Robust and Efficient Deep Learning Methods for Vision-based Action Recognition

Yuecong Xu

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2021

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

16 Sep. 2021
.....
Date

.....
Yuecong Xu

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

16 Sep. 2021

Date

Assoc. Prof. Kezhi Mao

Authorship Attribution Statement

This thesis contains material from 6 papers published or under review in the in the peer-reviewed journals and conferences in which I am listed as an author.

Chapter 3 has been accepted for publication as Yuecong Xu, Jianfei Yang, Kezhi Mao, Jianxiong Yin and Simon See. "Exploiting Inter-Frame Regional Correlation for Efficient Action Recognition." Expert Systems with Applications 178 (2021): 114829.

The contributions of the co-authors are as follows:

- I proposed the idea, designed the experiments and prepared the manuscript.
- Jianxiong Yin, Jianfei Yang and I conducted the experiments.
- Simon See provided hardware and technical support for conducting the experiments.
- The manuscript was revised by Kezhi Mao and Jianfei Yang.

Chapter 4 has been accepted for publication as Yuecong Xu, Haozhi Cao, Jianfei Yang, Kezhi Mao, Jianxiong Yin, and Simon See. "PNL: Efficient long-range dependencies extraction with pyramid non-local module for action recognition." Neurocomputing 447 (2021): 282-293.

The contributions of the co-authors are as follows:

- I proposed the idea, designed the experiments and prepared the manuscript.
- Haozhi Cao, Jianfei Yang and I conducted the experiments.
- Jianxiong Yin and Simon See provided hardware and technical support for conducting the experiments.
- The manuscript was revised by Kezhi Mao and Jianfei Yang.

Chapter 5 contains research results published as Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "ARID: A New Dataset for Recognizing Action in the Dark." Deep Learning for Human Activity Recognition, DL-HAR 2021, Communications in Computer and Information Science 1370 (2021).

And a paper under review: Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "ARID: A Comprehensive Study on Recognizing Actions in the Dark and A New Benchmark Dataset."

The contributions of the co-authors are as follows:

- I proposed the idea, designed the experiments and prepared the manuscript.
- Jianfei Yang, Kezhi Mao and I discussed the idea.

- Haozhi Cao, Jianfei Yang, Jianxiong Yin and I conducted the experiments
- Simon See provided hardware and technical support for conducting the experiments.
- The manuscript was revised by Kezhi Mao, Simon See and Jianfei Yang.

Chapter 6 contains a paper under review as Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "Aligning Correlation Information for Domain Adaptation in Action Recognition."

The contributions of the co-authors are as follows:

- I proposed the idea, designed the experiments and prepared the manuscript.
- Jianfei Yang and I discussed the idea.
- Haozhi Cao and I conducted the experiments.
- Jianxiong Yin and Simon See provided hardware and technical support for conducting the experiments.
- The manuscript was revised by Kezhi Mao and Jianfei Yang.

Chapter 7 is a paper under review as Yuecong Xu, Jianfei Yang, Haozhi Cao, Zhenghua Chen, Qi Li and Kezhi Mao. "Partial Video Domain Adaptation with Partial Adversarial Temporal Attentive Network."

The contributions of the co-authors are as follows:

- I proposed the idea, designed the experiments and prepared the manuscript.
- Haozhi Cao, Qi Li, Jianfei Yang and I conducted the experiments.
- The manuscript was revised by Kezhi Mao, Zhenghua Chen and Jianfei Yang.

16 Sep. 2021

.....

Date

.....

Yuecong Xu

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor, Prof. Kezhi Mao, for his professional guidance, continuous encouragement while providing academic freedom throughout my Ph. D. study. His incisive insights in machine learning have greatly inspired my explorations and have encouraged me to forge ahead into challenging research. All the methods and ideas presented are impossible without his valuable guidance and constant support. His attitude towards solid research set a good example of what a true researcher should be.

I would like to give special gratitude to Dr. Jianfei Yang, for his unselfish guidance and sincere encouragement throughout the course of my Ph. D. study. Dr. Yang has spent large amounts of time beyond his own research in numerous discussions on research ideas with me. Many of the ideas presented would not be possible if it were not for his constructive suggestions and assistance. To me, he is not only a collaborator but also a mentor who shares his experience selflessly.

I would also like to give my great gratitude to Mr. Haozhi Cao, for his endless support and patience during our collaboration. Sharing similar experiences, his optimism towards research has been a great source of encouragement whenever I face difficulties during my own research. I wish him all the best in his future path towards obtaining a Ph. D. degree.

Meanwhile, I would like to express my thanks to my seniors, colleagues, and friends, Dr. Zhenghua Chen, Dr. Dongzhe Wang, Dr. Pengfei Li, Dr. Zuobin Wu, Dr. Jie Ding, Dr. Qinxu Ding, Dr. Kemi Ding, Dr. Chao Deng, Dr. Xiaolei Li, Dr. Xiaokang Liu, Dr. Jiaqi Yan, Dr. Qianwen Xu, Dr. Ci Chen, Dr. Chongyi Li, Dr. Le Zhang, Dr. Thien-Minh Nguyen, Dr. Shenghai Yuan, Qi Li, Jiaheng Zhang, Xinyao Li, Minghui Hu, Xu Fang, Jingsong He, Muqing Cao, Hanjie Qian, He Huang, Ruikang Luo, Kai Wang, Xu Yang, Huaizheng Zhang and Yanxi Wang, for their encouragement, support, and friendship. Special thank goes to Dikai Liu for our unforgettable memories in various competitions

around the globe. Special thanks go to Jianxiong Yin and Prof. Simon See for their continuous support and help in realizing my ideas.

Last but not least, I would like to sincerely thank my beloved mother, for her unconditional love and support throughout my Ph. D. study. Her career as a university lecturer is what ignites me in pursuing a Ph. D. degree. My greatest thank and love to her.

Summary

Vision-based action recognition, which performs action recognition based solely on RGB frames, has received strong research interest thanks to its wide applications in various fields, e.g. surveillance, smart homes, and autonomous driving. Significant progress has been made in vision-based action recognition thanks to the development of recognition technologies, particularly deep learning methods which have proven their effectiveness in visual recognition tasks, such as image classification. Compared to static images, videos contain additional information due to the additional temporal dimension, which includes both temporal and spatiotemporal correlation features. Therefore, the key to robust and efficient action recognition lies in the effective and efficient utilization of the temporal and spatiotemporal correlation features embedded within videos.

In this thesis, we first investigate extracting temporal features in a robust and efficient manner. While methods for extracting temporal features were proposed, these methods either require the computation or estimation of optical flow, which demand high computational power and large storage resource; or extracts only linear feature along the temporal dimension which results in inferior performances. To extract temporal features without the utilization of optical flow, we propose Attentive Correlated Temporal Feature (ACTF) which leverages inter-frame correlation feature and exploits both bilinear and linear correlations between successive frames on the regional level. By excluding optical flow estimation or calculation, ACTF can be combined with any spatial feature extraction network under the two-stream structure for end-to-end training. Meanwhile, capturing long-range spatiotemporal dependencies is an effective strategy for extracting spatiotemporal correlation features. Previous works have proposed methods utilizing either hand-crafted features or stacks of convolution or recurrent modules, both of which are computationally inefficient, and cause difficulty in network optimization. While the more recent non-local block inspired by the non-local means could extract long-range dependencies without affecting the networks' optimization, it significantly increases the parameter size and computational cost of the inserted networks. To extract robust long-range dependencies more efficiently, we explore on further improving

the non-local neural network by proposing a novel long-range spatiotemporal dependencies extraction module, the Pyramid Non-Local (PNL) module. It extends the original non-local block by incorporating regional feature correlations at multiple scales. PNL upscales the effectiveness of the original non-local block by additionally addressing the spatiotemporal correlations between different regions, while improving the efficiency of the original non-local block with a significant decrease in computation cost.

Besides the development of recognition technologies, the progress made in vision-based action recognition could also be attributed to the development of large-scale video datasets, which enable the effective training of deep learning models. However, it could be observed that the majority of current research focuses on videos in normal illumination, partly due to the fact that current benchmark datasets for vision-based action recognition are normally collected from web videos shot mostly under normal illumination. Yet, we argue that vision-based action recognition should not be constrained in normal illuminated videos. Vision-based action recognition in dark videos are also useful in various scenarios, e.g., night surveillance and self-driving at night. Such a task has rarely been researched, partly due to the lack of sufficient datasets for such a task. To this end, this thesis bridges the gap of the lack of data and pioneers vision-based action recognition in dark videos by collecting a novel dataset: the Action Recognition in the Dark (ARID) dataset. In this thesis, the ARID dataset is analyzed thoroughly with a comprehensive benchmark of current deep learning methods.

Meanwhile, though the introduction of ARID pioneers vision-based action recognition in dark videos and bridges the gap between the absence of dark video datasets with the need for research, the scale of such a dataset is relatively small compared to current large-scale video datasets. One solution for training robust models for domains with less labeled data would be to transfer models learned in well-labeled domains. However, models trained in one domain would not generalize well in the other domain due to domain shift across domains, which is presented as distribution discrepancy between different domains. Domain adaptation (DA) approaches address domain shifts and enable networks to be applied to different scenarios. Although various image DA approaches have been proposed in recent years, there is limited research towards Video Domain Adaptation (VDA), owing to the complexity in adapting the different modalities of features in videos, which includes both temporal features and spatiotemporal correlation features. We argue that the correlation features are highly associated with action classes and proven their effectiveness in accurate video feature extraction through supervised

vision-based action recognition. Yet correlation features of the same action would differ across domains due to domain shift. Adversarial Correlation Adaptation Network (ACAN) is developed in this thesis to align action videos by aligning pixel correlations, while a novel HMDB-ARID dataset with a larger domain shift is built in an effort to leverage current datasets for vision-based action recognition in dark videos.

We observe that while VDA methods enable the learning of transferable features across domains, these methods generally assume that the video source and target domains share an identical label space. Such an assumption may not hold in real-world applications. Instead, Partial Domain Adaptation (PDA) is a practical and general domain adaptation scenario, which relaxes the fully shared label space assumption such that the source label space subsumes the target one, and is more challenging than DA due to negative transfer caused by source-only classes. For videos, such negative transfer could be triggered by both spatial and temporal features, which leads to an even more challenging Partial Video Domain Adaptation (PVDA) problem. This thesis pioneers the PVDA problem by proposing a novel Partial Adversarial Temporal Attentive Network (PATAN) by utilizing both spatial and temporal features for filtering source-only classes. This thesis further introduces new benchmarks to facilitate research on PVDA problems, covering a wide range of PVDA scenarios.

In summary, this thesis contributes to robust and efficient vision-based action recognition by introducing two algorithms for extracting robust and efficient temporal or spatiotemporal correlation features and pioneering in the research of robust vision-based action recognition in dark videos, breaking through the constraint of current vision-based action recognition research conducted on only normal illuminated videos.

