmed videos is an obvious future work [201].

7.3.2 Long-term Perspectives

- **NAS for Multi-modal video representation** - One of the limitation of the current state-of-the-art multi-modal fusion mechanisms involve handcrafted choice of feature space where the fusion must take place. In fact, such fusion strategies do not take into account other possibilities of fusing modalities like (i) at which layers the fusion must take place, (ii) which convolutional operations and activations each modalities must undergo before and after fusion. Some automated techniques [202, 8] have been applied to solve this problem using *AutoML* but limited to RGB and optical flow modalities. Thus, one possible research direction can be towards searching models using NAS (Neural Architecture Search) for combining modalities like RGB and 3D poses for recognition of ADL. The challenges are to (i) minimize the searching space complexity, (ii) taking benefit of currently available models pre-trained on Kinetics instead of using models from scratch.

- **Towards hard-attention using RL** - One of the primary benefit of using soft-attention is its formulation which is differentiable. This enables these mechanisms to be trained end-to-end. At the same time, hard attention has its own advantages

with only processing the most meaningful part of the video ignoring the rest. On the other side, current state-of-the-art 3D CNNs for action recognition are either fed with random crops from a video or tracks of human body while training them. But as we know data augmentation is a complex process for video based networks. Random cropping is not often optimal for ADL whereas human crops are not optimal for sport videos, while training action recognition systems. Thus, to have a generic model with deeper understanding of the semantics of the actions, a step towards using Reinforcement Learning (RL) for choosing these crops of image set (stack of images) can be a possible solution. The other solution can be to use a similar concept but with a differential loss. This might lead to an extension of spatial transformers [109] in the temporal domain. However, the challenge is that unlike in the image domain, the transformations in a video must not be an affine transformation to select the RoI in a video. Rather such transformations should be dependent on the human tracklets.

- **View Adaptive Action Recognition** - View adaptation is one of the challenge in action recognition. We know that CNNs are not view-adaptive. For image classification networks, we often perform data augmentation technique like rotation specifically to mitigate the problem of variance in view. However, for video based networks, this is still a challenging task. We know that 3D poses are robust to views and thus, somewhat mitigates this problem of view variance in action recognition. But what about view-invariant property of RGB modality? One possible future direction is to generate RGB images from a view that facilitate action recognition using adversarial loss.

- **Domain Adaptation for Action Recognition** - The limitation of the current action recognition algorithms is that they exhibit environmental bias. Training a model in one environment and deploying in another results in a drop in action classification accuracy due to an unavoidable domain shift. Learning representations that generalize over source and target distributions for recognizing action is still an open research problem. Utilizing modality like 3D poses which are robust to domain shifts, illumination and views are an obvious choice for some sort of control vectors to accomplish this aforementioned task.

- **Weakly-supervised Action Detection** - With Action detection algorithms advancing in the current era, weakly supervised action detection under the absence of temporal annotations is a possible research direction. Current weakly-supervised action detection algorithms use Multiple Instance Learning (MIL) to learn the actionness in an untrimmed video as in [203]. But most of these algorithms are limited to videos

having the same action instances repeated in an untrimmed video. Consequently, we aim at using video-level labels to disambiguate set of actions occurring in a video.

Bibliography

[1] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, pp. 568–576, 2014. (Cited on pages xiii, 16, 17, 18, 19, 20, 29, 49, 50 and 140.)

[2] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-Scale Video Classification with Convolutional Neural Networks," in *CVPR*, 2014. (Cited on pages xiii, 18, 19 and 20.)

[3] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4724–4733, IEEE, 2017. (Cited on pages xiii, xiv, 9, 18, 20, 21, 24, 29, 34, 60, 67, 70, 72, 77, 78, 90, 92, 126, 136, 137, 138, 140, 141, 142 and 145.)

[4] N. Hussein, E. Gavves, and A. W. M. Smeulders, "Timeception for complex action recognition," *CoRR*, vol. abs/1812.01289, 2018. (Cited on pages xiii, 22, 25, 68 and 116.)

[5] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+d: A large scale dataset for 3d human activity analysis," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. (Cited on pages xiii, xvii, 16, 17, 24, 26, 34, 35, 36, 37, 38, 49, 55, 117, 120, 126, 137 and 140.)

[6] Q. Ke, M. Bennamoun, S. An, F. A. Sohel, and F. Boussaïd, "A new representation of skeleton sequences for 3d action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 4570–4579, IEEE Computer Society, 2017. (Cited on pages xiv, 24 and 27.)

[7] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "MARS: Motion-Augmented RGB Stream for Action Recognition," in *CVPR*, 2019. (Cited on pages xiv, 20 and 29.)

[8] M. S. Ryoo, A. Piergiovanni, M. Tan, and A. Angelova, "Assemblenet: Searching for multi-stream neural connectivity in video architectures," in *International Conference on Learning Representations*, 2020. (Cited on pages xiv, 29 and 151.)

