國立臺灣大學電機資訊學院資訊網路與多媒體研究所
碩士論文
Graduate Institude of Networking and Multimedia
College of Electrical Engineering and Computer Science
National Taiwan University
Master Thesis

融入先驗知識以增強動作識別中跨領域少樣本學習能力

Incorporating Prior Knowledge to Enhance Cross-Domain Few-Shot Learning in Action Recognition

金明毅
Ming-Yi Chin

指導教授：許永眞博士
Advisor: Jane Yung-jen Hsu, Ph.D.

中華民國 112 年 7 月
July, 2023

doi:10.6342/NTU202303375

誌謝

在這次的研究中，我最衷心感謝的是我的指導教授許永眞博士。他在我整個研究路上提供了無私的指導和支持，給予我許多寶貴的建議，使我得以解決眾多迷津。我深感幸運有這樣一位優秀的教授指引我的學術成長。兩年前，他慷慨地收留了我，即使我並非台大大學部畢業，也不屬於資工相關科系，這份信任和機會讓我感激不盡。

同樣地，我要特別感謝我的同學們。和他們一起度過的這兩年研究生涯是充滿歡樂和意義的。彼此間的陪伴和支持使我能夠輕鬆地釋放壓力，一起出遊、一起運動成爲我們最珍貴的回憶。這份群體的溫暖和友情讓我的研究生活更加豐富多彩。

最後，我要由衷地感謝我的父母親。感謝他們的理解和支持，讓我得以追尋我的夢想，繼續留在學校深造並從事研究。他們的無條件信任和持續鼓勵是我前進的動力，我深感榮幸能夠有他們作爲我的支持者。

在此，我向所有幫助過我的人致以最誠摯的謝意。沒有你們的協助和鼓勵，我無法順利進入台大、加入iAgents Lab，並順利畢業。這份論文的完成是你們支持的結晶，我將永遠心存感激。這個過程塑造了我，成就了我，我衷心感謝每一位在我學術旅程中伸出援手的人

doi:10.6342/NTU202303375

Abstract

Action recognition, a critical domain in video understanding, typically requires a substantial amount of training data. To address this, we employ few-shot learning methods. However, these methods, primarily designed for same-domain scenarios, face challenges when applied to real-world, cross-domain situations. In this study, we introduce a novel data representation, 'Trajectory', and a 'Cross-Similarity Attention (CSA) Block', both informed by prior knowledge specific to action recognition and easily integrated into existing few-shot learning methods.

The 'Trajectory' method, a new data representation for skeleton data, leverages spatial information to compensate for the temporal data loss due to video sampling. This approach allows us to achieve comparable results to those obtained with more frames but with fewer frames and less computational resources.

The CSA Block utilizes the unique characteristics of skeleton data for enhanced comparison of spatial and temporal similarities, enabling metric learning to generate better embedding.

We also incorporate visual prompt learning for fine-tuning during the adaptation to new domains. Our method demonstrates robust performance not only on open datasets but also in real-world scenarios, such as the infant action data collected in our lab's AIMS project. This underlines its practical applicability and potential in addressing real-world action recognition challenges with limited labeled data.

Keywords: *Action Recognition, Few-Shot learning, Cross Domain, Visual Prompt Learning, Skeleton Data*

doi:10.6342/NTU202303375

摘要

行動識別是視頻理解中的關鍵領域，通常需要大量的訓練數據。爲了解決這個問題，我們採用了少樣本學習方法。然而，這些方法主要設計用於同領域的場景，當應用到現實世界的跨領域情況時，會面臨挑戰。在本研究中，我們引入了一種新的數據表示方式——“軌跡 (Trajectory)”，以及一個“交叉相似性注意力（CSA）塊”，它們都是基於行動識別的先驗知識，並且可以輕鬆地整合到現有的少樣本學習方法中。

“軌跡 (Trajectory)”方法是一種新的骨骼數據表示方式，利用空間信息來彌補由於視頻採樣導致的時間數據損失。這種方法使我們能夠使用更少的幀數和更少的計算資源達到與更多幀數相比的結果。 CSA塊利用骨骼數據的獨特特性來增強空間和時間相似性的比較，從而使度量學習能夠生成更好的嵌入。

我們還將視覺提示學習整合到新領域適應的微調過程中。我們的方法不僅在開放數據集上表現出強大的性能，還在現實世界的場景中，如我們實驗室收集的AIMS項目中的嬰兒行爲數據，也展現了出色的表現。這突顯了它在解決具有有限標籤數據的實際行動識別挑戰的實用應用性和潛力。

關鍵字: 動作識別, 少樣本學習, 跨領域學習, 視覺提示任務, 骨架資料

doi:10.6342/NTU202303375

Contents

doi:10.6342/NTU202303375

doi:10.6342/NTU202303375

doi:10.6342/NTU202303375

List of Tables

doi:10.6342/NTU202303375

doi:10.6342/NTU202303375

List of Figures

doi:10.6342/NTU202303375

Chapter 1

Introduction

This chapter begins by introducing the background and applications of the action recognition task. As deep convolutional neural networks are commonly used for action recognition, they require a large amount of labeled data. To avoid collecting large amounts of data, directly applying previous few-shot learning methods in real-life scenarios can result in many problems. This chapter will uncover these problems and propose solutions to address them.

1.1 Background

In recent years, research on topics related to human behavior has been gaining popularity in the field of computer vision. Applications such as intelligent video surveillance and environmental home monitoring[5] [6], intelligent human–machine interfaces[7], and identity recognition[8] have made great progress. Among these tasks, action recognition is a fundamental task that has attracted the attention of many scholars. The aim of action recognition is to recognize actions performed by individuals based on their movements, the objects they interact with, or even the surrounding scene in the given video. Such techniques have wide-ranging applications. For example, our laboratory utilized action recognition technology in a project to assess whether a child can perform a certain action, in order to detect

doi:10.6342/NTU202303375

developmental delays and facilitate early treatment.

Previous research on action recognition can be mainly categorized into two groups: representation-based solutions and deep network-based solutions. Representation-based solutions classify actions based on holistic or local representations, such as histograms of oriented gradients (HOG)[9], scale-invariant feature transform (SIFT)[10], and spatio-temporal interest points (STIP)[11]. However, these methods often suffer from limited discriminative power and require manual feature engineering. On the other hand, deep network-based[12] solutions have shown great potential in addressing these limitations by learning features automatically from the input data. These approaches typically utilize deep convolutional neural networks (CNNs) to extract high-level spatial and temporal features from videos, which can then be used to classify actions.

1.2 Motivation

Recent advancements in deep learning have enabled the development of more complex and sophisticated models, such as 3D-CNNs[12], two-stream CNNs[13], and attention-based models[14]. These models have achieved state-of-the-art performance on large-scale action recognition datasets, such as Kinetics[15], FineGYM[16], and NTURGB+D120[3]. Nevertheless, these approaches heavily depend on large amounts of labeled data, making it challenging to collect sufficient data for rare or novel actions.

To tackle this challenge, few-shot learning has emerged as a promising research direction for action recognition. Few-shot learning aims to learn from a limited number of labeled examples and generalize to new, unseen examples. However, previous few-shot learning methods[17][18] typically pretrain and test on the same dataset, which is difficult to achieve in real-life scenarios. For example, in our laboratory's

doi:10.6342/NTU202303375

aims project, it is extremely challenging to first collect a large number of labeled infant actions and then test on new infant actions in a same-domain setting. In cross-domain scenarios, previous few-shot learning methods have demonstrated inferior performance compared to baseline approaches.

In addition, action recognition typically involves the use of videos to capture temporal features. This also implies that an equal number of frames must be sampled and fed into the model. Our experiments have shown that when the number of frames is reduced, the impact on performance is not significant in the same-domain scenario. However, in the cross-domain scenario, as the number of frames decreases, the performance drops accordingly, although the issues mentioned above can be addressed by collecting more data, the time required for such data collection may not be feasible in real-world scenarios.

1.3 Proposed Method

To address the performance degradation caused by limited frame sampling in cross-domain few-shot learning, we propose a method that integrates information from multiple frames into a single frame, termed trajectory. This approach leverages spatial space to compensate for missing temporal information while maintaining the original input dimension and without causing additional inference time.

Additionally, we exploit prior knowledge from skeleton data, which reveals that similar actions tend to exhibit similar action patterns at corresponding frames and positions, while videos of different actions display more significant differences in action patterns at corresponding positions. This prior knowledge assists the model in generating more distinctive embeddings.

To evaluate the effectiveness of the proposed method, we conducted experiments on established action recognition datasets including NTURGB+D120[3], FineGym[16], and our laboratory's AIMS project dataset. The experimental results

doi:10.6342/NTU202303375

demonstrate that our proposed approach significantly outperforms conventional few-shot learning methods in terms of recognition accuracy.

1.4 Outline of the thesis

This thesis is organized into six chapters. Chapter 2 reviews related work on action recognition, human pose estimation, and few-shot learning methods. Chapter 3 outlines the research objective and defines the problem. The proposed methodology is introduced in chapter 4, including an algorithm for trajectory extraction from videos and a design for a cross-attention block to generate more distinctive embeddings. Chapter 5 details the experimental setup and presents results that validate the proposed method's effectiveness. Finally, Chapter 6 summarizes the research contributions and concludes the thesis.

doi:10.6342/NTU202303375

Chapter 2

Literature Review

This chapter initially presents various few-shot learning methods, such as Distance Metric Learning Based, Augment Dataset by Prior Knowledge, and those applied in cross-domain scenarios. In Section 2.2, we concentrate on action recognition, introducing three common types of data modalities. Finally, in section 2.3, we discuss several existing methods of few-shot learning in action recognition and highlight the differences between our proposed method and the current methodologies.

2.1 Few shot learning

Machine learning has revolutionized data-intensive applications, but its performance can be hindered by small training datasets. To address this issue, Few-Shot Learning (FSL) has emerged as a promising solution. In this context, we will discuss several FSL methods and cross-domain FSL in the following paragraphs.

2.1.1 Initialization based

Initialization based methods are primarily concerned with learning an effective feature representation or model initialization that can be fine-tuned using a limited

doi:10.6342/NTU202303375

number of labeled examples. One prevalent approach in this category is transfer learning [19], in which a model pre-trained on a large dataset serves as the initial setup before being fine-tuned on the target task within a few-shot learning context. Another approach is meta-learning [20], which focuses on obtaining an initialization that is readily adaptable to new tasks. For instance, MAML [20] learns a model initialization that enables classifier training for novel classes using a minimal number of labeled examples and gradient update steps. This method essentially addresses the few-shot learning challenge by "learning to fine-tune."

2.1.2 Distance Metric Learning Based

Distance metric learning methods aim to learn a similarity metric between examples, such that examples from the same action class have high similarity, while those from different classes have low similarity. One popular method in this category is Siamese networks [21], which learn to compare two examples by passing them through identical neural networks and computing a similarity score. Another approach is prototypical networks [17] further extend this idea by learning a prototype representation for each class and computing distances between the query example and prototypes for classification.

An additional method, known as Learning to Compare: Relation Network (RN) [18], addresses few-shot learning by introducing a relation network. This approach learns to compare a query example with a small support set of labeled examples by incorporating two embedded networks: an embedding network and a relation network. The embedding network is responsible for learning a feature representation of the input examples, while the relation network learns to compare the query example and the support set examples in the embedded space. By computing a relation score for each pair of query and support examples, the model can effectively classify query examples in few-shot learning scenarios. This method contributes to the development of more sophisticated and adaptable distance metric learning techniques for

doi:10.6342/NTU202303375

few-shot classification tasks.

2.1.3 Augment Dataset by Prior Knowledge

Methods in this category aim to leverage prior knowledge or additional information to augment the limited labeled data available for few-shot learning. One approach is to use unsupervised learning techniques such as clustering [22] to learn additional information from the unlabeled data. Another strategy is to incorporate auxiliary tasks, such as pose estimation [23] or object detection [24], which can provide additional information about the scene and help improve action recognition. Some methods also make use of external knowledge sources, such as semantic information [25] or textual descriptions [26], to provide richer context and guide the learning process in few-shot scenarios.

2.1.4 Cross-Domain Few-Shot Learning

Early few-shot learning methods [20] [17] [18] focused on learning-to-learn by training deep networks in a single domain, such as Omniglot [27], miniImageNet [28]. Although they achieved significant progress, these methods struggled in more challenging cross-domain few-shot tasks where test data comes from unknown or previously unseen domains.

Previous research has shown that in cross-domain scenarios, FSL methods [20] [17] [18] generally perform worse than baseline approaches [29] [4]. The baseline methods involve pretraining a model on a large dataset, freezing the feature extraction during testing, and training a new classifier using the support set. Subsequent studies have explored using adapters for fine-tuning in cross-domain few-shot learning tasks [30], These approaches aim to overcome the limitations of earlier FSL methods and demonstrate better adaptability to previously unseen domains.

doi:10.6342/NTU202303375

2.2 Action Recognition

In the field of action recognition, there have been significant advancements over the past few years, with several methods focusing on different data modalities. These primarily include RGB frames, optical flows, and human skeleton data. Each of these modalities has its own advantages and challenges. We discuss each of these in the following sections.