Contents

List of Figures

List of Tables

Symbols and Acronyms

Symbols

$\mathcal{H}$	the hypothesis space
$\mathcal{H}\Delta\mathcal{H}$	the symmetric difference hypothesis space with respect to the hypothesis space $\mathcal{H}$
$\epsilon(h)$	the expected error of hypothesis h
$\boldsymbol{C}_0$	a constant term
$\mathbb{R}^{N\times M}$	set of real $N \times M$ dimensional matrices
$\mathbb{R}^N$	set of real N dimensional vectors
$\langle \cdot \rangle$	the inner product of two vectors
$\oplus$	the concatenation operation
$\sum_{i=a}^{b} g(i)$	the sum of function $g(i)$ for variable i with values from a to b
$f(\cdot, \cdot)$	a pairwise function f
$\sigma(\cdot)$	the softmax function
$\lceil \cdot \rceil$	the ceiling function (largest integer less than $\cdot$)
$\mathcal{N}(\mu, \sigma^2)$	a normal distribution with mean μ and standard deviation σ
$G(\cdot; \theta)$	a function G with parameters θ
$\| \cdot \|^2$	the L2-norm of a matrix or vector in the Euclidean space
$\lvert \cdot \rvert$	the number of elements in a space

Acronyms

CNN	Convolutional Neural Network
DT	Dense Trajectory
SIFT	Scale-invariant feature transform
GLOH	Gradient Location and Orientation Histogram
ARID	Action Recognition in the Dark

DA	Domain Adaptation
VDA	Video Domain Adaptation
PVDA	Partial Video Domain Adaptation
ACTF	Attentive Correlated Temporal Feature
NLBlock	Non-Local block
PNL	Pyramid Non-Local
ACAN	Adversarial Correlation Adaptation Network
STIP	Space-Time Interest Points
SVM	Support Vector Machine
MEI	Motion Energy Image
MHI	Motion History Image
MC	Motion Context
HOG	Histogram of Oriented Gradients
BOVW	Bag-of-Visual Words
iDT	improved Dense Trajectories
MBH	Motion Boundary Histogram
SFV	Stacked Fisher Vectors
DNN	Deep Neural Networks
TSN	Temporal Segment Network
RNN	Recurrent Neural Network
LRCN	Long-term Recurrent Convolutional Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
CSN	Channel-Separated Convolutional Network
NLNet	Non-Local Neural Network
CGNL	Compact Generalized Non-Local
GC	Global Context
SID	See-in-the-Dark
ExDARK	Exclusively Dark
DRV	Dark Raw Video
SMOID	See-Moving-Objects-in-the-Dark
ND	Neutral Density
SDA	Supervised Domain Adaptation
UDA	Unsupervised Domain Adaptation
VUDA	Video Unsupervised Domain Adaptation

GAN	Generative Adversarial Network
DANN	Domain-Adversarial Neural Network
GRL	Gradient Reversal Layer
MDD	Margin Disparity Discrepancy
ADDA	Adversarial Discriminative Domain Adaptation
CADA	Consensus Adversarial Domain Adaptation
AADA	Asymmetric Adversarial Domain Adaptation
MMD	Maximum Mean Discrepancy
RKHS	Reproducing Kernel Hilbert Space
DAN	Deep Adaptation Network
SISS	Statistically Invariant Sample Selection
SIE	Statistically Invariant Embedding
AMLS	Action Modeling Latent Subspace
DAAA	Deep Adversarial Action Adaptation
TCoN	Temporal Co-attention Network
ICCF	Inter-frame Corresponding-regional Correlation Feature
IMF	Inter-frame Mean Feature
TS	Two-Stream
HE	Histogram Equalization
GIC	Gamma Intensity Correction
Improv.	Improvement
CAM	Class Activation Maps
PCD	Pixel Correlation Discrepancy
PATAN	Partial Adversarial Temporal Attentive Network

Chapter 1

Introduction

1.1 Motivation and Objectives

As one of the cornerstone tasks for video understanding problems, vision-based action recognition based on RGB frames has received considerable attention from the vision community in recent years, whose goal is to identify actions in unlabeled test videos utilizing trained classifiers. The rise in the attention of vision-based action recognition is attributed to both its increasing applications in fields such as surveillance, smart home, and autonomous driving, and the development of vision-based action recognition technologies. More recently, deep learning methods, such as convolutional neural networks (CNN), has achieved significant results compared to previous methods in visual recognition tasks, e.g. image classification. The success of CNN in tasks such as image classification prompted researchers to apply CNN in vision-based action recognition. Compared to image classification or other static image-based visual recognition tasks, vision-based action recognition is more challenging due to the higher complexity of video data where videos contain additional information thanks to the additional temporal dimension. More specifically, videos contain not only spatial features as in images but also temporal and spatiotemporal correlation features. Robust and efficient action recognition methods, therefore, require the extraction of robust and efficient temporal and spatiotemporal correlation features in videos.

From the perspective of extracting temporal features, there have been multiple methods proposed in the past decade. Most of these methods could be organized into two categories: two-stream or 3D Convolutional Neural Network(3D-CNN), where both have

their own advantages and drawbacks. On one hand, two-stream methods achieve state-of-the-art performances at the cost of high computation power due to their reliance on optical flow estimation for robust temporal features. On the other hand, 3D-CNNs are computationally more efficient and can be trained in an end-to-end manner, but their efficiencies are achieved at a cost of inferior action recognition accuracies. As such, there is a need for temporal features to be both robust and efficient by excluding the involvement of optical flow. In this thesis, we explore a novel temporal feature that leverages regional correlation across successive frames in Chapter 3.

From the perspective of extracting spatiotemporal correlation features, one effective strategy is by extracting long-range spatiotemporal dependencies. In previous methods, such dependencies have been modeled through hand-crafted features such as SIFT, GLOH, or Dense Trajectory (DT). With convolutional and recurrent modules being utilized as the predominant building block for video feature extractor instead of the above hand-crafted methods, long-range dependencies are captured through a stack of multiple building blocks due to the fact that each building block could extract only local spatiotemporal dependencies. More recently, the non-local block was proposed to capture long-range dependencies directly without having to stack multiple building blocks, and networks with non-local blocks inserted have achieved significant improvement in vision-based action recognition. Despite its effectiveness, the insertion of non-local block greatly increases the parameter size and computational cost of the inserted networks owing to the fact that long-range dependencies are captured through pixel-level correlations. We argue that instead of pixel-level correlations, humans recognize actions by focusing on region-level correlations. In this thesis, we explore on improving the effectiveness and efficiency of the non-local block by proposing a novel long-range spatiotemporal dependencies extraction module based on multi-scaled region-level correlations in Chapter 4.

The development of deep learning methods constitutes only part of the progress made in vision-based action recognition. The availability of large-scale video datasets, such as HMDB51, UCF101, Activity-Net, Kinetics, and Something-Something, helped pushed the boundaries of the task of action recognition by enabling deep learning models to be trained on a vast amount of videos. The resulting trained models are able to generalize well to unseen testing videos thanks to the diversity of the large-scale video datasets. However, although these datasets involve an abundant scale of videos, they are mostly

collected from web videos due to the ease of obtaining web videos through mass downloading. Therefore, it could be noticed that the majority of current research may only apply to videos shot under normal illumination. We argue that the robustness of deep learning methods towards videos shot in different environment, e.g., towards videos shot in low illumination or under hazy conditions, is an important aspect of method robustness. While vision-based action recognition in dark videos has rarely been studied, such a task is useful in various real-world applications, including night surveillance and nighttime autonomous driving. Compared to the abundance of data for videos in normal illumination, there are no public datasets dedicated to human actions in the dark, which partly explains the lack of research in this task. Therefore this thesis pioneers vision-based action recognition in the dark by introducing a novel Action Recognition in the Dark (ARID) dataset in Chapter 5, and focused on understanding this dataset through benchmarking of current action recognition methods and thorough analysis.

Compared to the public datasets with videos mostly in normal illumination, the scale of ARID is much smaller. This hinders the ability of models trained on ARID to generalize to other test videos, and limits the complexity of the method trained as methods with higher complexity tend to overfit on small datasets. While it is possible theoretically to simply increase the scale of the dataset, it is worth noted that annotations of video data are very costly. One strategy for training robust deep learning models in dark videos without the cost of annotation is by transferring models trained in current large-scale datasets to dark videos. The main challenge of such a strategy lies in the difference in data characteristics and distributions over the different datasets, defined as the domain shift between different domains. To bridge domain shift, domain adaptation (DA) approaches are proposed to transfer models trained from a label-rich source domain, in this case, datasets of videos in normal illumination, to a label-scarce or even unlabeled target domain, in this case, datasets of dark videos. While multiple image-based DA approaches have been developed, methods for Video Domain Adaptation (VDA) are scarce, due to the complexity of video data compared to image data. Earlier works for VDA proposed approaches that are adopted directly from image-based DA approaches while utilizing 3D-CNNs instead of 2D-CNNs for images. Subsequently, improvements are made in VDA through aligning data of the different domains along the temporal direction, yet these improvements have not touched on the alignment of the spatiotemporal correlation feature of a video. As stated above, we believe that the spatiotemporal correlation feature is also a key feature of a video and is highly associated with the action in the video. It is therefore intuitive to align correlation features in a robust VDA approach,

which is explored in Chapter 6. Furthermore, one other drawback of current VDA approaches is the fact that these approaches all assume that source and target domains share an identical label space. Given that the target domain is usually label-scarce or even unlabeled, such an assumption may not hold in real-world applications. Instead, it is more feasible to assume that the source domain is related to large-scale datasets while the target domain is related to small-scale datasets. This scenario is referred to as Partial Domain Adaptation (PDA), which assumes that the target label space is a subspace of the source label space. Such an assumption has not been discussed for videos, therefore in this thesis, we formulated Partial Video Domain Adaptation (PVDA), pioneering in PDA problem for videos. The key towards PVDA is similar to PDA, which is to eliminate negative transfer brought by source-only classes and is achieved through class filtration. However, PVDA is more challenging considering the fact that negative transfer could be triggered by spatiotemporal domain shift instead of only spatial domain shift for images. In Chapter 7 of this thesis, our pioneer research in PVDA proposes a novel approach utilizing both spatial and temporal features for class filtration. Meanwhile, to explore how to leverage current large-scale datasets for videos shot in the adverse environment with a focus on dark videos, we further propose novel cross-domain datasets for both VDA and PVDA problems.

The structure of this thesis is shown in Figure 1.1, where the core concept is robust and efficient vision-based action recognition. To this end, we study on approaches for robust and efficient extraction of temporal and spatiotemporal correlation features, while pioneering the study of robust vision-based action recognition methods in dark videos. For robust and efficient feature extraction, we proposed novel approaches for temporal and spatiotemporal correlations respectively. For research in dark videos, we propose a novel dataset while also focusing on transferring models from current public large-scale datasets to dark videos while overcoming domain shifts across the different datasets. The objective of this thesis is to empower robust and efficient vision-based action recognition by means of feature extraction and research into videos of low-illumination (e.g. dark videos).

1.2 Major Contributions

Our main contributions can be stated as follows:

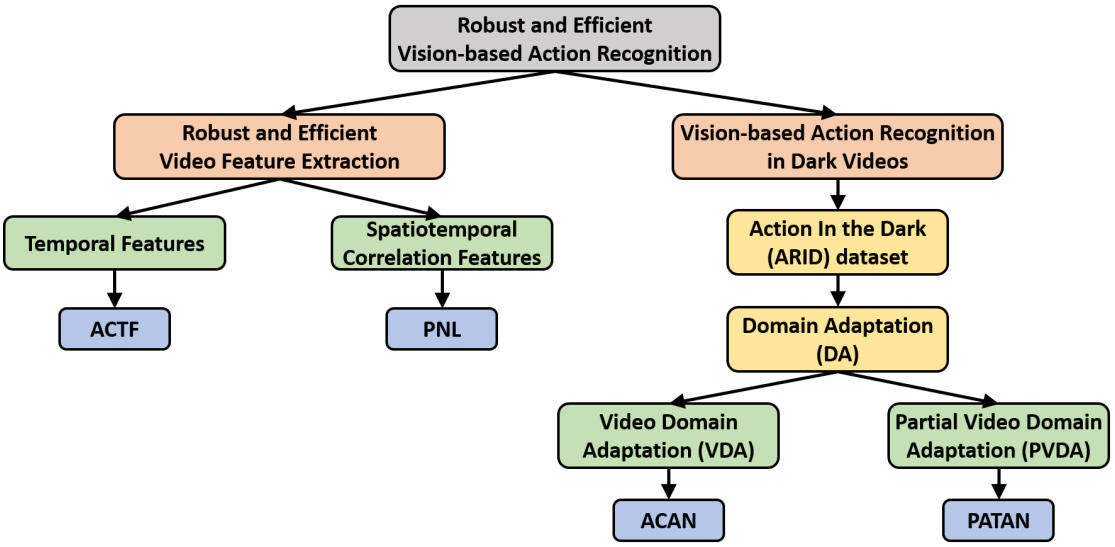


FIGURE 1.1: Illustration of the structure of this thesis.

Efficient Temporal Feature Extraction with Inter-frame Regional Correlation. Existing vision-based action recognition methods tend to extract robust temporal features through extracting optical flow thanks to its capacity of capturing pixel-level correlation information across consecutive frames. Yet, such temporal feature extraction is of high computation cost. To extract robust temporal features efficiently, in this thesis, we propose a novel approach named Attentive Correlated Temporal Feature (ACTF), exploiting inter-frame corresponding-regional correlation for implicit extraction of temporal feature without utilizing optical flow. Extensive experiments show that ACTF leads to competitive performance compared to optical-flow-based methods while being more efficient by demanding less computation and memory.