[9] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Mining actionlet ensemble for action recognition with depth cameras," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. (Cited on pages xiv, xix, 34, 37, 38 and 52.)

[10] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action Recognition by Dense Trajectories," in *IEEE Conference on Computer Vision & Pattern Recognition*, (Colorado Springs, United States), pp. 3169–3176, June 2011. (Cited on pages xiv, xix, 16, 49, 51, 52, 141 and 142.)

[11] S. Das, M. Koperski, F. Brémond, and G. Francesca, "Deep-temporal lstm for daily living action recognition," *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, 2018. (Cited on pages xiv and 52.)

[12] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008. (Cited on pages xv, 62 and 63.)

[13] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," in *IJRR*, 2013. (Cited on pages xix, 34, 37 and 52.)

[14] M. Koperski, *Human action recognition in videos with local representation*. Theses, Université Côte d'Azur, Nov. 2017. (Cited on pages xix and 52.)

[15] "Department of economic and social affairs population of united nations, the world population aging report." `https://www.un.org/en/development/desa/population/`. Accessed Feb. 28th, 2020. (Cited on page 4.)

[16] G. A. Sigurdsson, O. Russakovsky, and A. Gupta, "What actions are needed for understanding human actions in videos?," in *International Conference on Computer Vision (ICCV)*, 2017. (Cited on pages 4 and 6.)

[17] J. Munro and D. Damen, "Multi-modal Domain Adaptation for Fine-grained Action Recognition," in *Computer Vision and Pattern Recognition (CVPR)*, 2020. (Cited on page 5.)

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012. (Cited on page 6.)

[19] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," in *British Machine Vision Conference*, 2014. (Cited on pages 6 and 25.)

[20] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009. (Cited on pages 6, 20, 23, 90 and 126.)

[21] T.-Y. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014. (Cited on page 6.)

[22] K. Soomro, A. Roshan Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," 12 2012. (Cited on pages 6, 20, 22 and 50.)

[23] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: a large video database for human motion recognition," in *2011 International Conference on Computer Vision*, pp. 2556–2563, IEEE, 2011. (Cited on pages 6, 34 and 50.)

[24] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017. (Cited on pages 6, 20, 90 and 126.)

[25] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *European Conference on Computer Vision(ECCV)*, 2016. (Cited on pages 6, 34 and 36.)

[26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012. (Cited on page 14.)

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, pp. 1735–1780, Nov. 1997. (Cited on pages 14, 23 and 115.)

[28] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 257–267, March 2001. (Cited on page 15.)

[29] E. Shechtman and M. Irani, "Matching local self-similarities across images and videos," in *IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07)*, June 2007. (Cited on page 15.)

[30] Y. Ke, R. Sukthankar, and M. Hebert, "Spatio-temporal shape and flow correlation for action recognition," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2007. (Cited on page 15.)

[31] M. D. Rodriguez, J. Ahmed, and M. Shah, "Action mach a spatio-temporal maximum average correlation height filter for action recognition," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008. (Cited on page 15.)

[32] I. Laptev and T. Lindeberg, "Space-time interest points," in *ICCV*, 2003. (Cited on page 16.)

[33] C. Harris and M. Stephens, "A combined corner and edge detector," in *In Proc. of Fourth Alvey Vision Conference*, pp. 147–151, 1988. (Cited on page 16.)

[34] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proceedings of the 14th International Conference on Computer Communications and Networks*, ICCCN â05, (USA), p. 65â72, IEEE Computer Society, 2005. (Cited on page 16.)

[35] M. Bregonzio, Shaogang Gong, and Tao Xiang, "Recognising action as clouds of space-time interest points," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1948–1955, June 2009. (Cited on page 16.)

[36] K. Rapantzikos, Y. Avrithis, and S. Kollias, "Dense saliency-based spatiotemporal feature points for action recognition," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1454–1461, June 2009. (Cited on page 16.)

[37] P. Matikainen, M. Hebert, and R. Sukthankar, "Trajectons: Action recognition through the motion analysis of tracked features," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, pp. 514–521, Sep. 2009. (Cited on page 16.)

[38] R. Messing, C. Pal, and H. Kautz, "Activity recognition using the velocity histories of tracked keypoints," in *2009 IEEE 12th International Conference on Computer Vision*, pp. 104–111, Sep. 2009. (Cited on page 16.)

[39] Ju Sun, Xiao Wu, Shuicheng Yan, L. Cheong, T. Chua, and Jintao Li, "Hierarchical spatio-temporal context modeling for action recognition," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2004–2011, June 2009. (Cited on page 16.)

[40] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *IEEE International Conference on Computer Vision*, (Sydney, Australia), 2013. (Cited on pages 16, 51, 54 and 115.)

[41] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAIâ81, (San Francisco, CA, USA), p. 674â679, Morgan Kaufmann Publishers Inc., 1981. (Cited on page 16.)