2.2.1 RGB Frames

RGB frames, the standard color images captured by video cameras, serve as the most prevalent data modality in action recognition. These frames contain color information which can encapsulate fine details of human actions and specific objects. Early methods in this area concentrated on the extraction of hand-engineered features from these frames [31].However, with the emergence of deep learning, Convolutional Neural Networks (CNNs) have ascended to the primary method for extracting features from RGB frames [32]. This shift can be attributed to CNN's ability to learn highly discriminative features from the data, surpassing hand-engineered features in performance.

Various architectures have been proposed to leverage the rich information in RGB frames. Two-stream networks [33], for instance, have been devised to separately handle static spatial features and dynamic temporal features in the video data. In a different approach, 3D ConvNets [12] [34] are employed to directly learn spatio-temporal features, integrating both spatial and temporal information in a unified framework.The widespread use of RGB frames can be credited to the simplicity of the method and the omnipresence of RGB cameras in most devices, making it a reliable and accessible choice for action recognition tasks.

doi:10.6342/NTU202303375

2.2.2 Optical Flows

Optical flow is a method to estimate the motion between two consecutive frames. It has been widely used in action recognition to capture the dynamic information in videos. In the two-stream network [13], one of the streams takes optical flow as input to capture the motion information. More recent methods like I3D [12] and R(2+1)D [35] incorporate optical flow information into the 3D convolutions to learn spatio-temporal features more effectively.

2.2.3 Human Skeleton Data

The evolution of pose estimation techniques [36], [37] has made it more feasible to acquire human skeleton data, essentially the coordinates of human joints. This type of data brings an advantage to the table by being less influenced by background or unrelated objects present in the video. It has become an increasingly popular medium for compactly representing human actions, with a focus on the human body's joints and their interconnections, depicted as an arrangement of points or lines.

Methods such as PoseC3D[38], and GCN [39] have been introduced to learn effectively from skeleton data. These methodologies have recorded state-of-the-art results across numerous benchmarks, establishing that skeleton data can be a potent supplement to RGB frames and optical flow in the realm of action recognition.This paper will further explore the usage of human skeleton data in the context of action recognition.

2.2.4 Pose MoTion Representation for Action Recognition[1]

The PoTion [1] representation is a method used to analyze videos by extracting joint heatmaps from each frame and then colorizing them based on their relative time within the video clip. This technique allows for a fixed-dimension representation of

doi:10.6342/NTU202303375

the video clip.

To create the PoTion representation, the system first processes the video and identifies the joints' positions in each frame. These joint positions are then used to generate heatmaps, which highlight the intensity and location of each joint within the frame.

Once the heatmaps are obtained, the system assigns a unique color to each joint based on its relative time in the video clip. This means that the colors of the heatmaps change over time, providing a visual indication of how each joint's movement evolves throughout the clip.

After colorizing the joint heatmaps, the system aggregates them to form a single representation for the entire video clip. This aggregation process ensures that the PoTion representation has a fixed dimension, making it suitable for further analysis and comparison with other video clips.

2.3 few shot learning on action recognition

Few-Shot Learning (FSL) has demonstrated significant progress within image-based tasks, yet its deployment in video-based tasks, including action recognition, has seen less exploration. The unique challenge of integrating FSL into action recognition is the need to grapple with the greater complexity and dimensionality inherent to video data, introducing a level of intricacy exceeding that of two-dimensional image tasks.

Traditional methods primarily focus on metric-based learning, often leveraging 2D CNNs to extract features from RGB frames. These approaches usually rely on identifying distinct objects within these frames to classify actions. For instance, OTAM [40] maintains the frame sequence in video data and measures distances using ordered temporal alignment. Similarly, TRX [41] identifies actions by aligning multiple tuples of differing sub-sequences. These approaches predominantly emphasize

doi:10.6342/NTU202303375

the independent learning of video embeddings. Contrarily, our proposed methodology is anchored in human skeleton data, utilizing a 3D CNN, and centers exclusively on human body movements. This approach more closely mirrors real-world scenarios. For example, within our lab's AIMS project, we assess the standardization of infant movements based exclusively on body posture, with no reliance on any auxiliary objects for observation. This context underscores the applicability and relevance of our method in real-life applications, thereby highlighting its potential value.

doi:10.6342/NTU202303375

Chapter 3

Problem Definition

In the domain of action recognition, the task is to identify and categorize the actions performed in a given video sequence. However, this task becomes significantly more challenging when applied within the context of few-shot learning, particularly when the number of input frames per video is constrained.

The problem can be formally defined as follows:

Given a support set S of size k, which consists of k labeled videos, each representing a unique action class, and a query set Q, which contains unlabeled videos, the goal is to correctly classify the actions in the videos of the query set Q. Each video, both in the support set S and the query set Q, is limited to a predefined number of frames.

This problem poses unique challenges:

- **Limited data:** In a few-shot learning scenario, only a small number of examples are available for each action class during training. This limited data availability makes it challenging to learn a robust model that generalizes well to unseen actions.

doi:10.6342/NTU202303375

- **Limited input frames:** When the number of input frames per video is restricted, the model has fewer opportunities to observe and understand the temporal dynamics of each action, which is crucial for accurate action recognition.

- **High variability:** Actions can be performed by different individuals, in different environments, and with varying speeds and styles, leading to high intra-class variability.

- **Complex temporal structures:** Unlike static images, videos encompass temporal information, adding an additional level of complexity to the learning task.

Our objective is to develop a few-shot learning model and a new data representation that can effectively address these challenges, and accurately recognize actions in videos despite the constraints on the number of input frames and the scarcity of training examples.

doi:10.6342/NTU202303375

Chapter 4

Methodology

In this chapter, we first present the pilot study results on few-shot action recognition using existing few-shot learning methods, identifying the limitations and challenges these methods face, particularly their performance degradation in cross-domain scenarios. We then introduce a novel data representation technique, "Trajectory," devised to leverage unique data properties to address the issues of cross-domain performance and limited frame sampling. Additionally, we develop a new few-shot learning model designed to address these identified challenges effectively. This approach is particularly relevant as conventional meta-learning methods often struggle to adapt to novel classes that substantially deviate from their training data. Our proposed model offers an innovative solution to these problems, thereby enhancing the performance and adaptability of few-shot learning in action recognition tasks.

doi:10.6342/NTU202303375

4.1 Pilot Study: Exploring the Shortcomings of Existing Few-Shot Learning Techniques in Action Recognition

Few-shot learning has emerged as a promising research avenue in the field of action recognition, aiming to learn from a limited number of labeled examples and generalize to new, unseen instances. Nevertheless, existing few-shot learning methods[17, 18] typically pretrain and test on the same dataset, which is often impractical in real-world scenarios. For instance, in our laboratory's aims project, the task of collecting a large number of labeled infant actions and subsequently testing on new infant actions within the same domain proves to be particularly challenging. Furthermore, these methods have demonstrated inferior performance in cross-domain scenarios compared to baseline approaches[4], as illustrated in Table4.1 and Figure4.1.

Action recognition generally involves the use of videos to capture temporal features, implying that an equal number of frames must be sampled and fed into the model. As evidenced by our experiments, when the number of frames is reduced (see Table 4.2), the impact on performance is not significant in the same-domain scenario. However, in cross-domain scenarios, performance deteriorates as the number of frames decreases. While the aforementioned challenges could theoretically be addressed by collecting more data, the time required for such data collection often proves impractical in real-world scenarios.

Method	NTURGB+D120[3]	NTU→ FineGYM[16]	NTU → AIMS
Baseline	83.22%	68.18%	57.74%
Relation network(Meta-Learning) [18]	85.36%	61.20%	46.20%

Table 4.1: 5-shot accuracy in different scenarios with a C3D backbone[2].

This pilot study underscores the shortcomings of current few-shot learning

doi:10.6342/NTU202303375

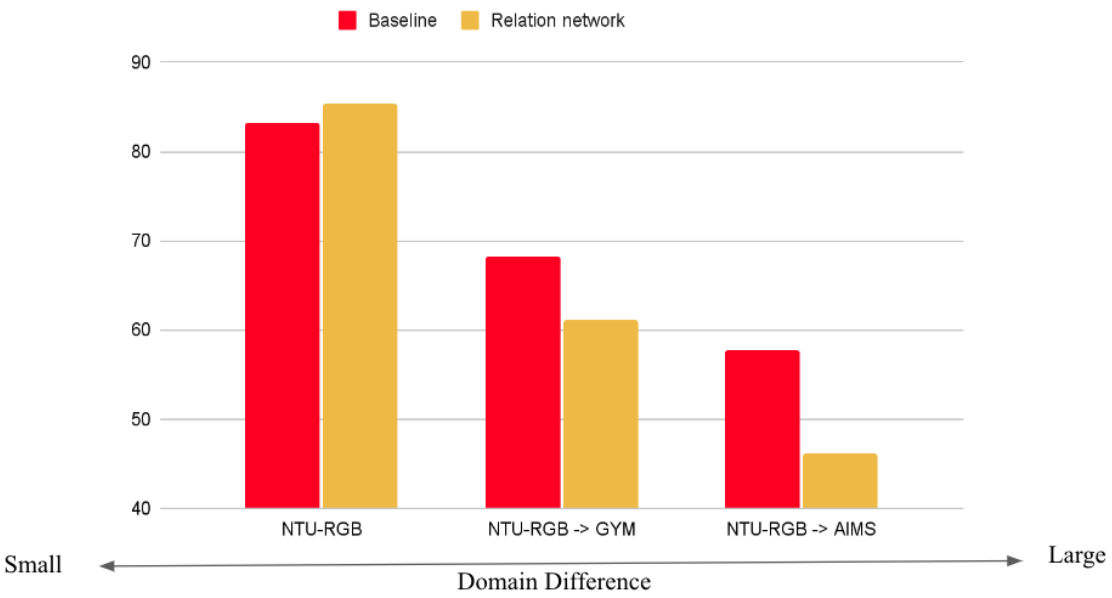


Figure 4.1: 5-shot accuracy in different scenarios with a C3D backbone[2]. The Baseline[4] model performs relatively well with larger domain differences.

Test dataset	12 frames	24 frames	48 frames
NTU-RGB+D120(Same domain)	83.22%	83.45%	84.77%
FineGYM(Cross domain)	68.18%	71.33%	72.68%
AIMS(Cross domain)	57.74%	58.38%	62.35%

Table 4.2: 5-shot accuracy in different frame sampling with baseline method, pretraining on NTURGB+D120(first 80 classes)[3] and testing on the different dataset (e.g., NTURGB+D120, FineGYM), the performance tends to decrease due to the limited number of sampled frames in cross-domain scenarios.

techniques in the realm of action recognition, particularly their inability to perform well in cross-domain scenarios and the dependency on equal frame sampling from videos. Further investigations are required to develop methods that are more suited for real-world applications, overcoming the challenges of data collection, and ensuring robust performance across different domains and varying number of frames.

doi:10.6342/NTU202303375

4.2 The Trajectory - A Novel Data Representation for few shot learning in Action Recognition

Temporal information is paramount in the field of action recognition, significantly shaping the understanding and interpretation of dynamic actions. With our innovative methodology, we introduce 'Trajectory', a data representation specifically designed to encapsulate and relay these temporal dynamics, particularly focusing on joint movements across time. In essence, a 'Trajectory' is a representation of the path traced by each joint within a specified interval. Preparing a 'Trajectory' for model input involves three crucial steps, detailed in the subsequent paragraphs and illustrated in Figure 4.2.