Efficient Spatiotemporal Correlation Feature Extraction via Pyramid Non-Local Module. Robust vision-based action recognition requires not only robust temporal features but also robust spatiotemporal correlation features. While utilizing Non-Local block (NLBlock) inspired by the non-local mean to extract long-range spatiotemporal dependencies is an effective strategy, it significantly increases network size and computation cost owing to its capturing long-range dependencies on the pixel level. Moreover, humans recognize actions through correlations between larger regions intuitively. To this end, we propose Pyramid Non-Local (PNL) module which extends the NL block by incorporating regional feature correlations at multiple scales. The PNL module attends to the interactions between different regions, while comprehensive analysis over computation cost corroborates the higher efficiency of the PNL module compared to the NL block.

Action Recognition in the Dark Dataset. The robustness of vision-based action recognition methods towards videos shot in adverse environments, e.g. low illumination or haze, is also an important aspect of method robustness. Yet little research has touched on the task of vision-based action recognition in the dark, partly due to the lack of relevant datasets as most public datasets consist of web videos shot under normal illumination. In this thesis, we pioneer the task of vision-based action recognition in the dark by introducing a novel dataset: the Action Recognition In the Dark (ARID) dataset. To our knowledge, ARID is the first dataset dedicated to human actions in dark videos. Extensive analysis and experiments are conducted to corroborate the necessity of ARID and to reveal challenges in vision-based action recognition in the dark.

Aligning Correlation Information in Videos for Video Domain Adaptation. Transferring models trained in the well-labeled source domain to label-scarce or unlabeled target domain is an effective strategy to cope with the scarcity of labeled dark videos, as ARID relatively small-scale compared with existing public video datasets. Domain Adaptation (DA) tackles the problem of domain shift, yet existing visual DA approaches have been mainly image DA approaches, while there is limited research towards Video Domain Adaptation (VDA). VDA is indeed more challenging thanks to the complexity in adapting the different modalities of features in videos. Current VDA approaches either extend from image DA approaches or only focused on the alignment of temporal features in videos. To this end, we propose an Adversarial Correlation Adaptation Network (ACAN) to align videos across domains by aligning their spatiotemporal pixel correlations. Further, a novel and more challenging cross-domain video dataset is introduced to leverage current datasets for dark vision-based action recognition. Extensive experiments demonstrate the effectiveness of the proposed network across multiple video cross-domain datasets.

Partial Adversarial Temporal Attentive Network for Partial Video Domain Adaptation. In real-world applications, the constraint of shared label space between the source and target domains as assumed in DA and VDA approaches may not hold. Instead, it is more feasible to transfer representations learnt in the larger source domain to the smaller target domain, where the source domain subsume categories of the target domain. For video data, such a scenario is formulated as Partial Video Domain Adaptation (PVDA) and is pioneered in this thesis. To tackle the challenge of negative transfer of PVDA brought by the source-only classes, we propose a Partial Adversarial Temporal Attentive Network (PATAN). PATAN addresses the negative transfer in PVDA by full

utilization of temporal features in filtering out source-only classes while constructing robust video temporal features. Additionally, we introduce several PVDA benchmarks while demonstrating the effectiveness of PATAN on these benchmarks by conducting extensive experiments.

1.3 Outline of the Thesis

The remainder of this thesis is organized as follows:

Chapter 2 presents a comprehensive literature review on various methods for vision-based action recognition. Particularly, deep learning methods are focused, which is closely related to this thesis. Recent visual domain adaptation approaches, which include both image and video domain adaptation are also reviewed in this chapter. Challenges and drawbacks of these methods and approaches are pointed out, from which research in this thesis is inspired. Further, relevant benchmark datasets, including vision-based action recognition, dark visual, and cross-domain video benchmark datasets, are also surveyed and discussed.

Chapter 3 proposes a novel robust and efficient temporal feature extraction method for vision-based action recognition. The novel method exploits corresponding regional correlation across successive frames for implicit capturing of temporal information.

Chapter 4 proposes a pyramid-structured module for efficient capturing of long-range spatiotemporal dependencies, an effective form of spatiotemporal correlation feature. The new module aims to capture robust and efficient long-range spatiotemporal dependencies by incorporating multi-scaled regional feature correlations.

Chapter 5 introduces a new dataset, namely the Action Recognition In the Dark (ARID) dataset, dedicated to the task of vision-based action recognition in dark videos. In this chapter, the ARID dataset is analyzed thoroughly both statistically and by extensive benchmark experiments. The investigation on ARID also reveals challenges in the task of vision-based action recognition in dark videos as well as potential methods for improving current deep learning methods for action recognition.

Chapter 6 proposes a new domain adaptation approach for Video Unsupervised Domain Adaptation (VUDA), which aims to tackle domain shift by aligning video spatiotemporal correlation features across domains. Meanwhile, a more challenging cross-domain

dataset is also introduced and is the first cross-domain dataset that includes videos shot under different illumination. Extensive experiments are discussed with respect to current and novel VUDA datasets.

Chapter 7 proposes Partial Video Domain Adaptation (PVDA), a more realistic video-based domain adaptation scenario, where the target label space is a subspace of the source label space. In this chapter, a new network that utilizes both spatial and temporal features for class filtration is presented to tackle the PVDA problem, while several PVDA benchmarks are introduced to facilitate future PVDA research. Quantitative results of our proposed method on the PVDA benchmarks are detailed and discussed.

Chapter 8 concludes this thesis and discusses some potential future research.

Chapter 2

Literature Review

In this thesis, we focus on vision-based action recognition which performs action recognition based on RGB frames solely. In this chapter, we comprehensively review the related fields with respect to our research. Our review is conducted from three perspectives: vision-based action recognition, vision-based action recognition datasets and dark visual datasets, and domain adaptation. Since this thesis focuses on deep learning methods, we start this chapter with a review of vision-based action recognition methods in Section 2.1, where traditional handcrafted methods are presented and deep learning methods are discussed comprehensively. In Section 2.2, current benchmark datasets which include both vision-based action recognition benchmark datasets and dark visual benchmark datasets are introduced and discussed. Lastly, in Section 2.3, we survey video unsupervised domain adaptation methods with reviews on the learning theory of domain adaptation and general domain adaptation methods, followed by the relevant cross-domain video benchmark datasets.

2.1 Overview of Vision-based Action Recognition Methods

As one of the basic research tasks in video analysis, action recognition has been studied by the community over the past few decades. Different modalities have been used for action recognition, most notably skeleton data (corresponding to skeleton-based action recognition) and RGB frames (corresponding to vision-based action recognition).

While skeleton-based action recognition is more robust to changing conditions involving body scales, view points and motion speed thanks to the more accurate topological representation of human with skeleton data, acquiring skeleton data usually requires additional sensors, such as depth sensor, which might not be readily available to scenarios in the wild. Moreover, compared to vision-based action recognition based on only RGB-frames, skeleton-based action recognition requires accurate human pose estimation, which increases computation cost. Therefore, in this thesis we focus on vision-based action recognition based solely on RGB frames.

In earlier works, vision-based action recognition methods aim to design handcrafted features to represent action robustly and effectively. However, due to the negative effects posed by factors such as camera movement and occlusion, action recognition performances of methods based on handcrafted features are limited. More recently, following the success of deep learning methods in image-based tasks, deep learning methods have also been applied to vision-based action recognition, and have achieved remarkable progress. In this section, we briefly introduce vision-based action recognition methods utilizing handcrafted features while comprehensively review vision-based action recognition methods based on deep learning techniques.

2.1.1 Methods with Handcrafted Features

Handcrafted features designed for vision-based action recognition methods aim to capture object (usually human) movements, as well as spatiotemporal changes in the action videos. Roughly, the handcrafted features could be categorized into three types: spatiotemporal volume-based features, Space-Time Interest Points-based (STIP-based) features, and trajectory-based features. To perform action recognition, classic machine learning algorithms such as Support Vector Machine (SVM) are applied to the extracted features.

Spatiotemporal Volume-based Features. Spatiotemporal volume-based features could be viewed as three-dimensional spatiotemporal templates, therefore methods utilizing such features are simply template matching methods. Spatiotemporal volume-based features are one of the earliest features dedicated to vision-based action recognition. Motion Energy Image (MEI) and Motion History Image (MHI) introduced in [3] both belong to this category and are the representative work for this category. Later works as in [4, 5] are volumetric extensions of the MEI and MHI templates, which enhances

the original MEI and MHI templates by adding robustness towards viewpoint variations. Further, a Motion Context (MC) feature [6] is proposed based on Scale-Invariant Feature Transform (SIFT) and the location of the human body obtained using polar coordinates in MHI. Subsequently, 3D-HOG [7], a spatiotemporal extension of the Histogram of Oriented Gradients (HOG) [8] is proposed which shows promising results among methods in this category. Though progress has been made within this category, methods with spatiotemporal volume-based features require clear human contour, and would not work well under complex scenes or if the human body is occluded.

STIP-based Features. STIP-based features used in vision-based action recognition are heavily inspired by similar features used in image-based tasks. Action recognition methods using STIP-based features are therefore mostly extended from image-based methods. The key for extracting robust STIP-based features lies in extracting accurate "Interest Points" [9], which are generally referred to as the position that changes significantly across the spatiotemporal dimension [10]. One popular idea extends the Harris corner detector [11] to 3D-Harris detector [12, 13], which identifies points with significant spatial variations and varied motions. Similarly, the Hessian detector [14] used in image interest point detection is also extended to the temporal dimension, and the 3D-Hessian detector [15] is proposed for interest point detection across spatiotemporal dimensions. Upon the detection of the interest points with the 3D interest point detectors, the feature descriptors are computed and a dictionary that represents the actions is finally learnt. More recently, the STIP-based features are fused in a hybrid manner [16] and are encoded with super vector-based encoding methods (e.g. Fisher Vector [17] and Super Vector Coding [18]), achieving better performance than previous methods in several vision-based action recognition benchmarks. Further, the 3D-Harris interest point detector is also combined with 3D Scale-Invariant Feature Transform descriptor for action feature representation [19], with action videos represented by the traditional Bag-of-Visual Words (BOVW) approach, rendering satisfactory results in early benchmarks. Compared to earlier methods utilizing spatiotemporal volume-based features, methods with STIP-based features do not require the pre-processing of videos which include but not limited to background segmentation and human detection, and are more robust to scale and rotation variance. However, these methods are still sensitive to the change of camera view and background motion.

Trajectory-based Features. Trajectory-based features are constructed by utilizing the path of key points (i.e. joints of the human skeleton). The seminal work for this category

is the improved Dense Trajectories (iDT) proposed in [20], which integrates features such as HOG and the Motion Boundary Histogram (MBH) [21], with the trajectories extracted through densely sampled key points utilizing optical flow. Subsequently, a number of works have been proposed with the goal of improving iDT or further incorporating iDT with other features. One example is the SDT feature proposed in [22] which introduced a spectral divisive clustering algorithm to extract local trajectories at different motion hierarchies. Further improvements of iDT are introduced in [23] through explicit camera motion estimation along with a human detector, which improves over iDT on its robustness towards camera movement and background trajectories. Meanwhile, the Stacked Fisher Vectors (SFV) [24] is proposed by stacking multiple layers of Fisher Vectors obtained on top of iDT features. Compared to previous methods, methods using trajectory-based features are robust towards camera view change, yet these methods require accurate human skeleton models and the tracking of the joints in the skeleton models, which requires a much higher computation cost.

2.1.2 Deep Learning Methods

Inspired by the biological neural systems [25], deep learning methods based on Deep Neural Networks (DNNs) have been proposed to extract data features in an automatic manner. Among the various types of DNNs, Convolutional Neural Networks (CNNs) yield exceptional performances in image-based tasks, such as image classification and object detection. CNNs consist of multiple layers which include convolution, pooling, activation, and fully connected layers. One of the pioneering works is LeNet [1] as shown in Figure 2.1, which has been successfully applied to hand-written digit classification. Over the past few years, deeper and more complex CNNs [26–29] have been subsequently introduced and achieved satisfying results on large-scale datasets such as ImageNet [30].