[42] D. Hogg, "Model-based vision: a program to see a walking person," *Image and Vision Computing*, vol. 1, no. 1, pp. 5 – 20, 1983. (Cited on pages 16 and 54.)

[43] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld, "Learning realistic human actions from movies," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pp. 1–8, IEEE, 2008. (Cited on pages 16 and 54.)

[44] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *European conference on computer vision*, pp. 428–441, Springer, 2006. (Cited on pages 16 and 54.)

[45] G. Cheron, I. Laptev, and C. Schmid, "P-cnn: Pose-based cnn features for action recognition," in *ICCV*, 2015. (Cited on pages 16, 17, 19, 29, 53, 70 and 72.)

[46] Y. Wu, "Mining actionlet ensemble for action recognition with depth cameras," in *CVPR*, 2012. (Cited on pages 16, 17, 134 and 136.)

[47] B. Amor, J. Su, and A. Srivastava, "Action Recognition Using Rate-Invariant Analysis of Skeletal Shape Trajectories," *PAMI*, vol. 38, pp. 1–13, Jan. 2016. (Cited on pages 16 and 17.)

[48] F. Negin, F. Özdemir, C. B. Akgül, K. A. Yüksel, and A. Erçil, "A decision forest based feature selection framework for action recognition from rgb-depth cameras," in *ICIAR*, 2013. (Cited on pages 16 and 17.)

[49] R. Vemulapalli and R. Chellappa, "Rolling rotations for recognizing human actions from 3dskeletal data," in *CVPR*, 2016. (Cited on pages 16 and 17.)

[50] D. Wu and L. Shao, "Leveraging hierarchial parametric networks for skeletal joints based action segmentation and recognition," in *CVPR*, 2014. (Cited on pages 16 and 17.)

[51] C. Liu, Y. Hu, Y. Li, S. Song, and J. Liu, "Pku-mmd: A large scale benchmark for skeleton-based human action understanding," in *Proceedings of the Workshop on*

Visual Analysis in Smart and Connected Communities, VSCC '17, (New York, NY, USA), pp. 1–8, ACM, 2017. (Cited on pages 16 and 17.)

[52] S. Zhang, X. Liu, and J. Xiao, "On geometric features for skeleton-based action recognition using multilayer lstm networks," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 148–157, March 2017. (Cited on pages 16, 17, 49 and 55.)

[53] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive recurrent neural networks for high performance human action recognition from skeleton data," in *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. (Cited on pages 16, 17, 55 and 137.)

[54] J. Liu, A. Shahroudy, D. Xu, and G. Wang, "Spatio-temporal lstm with trust gates for 3d human action recognition," in *Computer Vision – ECCV 2016* (B. Leibe, J. Matas, N. Sebe, and M. Welling, eds.), (Cham), pp. 816–833, Springer International Publishing, 2016. (Cited on pages 16, 32, 55, 137 and 140.)

[55] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR*, 2011. (Cited on pages 17 and 126.)

[56] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pp. 1933–1941, IEEE, 2016. (Cited on pages 17, 19, 29 and 50.)

[57] J. Zhao and C. G. M. Snoek, "Dance with flow: Two-in-one stream action detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. (Cited on page 19.)

[58] S. Das, M. Koperski, F. Bremond, and G. Francesca, "Action recognition based on a mixture of rgb and depth based skeleton," in *AVSS*, 2017. (Cited on pages 19, 20, 49, 134, 135 and 136.)

[59] L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. Gehler, and B. Schiele, "Deepcut: Joint subset partition and labeling for multi person pose estimation," in *CVPR*, 2016. (Cited on page 19.)

[60] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," *arXiv preprint arXiv:1611.08050*, 2016. (Cited on pages 19, 53 and 60.)

[61] Z. Wu, X. Wang, Y. Jiang, H. Ye, and X. Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," *CoRR*, vol. abs/1504.01561, 2015. (Cited on pages 19 and 20.)

[62] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, "Sequential deep learning for human action recognition," in *Human Behavior Understanding* (A. A. Salah and B. Lepri, eds.), (Berlin, Heidelberg), pp. 29–39, Springer Berlin Heidelberg, 2011. (Cited on page 20.)

[63] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. (Cited on pages 20 and 55.)

[64] J. Yue-Hei, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *CVPR*, 2015. (Cited on page 20.)

[65] F. Li, C. Gan, X. Liu, Y. Bian, X. Long, Y. Li, Z. Li, J. Zhou, and S. Wen, "Temporal modeling approaches for large-scale youtube-8m video understanding," *CoRR*, vol. abs/1707.04555, 2017. (Cited on page 20.)

[66] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised learning of video representations using lstms," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICMLâ15, p. 843â852, JMLR.org, 2015. (Cited on page 20.)