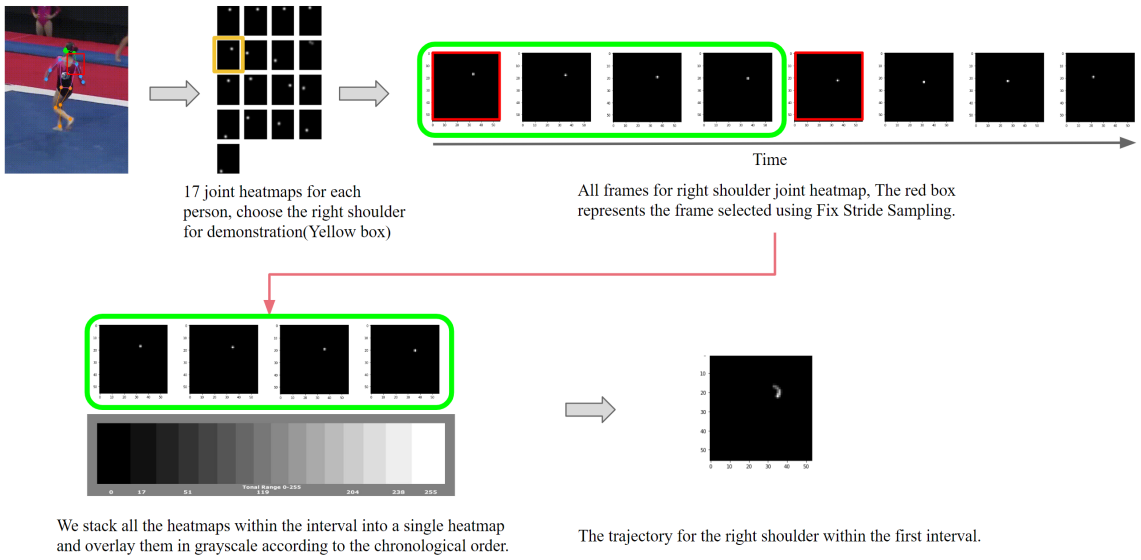


Figure 4.2: The pipeline of the trajectory generation process.

doi:10.6342/NTU202303375

4.2.1 Pose Extraction

The primary objective of this phase is to transfigure RGB video data into joint heatmaps. The process initiates with object detection [42] [43] to pinpoint each individual's bounding box in every frame, represented as (x_1, y_1, x_2, y_2). In this notation, x_1 and y_1 stand for the vertical and horizontal coordinates of the bounding box's top-left point, while x_2 and y_2 indicate the bottom-right point coordinates of the bounding box.

Subsequently, the extraction of the pose within the identified bounding box takes place, aiming to locate human joints' positions. For this, we employ 2D Top-Down pose estimators [23]. The estimated heatmaps are commonly preserved as coordinate-triplets (x, y, c), where c symbolizes the peak score of the heatmap and (x, y) is the corresponding coordinate set of c. By adopting this approach, the coordinate-triplets (x, y, c) help substantially reduce storage requirements while ensuring minimal compromise on performance.

4.2.2 Pose to 3D Heatmap

Following the extraction of 2D poses from the video frames, our next step is to transmute these poses into a three-dimensional heatmap volume. We define a 2D pose in the context of a heatmap with the dimensions of $K \times H \times W$, where K denotes the total number of joints, while H and W are reflective of the frame's height and width correspondingly.

The heatmap output from the Top-Down pose estimator serves conveniently as the target heatmap. Yet, to align this heatmap with the original frame in relation to its bounding box, it's necessary to implement zero-padding. Given that we only have the triple-coordinate representation (x_k, y_k, c_k) of the skeleton joints, we can derive a joint heatmap, J, by amalgamating K Gaussian maps, each centered around a

doi:10.6342/NTU202303375

particular joint.

The final process involves creating a 3D heatmap volume by aligning all the heatmaps along the temporal axis, resulting in a volume size of K × T × H × W.

To improve computational efficiency in our methodology, we employ a technique referred to as Centered Cropping. Maintaining a heatmap of the complete frame size can be computationally expensive, particularly when the subjects of interest inhabit only a minor fraction of the frame. To address this, we discern the smallest bounding box encapsulating all 2D poses throughout the frames. Consequently, all frames are cropped in accordance with this bounding box and resized to a predetermined target size. This approach enables us to decrease the spatial dimensions of the 3D heatmap volume while retaining all 2D poses and their associated movements.

4.2.3 3D Heatmap to Trajectory

The dimensionality of the 3D heatmap volume can also be reduced along the temporal axis by selecting a subset of frames. To accomplish this, we divide the video into N segments of equal length. Suppose each segment contains T frames. We then partition the grayscale range from 0 to 255 into T equal segments. Subsequently, each frame within the segment is multiplied by its corresponding grayscale value, in chronological order, and these are stacked to create a single frame. This method not only allows us to decrease the dimensionality of the heatmap volume, but it also preserves the temporal information, preventing the loss of valuable data that could occur due to sampling.

doi:10.6342/NTU202303375

4.3 CROSS Similarity Attention Block

Our proposed network architecture is built upon the relation network [18] with Convolutional C3D [2] as its backbone. Our unique innovation lies in the incorporation of the Cross Similarity Attention Block, strategically positioned after the second convolutional block, as depicted in Figure 4.3. This placement decision is rooted in our observation that structural patterns are better preserved towards the front of the network, while the latter part corresponds to a dimension of a highly nonlinear feature space. This makes the comparison of unseen classes more challenging.

We further leverage our architecture to extract patterns from skeleton data. As a result of preprocessing steps such as object detection and centered cropping, the human subject consistently occupies the center of the frame. This results in the same action across different videos presenting similar action patterns at corresponding frames and positions. On the contrary, distinct actions usually exhibit noticeable disparities in their patterns at the same locations, as illustrated in Figures 4.5 and 4.6.

Based on our observations, we developed the Cross Similarity Attention Block, as shown in 4.4. This essential component is specifically designed to compare two videos by taking into account their temporal and spatial proximity. To accomplish this goal, the Cross Similarity Attention Block consists of three crucial steps:

1. Spatio-temporal Cross Similarity.
2. Feature Extraction.
3. Feature Aggregation.

These steps are responsible for identifying and enhancing the spatio-temporal similarities present in two videos being compared. By systematically processing the input videos through each step, our Cross Similarity Attention Block can efficiently distinguish and analyze the unique patterns of different actions, ultimately

doi:10.6342/NTU202303375

improving the overall performance of our network architecture. Details on the implementation and functioning of these steps are provided in the subsequent sections.

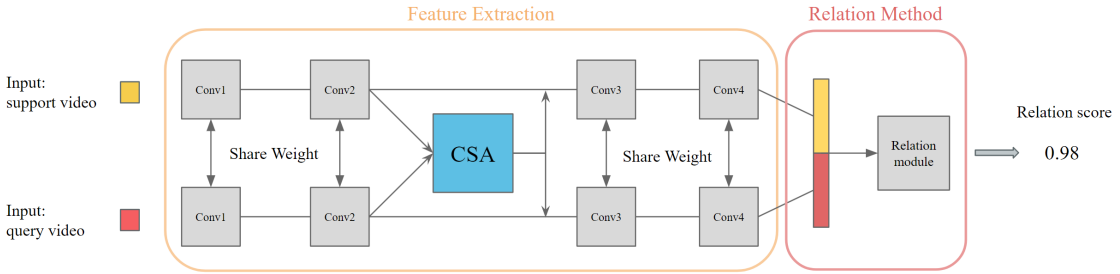


Figure 4.3: **Overall architecture of our model.** Integrating our novel Cross-Similarity Attention Block into a relation network with a C3D backbone, placed strategically after the second convolutional block.

4.3.1 Spatio-temporal Cross Similarity

Given two separate sequences of frames that constitute two videos, we denote the feature maps of these videos, each with T frames, as Q and S, which are both within $\mathbb{R}^{T\times X\times Y\times C}$. Applying the Cross Spatio-temporal similarity transformation to Q and S results in a 6D tensor C, which belongs to $\mathbb{R}^{T\times X\times Y\times L\times U\times V}$. The constituents of this tensor are defined as follows.

$$C_{t,x,y,l,u,v} = sim(Q_{t,x,y}, S_{t+l,x+u,y+v}) \tag{4.1}$$

where $sim(,)$ represents a similarity metric such as cosine similarity. Here, (t, x, y) signifies a query coordinate, while (l, u, v) represents a spatio-temporal offset from the query. To prioritize locality, both spatial and temporal offsets are restricted to their respective neighborhoods.

For spatial dimensions, the offset (u, v) is confined to the immediate vicinity: $(u, v) \in [dU, dU] \times [dV, dV]$. Consequently, $U = 2dU + 1$ and $V = 2dV + 1$ are defined. This spatial neighborhood constraint ensures that only nearby regions of the frame are

doi:10.6342/NTU202303375

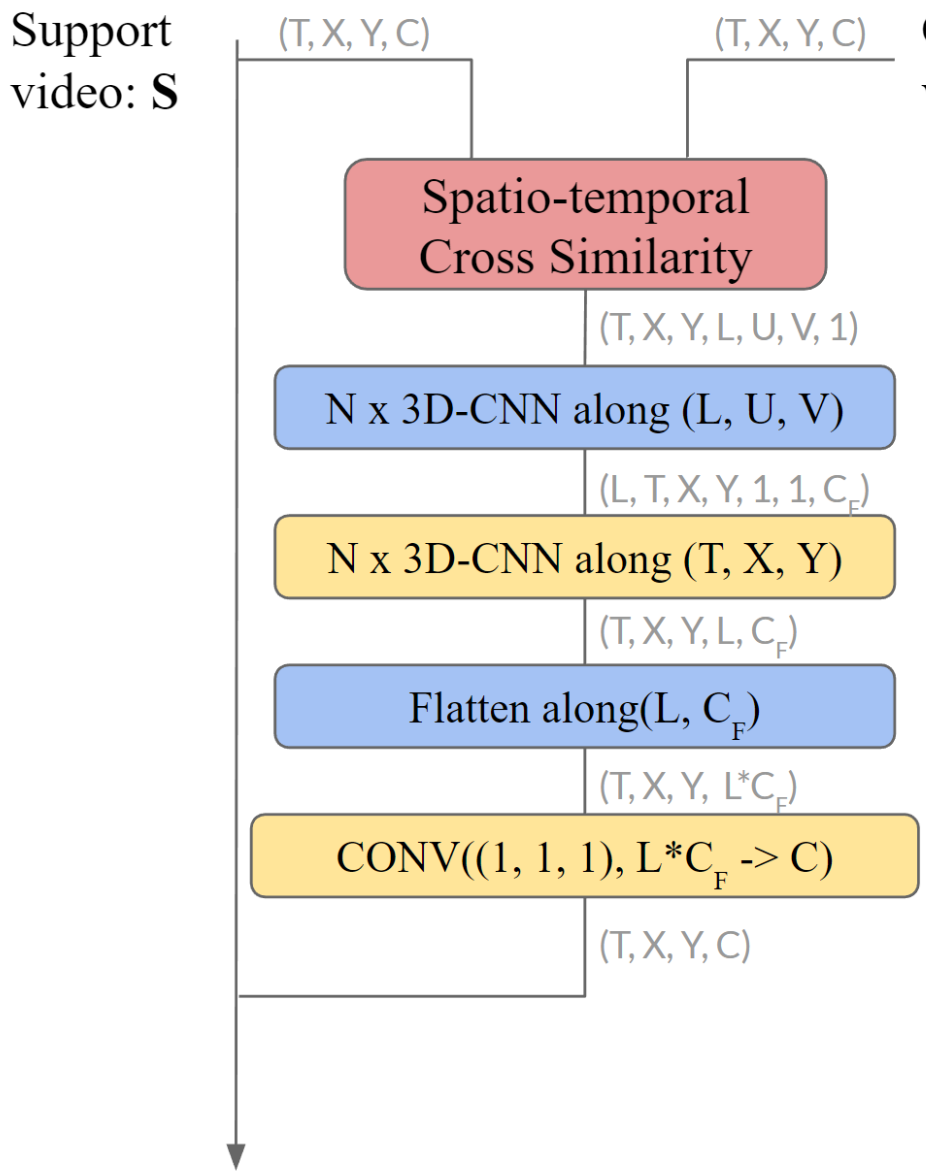


Figure 4.4: Detailed internal architecture and dimensional variations of Input Support Video and Query Video in the Cross Similarity Attention block.

compared, facilitating focus on local spatial patterns.

Temporally, the offset l is limited to the temporal vicinity: $l \in [dL, dL]$, thereby defining $L = 2dL + 1$. This temporal neighborhood constraint ensures only frames within a specific temporal proximity are considered for comparison, aiding in the

doi:10.6342/NTU202303375

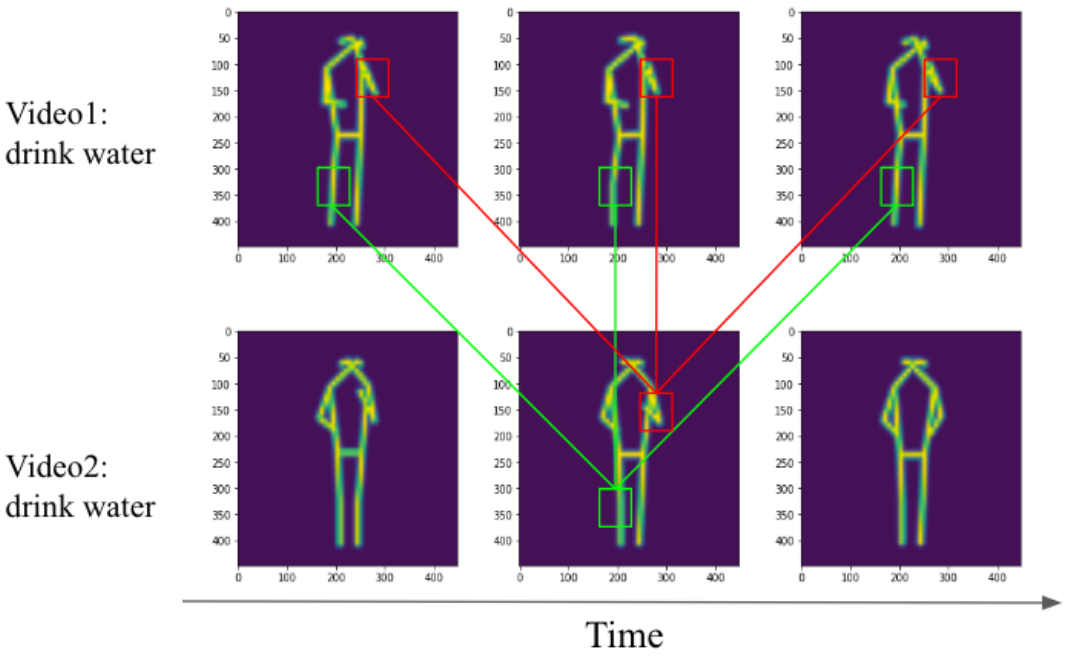


Figure 4.5: The same action in different videos tends to have similar action patterns at corresponding close frames and corresponding spatial positions.

highlighting of local temporal patterns. This approach effectively captures spatio-temporal similarities while preserving the local context.