As videos could be viewed as a collection of multiple images placed sequentially across time, the success of CNNs in image tasks could also be extended to videos. From the type of CNNs utilized, deep learning methods for vision-based action recognition could be generally grouped into two categories: 2D-CNN based methods and 3D-CNN based methods.

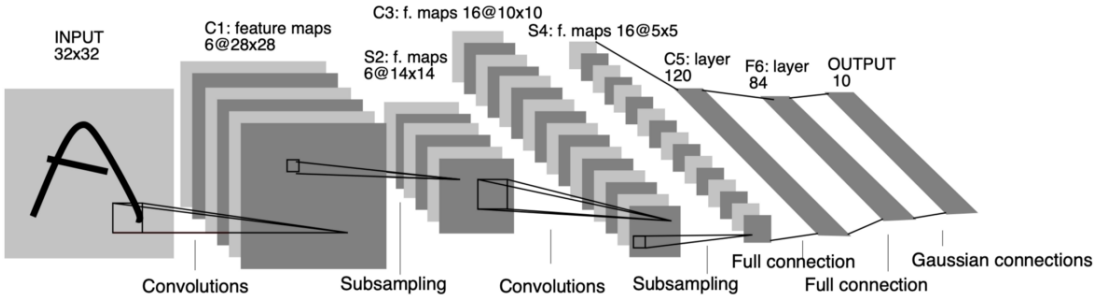


FIGURE 2.1: Structure of LeNet. (Figure source: Lecun et al. [1])

2.1.2.1 2D-CNN based Methods

Early deep learning methods for action recognition, such as those proposed in [2, 31] utilize 2D-CNNs to extract features from videos. In both methods, a video frame is sampled and used as input to a 2D-CNN for feature extraction. Apparently, the feature of a single sampled frame could at most represent the feature of the video across the spatial dimensions, and would not be able to represent video information along the temporal dimension. Previous researchers suggest that human would also process videos in a two-stream manner: the Ventral Stream processes object attributes such as object appearances and object colors; while the Dorsal Stream processes the motions and locations of the object [32]. Inspired by such research, a separate stream is added to extract temporal features embedded in videos, utilizing optical flow [2, 31]. More specifically, optical flow frames are computed using algorithms such as TV-L1 [33] and are stacked up along the channel dimension, forming the input of the temporal stream for these methods. The structure of the spatial stream and the temporal stream is usually similar or even identical (e.g., in [2]), yet they are trained separately, and their parameters are not shared. To obtain the overall classification results, the softmax scores of each individual stream are combined with a late fusion strategy. A typical structure of the two-stream network is presented in Figure 2.2.

Subsequently, multiple networks have been proposed to improve on the early two-stream networks. One improvement concerns the fusion strategy between the spatial and temporal streams [31], where the features extracted from each stream are fused before obtaining the softmax scores, forming the overall feature of the video. The overall feature is then classified to obtain the action of the video. Another simple improvement is proposed in ST-ResNet [34], where the ResNet [28] is used as the 2D-CNN backbone for extracting both spatial and temporal features. The Temporal Segment Network

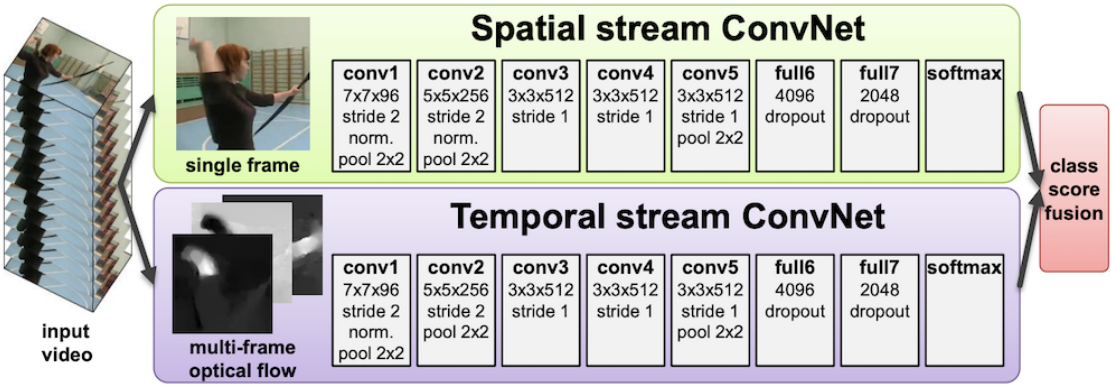


FIGURE 2.2: Typical structure of a two-stream network. (Figure source: Simonyan et al. [2])

(TSN) [35] is another improvement over the early two-stream networks by segmenting videos into clips, while the spatial and temporal features are extracted for each clip and the overall video feature is obtained through segmental consensus fusion. Subsequently, DOVF [36] extends from TSN by employing a two-stage classification strategy. More recently, TRN [37] is proposed and outperforms TRN through segmenting videos into clips of multiple temporal scales, while concatenating the features of each temporal frame for obtaining the features of each clip. Alternatively, handcrafted features such as iDT are aggregated with two-stream networks to achieve better performance [38].

Though the utilization of optical flow produces high-quality temporal features and therefore improves the overall video feature, computation of optical flow requires high computational power and large storage resource. Additionally, optical flow requires precomputation, therefore using optical flow for temporal feature extraction prohibits fully end-to-end training of the network. To address such limitations, subsequent works propose to estimate optical flow through a neural network that renders networks to be trained in an end-to-end manner. FlowNet [39] which learns optical flow from synthetic ground truth data is one example. Subsequently, MotionNet [40] estimates optical flow through the prediction of successive frames, LMoF [41] improves optical flow estimation robustness over blur videos through constructing a learnable directional filtering layer, TVNet [42] unfolds the TV-L1 [33] algorithm and formulates it with a neural network. More recently, Rep-Flow [43] further extends TVNet by constructing a convolutional layer for optical flow estimation. Even though the above methods enable networks utilizing optical flow for temporal feature extraction to be trained end-to-end, these methods are still intensive in computation power and memory requirement.

Meanwhile, there are also other methods built based on 2D-CNNs while avoiding the use of optical flow. One category involves the use of Recurrent Neural Networks (RNNs) and their variants for modeling temporal features in the form of sequence information. One seminal work is the Long-term Recurrent Convolutional Network (LRCN) proposed in [44], which extracts features of each frame with 2D-CNNs, while the overall video feature is modeled through a Long Short-Term Memory [45] (LSTM)-style RNN. LSTM is employed due to the fact that compared to vanilla RNN, LSTM is more capable of modeling long sequences. A similar approach is proposed in [46], where the features of each frame are extracted with 2D-CNNs, while the overall feature is obtained by a combination of a feature pooling layer and LSTM. More recently, Shi *et al.* proposed ShuttleNet [47] which is constructed by Gated Recurrent Units [48] (GRU), a variant of RNN, that are loop connected. Even though these methods could model video features without requiring the estimation of optical flow, RNNs are prone to vanishing or exploding gradient problem, while the training of RNNs is slow and complex.

2.1.2.2 3D-CNN based Methods

A more direct approach towards applying deep learning to vision-based action recognition is to extract temporal features jointly with spatial features, applying convolutional operations on both spatial and temporal dimensions. To achieve such operations, the 3D-CNN is first introduced in [49], which is constructed with 3 convolutional layers, 2 subsampling layers, and 1 fully connected layer, performed on videos pre-processed with a hardwired layer. The convolutional kernels used in [49] are 3D kernels, where the filters are extended along the temporal dimension. Empirically, 3D-CNN outperforms 2D-CNN without optical flow by a noticeable margin. Subsequently, a slow fusion strategy is proposed in [50] that fuses video features obtained from multiple clips progressively utilizing 3D-CNN for feature extraction of each clip. Further, C3D [51] is introduced as a generic video feature extractor, with full video frames as input while employing $3 \times 3 \times 3$ homogeneous convolutional kernels. C3D is a much deeper network compared to previous methods and outperforms them significantly. With larger and deeper networks such as VGG [26], ResNet [28] and ResNext [29] introduced and achieved outstanding performances, this progress is also employed in 3D-CNNs for vision-based action recognition. I3D [52], 3D-ResNet [53] and 3D-ResNext [54] are built by expanding the convolutional kernels of their 2D-CNN counterparts to the temporal dimension and are all deeper and larger 3D-CNNs compared to C3D.

Though 3D-CNNs are able to model temporal features with spatial features jointly, avoiding the use of optical flow, their parameter size is much larger than their 2D counterparts. The larger parameter size results in increase computation and difficulty for training. To address such issues, I3D [52] propose to initialize 3D-CNNs by inflating weights of 2D-CNNs trained for the image classification task. Meanwhile, R(2+1)D [53] propose to improve 3D-CNN efficiency by separating spatial convolution operation with temporal convolution operation. This is achieved by splitting the 3D convolutional kernels into 2D convolutional kernels for spatial convolution and 1D convolutional kernels for temporal convolution. This strategy is shared by S3D [55] and P3D [56]. Alternatively, Channel-Separated Convolutional Network (CSN) [57] shows that the efficiency and effectiveness of 3D-CNNs could also be improved by performing convolution operation across channels separately. MFNet [58] shares such a strategy, while including multiplexer modules to facilitate information flow across channels. More recently, SlowFast Network [59] is proposed to include a slow and fast pathway for modeling spatial and temporal semantics separately. Both pathways are constructed with 3D convolutional kernels, with the fast pathway being lightweight by using a fraction of channels, and the slow pathway requiring less computation by using a fraction of temporal frames.

In general, 3D-CNNs benefit from end-to-end training and requires only RGB input. However, the overall temporal feature extracted by 3D-CNNs is usually obtained through pooling along the temporal dimension, and thus only linear temporal feature is preserved. This results in partial loss of temporal information during feature extraction and ultimately results in inferior performances for some 3D-CNNs when compared to methods utilizing optical flow. The issue of extracting effective temporal features while excluding optical flow estimation inspires our work in Chapter 3.

2.1.2.3 The Non-Local Block

It can be observed from our review in Section 2.1.2.2 that previous 3D-CNNs did not address spatiotemporal correlation features embedded within videos. Therefore, one key approach to further improve the ability of modeling video features by 3D-CNNs is to incorporate spatiotemporal correlation features within the feature extraction network.

One typical form of spatiotemporal correlation is long-range spatiotemporal dependencies. Inspired by the non-local means for the image filtering task [60, 61], the Non-Local neural network [62] (NLNet), equipped with Non-Local block (NLBlock), is proposed to

capture long-range dependencies directly. NLBlock captures spatiotemporal long-range dependencies through direct modeling of the correlations of every single pixel at any spatiotemporal location. Without bells and whistles, the insertion of NLBlocks significantly improves the performance of vision-based action recognition for existing networks. Subsequently, multiple variants of the NLBlock have also been introduced to further improve the effectiveness of captured spatiotemporal long-range dependencies. Among these improvements is Compact Generalized Non-Local operation [63] (CGNL), which exploits cross-channel correlations on top of the original non-local block. Meanie, the Double-Attention module [64] computes correlations of features from a compact bag. More recently, Global Context (GC) block [65] is proposed as a lightweight alternative to the NLBlock by introducing a query-independent attention map, and combining it with squeeze-excitation block [66]. Despite the above variants improve from NLBlock, one constraint of these methods concerns the fact that long-range spatiotemporal dependencies are all extracted on the pixel level, while it is more intuitive to recognize actions by focusing on correlations between regions. In other words, these methods have not considered the use of spatiotemporal correlations at the regional level. This issue motivates the work in Chapter 4.

2.2 Overview of Benchmark Datasets

The development of various benchmark datasets is a key driving force for the rapid progress of deep learning methods in vision-based action recognition under both supervised and cross-domain settings. In this section, we present benchmark datasets for vision-based action recognition, as well as dark visual benchmark datasets.