[67] C. Sun, S. Shetty, R. Sukthankar, and R. Nevatia, "Temporal localization of fine-grained actions in videos by domain transfer from web images," in *Proceedings of the 23rd ACM International Conference on Multimedia*, MM â15, (New York, NY, USA), p. 371â380, Association for Computing Machinery, 2015. (Cited on page 20.)

[68] C. Gan, T. Yao, K. Yang, Y. Yang, and T. Mei, "You lead, we exceed: Labor-free video concept learning by jointly exploiting web videos and images," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 923–932, June 2016. (Cited on page 20.)

[69] Y. Bian, C. Gan, X. Liu, F. Li, X. Long, Y. Li, H. Qi, J. Zhou, S. Wen, and Y. Lin, "Revisiting the effectiveness of off-the-shelf temporal modeling approaches for large-scale video classification," *CoRR*, vol. abs/1708.03805, 2017. (Cited on page 20.)

[70] S. Abu-El-Haija, N. Kothari, J. Lee, P. Natsev, G. Toderici, B. Varadarajan, and S. Vijayanarasimhan, "Youtube-8m: A large-scale video classification benchmark," *arXiv preprint arXiv:1609.08675*, 2016. (Cited on pages 20 and 22.)

[71] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, (Washington, DC, USA), pp. 4489–4497, IEEE Computer Society, 2015. (Cited on pages 20, 30, 67, 115 and 137.)

[72] M. Lin, Q. Chen, and S. Yan, "Network in network," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2014. (Cited on page 20.)

[73] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. (Cited on pages 20, 24, 33 and 67.)

[74] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XV* (V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, eds.), vol. 11219 of *Lecture Notes in Computer Science*, pp. 318–335, Springer, 2018. (Cited on pages 21 and 24.)

[75] A. Miech, I. Laptev, and J. Sivic, "Learnable pooling with context gating for video classification," *CoRR*, vol. abs/1706.06905, 2017. (Cited on page 21.)

[76] K. Hara, H. Kataoka, and Y. Satoh, "Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet?," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. (Cited on pages 21 and 33.)

[77] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. (Cited on pages 21 and 24.)

[78] C. Feichtenhofer, "X3d: Expanding architectures for efficient video recognition," 2020. (Cited on pages 21 and 24.)

[79] B. Zhou, A. Andonian, and A. Torralba, "Temporal relational reasoning in videos," in *ECCV*, 2017. (Cited on pages 22 and 116.)

[80] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Val Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *ECCV*, 2016. (Cited on pages 22 and 116.)

[81] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. V. Gool, "Temporal segment networks: Towards good practices for deep action recognition," *ArXiv*, vol. abs/1608.00859, 2016. (Cited on pages 22 and 116.)

[82] S. Song, N.-M. Cheung, V. Chandrasekhar, and B. Mandal, "Deep adaptive temporal pooling for activity recognition," in *Proceedings of the 26th ACM International Conference on Multimedia*, MM '18, (New York, NY, USA), pp. 1829–1837, ACM, 2018. (Cited on page 22.)

[83] X. Wang and A. Gupta, "Videos as space-time region graphs," in *ECCV*, 2018. (Cited on pages 22 and 33.)

[84] C.-Y. Wu, C. Feichtenhofer, H. Fan, K. He, P. Krahenbuhl, and R. Girshick, "Long-term feature banks for detailed video understanding," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. (Cited on page 23.)

[85] X. Wang, R. B. Girshick, A. Gupta, and K. He, "Non-local neural networks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7794–7803, 2018. (Cited on pages 23, 24, 33, 34, 68, 92, 141 and 142.)

[86] V. Veeriah, N. Zhuang, and G.-J. Qi, "Differential recurrent neural networks for action recognition," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV â15, (USA), p. 4041â4049, IEEE Computer Society, 2015. (Cited on page 24.)

[87] Yong Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1110–1118, June 2015. (Cited on page 24.)

[88] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAIâ16, p. 3697â3703, AAAI Press, 2016. (Cited on page 24.)

[89] C. Si, W. Chen, W. Wang, L. Wang, and T. Tan, "An attention enhanced graph convolutional lstm network for skeleton-based action recognition," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1227–1236, June 2019. (Cited on pages 24, 137 and 138.)

[90] R. Pascanu, Ãaglar GülÃ§ehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," *CoRR*, vol. abs/1312.6026, 2013. (Cited on page 24.)

[91] Y. Du, Y. Fu, and L. Wang, "Skeleton based action recognition with convolutional neural network," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pp. 579–583, Nov 2015. (Cited on pages 24 and 25.)

[92] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, vol. 68, pp. 346 – 362, 2017. (Cited on pages 24 and 25.)

[93] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016. (Cited on page 25.)

[94] Y. Tang, Y. Tian, J. Lu, P. Li, and J. Zhou, "Deep progressive reinforcement learning for skeleton-based action recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5323–5332, June 2018. (Cited on pages 25 and 28.)

[95] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with directed graph neural networks," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. (Cited on pages 25, 28, 137 and 138.)