4.3.2 Feature Extraction

For the 6D tensor C in $\mathbb{R}^{T\times X\times Y\times L\times U\times V}$, we use a 3D Convolutional Neural Network (CNN) to extract C_F-dimensional features for every spatio-temporal position. This process produces a C_F-dimensional feature at each spatio-temporal coordinate (t, x, y) and for each temporal offset l, resulting in a new tensor F within $\mathbb{R}^{T\times X\times Y\times L\times C_F}$. Crucially, this tensor maintains translational equivariance in space, time, and temporal offset.

doi:10.6342/NTU202303375

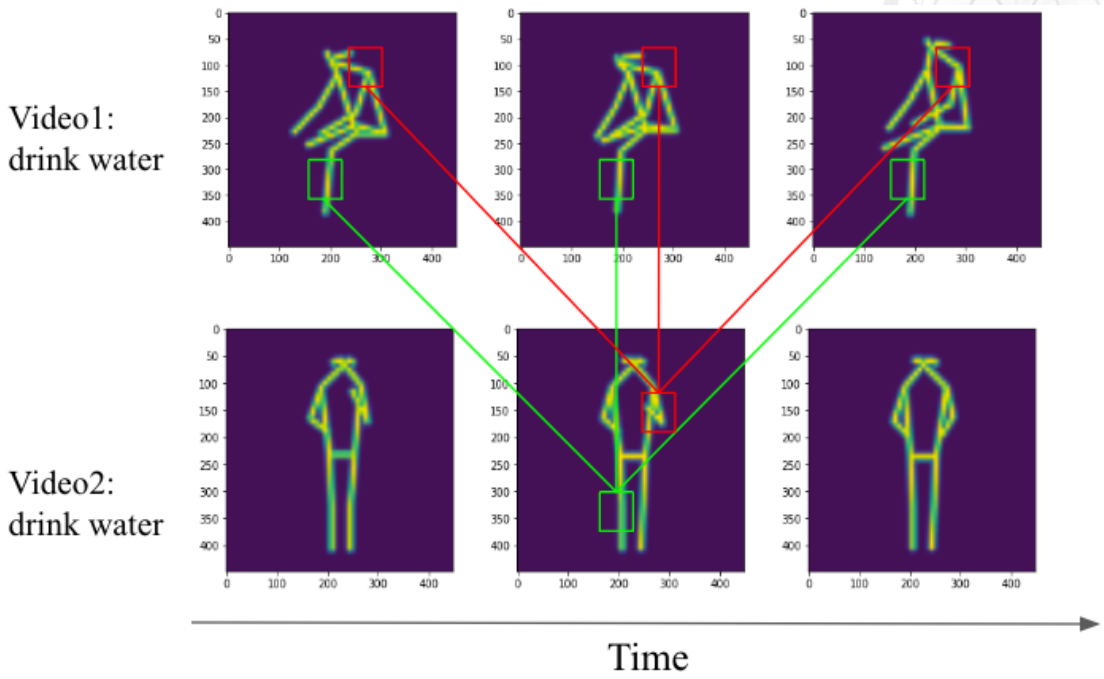


Figure 4.6: Videos of different actions tend to have more significant differences in action patterns at corresponding positions.

4.3.3 Feature Aggregation

At this stage, we consolidate the extracted spatio-temporal similarity features, represented as F in $\mathbb{R}^{T\times X\times Y\times L\times C_F}$, aiming to reincorporate them into the original input stream with dimensions (T, X, Y, C). We initiate this process by applying spatio-temporal convolution kernels along the (t, x, y) dimension of F. The convolution layer, denoted as $h()$, can be formulated as:

$$h(F) = ReLU(Conv(F, K_i)), \tag{4.2}$$

where K_i is a multi-channel convolution kernel that belongs to $R^{T_k\times X_k\times Y_k\times 1\times C_F\times C_B}$. Subsequently, we reshape the volume of dimensions (L, C_B) into LC_B-dimensional vectors, resulting in a new tensor, F, which belongs to $\mathbb{R}^{T\times X\times Y\times LC_B}$. We then apply a $1\times 1\times 1$ convolution layer to produce the final output. This convolution layer not only integrates features from various temporal offsets but also adjusts its channel

doi:10.6342/NTU202303375

dimension to match that of the original inputs, Q or S.

4.3.4 Objective function

Our proposed model architecture is depicted in Figure 4.3. It consists of two primary components: a feature extraction module, denoted as f_ϕ, and a relation module, denoted as g_ϕ. To optimize our model, we train it by minimizing a combination of two losses: the anchor-based classification loss $\mathcal{L}_{anchor}$ and the metric-based classification loss $\mathcal{L}_{metric}$.

The feature extraction module f_ϕ is designed to transform the samples x_i from the support set S and x_j from the query set Q into feature maps $f_\phi(x_i)$ and $f_\phi(x_j)$. The resulting transformations enable the network to extract complex patterns from the data, providing a richer and more detailed representation of the samples.

$$C_{ij} = C(f_\phi(x_i), f_\phi(x_j)) \tag{4.3}$$

Here, $C(f_\phi(x_i), f_\phi(x_j))$ represents the chosen operator to combine these feature maps. While we currently use depth-wise concatenation of the feature maps for the operator $C(\cdot, \cdot)$, future work could explore alternative methods to evaluate their impact on the overall network performance.

The combined feature map C_{ij} is then processed by the relation module g_ϕ to generate a scalar relation score $r_{i,j}$, which ranges between 0 and 1. The relation score represents the degree of similarity between the support sample x_i and the query sample x_j.

$$r_{i,j} = g_\phi(C_{ij}) \tag{4.4}$$

The Binary Cross-Entropy with Logits Loss (BCEWithLogitsLoss) is employed as our objective function during the training phase. We aim to regress the predicted

doi:10.6342/NTU202303375

relation score, $r_{i,j}$, to the ground truth, where matched pairs (i.e., $y_i = y_j$) have a similarity score of 1, and mismatched pairs have a similarity score of 0.

$$\mathcal{L}_{metric} = \underset{\phi,\varphi}{\operatorname{argmin}} \sum_{i=1}^{m} \sum_{j=1}^{n} [y_{i,j} \cdot \log(\sigma(r_{i,j})) + (1 - y_{i,j}) \cdot \log(1 - \sigma(r_{i,j}))] \tag{4.5}$$

In the above equation, σ is the sigmoid function, which maps the input relation score into the range between 0 and 1.

In addition to the metric-based classification loss L_{metric}, we also compute an anchor-based classification loss L_{anchor} using a fully connected classification layer added after the feature extraction of the support set $f_\phi(x_i)$. The purpose of this loss is to guide the model to accurately classify a query sample belonging to a class c in C_{train}.

$$\mathcal{L}_{\text{anchor}} = -\log\left(\frac{\exp\left(\mathbf{w}^T c f\phi(x_i) + b_c\right)}{\sum_{c'=1}^{|C_{\text{train}}|} \exp\left(\mathbf{w}^T c' f\phi(x_i) + b_{c'}\right)}\right), \tag{4.6}$$

In the above equation, $\mathbf{w}_1, \ldots, \mathbf{w}_{|C_{\text{train}}|}$ and $b_1, \ldots, b_{|C_{\text{train}}|}$ represent the weights and biases in the fully connected layer. This equation models the standard cross-entropy loss for multi-class classification.

In summary, the training of our model involves the minimization of the combined loss function, $\mathcal{L}$:

$$\mathcal{L} = \mathcal{L}_{\text{metric}} + \lambda \mathcal{L}_{\text{anchor}}$$

Here, λ is a hyperparameter that helps balance the two loss terms. It's important to note that the fully-connected layer used in calculating $\mathcal{L}_{anchor}$ is not included during inference. This approach ensures that the model is accurately classifying

doi:10.6342/NTU202303375

query samples while maintaining efficiency during the inference phase.

4.4 Cross domain adaptation

As the domain difference grows larger, the adaptation based on a few novel class instances becomes more important, While there exist many design choices, we introduce three methods for adaptation in this work.

4.4.1 Fine-tuning the relation module head.

During the Meta-testing phase, we maintain the backbone feature extraction as constant and solely fine-tune the relation module utilizing the support set.

4.4.2 Parameter Efficient Fine-tuning.

However, the task of effectively learning a relation module is laden with challenges due to two primary reasons.

1. Predicting high-dimensional weights becomes a daunting learning problem, particularly when each weight corresponds to a dimension in a highly nonlinear feature space.

2. The scarcity of samples in the support set further compounds the difficulty of this task, leading to potential overfitting due to a high parameter-sample ratio.

To address these challenges, our method incorporates the Cross Similarity Attention Block after the second convolutional block. This strategic placement is primarily grounded on the observation that structural patterns are more effectively preserved

doi:10.6342/NTU202303375

closer to the front of the network, as illustrated in **??**. During the meta-testing phase, our approach to fine-tuning the Cross Similarity Block offers two key advantages:

1. It involves fewer parameters compared to an auxiliary network, thereby reducing the risk of overfitting.

2. It promotes the retention of more structural pattern information, thereby ensuring a more robust learning process when dealing with unseen classes.

4.4.3 Visual Prompt Tuning

To further refine our model, we introduce the Visual Prompt Tuning mechanism. This mechanism is implemented by appending a 'prompt', of the same size as the tensor, after the Cross Similarity Attention Block. The aim of this prompt is to emphasize key patterns and guide the network towards more relevant features. In this setup, the input, denoted as 'h', is transformed according to the following equation:

$$\text{Output} = h + \alpha \cdot \text{prompt} \tag{4.7}$$

Here, the prompt is of the same size as the input 'h'. The scalar α is a learnable parameter that scales the contribution of the prompt. This approach allows the prompt to exert a controlled influence on the final output, thereby facilitating the model in identifying key patterns more effectively. During the meta-testing phase, we undertake a dual fine-tuning process. In addition to fine-tuning the Cross Similarity Block, we also adjust the parameters of the Visual Prompt Tuning mechanism, particularly the scaling factor α, as illustrated in 4.7.

doi:10.6342/NTU202303375

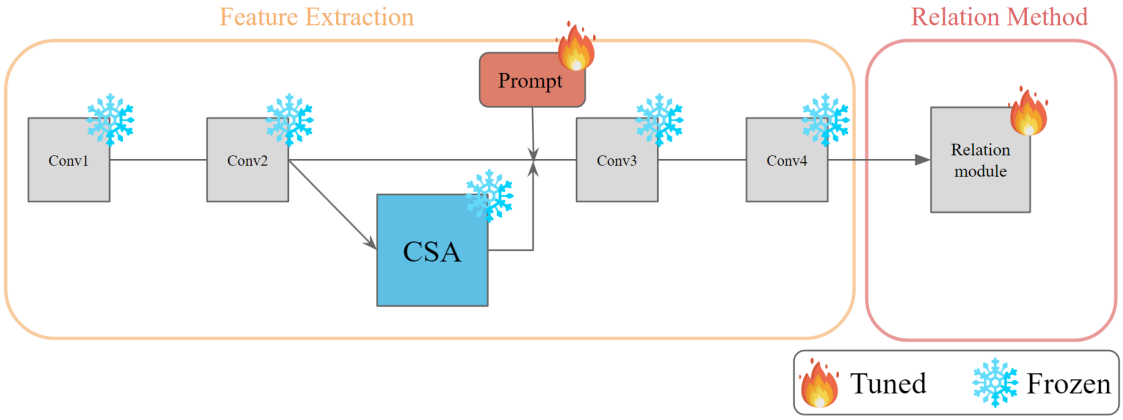


Figure 4.7: Visual Prompt Tuning.

doi:10.6342/NTU202303375

Chapter 5

Experiments

5.1 Experiment Setup

5.1.1 Datasets and scenarios

We evaluate the few-shot classification using three human-centered datasets: NTU-RGB+D120[3], FineGym[16], and our laboratory's project dataset called AIMS, which is specifically designed for detecting developmental delays in infants.