2.2.1 Vision-based Action Recognition Benchmark Datasets

Vision-based action recognition aims to utilize videos for recognizing actions. To evaluate the performance of the various methods as presented in Section 2.1, a number of datasets are established. Earlier vision-based action recognition benchmark datasets include KTH [67], Weizmann [68], and IXMAS [69]. In general, these datasets contain a relatively small number of action classes, while datasets such as KTH, Weizmann, and IXMAS are collected offline without using public available videos. The KTH [67]

dataset contains 6 different categories, while each category contains 4 background scenarios with 25 actors, resulting in a dataset with a total of 2,391 single-person videos shot under 25 Frames per Second (FPS) and a single resolution of 160×120 pixels. The Weizmann dataset [68] is similar to KTH in that videos in Weizmann are also single-person videos, yet the Weizmann dataset contains more action categories with less video. In total, the Weizmann datasets consisted of 90 videos under 10 categories, with all videos shot under a single resolution of 180×144. One common constraint concerning both datasets is the fact that both datasets are based on a single camera shot under a single viewpoint. Meanwhile, the IXMAS dataset [69] addresses such constraint by constructing a multi-view action dataset taken from five cameras at different angles. The entire IXMAS dataset contains 1,148 videos, divided into 11 actions.

Performances on these previous datasets are mostly saturated, partly due to their small scale. Larger datasets such as Hollywood2 [70], Olympic Sports dataset [71], HMDB51 [72], UCF50 [73] and UCF101 [74] are introduced, where videos are collected from public video platforms such as YouTube. More specifically, videos in the Hollywood2 dataset [70] are collected from 69 Hollywood movies by means of automatic script-to-video alignment. The dataset is composed of 3,669 videos categorized into 12 action classes. The Olympic Sports dataset [71] contains sports videos from 16 sport categories, where all videos are obtained from YouTube. HMDB51 [72] is an even larger dataset collected from various video platforms including YouTube and Google, containing 7,000 videos under 51 action categories. HMDB51 is more challenging compared to all previous datasets, thanks to the fact that videos in HMDB51 tend to contain more complex backgrounds, while similar scenes may appear in different actions. HMDB51 is still considered a challenging benchmark dataset and progress has been made constantly throughout the past decade. The UCF50 dataset [73] is similar in both its collection method and scale compared to HMDB51, with 6,676 videos in 50 categories. The UCF101 [74] is the successor of the UCF50 dataset, extending the number of action classes from 50 to 101, and contains a total of 13,320 videos which is the largest dataset at the time of its introduction. The 101 action classes are further divided into 5 categories: human-object interaction actions, body-motion only actions, human-human interaction actions, playing musical instruments, and sports.

More recently, the Something-Something (V1/V2) dataset [75] is introduced as a large-scale dataset, with a total of 174 classes, its first version (V1) contains 108,499 videos while its second version (V2) contains 220,847 videos. The Something-Something

dataset is created by a large number of crowd workers, and its action categories are more fine-grained (e.g. "Pushing something from right to left" instead of "Pushing"). The introduction of the Something-Something dataset allows methods to develop a fine-grained understanding of actions. At the same time, the Kinetics dataset [76] with 400 action classes and over 160,000 videos all collected from YouTube is one of the largest datasets in terms of the number of action categories. It has become the primary choice in action recognition studies thanks to its scale. The introduction of these larger datasets help pushed the boundaries of vision-based action recognition, and lead to the introduction of more sophisticated models. However, large datasets such as Kinetics requires large storage and are time-consuming without powerful computation tools (i.e. GPUs). To address such limitations, the MiniKinetics [55] dataset is introduced with halved action categories (i.e. 200 classes) and a total of 80,000 videos for training and 5,000 videos for validation.

Although the datasets as reviewed above involve an abundant scale of different action categories, these actions are mostly collected from online platforms thanks to the ease of obtaining web videos through mass downloading. These web videos are mostly recorded under normal illumination, limiting the capability of current vision-based action recognition methods to videos with normal illumination. This limitation motivates the work in Chapter 5.

2.2.2 Dark Visual Benchmark Datasets

Recently, there has been a rapid increase of research interest with regards to computer vision tasks in adverse environments, such as face recognition in the dark [77–79]. The rise in research for visual tasks under dark environments is partly supported by the various dark visual datasets introduced. Among these, most datasets focused on image enhancement and denoising tasks, where the goal is to visually enhance dark images for a clearer view. These include the LOw Light paired (LOL) dataset [80], ReNOIR [81], See-in-the-Dark (SID) [82], and Exclusively Dark (ExDARK) [83]. Generally, images in these datasets are shot in a low illumination environment, where it is difficult for the human naked eye to observe and identify objects. More specifically, LOL dataset [80] contains 500 low/normal-light image pairs, where low-light images are collected by alternating exposure time and ISO values. ReNOIR [81] is constructed in a similar manner with

normal/low-light image pairs, shot under 120 scenes with 2 normal images and 2 low-light images acquired for each scene. The low-light images in the two datasets above are collected in normal conditions and these images are acquired by tuning camera parameters. Alternatively, the SID [82] dataset collects raw low-light images collected under dark environments, while the corresponding reference images obtained by extending the exposure time to 10 to 30 seconds. The SID dataset contains 5,094 raw low-light images each with a corresponding reference image. More recently, the ExDARK [83] dataset is introduced with a larger scale (7,363 low-light images) with both image class level and local object annotations. It includes 10 types of low-light conditions and contains 12 different types of objects, which results in a total of 23,710 object instances annotated.

More recently, the task of low-light enhancement has been further expanded to the video domain, where the goal is to process dark videos for clearer video frames with higher visibility. Relevant datasets include the Dark Raw Video (DRV) [84] and the See-Moving-Objects-in-the-Dark (SMOID) [85] datasets. The collection method for the DRV dataset [84] is similar to that of the SID dataset, and is the first public dataset with real-world low-light raw videos. It contains a total of 202 low-light raw videos. The reference of the raw videos is set to be the long-exposure image of the videos' first frames. Due to the requirement of a reference image, most videos captured for the DRV dataset are shot in static scenes without or with little motion. Meanwhile, the SMOID dataset [85] collects 179 normal/low-light video pairs of street views with moving vehicles and pedestrians. The video pairs are collected simultaneously via synchronized cameras built into an integrated camera system. The low-light videos are captured by a Neutral Density (ND) filter which is able to diminish light intensity. Although the above datasets are both datasets for dark videos, their research focus is more on enhancing the visibility of video frames. Meanwhile, it can be observed that clearer video frames may break the original pixel distribution, thus there is no guarantee that clearer videos would result in higher action recognition accuracy. Furthermore, the scenes where these videos are collected are either static scenes or street views, which do not include specific human actions, thus are not suitable for the task of vision-based action recognition in dark videos. Such limitations further inspire our work presented in Chapter 5.

2.3 Overview of Video Unsupervised Domain Adaptation

A majority of current vision-based action recognition methods assume that the distribution of the testing data is in line with that of the training data. However, in real-world applications where labels may not be sufficiently available (e.g. vision-based action recognition in dark videos), such assumption may not be fully applicable. Significant decreases in methods' performances are observed when applying methods directly from one domain (e.g. normal-illuminated videos) to another (e.g. dark videos), which are the results of the different data distributions across different domains, known as domain shift. To address domain shift, Domain Adaptation (DA) aims to adapt models to the label-scarce target domain (denoted as Supervised Domain Adaptation (SDA)) or unlabeled target domain (denoted as Unsupervised Domain Adaptation (UDA)) utilizing labeled source domain. In this thesis, we mainly focus on Video Unsupervised Domain Adaptation (VUDA). We begin this section with the learning theory and general approaches of UDA, followed by a brief review of current VUDA methods. Last but not least, cross-domain video benchmark datasets used for evaluating VUDA approaches in the task of cross-domain action recognition are also reviewed and discussed.

2.3.1 Learning Theory and General Approaches of Unsupervised Domain Adaptation

In general, the goal of deep learning-based domain adaptation methods is to minimize the discrepancy between source and target domains. The theoretical upper bound for the expected error of the target samples $\epsilon_T(h)$ in the learning theory of domain adaptation is derived in [86] and formulated as follows:

Theorem 2.3.1. Let $\mathcal{H}$ be a hypothesis space, and $\mathcal{U}_S$, $\mathcal{U}_T$ be samples drawn from distributions p_S of source domain and p_T of target domain respectively. For hypothesis $h \in \mathcal{H}$, the expected error of the target samples is bounded by:

$$\epsilon_T(h) \leq \epsilon_S(h) + \frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T) + \boldsymbol{C}_0 + \lambda, \tag{2.1}$$

where $\epsilon_S(h)$ denotes the source error while $\frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T)$ is the empirical $\mathcal{H}\Delta\mathcal{H}$-divergence on samples $\mathcal{U}_S, \mathcal{U}_T$ drawn from distributions p_S, p_T. λ is the error of an ideal

hypothesis for both source and target domains, while $\boldsymbol{C}_0$ is a constant term determined by the complexity of the hypothesis space $\mathcal{H}$.

Theorem 2.3.1 suggests that the target error should be bounded by a combination of the source error, a discrepancy term, the complexity of the hypothesis space, as well as the error of the ideal hypothesis. For a fixed set of source data, the source error is fixed. Similarly, for a fixed model, the complexity of its corresponding hypothesis space is also fixed. Meanwhile, the error of an ideal hypothesis for both domains is considered to be small and negligible. Therefore the main factor that impacts the target error is the discrepancy term that computes the discrepancy between the distribution of source and target domain data. To minimize the discrepancy term for minimal target error, current general approaches could be mainly split into two categories: adversarial-based approaches and global statistic-based approaches.

Adversarial-based Domain Adaptation. Motivated by the success of the Generative Adversarial Network (GAN) [87], the adversarial-based domain adaptation approaches perform domain adaptation utilizing additional domain discriminators that are trained with the feature generators in an adversarial manner [88], and construct adversarial losses for UDA [89]: the domain discriminators learn to discriminate source features from the target features, whereas the feature generators learn to deceive the domain discriminators. Adversarial-based domain adaptation methods were first introduced in Domain-Adversarial Neural Network (DANN) [89, 90], where the adversarial training of domain discriminator and feature generators are achieved by applying a Gradient Reversal Layer (GRL) on top of the domain discriminator. Such domain adaptation method is attractive due to its excellent alignment performance and the flexibility of the GRL, which could be utilized in a plug-and-play manner. The success of DANN motivates many subsequent UDA methods. Among these works, PixelDA [91] is introduced to leverage GAN for both feature-level and pixel-level domain adaptation, which results in notable improvements. Similarly, CyCADA [92] also leverages pixel-level domain adaptation, while incorporating cycle loss over source data as well as semantic consistency losses over both source and target domains. The additional losses yield significant improvements in domain adaptation, however, it is achieved at a cost of high network complexity. Zhang et al. proposed a novel Margin Disparity Discrepancy (MDD) [93] for distribution comparison based on minimax optimization utilizing GRL. To further

improve the effectiveness of DANN, a discriminator gate [94] is proposed to filter potential negative transfer by reducing the bias between the source and target risks. Meanwhile, Adversarial Discriminative Domain Adaptation (ADDA) [95] unties the process of feature representation learning and domain discrimination. ADDA performs discriminative representation learning first followed by the mapping of the target data to the learnt representation space by asymmetric mapping learnt through domain adversarial loss. ADDA thus performs domain adaptation in a non-end-to-end manner. A similar strategy is adopted for Consensus Adversarial Domain Adaptation (CADA) [96], while CADA improves on ADDA by fine-tuning the trained source feature generator simultaneously with the training of the target feature generator during the adversarial learning. In this manner, CADA enables both the target data and source data to be embedded to a common domain-invariant feature space defined by both source and target domains. More recently, to tackle the uncertainty issue brought by adversarial domain adaptation to feature learning, Asymmetric Adversarial Domain Adaptation (AADA) [97] is proposed where a novel asymmetric learning scheme is designed to fix source features while encouraging target features to approach the fixed source features, which is achieved by an autoencoder-based domain discriminator that embeds source domain only.