[96] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *AAAI*, 2018. (Cited on pages 25 and 28.)

[97] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *CVPR*, 2019. (Cited on pages 25 and 140.)

[98] M. Zolfaghari, G. L. Oliveira, N. Sedaghat, and T. Brox, "Chained multi-stream networks exploiting pose, motion, and appearance for action classification and detection," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 2923–2932, IEEE, 2017. (Cited on pages 30, 50 and 137.)

[99] H. Rahmani and M. Bennamoun, "Learning action recognition model from depth and skeleton videos," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5833–5842, Oct 2017. (Cited on page 30.)

[100] G. Liu, J. Qian, F. Wen, X. Zhu, R. Ying, and P. Liu, "Action recognition based on 3d skeleton and rgb frame fusion," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 258–264, Nov 2019. (Cited on pages 30 and 31.)

[101] A. Shahroudy, G. Wang, and T. Ng, "Multi-modal feature fusion for action recognition in rgb-d sequences," in *2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pp. 1–4, May 2014. (Cited on pages 30 and 31.)

[102] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015. (Cited on pages 30 and 103.)

[103] Z. Luo, J.-T. Hsieh, L. Jiang, J. Carlos Niebles, and L. Fei-Fei, "Graph distillation for action detection with privileged modalities," in *The European Conference on Computer Vision (ECCV)*, September 2018. (Cited on pages 30, 31 and 50.)

[104] J.-M. Perez-Rua, V. Vielzeuf, S. Pateux, M. Baccouche, and F. Jurie, "Mfas: Multimodal fusion architecture search," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. (Cited on page 30.)

[105] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, (Cambridge, MA, USA), pp. 2204–2212, MIT Press, 2014. (Cited on pages 31 and 32.)

[106] F. Baradel, C. Wolf, J. Mille, and G. W. Taylor, "Glimpse clouds: Human activity recognition from unstructured feature points," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. (Cited on pages 31, 33, 34, 40, 77, 81, 82, 83, 85, 136, 137, 138 and 145.)

[107] S. Yeung, O. Russakovsky, G. Mori, and L. Fei-Fei, "End-to-end learning of action detection from frame glimpses in videos," *arXiv preprint arXiv:1511.06984*, 2015. (Cited on page 32.)

[108] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang, "Residual attention network for image classification," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6450–6458, July 2017. (Cited on page 32.)

[109] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu, "Spatial transformer networks," *ArXiv*, vol. abs/1506.02025, 2015. (Cited on pages 32 and 152.)

[110] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017. (Cited on page 32.)

[111] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, A. Ku, and D. Tran, "Image transformer," in *ICML*, 2018. (Cited on page 32.)

[112] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," in *International Conference on Learning Representations (ICLR) Workshop*, May 2016. (Cited on pages 32, 34 and 67.)

[113] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "An end-to-end spatio-temporal attention model for human action recognition from skeleton data," in *AAAI Conference on Artificial Intelligence*, pp. 4263–4270, 2017. (Cited on pages 32, 34, 40, 74, 80, 81, 86, 98, 117, 122 and 137.)

[114] F. Baradel, C. Wolf, and J. Mille, "Human action recognition: Pose-based attention draws focus to hands," in *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 604–613, Oct 2017. (Cited on pages 32, 33, 34, 70, 74, 81, 82, 83, 93, 122 and 137.)

[115] F. Baradel, C. Wolf, and J. Mille, "Human activity recognition with pose-driven attention to rgb," in *The British Machine Vision Conference (BMVC)*, September 2018. (Cited on pages 33, 34, 40, 67, 77, 81, 82, 83, 85 and 137.)

[116] DARPA and Kitware, "Virat video dataset." `http://www.viratdata.org/`. Accessed Feb. 28th, 2019. (Cited on page 34.)

[117] S. M. Safdarnejad, X. Liu, L. Udpa, B. Andrus, J. Wood, and D. Craven, "Sports videos in the wild (svw): A video dataset for sports analysis," in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 1, pp. 1–7, IEEE, 2015. (Cited on page 34.)

[118] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, C. Schmid, and J. Malik, "Ava: A video dataset of spatio-temporally localized atomic visual actions," *Conference on Computer Vision and Pattern Recognition(CVPR)*, 2018. (Cited on page 34.)

[119] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, and M. Wray, "Scaling egocentric vision: The EPIC-KITCHENS dataset," *CoRR*, vol. abs/1804.02748, 2018. (Cited on pages 34 and 36.)

[120] R. Goyal, S. E. Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fründ, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic, "The "something something" video database for learning and evaluating visual common sense," *CoRR*, vol. abs/1706.04261, 2017. (Cited on page 34.)

[121] M. Rohrbach, A. Rohrbach, M. Regneri, S. Amin, M. Andriluka, M. Pinkal, and B. Schiele, "Recognizing fine-grained and composite activities using hand-centric

features and script data," *International Journal of Computer Vision*, pp. 1–28, 2015. (Cited on pages 34 and 36.)