- **NTU-RGB+D120**[3] dataset, extensively used in computer vision research, is particularly renowned for action recognition tasks. This comprehensive dataset incorporates 114,480 Full HD videos that span across 120 distinct action categories, enacted by a variety of subjects. It captures an array of actions ranging from mundane activities such as drinking water, standing up, to waving hands, Figure 5.1 showcases a selection of samples from the NTU-RGB+D120[3] dataset.

  The unique data collection strategy involved using three cameras positioned at different angles to capture videos, resulting in significant variations in the dataset due to differing viewpoints and performers. This added complexity

doi:10.6342/NTU202303375

makes the dataset more challenging and an accurate representation of real-world scenarios.

Apart from the RGB videos, the NTU-RGB+D120 dataset further includes multiple modalities such as 3D skeleton data, depth maps, and infrared (IR) images. Captured using the Microsoft Kinect v2 sensor, these additional modalities contribute to providing a more holistic set of information for action recognition algorithms.

In terms of data allocation, the 120 action classes are divided into training, validation, and testing sets with 65, 15, and 40 classes respectively, ensuring a random and balanced distribution.

- **FineGym**[16] is a unique and valuable action recognition dataset that is built upon gymnasium videos. It stands out among other existing datasets due to its richness, quality, and diversity of content. What sets FineGym apart is its meticulous temporal annotations at both the action and sub-action levels, which follow a three-level semantic hierarchy. This means that each action in the dataset is annotated not only with the overall action label but also with detailed annotations of the elementary sub-actions that compose it. For example, an action like "balance beam" is broken down into sub-actions such as "leap-jumphop", "beam-turns", "flight-salto", "flight-handspring", and "dismount". These sub-actions are further annotated with finely defined class labels. This level of granularity presents significant challenges in action recognition, as it requires algorithms to understand the temporal structures within a coherent action and differentiate between subtly different action classes. With a collection of 29,000 videos and 99 fine-grained gymnastic action classes, FineGym provides an invaluable resource for researchers and practitioners to advance the field of fine-grained action recognition.

- **Alberta Infant Motor Scale (AIMS)** is a standardized evaluation tool

doi:10.6342/NTU202303375

for assessing the gross motor development of infants aged between 0 and 18 months. This assessment comprises 58 movement test items, categorized into four primary positions: prone, supine, sitting, and standing, as depicted in Figure 5.1,5.2. AIMS is commonly employed to identify potential developmental delays in infants.

To facilitate our research, we have developed a unique dataset, termed the AIMS dataset, that encompasses 31 distinct infant movement types. These movements are classified as follows: 9 from the prone position, 3 from the supine position, 7 from the sitting position, and 12 from the standing position, as detailed in the following Table 5.1. Each category of movement comprises 200 samples, thereby totaling 6,200 video clips in the entire dataset.

The recording setup for data collection consisted of five cameras mounted on tripods at a height of 0.6 meters. These cameras were strategically positioned around the mattress, with each one set 1.5 meters apart and at an angle of 45° from the next, to ensure comprehensive coverage of the infant's movements.

5.1.2 Implement detail

The implementation details of meta-training (pretraining) involve the following steps: Pretraining is performed on the first 80 classes of the NTURGB+D 120 dataset. The pretraining process adopts a 5-way 5-shot setting, meaning the model is trained using only a limited number of examples (5 shots) from a few classes (5-way) to encourage better generalization. During the meta-test phase, the model is tested on classes that were completely unseen during the pretraining stage. This evaluation ensures that the model's performance is assessed on new and unfamiliar classes, thus examining its ability to adapt and generalize effectively.

doi:10.6342/NTU202303375

5.1.3 Evaluation Scenarios

Same-Domain Action Recognition

In this scenario, we employ the NTU-RGB+D120 dataset, partitioning the action classes into training (65 classes), validation (15 classes), and testing (40 classes) subsets.

Cross-Domain Action Recognition

We conduct two tests to assess the model's ability to transfer learning:

1. **NTU-RGB+D120 to FineGym**: We evaluate how effectively a model trained on the NTU-RGB+D120 dataset can adapt to the FineGym dataset.

2. **NTU-RGB+D120 to AIMS**: This test measures the model's ability to transfer learning from the NTU-RGB+D120 dataset to the AIMS dataset.

These scenarios collectively enable us to evaluate our model's performance and versatility across different domains, thus assessing its potential for real-world applications.

Movement Types in AIMS Dataset	
Position	**Specific Movements**
Prone	Forearm support (1), Forearm support (2), Extended arm support, Pivoting, Four point kneeling (1), Four point kneeling to sitting or half-sitting, Reciprocal creeping (1), Reciprocal creeping (2), reaching from forearm support
Supine	Supine lying (4), Rolling from supine to prone with rotation, Hands to knees
Sitting	Sitting with propped arm, Sitting with arm support, Sitting without arm support (1), Sitting to four point kneeling, Sitting without arm support(2), Sitting with support, Pull to sit
Standing	Support standing 2, Support standing 3, Pulls to stand, Cruising without rotation, Controlled lowering through standing, Stand alone, Early stepping, Walk alone, Pulls to stand with support, Supported standing with rotation, Standing from modified squat, Squat

Table 5.1: Detailed movement types per position in the AIMS dataset

doi:10.6342/NTU202303375

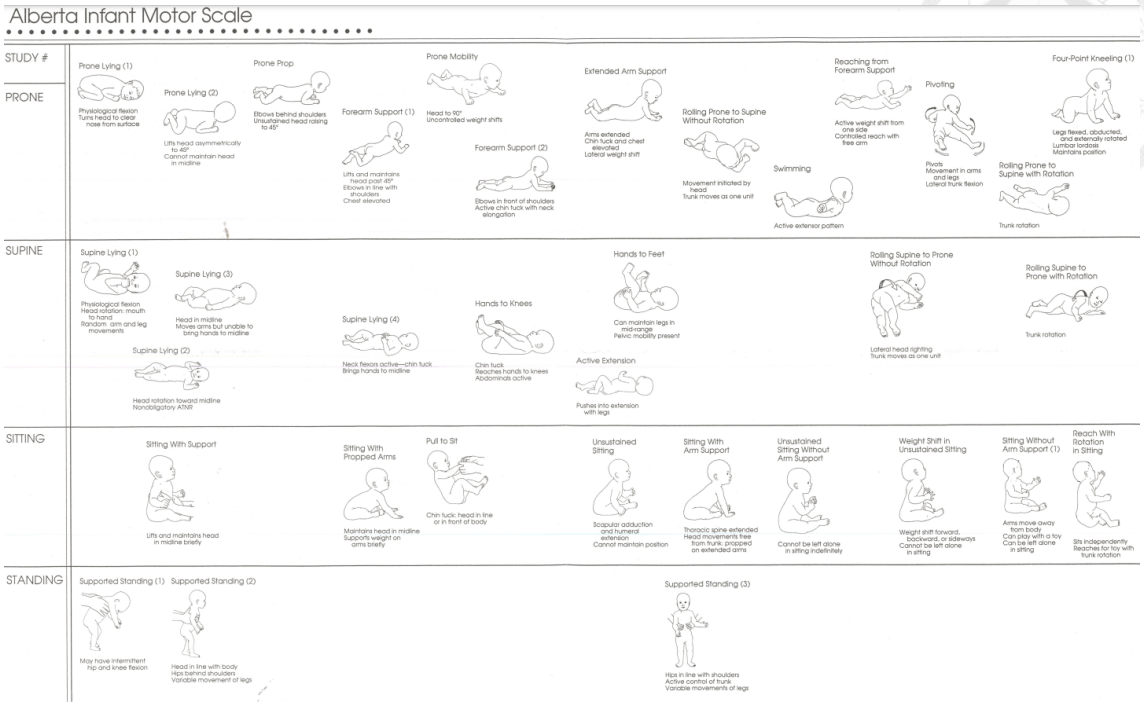


Figure 5.1: Albert Infant Motor Scale.

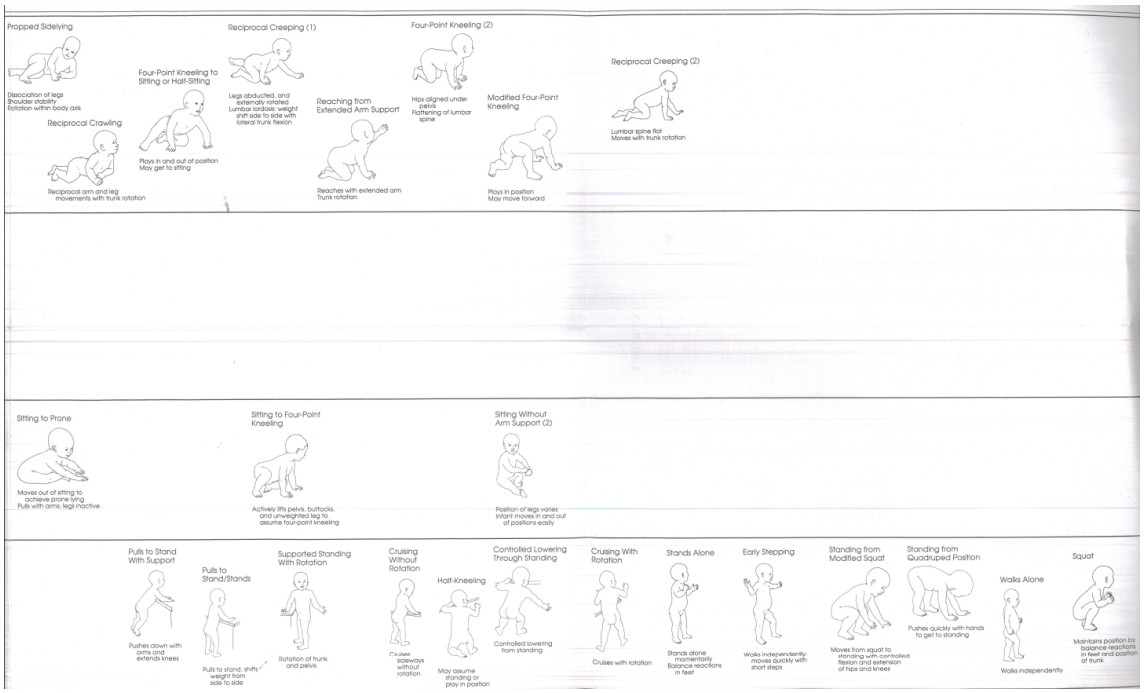


Figure 5.2: Albert Infant Motor Scale.

doi:10.6342/NTU202303375

5.1.4 Data Augmentation

For spatial augmentation, we apply a series of transformations to augment the data. Firstly, we resize the video frames to a resolution of 64x64 pixels. Then, we perform random cropping to obtain a size of 56x56 pixels, selecting a random region of interest within the frame. Additionally, we apply random horizontal flipping to further increase the diversity of the data.

In terms of temporal augmentation, we employ different frame sampling techniques based on the specific experiment settings. We sample 12, 24, or 48 frames from each video, ensuring a varied representation of the temporal information in the dataset.

5.1.5 Competitor Methods

Our approach is compared to rival methods that present significant challenges. **State-of-the-art action recognition methods.** The state-of-the-art model for general action recognition is represented by PoseC3D[38]. To adapt the model to Few- Shot Learning (FSL) using the Baseline method and RelationNet, we choose theC3D backbone, resulting in the model referred to as PoseC3D and RelationNet, respectively.

State-of-the-art few-shot action recognition methods. STRM[44], a state-of-the-art model for few-shot action recognition, adopts ResNet-50 as its backbone, pretrained on ImageNet, to serve as the feature extractor. In this model, an adaptive maxpooling operation is applied to reduce the spatial resolution to $P = 4$. Regarding the TRM, we utilize $\Omega = 2$ for our evaluations, as specified in the research paper.

doi:10.6342/NTU202303375

5.1.6 Evaluation Metrics

In our study, we employ classification accuracy as the primary evaluation metric for the few-shot classification task. This metric quantifies the ratio of samples correctly classified to the total number of samples within the dataset. The computation involves comparing the predicted class label with the actual ground truth label for each individual sample, establishing whether they coincide. The accuracy calculation is made based on the prediction with the highest probability for each sample, signifying the most confident classification.

Additionally, to compare the stability of different methods, we also calculate the standard deviation for each experiment. This statistical measure provides further insight into the consistency of our models, highlighting their ability to produce reliable results across multiple runs. Thus, a method with lower standard deviation is considered more stable and reliable.

5.2 Implementation Details

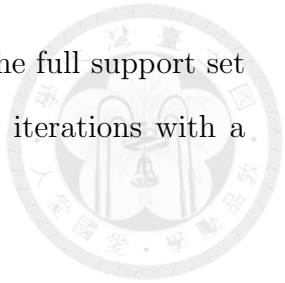
Throughout the development process, we've adhered to a consistent approach for the training phase of both the Baseline and our proposed method, totalling 40 epochs.