Global Statistics-based Domain Adaptation. Besides adversarial-based domain adaptation approaches, global statistics-based domain adaptation approaches are the other major category of domain adaptation approaches, where they aim to measure the discrepancy between source and target domains by statistics, and the computed discrepancy is minimized to minimize target error. One seminal work under this category is the Maximum Mean Discrepancy (MMD) [98], which has proven its effectiveness in the two-sample testing. The two-sample testing concerns the probability distributions p_S and p_T for source and target domains respectively, where acceptance or rejection decisions are made for a null hypothesis $p_S = p_T$, given source and target data samples. MMD has been utilized by Pan et al. in transfer component analysis [99] which learns transfer components across the source and target domains in a Reproducing Kernel Hilbert Space (RKHS). The multiple kernel variant of MMD denoted as MK-MMD, is proposed in [100], and is formalized to jointly maximize the two-sample testing power while minimizing the error caused by rejecting a false null hypothesis. Subsequently, Long *et al.* proposed Deep Adaptation Network (DAN) [101] which explores the idea of MK-MMD for learning transferable features in deep networks. Meanwhile, Statistically Invariant Sample Selection (SISS) and Statistically Invariant Embedding (SIE) [102] are

proposed to measure the discrepancy between source and target domains on the Riemannian manifold, while CORAL [103] and Deep CORAL [104] are proposed to measure discrepancy through measuring the difference in feature covariance matrices.

2.3.2 Current Video Unsupervised Domain Adaptation methods

In recent years, there has been a rapid rise of research interest in domain adaptation, especially image-based domain adaptation, where the methods described above (e.g. DANN [89, 90]) have been applied to various image-based tasks, including but not limited to image recognition [105–107], object detection [108–110], and semantic segmentation [111–113]. Despite the remarkable progress made in image-based domain adaptation, there have been few works on VUDA. One major factor for the lack of relevant research is the fact that videos contain data with more modalities compared to images, which includes both temporal and spatiotemporal correlation features, thus complicating the overall adaptation process. Recently, Jamal et. al introduce Action Modeling on Latent Subspace (AMLS) and Deep Adversarial Action Adaptation (DAAA) [114] for VUDA. In particular, the AMLS models target domain videos as a sequence of points on a latent subspace while adaptive kernels are learned between source and target domain points on the latent subspace, while DAAA is an end-to-end adversarial-based framework that adapts adversarial-based domain adaptation methods to videos by utilizing 3D-CNNs as feature generators.

Subsequent VUDA approaches improve on DAAA by focusing on improving source and target video alignment along the temporal direction. TA^{3}N [115] is proposed to align video features along the temporal direction by applying attention mechanisms to video segments sampled across the temporal direction, whose features are extracted with TRN [37], thus resulting in the dynamic alignment of the different video segments. Similarly, Temporal Co-attention Network (TCoN) [116] is further proposed to align the distributions of video features using a novel cross-domain co-attention mechanism across the temporal dimension. Meanwhile, SAVA [117] is proposed to align video feature by utilizing the auxiliary task of clip order prediction [118]. Though there has been certain progress in VUDA, the above methods failed to explore the alignment of spatiotemporal features embedded in videos, which is also a key element in the robust modeling of videos. Such limitation inspires our work in Chapter 3. Furthermore, the above VUDA

approaches all assume that source and target domains share the same label space, which may not apply in real-world applications, given the fact that domain adaptation is often used in scenarios where the source label space subsumes the target one. This further limitation further motivates our work in Chapter 7.

2.3.3 Cross-Domain Video Benchmark Datasets

Besides the complexity of aligning video data of the different domains, the lack of VUDA research is also partly contributed by the fact that there are very limited cross-domain benchmark datasets available. The two most widely used cross-domain video benchmark datasets are the UCF-Olympic [119] dataset, built across UCF50 [73] and the Olympic Sports dataset [71]; and the UCF-HMDB$_{\text{small}}$ [119] dataset, built across UCF50 [73] and HMDB51 [72]. Both cross-domain datasets are of very small scale. More specifically, the UCF-Olympic [119] dataset is constructed by including 6 action classes, with a total of 841 videos from UCF50 and 304 videos from the Olympic Sports dataset. The UCF-HMDB$_{\text{small}}$ [119] dataset are constructed from 5 action classes of both UCF50 and HMDB51 datasets, containing 671 videos from UCF50 and 500 videos from HMDB51. To further facilitate research on VUDA, larger cross-domain video datasets are introduced, with UCF-HMDB$_{\text{full}}$ [115] being the most commonly used benchmark dataset. Similar to the UCF-HMDB$_{\text{small}}$ dataset, it is also built across UCF50 and HMDB51, yet it includes 12 classes, which is more than doubled that of the UCF-HMDB$_{\text{small}}$ dataset. This dataset incorporates 2009 videos from UCF50 and 1200 videos from HMDB51. A more detailed comparison of the above cross-domain datasets is presented in Table 2.1. One common limitation across the above cross-domain datasets lies in the fact that both domains included are based on current well-established vision-based action recognition datasets, whose videos are of similar video statistics due to their common video collection method (i.e. through public video platforms). Therefore it can be argued that domain shift across domains may not be remarkable. In this thesis, we seek to tackle such limitations through the introduction of larger cross-domain video benchmark datasets with more significant domain shifts in both Chapter 6 and Chapter 7.

Statistics	*UCF-HMDB$_{small}$*	*UCF-Olympic*	*UCF-HMDB$_{full}$*
Video Length (seconds)	1-21	1-39	1-33
Video Classes #	5	6	12
Training Video #	UCF:482/HMDB:350	UCF:601/Olympic:250	UCF:1438/HMDB:840
Validation Video #	UCF:189/HMDB:150	UCF:240/Olympic:54	UCF:571/HMDB:360

TABLE 2.1: Comparison of current cross-domain video benchmark datasets.

2.4 Conclusion

In this chapter, we have surveyed vision-based action recognition methods, video unsupervised domain adaptation approaches, and various benchmark datasets. For vision-based action recognition methods, both handcrafted feature-based methods and deep learning methods are reviewed. The limitation of current deep learning methods includes the requirement of optical flow estimation through handcrafted feature or neural network for 2D-CNNs, potential partial loss of temporal information for 3D-CNNs, and a lack of efficiency for the Non-Local Blocks. These limitations motivate the introduction of more robust and efficient temporal feature and spatiotemporal correlation feature extraction methods. Meanwhile, a review of the current vision-based action recognition benchmark dataset and dark visual datasets reveals the lack of sufficient data for vision-based action recognition in dark videos, which inspires the introduction of such a dataset in our work. To tackle the lack of sufficient labeled data in dark videos further motivates the need of applying VUDA approaches for transferring models trained on current vision-based action recognition datasets to dark videos. To deal with this need, domain adaptation approaches, especially video unsupervised domain adaptation approaches have been reviewed. There are currently very limited VUDA approaches, due to the complexity of video data compared to image data and a lack of cross-domain video datasets. Current VUDA approaches also lack the ability to effectively align source and target video data due to their failure of exploiting spatiotemporal correlation features for alignment. These limitations have been pointed out in this chapter, which inspires the subsequent video domain adaptation research in this thesis.

Chapter 3

Exploiting Inter-Frame Regional Correlation for Efficient Vision-based Action Recognition

One of the key factors for robust and efficient vision-based action recognition lies in the extraction of temporal features in videos. In this chapter, we propose a novel temporal feature extraction method for efficient vision-based action recognition. The motivation and introduction are presented in Section 3.1, followed by a detailed introduction over the proposed temporal feature extraction method: Extraction of Attentive Correlated Temporal Feature in Section 3.2. Then the experiments and evaluations are conducted in terms of vision-based action recognition accuracy, as described in Section 3.3. We conclude this chapter in Section 3.4.

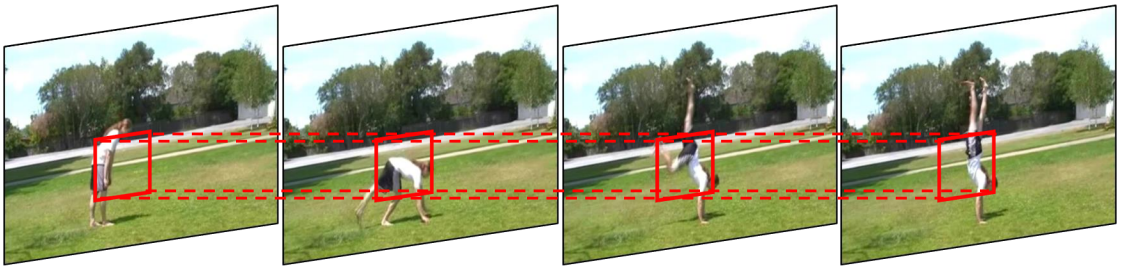
3.1 Introduction

Action recognition has received considerable attention from the vision community in recent years [120–124] thanks to its increasing applications in various fields, such as surveillance [125–127] and smart homes [128, 129] etc. Compared with static images, videos contain additional temporal information. Hence, extracting and handling temporal information is very critical in action recognition.

To extract temporal features underlying a video, a few methods have been proposed in the literature. Most of these efforts can be organized into two categories. The first category is the two-stream methods. A typical work in this category is the one proposed in [2], which conduct the classification using temporal features and spatial features separately. The two types of features are integrated through classification decision fusion. The second category is the 3D-CNN based methods, which extract spatial and temporal features jointly by expanding the convolution kernel of 2D-CNNs to the temporal dimension. A seminal work in this category is the C3D [51] network.

The methods in the two categories have their respective merits and limitations. The two-stream methods often produce the state-of-the-art performance, yet this is achieved at the cost of heavy reliance on accurate temporal feature. Therefore, the two-stream methods usually involve computation or estimation of optical flow, both of which require high computational power and large storage resource. Also, obtaining optical flow needs to be performed prior to the training of the network, thus methods utilizing optical flow cannot be trained end-to-end. On the other hand, the 3D-CNN based methods are computationally less demanding, yet their performances are usually inferior to that of the two-stream methods. A possible reason would be the temporal pooling used for dimension reduction towards the complete representation. Temporal pooling extracts only linear feature along the temporal dimension of the video through pooling operation. With only the linear feature being extracted, we argue that part of the temporal feature is lost during the pooling operation.

In this chapter, we present a novel method for temporal feature extraction, which achieves performance comparable to or even better than two-stream methods, yet demands less computational power. Intuitively, temporal feature of an action is related to the correlation of appearance between frames within a certain region. For instance, in Figure 3.1a, the indicated box across the series of frames shows how a person turns upside down, and is related to the action of ”Handstand”. Therefore, instead of using optical flow, our proposed method extracts temporal features by extracting the correlation of neighbouring frames with respect to the corresponding regions. The degree of change of appearance varies between different actions. For actions that are slower or more static, neighboring sampled frames could be very similar. One example is the action of ”Brushing Teeth” shown in Figure 3.1b. If linear correlation, such as the difference in RGB value is employed, the correlation extracted would fail to contain temporal information of the video.



(A) Action "Handstand"

(B) Action "Brushing Teeth"

FIGURE 3.1: Illustration of extracting inter-frame corresponding-regional correlation for action recognition. The temporal feature of an action is related to the correlation appearance between frames. Actions that are faster such as "Handstand" in (a) exhibits obvious change within the indicated box. Slower and more static actions such as "Brushing Teeth" in (b) shows little change between frames. To cope with both situations, bilinear operation is employed to extract the inter-frame corresponding-regional correlation

To cope with the various type of actions, the inter-frame correlation would thus be computed through bilinear operation. The complete temporal feature, named as Attentive Correlated Temporal Feature (ACTF), is obtained through attentive combination of the inter-frame corresponding-regional correlation feature and the inter-frame mean feature obtained through inter-frame temporal average pooling.

Our main contributions are summarized as follows:

* We propose a novel temporal feature extraction method: Attentive Correlated Temporal Feature (ACTF), for action recognition. First, ACTF exploits inter-frame corresponding-regional correlation to implicitly capture temporal information without the use of optical flow. Second, by excluding optical flow estimation or calculation, ACTF can be combined with any spatial feature extraction network under the two-stream structure to implement end-to-end training. Third, ACTF leads to performance comparable to or even better than optical flow-based methods, yet it demands less computation and memory due to the exclusion of optical flow.

* We conduct extensive experiments on two action recognition benchmark datasets: UCF101 [74] and HMDB51 [72] with a framework utilizing our proposed ACTF. The results demonstrate that our proposed ACTF brings noticeable improvements over baseline methods, achieving state-of-the-art performance for these datasets.

The rest of this chapter is organized as follows. In Section 3.2, we introduce the proposed Attentive Correlated Temporal Feature (ACTF) in detail. After that, we present and analyze the experimental results of our proposed ACTF feature, with a thorough ablation study on the design of ACTF. Finally, we conclude the chapter in Section 3.4.