[122] G. Vaquette, A. Orcesi, L. Lucat, and C. Achard, "The daily home life activity dataset: a high semantic activity dataset for online recognition," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pp. 497–504, IEEE, 2017. (Cited on page 34.)

[123] J. Sung, a. B. S. Colin Ponce, and A. Saxena, "Human activity detection from rgbd images," in *AAAI workshop*, 2011. (Cited on pages 34 and 37.)

[124] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu, "Cross-view action modeling, learning, and recognition," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2649–2656, June 2014. (Cited on pages 34, 37, 40 and 145.)

[125] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "Ntu rgb+d 120: A large-scale benchmark for 3d human activity understanding," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. (Cited on pages 34, 36, 37, 39 and 140.)

[126] Z. Zhang, "Microsoft kinect sensor and its effect," in *IEEE MultiMedia*, vol. 19, April 2012. (Cited on page 36.)

[127] B. Ni, G. Wang, and P. Moulin, "Rgbd-hudaact: A color-depth video database for human daily activity recognition," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, Nov 2011. (Cited on page 37.)

[128] Z. Cheng, L. Qin, Y. Ye, Q. Huang, and Q. Tian, "Human daily action analysis with multi-view and color-depth data," in *European Conference on Computer Vision(ECCV)*, 2012. (Cited on page 37.)

[129] S. M. Amiri, M. T. Pourazad, P. Nasiopoulos, and V. C. Leung, "Non-intrusive human activity monitoring in a smart home environment," in *2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom)*, 2013. (Cited on page 37.)

[130] L. Wang, Y. Qiao, and X. Tang, "Action recognition and detection by combining motion and appearance features," in *THUMOS*, 2014. (Cited on page 37.)

[131] H. Rahmani, A. Mahmood, D. Huynh, and A. Mian, "Histogram of oriented principal components for cross-view action recognition," in *TPAMI*, 2016. (Cited on page 37.)

[132] S. Das, R. Dai, M. Koperski, L. Minciullo, L. Garattoni, F. Bremond, and G. Francesca, "Toyota smarthome: Real-world activities of daily living," in *ICCV*, 2019. (Cited on pages 37, 39, 67, 81, 82, 83 and 141.)

[133] M. Koperski, P. Bilinski, and F. Bremond, "3D Trajectories for Action Recognition," in *ICIP*, 2014. (Cited on pages 37 and 38.)

[134] G. Rogez, P. Weinzaepfel, and C. Schmid, "LCR-Net++: Multi-person 2D and 3D Pose Detection in Natural Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. (Cited on pages 39, 84, 86, 126, 141 and 150.)

[135] P. Hu and D. Ramanan, "Finding tiny faces," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. (Cited on page 39.)

[136] S. Das, M. Thonnat, kaustubh Sakhalkar, M. Koperski, F. Brémond, and G. Francesca, "A new hybrid architecture for human activity recognition from rgb-d videos," *MultiMedia Modeling. MMM 2019*, 2019. (Cited on page 49.)

[137] M. Koperski and F. Bremond, "Modeling spatial layout of features for real world scenario rgb-d action recognition," in *AVSS*, 2016. (Cited on pages 55, 134, 135 and 136.)

[138] F. Chollet *et al.*, "Keras," 2015. (Cited on page 60.)

[139] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org. (Cited on page 60.)

[140] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Jan. 2014. (Cited on pages 62 and 91.)

[141] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, (Cambridge, MA, USA), pp. 3104–3112, MIT Press, 2014. (Cited on page 62.)

[142] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. (Cited on page 62.)

[143] S. Das, A. Chaudhary, F. Bremond, and M. Thonnat, "Where to focus on for human action recognition?," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 71–80, Jan 2019. (Cited on pages 67, 81, 82, 83 and 139.)

[144] S. Das, S. Sharma, R. Dai, F. Bremond, and M. Thonnat, "Vpn: Learning video-pose embedding for activities of daily living," 2020. (Cited on page 67.)

[145] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5534–5542, IEEE, 2017. (Cited on page 67.)

[146] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," *CoRR*, vol. abs/1812.02707, 2018. (Cited on page 68.)

[147] W. Wang, D. Tran, and M. Feiszli, "What makes training multi-modal networks hard?," *CoRR*, vol. abs/1905.12681, 2019. (Cited on page 68.)

[148] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang, "Jointly learning heterogeneous features for rgb-d activity recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 2186–2200, Nov 2017. (Cited on pages 70, 137 and 145.)

[149] F. Faugeras and L. Naccache, "Dissociating temporal attention from spatial attention and motor response preparation: A high-density eeg study," *NeuroImage*, vol. 124, pp. 947 – 957, 2016. (Cited on page 78.)