5.2.1 Baseline Method

For the Baseline models, we've opted to optimize performance via the Adam optimizer, chosen for its efficient computational properties. Parameters were set to $\beta_1 = 0.9$ and $\beta_2 = 0.999$, following conventional recommendations. We calculated loss using the cross-entropy metric and set the learning rate to 1e-3. After each training episode, the models were evaluated using a validation set. The model that achieved the highest accuracy on this validation set was then selected for further evaluation or deployment.

doi:10.6342/NTU202303375

In the fine-tuning stage of the Baseline method, we utilized the full support set to train a new classifier. This training was conducted over 100 iterations with a batch size of 4.



5.2.2 Proposed Method

For our proposed method, we've made a shift to the Stochastic Gradient Descent (SGD) optimizer, with a momentum of 0.9 and a weight decay of 5e-4. We also set the loss λ to 0.3, which has proven effective in enhancing the model's training performance.

In the fine-tuning stage for our proposed method, we made some modifications compared to the Baseline. We selectively fine-tuned our model over 100 iterations, but with an increased batch size of 10. For prompt turning, due to the greater number of parameters involved, we found that extending the training period to 1000 iterations resulted in optimal performance.

5.2.3 Meta-Testing Phase

The meta-testing phase was comprehensive, involving 600 separate experiments for all methods under consideration. Each experiment involved a randomly chosen subset of 5 classes from the novel classes pool. Within each chosen class, we further selected 5 instances to form the support set and 16 instances to form the query set. This design allowed us to extensively evaluate the methods across a variety of conditions and scenarios.

doi:10.6342/NTU202303375

5.3 Experiment Result

5.3.1 Performance Impact of Novel Data Representation 'Trajectory' in Few-Shot Learning in Action Recognition Tasks

We set out to investigate whether our novel data representation, termed "trajectory", can address the performance degradation associated with a decrease in the number of frames in few-shot learning tasks. Given that the data in such tasks is already limited, the loss of time-series information due to frame sampling reduction can significantly affect the model's performance. For our evaluation, we incorporate two different few-shot learning methods, the baseline method and the relation network method from the realm of metric learning, while consistently using C3D as the backbone.

Our evaluations provide insights into how the trajectory fares in this context. In the case of the same-domain scenario, it can be observed from the Table 5.2 that the decrease in accuracy due to reduced frame sampling is less pronounced when employing the trajectory. However, the increase in accuracy as a result of using the trajectory is minimal, and there is even a slight regression in performance at 48 frames.

In contrast, the trajectory's utility becomes evident in cross-domain scenarios. Specifically, for the NTU-RGB+D120 to FineGym transition, as seen in Table 5.3, the model's performance substantially deteriorates with frame sampling reduction if the trajectory is not utilized. Yet, with the implementation of the trajectory, there is a noticeable improvement in performance. Impressively, the model's performance at 12 and 24 frames approaches the level seen at 48 frames. This result signifies that we can achieve comparable performance with less data, significantly reducing the computation time.

Even in the case of a larger domain difference, such as the NTU-RGB+D120 to

doi:10.6342/NTU202303375

AIMS transition (as illustrated in Table 5.4), the trajectory still contributes to a slight improvement, though it is not as pronounced as in the FineGym scenario.

In conclusion, our novel data representation, the trajectory, demonstrates its effectiveness in mitigating performance decline due to frame sampling reduction. This effectiveness is particularly evident in cross-domain settings, where it not only preserves time-series information within the data but also enhances computational efficiency.efficiency.

	12 frames	24 frames	48 frames
PoseC3D	83.22%	83.45%	84.77%
PoseC3D + trajectory	**84.12% (+0.9%)**	**84.10%(+0.65%)**	**84.65%(-0.12%)**
RelationNet	85.36%	86.21%	86.73%
RelationNet + trajectory	**86.05% (+0.69%)**	**86.33%(+0.12%)**	**86.99%(+0.26%)**

Table 5.2: Performance of NTU-RGB+D 120 in a 5way-5shot scenario.

	12 frames	24 frames	48 frames
PoseC3D	68.18%	71.33%	72.68%
PoseC3D + trajectory	**73.58% (+5.4%)**	**74.22%(+2.89%)**	**73.10%(+0.42%)**
RelationNet	61.20%	64.32%	64.19%
RelationNet + trajectory	**65.41% (+4.21%)**	**65.73%(+1.41%)**	**65.27%(+1.08%)**

Table 5.3: Performance on cross domain of NTU-RGB+D 120 to FineGym in a 5way-5shot scenario.

	12 frames	24 frames	48 frames
PoseC3D	57.74%	58.38%	62.35%
PoseC3D + trajectory	**59.00% (+1.26%)**	**60.99%(+2.61%)**	**63.30%(+0.95%)**
RelationNet	46.20%	47.52%	49.76%
RelationNet + trajectory	**50.65% (+4.45%)**	**50.14%(+2.62%)**	**49.45%(-0.31%)**

Table 5.4: Performance on cross domain of NTU-RGB+D 120 to AIMS in a 5way-5shot scenario.

5.3.2 Performance Evaluation of Cross-Similarity Attention Block

The Cross-similarity Attention (CSA) block is a crucial component of our proposed model, aimed at improving performance in the context of various frame set-

doi:10.6342/NTU202303375

tings. To validate its effectiveness, we compared our model, referred to as "CSA", against two others previous, the baseline and relation network models.

In this evaluation, different configurations were set based on the number of frames. For 12 frames, the CSA model was configured with parameters L=3, U=9, V=9; for 24 frames, the model parameters were set at L=5, U=9, V=9; and for 48 frames, we used L=7, U=9, V=9. Furthermore, during the meta-testing phase, we adopted a fine-tuning approach for the relation module within our model.

Performance Analysis

This analysis was conducted across three distinct datasets: NTU-RGB+D 120, FineGym, and AIMS, thereby testing the effectiveness of the Cross-Similarity Attention (CSA) block under different frame and different domain settings.

- **NTU-RGB+D 120:** In the same domain scenario, as shown in Table 5.5, our CSA model outperformed the baseline by 6.32% at 12 frames and by 7.39% at 24 frames. This performance underscores the effectiveness of the CSA, especially in scenarios with lower frame counts.

- **NTU-RGB+D 120 → FineGym:** Transitioning to a cross-domain scenario with the FineGym dataset, as presented in Table 5.6, the CSA model continued to demonstrate its superiority. It improved the performance by 10.03% at 12 frames and by 8.72% at 24 frames compared to the baseline. This result underlines the CSA's ability to significantly enhance the model's accuracy even in cross-domain settings.

- **NTU-RGB+D 120 → AIMS:** Finally, in a more challenging cross-domain scenario where we transitioned from the NTU-RGB+D 120 dataset to the AIMS dataset (Table 5.7), the CSA model once again demonstrated exceptional performance. It outperformed the baseline by 7.06% at 12 frames and 6.92% at 24 frames. Despite the inherent difficulties in cross-domain scenar-

doi:10.6342/NTU202303375

ios, the CSA model, empowered by the CSA block, delivered a significant performance boost..

In conclusion, our proposed CSA model, equipped with the Cross-Similarity Attention block, consistently outperforms the baseline and relation network models across various domains and frame settings. This demonstrates the CSA block's effectiveness in aiding the model to utilize available data more efficiently, even when it's limited, improving accuracy and performance in diverse scenarios.

Furthermore, it's notable that the performance gains offered by our Auxiliary model are more substantial at lower frame counts. This reveals the CSA block's potential in improving the model's robustness against frame reduction, a prevalent issue in few-shot learning tasks. By mitigating performance degradation due to frame sampling reduction, the CSA block enables a more efficient use of the available data, particularly beneficial in cross-domain settings.

	12 frames	24 frames	48 frames
PoseC3D	83.22% ± 0.75%	83.45% ± 0.83%	84.77% ± 0.78%
RelationNet	85.36% ± 0.79%	86.21% ± 0.80%	86.73% ± 0.77%
CSA	**90.54% ± 0.96%**	**90.84% ± 0.76%**	**92.55% ± 0.90%**

Table 5.5: Performance Comparison in 5-way-5-shot Same-Domain Scenario: NTU-RGB+D 120 Dataset.

	12 frames	24 frames	48 frames
PoseC3D	68.18% ± 0.78%	71.33% ± 0.76%	72.68% ± 0.87%
RelationNet	61.20% ± 0.82%	64.32% ± 0.78%	64.19% ± 0.87%
CSA	**78.21% ± 0.81%**	**80.05% ± 0.79%**	**81.51% ± 0.91%**

Table 5.6: Performance Comparison in 5-way-5-shot Cross-Domain Scenario: NTU-RGB+D 120 to FineGym Dataset.

doi:10.6342/NTU202303375

	12 frames	24 frames	48 frames
PoseC3D	57.74% ± 0.83%	58.38% ± 0.84%	62.35% ± 0.79%
RelationNet	46.20% ± 0.85%	47.52% ± 0.74%	49.76% ± 0.85%
CSA	**64.80% ± 0.79%**	**67.20% ± 0.91%**	**68.11% ± 0.88%**

Table 5.7: Performance Comparison in 5-way-5-shot Cross-Domain Scenario: NTU-RGB+D 120 to AIMS.

5.4 Cross-domain Adaptation: Experimental Setup and Comparison

Adapting to new domains is crucial for deep learning models, particularly in scenarios where there is a substantial shift in data distribution. To assess different adaptation methods' efficacy in addressing these cross-domain challenges, we carried out a comprehensive set of experiments.

We compared three adaptation methods:

- Fine-tuning the relation module
- Parameter-efficient fine-tuning, which can further be divided into three sub-methods:
  - Fine-tuning the entire Cross-Similarity Attention (CSA) Block
  - Fine-tuning only the Feature Aggregation block within the CSA
  - Fine-tuning solely the point-wise convolution within the Feature Aggregation block of the CSA
- Visual prompt tuning

For both fine-tuning the relation module head and parameter-efficient fine-tuning, we used a support set with a batch size of 10 and fine-tuned for 100 epochs. This was done to achieve an optimal balance between performance and efficiency. On the other hand, visual prompt tuning involves learning additional parameters. To

doi:10.6342/NTU202303375

achieve the best results, we allowed for more extensive training, fine-tuning for 1000 epochs.

To achieve this, we conducted experiments on three datasets: the NTU-RGB+D 120 dataset (for the same domain), the FineGym dataset (for cross-domain), and the AIMS dataset (also for cross-domain). We measured the performance of each adaptation method on each dataset.

Our metrics for comparison included the accuracy of the method on both 12 frames and 24 frames data, the number of trainable parameters, and the time taken per iteration. The results of these experiments are reported in the following tables. The accuracy metrics give an insight into how well the method can recognize novel class instances in both same and cross-domain scenarios. Trainable parameters and time per iteration, on the other hand, provide a measure of the computational efficiency of each method. The aim was not only to find the method with the highest accuracy but also the one that provides a balance between computational efficiency and performance.

	12 frames	Trainable parameters	Times
Relation Module	90.54% ± 0.96%	4.20M	15.20s/it
CSA Block	93.33% ± 0.82%	0.87M	22.78s/it
Aggregation Block	92.64% ± 0.82%	0.24M	19.03s/it
Pointwise Conv	93.55% ± 0.98%	0.17M	18.14s/it
Visual Prompt	94.01% ± 0.72%	4.50M	117.29s/it

Table 5.8: Performance on the NTU-RGB+D 120 dataset in the same domain using different adaptation methods in a 5-way-5-shot scenario (12 frames).

Performance Analysis

- **Same Domain scenario(NTU-RGB+D 120):** In examining the performance on the NTU-RGB+D 120 dataset within the same domain (Table 5.8,5.9), all adaptation strategies showcase compelling performances. How-

doi:10.6342/NTU202303375

	24 frames	Trainable parameters	Times
Relation Module	90.84% ± 0.76%	4.20M	27.94s/it
CSA Block	93.65% ± 0.92%	0.95M	39.48s/it
Aggregation Block	93.19% ± 0.86%	0.31M	34.82s/it
Pointwise Conv	92.24% ± 0.89%	0.017M	34.04s/it
Visual Prompt	93.65% ± 0.67%	4.80M	187.21s/it

Table 5.9: Performance on the NTU-RGB+D 120 dataset in the same domain using different adaptation methods in a 5-way-5-shot scenario (24 frames).