3.2 Method

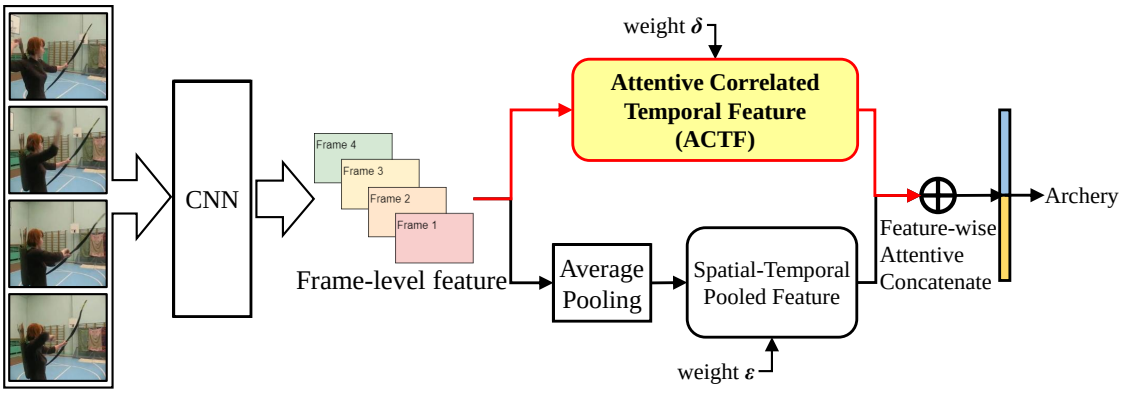


FIGURE 3.2: Detailed illustration of applying ACTF for action recognition. The sharp rectangles represent the networks or operations performed, while the rounded rectangles represent the resulting features. The overall framework takes the raw RGB frames as input. The feature of the RGB frames is extracted through a CNN (CNN). From the frame-level feature, we obtain the spatial-temporal pooled feature of the video through average pooling across both spatial and temporal dimensions. This feature is regarded as the spatial feature of the video. Simultaneously, we obtain the ACTF as the temporal feature of the video. Both features are combined attentively to form the whole representation of the video.

The primary goal of this chapter is to develop an effective video-based action recognition framework with focus on temporal feature extraction. The main idea of the proposed method is to explore correlation of successive frames within a certain region, which captures temporal information. The extracted correlation feature can work with various frame-level feature extraction networks that are normally convolutional neural networks (CNNs), e.g. C3D and 3D-ResNet. These networks normally adopt a simple temporal

pooling operation for obtaining the video representation. We propose an ACTF model to effectively extract the inter-frame corresponding-regional correlation feature and combine it with the feature obtained from simple temporal pooling. Next, we present a general action recognition framework that uses the proposed ACTF for temporal feature extraction, and then describe the details of the ACTF. The attention mechanism employed in ACTF will also be briefly explained.

3.2.1 General Framework for Action Recognition with ACTF

The prominent methods for action recognition employ multiple modality networks, e.g. two-stream convolutional networks [2]. In these networks, temporal and spatial features are extracted and processed separately. Figure 3.2 shows the overall framework in our study. Given an input video as a sequence of frames, the frame-level feature of each frame is first extracted through a convolutional neural network (CNN). The resulted frame-level feature is denoted by $\mathbf{F} \in \mathbb{R}^{t \times C_{out} \times H \times W}$, where t denotes the number of frames, C_{out} denotes the number of channels, and H, W are the height and width. Subsequently, we obtain two features from this frame-level feature, namely the Spatial-Temporal Pooled feature, and the ACTF feature. The Spatial-Temporal Pooled feature $\mathbf{V}_{stpooled}$ is obtained by performing spatial-temporal average pooling over the frame-level feature. The ACTF feature $\mathbf{V}_{actf}$ is obtained through an attentive concatenation of features obtained by performing both bilinear and linear operations on successive frames. Each of the two features characterizes a different perspective of the video. Performing average pooling over the frame-level feature results in a feature that provides a general appearance pattern of the video. Thus the Spatial-Temporal Pooled feature as shown in Figure 3.2 is referred to as the spatial feature of the video in this chapter. Meanwhile, the ACTF feature captures the correlation pattern of successive frames within a certain region, and is referred to as the temporal feature here. Both features have a dimension of C_{out}, *i.e.* $\mathbf{V}_{actf} \in \mathbb{R}^{C_{out}}$, and $\mathbf{V}_{stpooled} \in \mathbb{R}^{C_{out}}$.

In certain scenarios, the spatial feature of the video is sufficient to produce satisfactory action recognition result. This occurs when certain action types are associated with certain visual elements. Meanwhile, in other scenarios, temporal features play a more vital role. This occurs when visual elements of the video may appear in different actions. Thus, we adopt a feature-wise attentive concatenation method to dynamically combine the spatial and temporal features.

3.2.2 Extraction of ACTF Feature

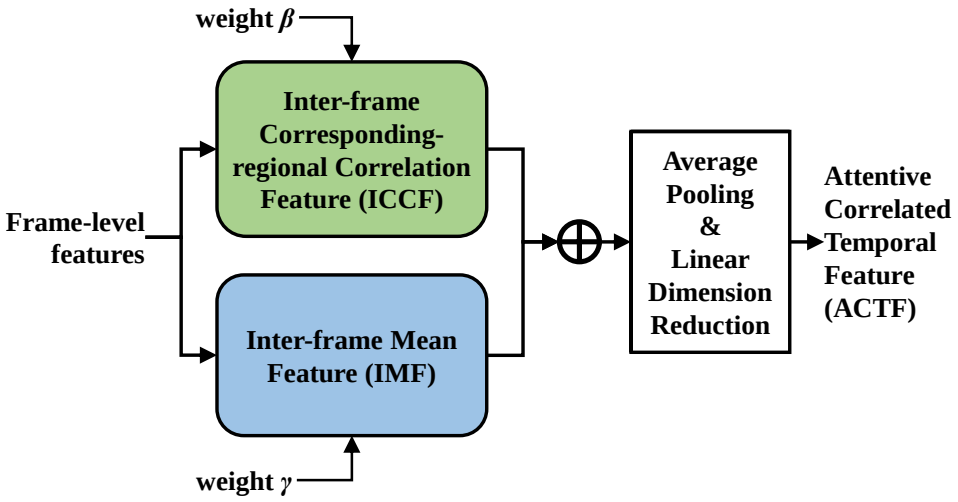


FIGURE 3.3: Illustration of the pipeline for extracting ACTF. From the frame-level feature extracted, we extract two forms of inter-frame correlation features. A bilinear inter-frame correlation feature, extracted as the Inter-frame Corresponding-regional Correlation Feature (ICCF), as well as a linear inter-frame correlation feature, extracted as the Inter-frame Mean Feature (IMF). The features are combined attentively to form the ACTF.

Previous works [2, 35, 130] show the importance of temporal feature in action recognition. However, most temporal information representations, such as optical flow used in two-stream convolutional network [2] or non-local operations in non-local 3D CNNs [62], are computationally expensive. This is due to the fact that both optical flow and non-local operations compute the correlations between successive frames on a pixel level. Computationally efficient RGB difference is employed in [35] to capture inter-frame relation, but shows inferior performance when used in combination with spatial feature. For slower actions where successive frames are very similar, RGB difference would return zero-valued correlation, and fail to capture the temporal feature, which might explain the inferior performance.

In this chapter, we propose to explore more sophisticated operations, such as bilinear function, on successive frames for temporal feature extraction. This is inspired by the bilinear operation used for fine-grained image recognition [131, 132], where the within-image bilinear operation is used to learn local pairwise feature correlation through the outer product at every single position of the image. In this chapter, the bilinear operation is extended across successive frames to discover inter-frame correlation within a certain region. Figure 3.3 shows the pipeline for extracting ACTF.

More specifically, given a video sequence, as described in Section 3.2.1, the frame-level feature of the video is extracted through a CNN, whose output is $\mathbf{F} \in \mathbb{R}^{t \times C_{out} \times H \times W}$. We then extract a bilinear inter-frame correlation feature, the Inter-frame Corresponding-regional Correlation Feature (ICCF), and a linear inter-frame feature, the Inter-frame Mean Feature (IMF). The extraction function for the ICCF is denoted by $\mathcal{P}_{bilinear}$, while the extraction function for the IMF is denoted by $\mathcal{P}_{mean}$.

We first describe $\mathcal{P}_{bilinear}$, which is the extraction function for the bilinear inter-frame correlation feature denoted as ICCF. Figure 3.4 shows the details of extracting the ICCF. Denote $\mathbf{f}_i \in \mathbb{R}^{C_{out} \times H \times W}$ as the frame-level feature extracted for frame i. To extract the bilinear inter-frame correlation feature, $\mathcal{P}_{bilinear}$ computes the pairwise bilinear correlation with respect to two successive frames within a certain region as follows:

$$\mathbf{b}_i = \mathcal{P}_{bilinear}(\mathbf{f}_i, \mathbf{f}_{i+1}). \tag{3.1}$$

Here $\mathbf{b}_i$ is the bilinear inter-frame correlation feature, and $\mathbf{b}_i \in \mathbb{R}^{C_{bilinear} \times H \times W}$, where $C_{bilinear}$ denotes the number of channels of the ICCF.

More specifically, at the spatial location of $\mathcal{S}$, the feature of the current frame and the next frame is denoted as $\mathbf{f}_{i,\mathcal{S}}$ and $\mathbf{f}_{i+1,\mathcal{S}}$. We denote the bilinear operation function at location $\mathcal{S}$ to be $\mathcal{B}_{\mathcal{S}}$, and is formulated by the following equation:

$$\mathcal{B}_{i,\mathcal{S}} = \mathbf{f}_{i,\mathcal{S}} {\mathbf{f}_{i+1,\mathcal{S}}}^T. \tag{3.2}$$

At the spatial location $\mathcal{S}$, the feature for frame i is of size $C_{out} \times 1$. Thus, from Equation 3.2, the bilinear inter-frame correlation feature at location $\mathcal{S}$ is of size $C_{out} \times C_{out}$. We then reshape it such that the result would be of size ${C_{out}}^2 \times 1$.

Although the bilinear inter-frame correlation feature obtained through Equation 3.2 is direct, such feature representation is very high-dimensional. In our case where C_{out} is around 750, the dimension of the bilinear inter-frame correlation feature at each spatial location is more than 500,000. Such high dimensional representation is impractical. Therefore, to obtain the desired bilinear correlation, we adopt a compact form of bilinear operation as implemented in [132].

The basis of the compact form of the bilinear operation is to find a low dimension projection function of $\mathcal{B}_{i,\mathcal{S}}$, denoted as $\mathcal{C}_{i,\mathcal{S}}$. The two functions are equivalent with respect to a linear kernel machine. Given two pairs of frames: frames $(i, i+1)$ and frames $(j, j+1)$,

a linear kernel machine is formulated as:

$$\begin{aligned}\langle \mathcal{B}_{i,\mathcal{S}}, \mathcal{B}_{j,\mathcal{S}} \rangle &= \langle \mathbf{f}_{i,\mathcal{S}} {\mathbf{f}_{i+1,\mathcal{S}}}^T, \mathbf{f}_{j,\mathcal{S}} {\mathbf{f}_{j+1,\mathcal{S}}}^T \rangle \\ &= \langle \mathbf{f}_{i,\mathcal{S}}, \mathbf{f}_{j,\mathcal{S}} \rangle^2.\end{aligned} \tag{3.3}$$

We then find a low dimension projection function as $\phi(\mathbf{f}_{i,\mathcal{S}}) \in \mathbb{R}^d$ such that $\langle \phi(\mathbf{f}_{i,\mathcal{S}}), \phi(\mathbf{f}_{j,\mathcal{S}}) \rangle \approx k(\mathbf{f}_{i,\mathcal{S}}, \mathbf{f}_{j,\mathcal{S}})$, where k is a polynomial kernel. Such projection function $\phi(\mathbf{f}_{i,\mathcal{S}})$ would allow us to approximate Equation 3.3 by:

$$\begin{aligned}\langle \mathcal{B}_{i,\mathcal{S}}, \mathcal{B}_{j,\mathcal{S}} \rangle &= \langle \mathbf{f}_{i,\mathcal{S}}, \mathbf{f}_{j,\mathcal{S}} \rangle^2 \\ &\approx \langle \phi(\mathbf{f}_{i,\mathcal{S}}), \phi(\mathbf{f}_{j,\mathcal{S}}) \rangle \\ &\equiv \langle \mathcal{C}_{i,\mathcal{S}}, \mathcal{C}_{j,\mathcal{S}} \rangle,\end{aligned} \tag{3.4}$$

where $\mathcal{C}_{i,\mathcal{S}} = \phi(\mathbf{f}_{i,\mathcal{S}})$ is the compact form of the bilinear operation $\mathcal{B}_{i,\mathcal{S}}$. Hence to obtain the compact form, we need to find the low dimension approximation of the polynomial kernel k. Here we utilize the Tensor Sketch approximation method proposed in [133]. Ultimately, our extraction function $\mathcal{P}_{bilinear}$ computed at each spatial location $\mathcal{S}$ for frame i and the successive frame $i+1$ is equivalent to its compact form $\mathcal{C}_{i,\mathcal{S}}$.