[150] A. Miech, I. Laptev, and J. Sivic, "Learning a text-video embedding from incomplete and heterogeneous data," *CoRR*, vol. abs/1804.02516, 2018. (Cited on page 88.)

[151] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, "Jointly modeling embedding and translation to bridge video and language," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. (Cited on page 88.)

[152] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. (Cited on pages 91 and 126.)

[153] S. Kim, M. Seltzer, J. Li, and R. Zhao, "Improved training for online end-to-end speech recognition systems," in *Proc. Interspeech 2018*, pp. 2913–2917, 2018. (Cited on page 103.)

[154] Y. Zhang and H. Lu, "Deep cross-modal projection learning for image-text matching," in *ECCV*, 2018. (Cited on page 103.)

[155] L. Wang, Y. Li, and S. Lazebnik, "Learning deep structure-preserving image-text embeddings," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5005–5013, June 2016. (Cited on page 103.)

[156] Y. Liu, Y. Guo, E. M. Bakker, and M. S. Lew, "Learning a recurrent residual fusion network for multimodal matching," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4127–4136, Oct 2017. (Cited on page 103.)

[157] W. J. Krzanowski, *Principles of Multivariate Analysis: A User's Perspective*. USA: Oxford University Press, Inc., 1988. (Cited on page 106.)

[158] L. Yao, A. Torabi, K. Cho, N. Ballas, C. J. Pal, H. Larochelle, and A. C. Courville, "Describing videos by exploiting temporal structure," *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4507–4515, 2015. (Cited on page 116.)

[159] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, vol. 68, pp. 346–362, 2017. (Cited on pages 117, 137, 140 and 145.)

[160] S. Das, M. Thonnat, and F. Bremond, "Looking deeper into time for activities of daily living recognition," in *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 487–496, 2020. (Cited on page 117.)

[161] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1510–1517, 2018. (Cited on page 125.)

[162] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012. (Cited on page 126.)

[163] Y. Zhu, W. Chen, and G. Guo, "Evaluating spatiotemporal interest point features for depth-based action recognition," *Image and Vision Computing*, vol. 32, no. 8, pp. 453 – 464, 2014. (Cited on pages 134 and 136.)

[164] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *Int. J. Rob. Res.*, vol. 32, pp. 951–970, July 2013. (Cited on pages 134 and 135.)

[165] O. Oreifej and Z. Liu, "Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences," in *CVPR*, 2013. (Cited on pages 134 and 136.)

[166] H. Jian-Fang, Z. Wei-Shi, L. Jianhuang, and Z. JianGuo, "Jointly learning heterogeneous features for RGB-D activity recognition," in *CVPR*, 2015. (Cited on pages 134, 136, 137 and 140.)

[167] L. Lin, K. Wang, W. Zuo, M. Wang, J. Luo, and L. Zhang, "A deep structured model with radius–margin bound for 3d human activity recognition," *International Journal of Computer Vision*, vol. 118, pp. 256–273, Jun 2016. (Cited on pages 134 and 135.)

[168] L. Rybok, B. Schauerte, Z. Al-Halah, and R. Stiefelhagen, "Important stuff, everywhere! activity recognition with salient proto-objects as context," in *IEEE Winter Conference on Applications of Computer Vision*, pp. 646–651, March 2014. (Cited on page 135.)

[169] L. Wang, Y. Qiao, and X. Tang, "Action Recognition With Trajectory-Pooled Deep-Convolutional Descriptors," in *CVPR*, 2015. (Cited on page 135.)

[170] H. S. Koppula and A. Saxena, "Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation," in *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pp. III–792–III–800, JMLR.org, 2013. (Cited on page 135.)

[171] T. Liu, X. Wang, X. Dai, and J. Luo, "Deep recursive and hierarchical conditional random fields for human action recognition," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1–9, March 2016. (Cited on page 135.)

[172] A. Shahroudy, T. T. Ng, Y. Gong, and G. Wang, "Deep multimodal feature analysis for action recognition in rgb+d videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2017. (Cited on pages 135, 136 and 137.)

[173] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, and P. Pala, "Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses," in *CVPRW*, 2013. (Cited on page 136.)

[174] A. Shahroudy, G. Wang, and T.-T. Ng, "Multi-modal feature fusion for action recognition in rgb-d sequences," in *ISCCSP*, 2014. (Cited on page 136.)

[175] L. Xia and J. Aggarwal, "Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera," in *CVPR*, 2013. (Cited on page 136.)

[176] L. Liu and L. Shao, "Learning discriminative representations from rgb-d video data," in *IJCAI*, 2013. (Cited on page 136.)

[177] X. Yang and Y. Tian, "Super normal vector for activity recognition using depth sequences," in *CVPR*, 2014. (Cited on page 136.)

[178] Y. Kong and Y. Fu, "Bilinear heterogeneous information machine for RGB-D action recognition," in *CVPR*, 2015. (Cited on page 136.)