	12 frames	Trainable parameters	Times
Relation Module	78.21% ± 0.81%	4.20M	15.21s/it
CSA Block	78.15% ± 0.80%	0.87M	22.86s/it
Aggregation Block	80.64% ± 0.96%	0.24M	19.08s/it
Pointwise Conv	80.67% ± 0.97%	0.17M	18.19s/it
Visual Prompt	84.25% ± 0.69%	4.50M	115.55s/it

Table 5.10: Performance on the NTU-RGB+D 120 to FineGym dataset in the cross domain using different adaptation methods in a 5-way-5-shot scenario (12 frames).

ever, Visual Prompt Tuning achieves the highest accuracy for both 12 and 24 frames at 94.01% and 93.65%, respectively. Despite delivering superior accuracy, Visual Prompt Tuning also requires the most considerable computational resources, reflected in both the highest FLOPS and the longest iteration time.

Conversely, the Pointwise Convolution emerges as a promising candidate, offering competitive accuracy figures (93.55% and 92.24% for 12 and 24 frames, respectively), whilst consuming less computational resources compared to other strategies. Thus, in terms of achieving a balance between performance and computational efficiency within the same domain, the Pointwise Convolution seems to be a preferred approach.

- **Cross Domain scenario(NTU-RGB+D 120 → FineGym):** Analysis with the FineGym dataset (Table 5.10, 5.11), the performance of the adaptation strategies differs. Visual Prompt Tuning maintains its leading position in terms of accuracy on the FineGym dataset. Yet, this superior performance comes at the expense of significantly higher computational demands.

doi:10.6342/NTU202303375

	24 frames	Trainable parameters	Times
Relation Module	80.05% ± 0.79%	4.20M	27.82s/it
CSA Block	77.48% ± 0.71%	0.95M	39.97s/it
Aggregation Block	78.55% ± 0.88%	0.31M	34.88s/it
Pointwise Conv	74.69% ± 0.80%	0.017M	34.82s/it
Visual Prompt	82.17% ± 0.74%	4.80M	177.65s/it

Table 5.11: Performance on the NTU-RGB+D 120 to FineGym dataset in the cross domain using different adaptation methods in a 5-way-5-shot scenario (24 frames).

	12 frames	Trainable parameters	Times
Relation Module	64.80% ± 0.79%	4.20M	15.41s/it
CSA Block	66.81% ± 0.82%	0.87M	22.98s/it
Aggregation Block	67.59% ± 0.99%	0.24M	19.01s/it
Pointwise Conv	63.32% ± 0.94%	0.17M	18.21s/it
Visual Prompt	69.50% ± 0.98%	4.50M	121.74s/it

Table 5.12: Performance on the NTU-RGB+D 120 to AIMS dataset in the cross domain using different adaptation methods in a 5-way-5-shot scenario (12 frames).

Interestingly, in the FineGym context, the Pointwise Convolution maintains competitive accuracy, even exceeding the performance of the Relation Module and the CSA Block. Given its significantly lower computational demands, this performance highlights the potential of the Pointwise Convolution strategy.

- **Cross Domain scenario(NTU-RGB+D 120 → AIMS):** The performance of Pointwise Convolution(Tabel 5.12,5.13), however, dips on the AIMS dataset, suggesting potential limitations in its adaptability to diverse cross-domain scenarios. Conversely, Visual Prompt Tuning retains its lead with the highest accuracy, albeit at a substantial computational cost.

In conclusion, while Visual Prompt Tuning consistently yields the highest accuracy across different scenarios, its substantial computational demands may limit its practicality. Conversely, Pointwise Convolution demonstrates a promising trade-off between accuracy and computational efficiency, especially in cross-domain settings. Future research should focus on improving its adaptability to enhance performance across a broader range of cross-domain scenarios..

doi:10.6342/NTU202303375

	24 frames	Trainable parameters	Times
Relation Module	67.20% ± 0.94%	4.20M	27.88s/it
CSA Block	64.43% ± 0.95%	0.87M	39.81s/it
Aggregation Block	66.67% ± 0.79%	0.31M	35.01s/it
Pointwise Conv	63.07% ± 0.81%	0.017M	33.71s/it
Visual Prompt	69.95% ± 0.95%	4.80M	192.92s/it

Table 5.13: Performance on the NTU-RGB+D 120 to AIMS dataset in the cross domain using different adaptation methods in a 5-way-5-shot scenario (24 frames).

5.5 System-Level Comparison

To validate the efficacy of our few-shot learning system in action recognition, we conducted a series of experiments utilizing our proposed novel data representation method, 'Trajectory,' and the Cross-Similarity Attention (CSA) block. We also incorporated visual prompt fine-tuning during the meta-testing adaptation phase. We contrasted our system's performance with traditional methods, which primarily rely on skeleton data input, as well as with methods that utilize earlier techniques. Our investigation included performance evaluation on 12-frame and 24-frame scenarios and examined variations in performance under same domain and cross-domain conditions.

Table 5.14 presents the system-level performance of different methods on the NTU-RGB+D 120 dataset in a 5-way-5-shot scenario. The results indicate that our method significantly outperforms the baseline and the relation network in both 12-frame and 24-frame settings. Specifically, our method achieves an accuracy of 94.57% in the 12-frame setting, which is an improvement of over 11% from the baseline. In the 24-frame setting, our method yields an accuracy of 93.26%, demonstrating its superior performance.

For cross-domain adaptation, we tested our method on the FineGym dataset, and the results are summarized in Table 5.15. Despite the challenges associated with cross-domain adaptation, our method once again proved to be effective. We achieved an accuracy of 84.61% in the 12-frame scenario and 83.25% in the 24-frame

doi:10.6342/NTU202303375

scenario, significantly outperforming the baseline and the relation network.

Finally, we examined the performance on the AIMS dataset, which is yet another cross-domain scenario. The results are presented in Table 5.16. Our method maintained its effectiveness, achieving an accuracy of 70.15% in the 12-frame scenario and 71.27% in the 24-frame scenario, once again outperforming both the baseline and the relation network.

In summary, the experimental results on both same domain and cross-domain scenarios demonstrate the effectiveness of our proposed few-shot learning system in action recognition. Our system consistently outperforms the baseline and the relation network across different scenarios, demonstrating its robustness and effectiveness. The novel data representation method, 'Trajectory,' the Cross-Similarity Attention (CSA) block, and the visual prompt fine-tuning during meta-testing adaptation collectively contribute to this superior performance. Therefore, our work provides a promising direction for future research in few-shot learning for action recognition.

	12 frames	24 frames
PoseC3D (Skeleton)	83.22 % ± 0.88%	83.45% ± 0.85%
RelationNet (Skeleton)	85.36% ± 0.79%	86.21% ± 0.83%
Ours (Skeleton)	94.57% ± 0.69%	93.26% ± 0.65%
STRM (RGB)	74.48% ± 0.86%	75.73% ± 0.91%

Table 5.14: System-Level performance of NTU-RGB+D 120 in a 5way-5shot scenario.

	12 frames	24 frames
PoseC3D (Skeleton)	68.18% ± 0.78%	71.33% ± 0.76%
RelationNet (Skeleton)	61.20% ± 0.86%	64.32% ± 0.82%
Ours (Skeleton)	84.61% ± 0.88%	83.25% ± 0.77%
STRM (RGB)	59.71% ± 0.98%	61.35% ± 0.85%

Table 5.15: System-Level Performance Comparison in 5-way-5-shot Cross-Domain Scenario: NTU-RGB+D 120 to FineGym Dataset.

doi:10.6342/NTU202303375

	12 frames	24 frames
PoseC3D (Skeleton)	57.74% ± 0.83%	58.38% ± 0.84%
RelationNet (Skeleton)	46.20% ± 0.93%	47.52% ± 0.89%
Ours (Skeleton)	70.15% ± 0.89%	71.27% ± 0.94%
STRM (RGB)	40.59% ± 0.81%	42.33% ± 0.95%

Table 5.16: System-Level Performance Comparison in 5-way-5-shot Cross-Domain Scenario: NTU-RGB+D 120 to AIMS.

5.6 Ablation Study

5.6.1 Investigating the Importance of Temporal Offset in the Cross Similarity Attention Block

In this section, we investigate the impact of different temporal offsets within the Cross Similarity Attention (CSA) block. The temporal offset is responsible for determining the range of each frame comparison. Our experimental setup utilizes a 24-frame input with a window size of U=9, V=9. During the meta-testing phase, we adopt the fine-tuning of the relation module setting. We conduct this investigation in a same domain context (NTU-RGB+D 120), as well as in cross-domain contexts (NTU-RGB+D 120 to FineGym, and a more realistic scenario of NTU-RGB+D 120 to AIMS dataset).

Range of L	FLOPS	NTU-RGB+D 120	NTU → FineGYM	NTU → AIMS
{0}	34.20G	88.21% ± 0.75%	76.55% ± 0.85%	61.89% ± 0.93%
{-1,0,1}	34.81G	89.05% ± 0.79%	78.23% ± 0.92%	63.40% ± 0.89%
{-2,-1,0,1,2}	35.49G	90.84% ± 0.76%	80.05% ± 0.79%	67.20% ± 0.94%

Table 5.17: Performance comparison with different sets of temporal offset.

From Table 5.18, we observe that a broader temporal offset L range in the Cross Similarity Attention (CSA) block improves the model's performance across all scenarios. When the L range is 0, the model provides decent accuracy. However, by increasing the temporal offset range, the model captures better the short-term and long-term action dynamics, which in turn leads to substantial accuracy improvement.

doi:10.6342/NTU202303375

However, a larger temporal offset range increases the computational demand (measured by FLOPS), as it requires more comparisons for each frame. This necessitates a careful balance between performance improvement and computational cost. In summary, this ablation study highlights the critical role of the temporal offset in the CSA block. A wider temporal offset helps to capture complex action dynamics and improve performance, but also demands more computational resources.

5.6.2 Investigating the Importance of Spatial Offset in the Cross Similarity Attention Block

This section delves into the influence of varying spatial offsets within the Cross Similarity Attention (CSA) block. The spatial offset defines the size of the spatial space for each frame-to-frame comparison. We configured our experimental setup with a 12-frame input and a temporal offset of L=3. During the meta-testing phase, we chose the fine-tuning of the relation module setting. We implemented this investigation in a same domain setting (NTU-RGB+D 120), as well as in cross-domain settings (NTU-RGB+D 120 to FineGym, and a more realistic situation of NTU-RGB+D 120 to AIMS dataset).

Range of U,V	FLOPS	NTU-RGB+D 120	NTU → FineGYM	NTU → AIMS
{3,3}	17.12G	94.94% ± 0.76%	80.53% ± 0.72%	65.51% ± 0.80%
{5,5}	17.18G	93.23% ± 0.81%	82.33% ± 0.89%	65.64% ± 0.82%
{9,9}	17.28G	90.54% ± 0.96%	81.05% ± 0.79%	65.80% ± 0.79%
{Whole Image}	17.34G	93.57% ± 0.46%	78.56% ± 0.78%	63.97% ± 0.74%

Table 5.18: Performance comparison with different sets of spatial offset.

From the results in Table 5.18, it is apparent that while spatial offsets do influence the performance of the model, they are not as impactful as temporal offsets. In the same-domain scenario (NTU-RGB+D 120), a spatial offset of 3,3 provides the highest accuracy of 94.94%. However, increasing the spatial offset to 9,9 leads to a decrease in performance. This suggests that a larger spatial offset does not necessarily lead to better performance in action recognition tasks.

doi:10.6342/NTU202303375

Likewise, in cross-domain scenarios ((NTU-RGB+D 120 to FineGym and (NTU-RGB+D 120 to AIMS), the performance doesn't improve significantly with an increase in spatial offset. Performance slightly dips when the 'whole image' is considered as the spatial offset.

Comparing these results to our previous ablation study on temporal offsets, it is clear that changes in temporal offsets have a more pronounced impact on the model's performance across various domains. This supports the understanding that in the context of few-shot learning for action recognition, the model might be more sensitive to the temporal dynamics within and between action sequences than to the spatial coverage.

Furthermore, these findings align with our observation that similar action patterns tend to occur in close spatial proximity. Consequently, smaller spatial offsets might be more suitable for capturing these patterns and effectively recognizing actions.

5.6.3 The Impact of Camera View on Few-Shot Learning in Action Recognition

In real-world scenarios, action videos are often captured from different camera angles, and the choice of camera view can potentially impact the performance of action recognition models. In this ablation study, we aim to examine the influence of camera view on the accuracy and robustness of action recognition in a few-shot learning setting.

Our experimental setup included data from two sources. The first source, NTU-RGB+D 120, comprises data captured from three different camera angles. The second source, the AIMS dataset, comprises data captured from five distinct camera angles, as shown in Figure 5.3. To maintain consistency and ensure the validity of our investigation, both support set and query set samples were sourced from the same camera angle. This controlled setup facilitated a focused examination of the

doi:10.6342/NTU202303375

influence of camera view variations on the performance of few-shot learning in action recognition. For our model settings, we adopted 12-frame inputs, a temporal offset of L=3, and spatial settings of U=9 and V=9.