[179] C. Lu, J. Jia, and C. K. Tang, "Range-sample depth feature for action recognition," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 772–779, June 2014. (Cited on page 136.)

[180] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 588–595, June 2014. (Cited on pages 137 and 145.)

[181] G. Evangelidis, G. Singh, and R. Horaud, "Skeletal quads: Human action recognition using joint quadruples," in *2014 22nd International Conference on Pattern Recognition*, pp. 4513–4518, Aug 2014. (Cited on page 137.)

[182] I. Lee, D. Kim, S. Kang, and S. Lee, "Ensemble deep learning for skeleton-based action recognition using temporal sliding lstm networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017. (Cited on pages 137 and 145.)

[183] J. Liu, G. Wang, P. Hu, L. Duan, and A. C. Kot, "Global context-aware attention lstm networks for 3d action recognition," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3671–3680, July 2017. (Cited on pages 137 and 140.)

[184] P. Wang, W. Li, C. Li, and Y. Hou, "Action recognition based on joint trajectory maps with convolutional neural networks," *Knowledge-Based Systems*, vol. 158, pp. 43 – 53, 2018. (Cited on page 137.)

[185] J. Zhu, W. Zou, L. Xu, Y. Hu, Z. Zhu, M. Chang, J. Huang, G. Huang, and D. Du, "Action machine: Rethinking action recognition in trimmed videos," *CoRR*, vol. abs/1812.05770, 2018. (Cited on pages 137, 139 and 145.)

[186] M. Liu and J. Yuan, "Recognizing human actions as the evolution of pose estimation maps," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. (Cited on pages 137, 138 and 140.)

[187] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," 2019. (Cited on pages 137 and 139.)

[188] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, "Skeleton-based action recognition using spatio-temporal lstm network with trust gates," *IEEE Transactions on*

Pattern Analysis and Machine Intelligence, vol. 40, no. 12, pp. 3007–3021, 2018. (Cited on page 140.)

[189] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "Learning clip representations for skeleton-based 3d action recognition," *IEEE Transactions on Image Processing*, vol. 27, pp. 2842–2855, June 2018. (Cited on page 140.)

[190] Z. Liu, H. Zhang, Z. Chen, Z. Wang, and W. Ouyang, "Disentangling and unifying graph convolutions for skeleton-based action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 143–152, 2020. (Cited on page 140.)

[191] M. S. Ryoo, A. Piergiovanni, J. Kangaspunta, and A. Angelova, "Assemblenet++: Assembling modality representations via attention connections," in *ECCV*, 2020. (Cited on pages 141 and 142.)

[192] B. Mahasseni and S. Todorovic, "Regularizing long short term memory with 3d human-skeleton sequences for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3054–3062, 2016. (Cited on page 142.)

[193] H. Rahmani and A. Mian., "3d action recognition from novel viewpoints," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1506–1515, June 2016. (Cited on pages 144 and 145.)

[194] R. Li and T. Zickler, "Discriminative virtual views for cross-view action recognition," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2855–2862, June 2012. (Cited on page 145.)

[195] Z. Zhang, C. Wang, B. Xiao, W. Zhou, S. Liu, and C. Shi, "Cross-view action recognition via a continuous virtual path," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2690–2697, June 2013. (Cited on page 145.)

[196] P. Zhang, C. Lan, W. Zeng, J. Xing, J. Xue, and N. Zheng, "Semantics-guided neural networks for efficient skeleton-based human action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. (Cited on page 145.)

[197] B. Li, O. I. Camps, and M. Sznaier, "Cross-view activity recognition using hankelets," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1362–1369, June 2012. (Cited on page 145.)

[198] A. Gupta, J. Martinez, J. J. Little, and R. J. Woodham, "3d pose from motion for cross-view action recognition via non-linear circulant temporal encoding," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2601–2608, June 2014. (Cited on page 145.)

[199] H. Rahmani and A. Mian, "Learning a non-linear knowledge transfer model for cross-view action recognition," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2458–2466, June 2015. (Cited on page 145.)

[200] D. Pavllo, C. Feichtenhofer, D. Grangier, and M. Auli, "3d human pose estimation in video with temporal convolutions and semi-supervised training," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. (Cited on page 150.)

[201] R. Dai, S. Das, S. Sharma, L. Minciullo, L. Garattoni, F. Bremond, and G. Francesca, "Activities de facto: Real-world untrimmed videos for activity detection," in *Anonymous conference (submitted)*, March 2020. (Cited on page 151.)

[202] A. Piergiovanni, A. Angelova, and M. S. Ryoo, "Evolving losses for unsupervised video representation learning," 2020. (Cited on page 151.)

[203] S. Majhi, S. Das, F. Bremond, R. Dash, and P. K. Sa, "Weakly-supervised joint anomaly detection and classification," in *Anonymous conference (submitted)*, April 2020. (Cited on page 152.)