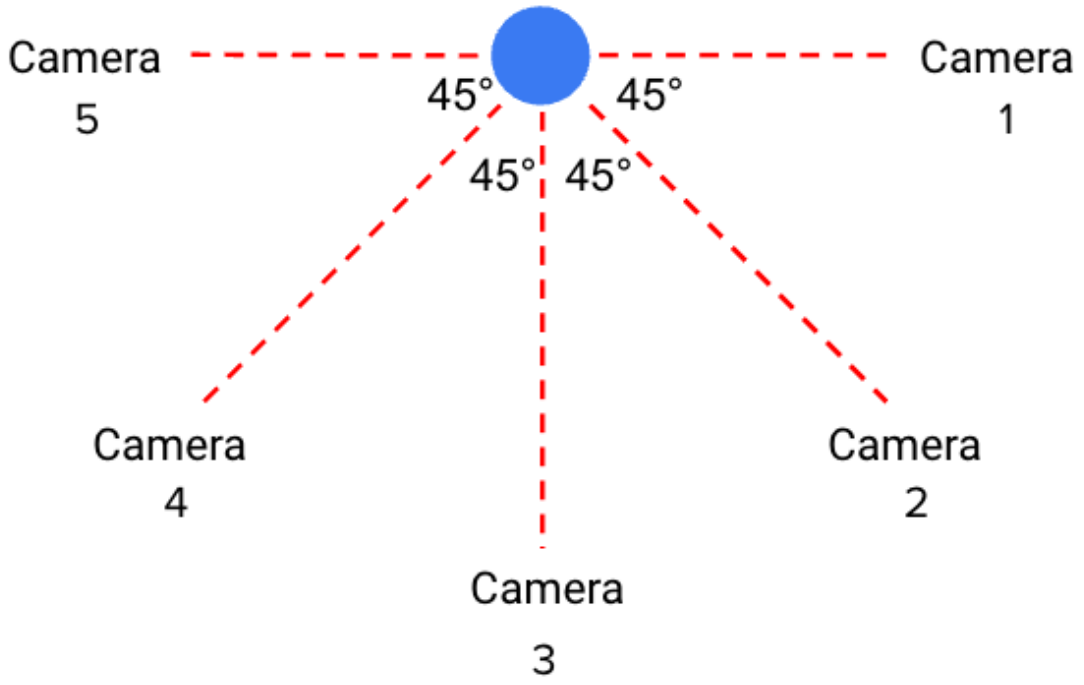


Figure 5.3: Camera Position Illustration for the AIMS dataset Shooting Scene.

	NTU-RGB+D 120	NTU → AIMS
Same View	80.59% ± 0.99%	55.65% ± 1.01%
Cross View	90.54% ± 0.96%	64.80% ± 0.94%

Table 5.19: Performance comparison under same view and cross view scenarios.

From Table 5.19, a striking observation emerges: the performance under the same view scenario is actually inferior to the cross view scenario for both datasets. This suggests that even with limited training data, incorporating multiple viewpoints can be beneficial for action recognition tasks. This could be attributed to the fact that different views provide more comprehensive information about the action, enhancing the model's ability to capture diverse patterns and nuances, and hence, improving its performance. Therefore, even under constraints of data availability, it would be

doi:10.6342/NTU202303375

advantageous to consider data from various angles for more robust and versatile action recognition.

5.6.4 Real-world Scenarios

In our study, we extended the testing of our novel system to include a more complex 10-way, 10-shot setup, which accurately reflects real-world scenarios. This more complex setup offers a broader perspective on the model's capacity to tackle intricate classification tasks, thereby validating its potential for real-world deployment and illustrating its robustness.

Our experiments are divided into two main scenarios, each corresponding to a different application environment: same-domain and cross-domain. In the same-domain scenario, we evaluated our system's performance within a single dataset, focusing on its capacity to learn effectively from limited examples and generalize to new instances within the same context. This serves as an assessment of the system's core few-shot learning capabilities.

In the cross-domain scenario, we challenged our system's adaptability by testing its ability to leverage knowledge from a source domain and apply it to a significantly different target domain. This scenario provides insight into the system's flexibility and adaptability, key traits for practical deployment in diverse real-world environments.

Each of these scenarios was iterated 100 times to ensure robust results. This level of repetition minimizes the potential influence of statistical outliers and random chance, producing results that accurately represent the model's capabilities. Additionally, these repeated iterations allow us to measure the model's performance variability, further bolstering the reliability of our findings.

doi:10.6342/NTU202303375

Scenario	Average Performance (%)	Standard Deviation (%)
NTU-RGB+D120	89.16	0.89
NTU-RGB+D120 → FineGym	76.74	1.45
NTU-RGB+D120 → AIMS	62.09	1.28

Table 5.20: Performance of our system in same-domain and cross-domain in Real-world Scenarios 10Way-10Shot.

doi:10.6342/NTU202303375

Chapter 6

Conclusion

In the field of action recognition, the scarcity and high collection and labelling costs of specific actions have necessitated the development of various few-shot learning algorithms. However, many of these methods excel only within the same domain while performance tends to degrade when applied in cross-domain scenarios, which are more prevalent in real-life situations.

This thesis addresses the aforementioned limitations by introducing a novel data representation named 'Trajectory' and proposing the Cross Similarity Attention Block that leverages the unique characteristics of skeleton data. Additionally, we incorporate a Visual Prompt to facilitate adaptation in cross-domain contexts. These innovative approaches significantly enhance the performance of existing few-shot learning methods in cross-domain scenarios, paving the way for more effective and adaptable action recognition systems in diverse real-life environments.

6.1 Contribution

We summarize the contribution of our work as follows:

- **Novel data representation:** 'Trajectory'. This representation addresses the problem of performance degradation due to video sampling issues, which

doi:10.6342/NTU202303375

cause temporal axis inconsistencies. With 'Trajectory', we can utilize fewer input dimensions, reducing computational requirements without compromising performance.

- **Improved Skeleton Data Classification Precision:** We developed the Cross Similarity Attention Block, a component that recognizes similar action patterns across different videos by considering their spatial and temporal proximity.

- **Innovative Methods for Domain Adaptation:** To address the crucial challenge of domain adaptation in cross-domain scenarios, we present three innovative methods: Fine-tuning the relation module head, Parameter Efficient Fine-tuning, and Visual Prompt Tuning. These strategies offer distinct benefits, collectively enhancing the robustness and adaptability of our model in diverse scenarios. we are the first to integrate visual prompt learning into few-shot learning for cross-domain situations, a breakthrough that our experiments have demonstrated to be highly effective.

6.2 Future Study

in ours study "Prompt learning" has not changed due to input data, but it might be possible to explore the direction of "conditional prompt learning." However, one important consideration is the potential challenge of learning in few-shot learning tasks, where the training samples are limited. In future work, it will be essential to investigate whether there are difficulties in learning under such conditions.

While our current research has significantly advanced the understanding and application of few-shot learning in action recognition using visual inputs, there remain several avenues for future exploration. Drawing inspiration from the success of CLIP (Contrastive Language-Image Pretraining)[26] in classifying images using textual descriptions, we believe that integrating language models with visual data

doi:10.6342/NTU202303375

can open up a new frontier in action recognition. The idea is to provide detailed, standardized textual descriptions of actions, which can then be used alongside visual data to enhance the recognition process.

Language models could be particularly beneficial in scenarios where the visual data is ambiguous, limited, or of poor quality, or when actions are complex and hard to distinguish based solely on visual cues. These models could potentially bring a more nuanced understanding of the actions, providing additional context and helping to bridge the gap between visual perception and semantic understanding. Overall, we believe that the integration of language models in action recognition is a promising direction for future research and can lead to the development of more robust, versatile, and intuitive action recognition systems.

doi:10.6342/NTU202303375

Bibliography

[1] V. Choutas, P. Weinzaepfel, J. Revaud, and C. Schmid, "Potion: Pose motion representation for action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7024–7033, 2018.

[2] A. Hernandez Ruiz, L. Porzi, S. Rota Bulò, and F. Moreno-Noguer, "3d cnns on distance matrices for human action recognition," in *Proceedings of the 25th ACM international conference on Multimedia*, pp. 1087–1095, 2017.

[3] 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," vol. 42, pp. 2684–2701, IEEE, 2019.

[4] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," *arXiv preprint arXiv:1904.04232*, 2019.

[5] J. K. Aggarwal and M. S. Ryoo, "Human activity analysis: A review," *Acm Computing Surveys (Csur)*, vol. 43, no. 3, pp. 1–43, 2011.

[6] M. Ziaeefard and R. Bergevin, "Semantic human activity recognition: A literature review," *Pattern Recognition*, vol. 48, no. 8, pp. 2329–2345, 2015.

[7] G. T. Papadopoulos, A. Axenopoulos, and P. Daras, "Real-time skeleton-tracking-based human action recognition using kinect data," in *MultiMedia Modeling: 20th Anniversary International Conference, MMM 2014, Dublin,*

doi:10.6342/NTU202303375

Ireland, January 6-10, 2014, Proceedings, Part I 20, pp. 473–483, Springer, 2014.

[8] S. N. Paul and Y. J. Singh, "Survey on video analysis of human walking motion," *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 7, no. 3, pp. 99–122, 2014.

[9] H. Rahmani, A. Mahmood, D. Q Huynh, and A. Mian, "Hopc: Histogram of oriented principal components of 3d pointclouds for action recognition," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, pp. 742–757, Springer, 2014.

[10] J. Wan, Q. Ruan, W. Li, G. An, and R. Zhao, "3d smosift: three-dimensional sparse motion scale invariant feature transform for activity recognition from rgb-d videos," *Journal of Electronic Imaging*, vol. 23, no. 2, pp. 023017–023017, 2014.

[11] D. Das Dawn and S. H. Shaikh, "A comprehensive survey of human action recognition with spatio-temporal interest point (stip) detector," *The Visual Computer*, vol. 32, pp. 289–306, 2016.

[12] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

[13] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *arXiv preprint arXiv:1406.2199*, 2014.

[14] S. Song, C. Lan, J. Xing, W. Zeng, and J. Liu, "Spatio-temporal attention-based lstm networks for 3d action recognition and detection," *IEEE Transactions on image processing*, vol. 27, no. 7, pp. 3459–3471, 2018.

doi:10.6342/NTU202303375

[15] 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.

[16] D. Shao, Y. Zhao, B. Dai, and D. Lin, "Finegym: A hierarchical video dataset for fine-grained action understanding," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2616–2625, 2020.

[17] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems*, vol. 30, 2017.

[18] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1199–1208, 2018.

[19] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *Advances in neural information processing systems*, vol. 27, 2014.

[20] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*, pp. 1126–1135, PMLR, 2017.

[21] G. Koch, R. Zemel, R. Salakhutdinov, *et al.*, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, vol. 2, Lille, 2015.

[22] S. Benaim and L. Wolf, "One-shot unsupervised cross domain translation," *advances in neural information processing systems*, vol. 31, 2018.

[23] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5693–5703, 2019.

doi:10.6342/NTU202303375

[24] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[25] W. Chen, C. Si, Z. Zhang, L. Wang, Z. Wang, and T. Tan, "Semantic prompt for few-shot image recognition," *arXiv preprint arXiv:2303.14123*, 2023.

[26] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021.

[27] B. M. Lake, R. Salakhutdinov, and J. B. Tenenbaum, "Human-level concept learning through probabilistic program induction," *Science*, vol. 350, no. 6266, pp. 1332–1338, 2015.

[28] O. Vinyals, C. Blundell, T. Lillicrap, D. Wierstra, *et al.*, "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.

[29] G. S. Dhillon, P. Chaudhari, A. Ravichandran, and S. Soatto, "A baseline for few-shot image classification," *arXiv preprint arXiv:1909.02729*, 2019.

[30] W.-H. Li, X. Liu, and H. Bilen, "Cross-domain few-shot learning with task-specific adapters," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7161–7170, 2022.

[31] 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, no. 3, pp. 257–267, 2001.

doi:10.6342/NTU202303375

[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[33] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6202–6211, 2019.

[34] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

[35] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 6450–6459, 2018.

[36] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII 14*, pp. 483–499, Springer, 2016.

[37] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.

[38] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2969–2978, 2022.

doi:10.6342/NTU202303375

[39] J. Lee, M. Lee, D. Lee, and S. Lee, "Hierarchically decomposed graph convolutional networks for skeleton-based action recognition," *arXiv preprint arXiv:2208.10741*, 2022.

[40] K. Cao, J. Ji, Z. Cao, C.-Y. Chang, and J. C. Niebles, "Few-shot video classification via temporal alignment," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10618–10627, 2020.

[41] T. Perrett, A. Masullo, T. Burghardt, M. Mirmehdi, and D. Damen, "Temporal-relational crosstransformers for few-shot action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 475–484, 2021.

[42] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[43] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[44] A. Thatipelli, S. Narayan, S. Khan, R. M. Anwer, F. S. Khan, and B. Ghanem, "Spatio-temporal relation modeling for few-shot action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19958–19967, 2022.

[45] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1010–1019, 2016.

doi:10.6342/NTU202303375