For image recognition tasks, features extracted through bilinear function go through a sum pooling operation to extract the complete representation of the image. However, if such a pooling method is used in videos, the temporal information may be lost. This conflicts with our goal of extracting temporal information through the bilinear inter-frame correlation feature. To dynamically combine all bilinear inter-frame correlation features temporally, we apply a temporal-wise attentive concatenation to each pair of successive frames. A learnable weight parameter α_i is assigned to each inter-frame correlation feature $\mathbf{b}_i$. Such attentive concatenation allows the extracted ICCF to focus on the pair of frames where the action most likely takes place. The result is a feature $\mathbf{B} \in \mathbb{R}^{(t-1) \times C_{bilinear} \times H \times W}$. For each pair of successive frames, $\mathbf{B}_i = \alpha_i \mathbf{b}_i$.

To extract the temporal feature of the video more accurately, besides the bilinear inter-frame correlation feature, we would also need the linear inter-frame correlation feature denoted as IMF. The IMF provides a baseline for the bilinear inter-frame correlation feature, and is important when actions are similar temporally but very different in appearance. Following this idea, we feed the frame-level feature $\mathbf{F}$ to extract the IMF in parallel with the ICCF.

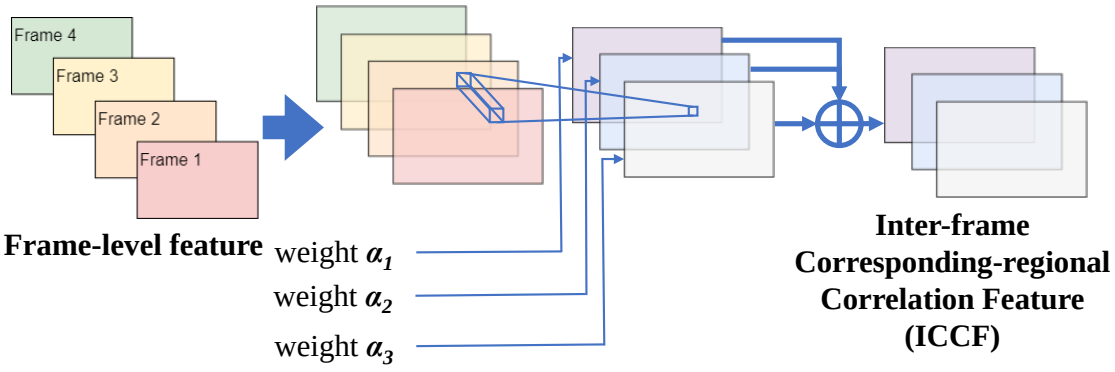


FIGURE 3.4: Illustration of the details for extracting the bilinear inter-frame correlation feature which is the ICCF. The pairwise bilinear correlation with respect to two successive frames within a certain region is computed for each pair of successive frames. The complete bilinear feature is extracted through temporal-wise attentive concatenation of each inter-frame correlation feature.

Unlike common temporal pooling layers where the average pooling is performed across the whole temporal dimension, the purpose of the extraction function $\mathcal{P}_{mean}$ is to capture the average of two successive frames. More specifically, the extraction function for IMF is an average pooling function with a kernel size of (k_i, k_h, k_w). Here k_h and k_w are the kernel size corresponding to the spatial dimensions. As we need to preserve all information along the spatial dimensions, hence $k_h, k_w = 1$. To obtain the average pooling along successive temporal features $\mathbf{f}_i$ and $\mathbf{f}_{i+1}$, the kernel size along the temporal dimension k_i is set to 2 instead of the whole temporal dimension length.

The IMF $\mathbf{L} \in \mathbb{R}^{(t-1)\times C_{out}\times H\times W}$ is computed by:

$$\mathbf{L} = \mathcal{P}_{mean}(\mathbf{F}). \tag{3.5}$$

Similar to the temporal-wise attentive concatenation for bilinear inter-frame correlation features, we adopt a feature-wise attentive concatenation approach to combine the bilinear inter-frame correlation feature with the linear inter-frame correlation feature. Each of the two types of features is assigned a separate weight parameter, denoted as β, γ respectively. This allows the network to dynamically focus on either feature for different actions. The result of the attentive concatenation $\mathbf{H} \in \mathbb{R}^{(t-1)\times C_{concat}\times H\times W}$ is obtained as follows:

$$\mathbf{H} = \beta\mathbf{B} \oplus \gamma\mathbf{L}, \tag{3.6}$$

where $\oplus$ denotes the concatenation operation along the feature channel dimension. C_{concat} is the total number of feature channels, which is the sum of C_{out} and $C_{bilinear}$.

The complete ACTF feature $\mathbf{V}_{actf}$ is obtained as follows:

$$\mathbf{V}_{actf} = \mathcal{L}(\mathcal{P}_{average}(\mathbf{H})), \tag{3.7}$$

where $\mathcal{P}_{average}$ is an average pooling operation with a kernel size of $((t-1), H, W)$ corresponding to the temporal and spatial dimensions respectively. ACTF feature summarizes both the ICCF and the IMF. $\mathcal{L}$ is a linear dimension reduction function, constructed as a multi-layer linear neural network. This allows $\mathcal{L}$ to be learnable and the overall system to be trainable in an end-to-end manner. The resulting feature is thus the ACTF feature $\mathbf{V}_{actf} \in \mathbb{R}^{C_{out}}$.

3.2.3 Attentive Concatenation of Features

Our network is designed to focus on the pairs of time steps which are more relevant to the action. Meanwhile, it is also designed to focus on the more important type of feature, *i.e.* the spatial or temporal feature . To achieve both goals, we adopt attentive concatenation at each location where different features are combined. In this section, we describe how to extract the ICCF by the temporal-wise attentive concatenation of bilinear inter-frame correlation features. The combination of features $\mathbf{V}_{stpooled}$ and $\mathbf{V}_{actf}$ mentioned in Section 3.2.1 as well as the combination of features $\mathbf{B}$ and $\mathbf{L}$ mentioned in Section 3.2.2 follow similar implementations.

The attentive concatenation of all $(t-1)$ bilinear correlation features is achieved by assigning each feature with a weight α_i for the i^{th} correlation feature $\mathbf{b}_i$. Inspired by the cascade attention network proposed in [134], we adopt an attentive concatenation approach for the computation of weight α_i. Formally, α_i is computed by:

$$\alpha_i = g(h((\mathcal{P}_{spatial}(\mathbf{b}_i))W)) \tag{3.8}$$

More specifically, given the i^{th} bilinear correlation feature $\mathbf{b}_i \in \mathbb{R}^{C_{bilinear} \times H \times W}$, $\mathcal{P}_{spatial}$ is a spatial average pooling function with kernel size (H, W). The output of $\mathcal{P}_{spatial}$ is a pooled feature vector $\mathbf{b}_{pooled,i} \in \mathbb{R}^{C_{bilinear}}$. $W \in \mathbb{R}^{C_{bilinear} \times 1}$ denotes a trainable parameter matrix, shared among all $(t-1)$ bilinear correlation features. The result of this matrix multiplication is a primitive weight parameter denoted as $\alpha_{prime,i}$.

To scale the primitive weight parameter $\alpha_{prime,i}$ to a range of $[0, 1]$, we apply a sigmoid function denoted as $h(\alpha_{prime,i})$, which is computed by:

$$h(\alpha_{prime,i}) = \frac{1}{1 + e^{-\alpha_{prime,i}}} \tag{3.9}$$

The weight α_i is then further processed from $h(\alpha_{prime,i})$ to satisfy $\Sigma\alpha_i = 1$. This is achieved by applying a softmax function denoted by $g(\cdot)$, and the weight α_i is calculated as follows:

$$\begin{aligned} \alpha_i &= g(h(\alpha_{prime,i})) \\ &= \frac{e^{h(\alpha_{prime,i})}}{\sum_{i=1}^{t-1} e^{h(\alpha_{prime,i})}} \end{aligned} \tag{3.10}$$

The weight α_i indicates the importance of the i^{th} bilinear inter-frame correlation feature, $\mathbf{b}_i$.

3.3 Experiments

In this section, we present our evaluation results of the proposed method. The evaluation is conducted through action recognition experiments on two public benchmark datasets. We present state-of-the-art results on a competitive architecture, and prove the novelties on another similar baseline. We also present detailed ablation study of the components of our proposed framework to verify our design.

3.3.1 Experimental Settings

We conduct experiments on two benchmark datasets of action recognition: UCF101 [74] and HMDB51 [72]. The UCF101 dataset contains 13,320 videos from 101 action categories while the HMDB51 dataset contains 6,766 videos from 51 action categories. We follow the experiment settings as in [51, 53, 58] that adopt the three training/testing splits for evaluation. We report the average top-1 accuracy of the three splits. Our proposed framework for temporal feature extraction can be used in any CNN based networks. To obtain the state-of-the-art result, we instantiate MFNet [58].

Our experiments are implemented using PyTorch [135]. Following the implementation in [58], the input is a frame sequence with each frame of size 224×224. The output

from MFNet [58] is a frame-level feature of size $8 \times 768 \times 7 \times 7$, where the number of output channels is 768 . Each frame is represented by a feature of size 7×7. We set the number of channels of our ICCF to 3,840. Thus, the size of $\mathbf{H}$, described in Section 3.2.2 of the chapter, is $7 \times 4608 \times 7 \times 7$. We design the linear dimension reduction function in Equation 7 as a three-layer linear neural network with RELU activation. For training, we utilize the pre-trained model of MFNet [58] trained on Kinetics [76], a large-scale human action dataset. To accelerate our training, the pre-trained model is used for the initialization of the network which includes our framework for temporal feature extraction. We use stochastic gradient descent algorithm [136] for optimization, setting the weight decay to 0.0001 and the momentum to 0.9. For both datasets, our initial learning rate is set to 0.005. For UCF101 [74] dataset, the learning rate is decreased for four times, while for HMDB51 [72] dataset, the learning rate is decreased for three times. The learning rate is decreased with a factor of 0.1.

To prove that our approach can be applied to other 3D CNN approaches, we also apply our proposed ACTF in another 3D CNN. We instantiate C3D [51], a classical 3D CNN baseline for action recognition. Our proposed ACTF is extracted after conv5 layer of the C3D network, in parallel with the spatial-temporal pooling layer pool5, as well as the linear layer that follows. We follow the setup as in [51], using stochastic gradient descent [136] with initial learning rate of 0.001. We compare the results of the C3D network with and without the temporal feature extracted by our proposed framework on HMDB51 dataset.

3.3.2 Results and Comparison

Table 3.1 shows the comparison of top-1 accuracy on UCF101 and HMDB51 datasets with other state-of-the-art methods including:

1. **Two-stream methods:** the original two-stream method (original TS) [2], Hidden Two-Stream (Hidden TS) [130], Long-term Temporal Convolutions (LTC) [137], ActionVLAD [138] and Temporal Segment Network (TSN) [35]
2. **Recurrent network-based methods:** ShuttleNet [47] and Att-ConvLSTM [139]
3. **3D CNNs-based methods:** C3D [51], TSN with RGB input [35], Res3D [140], ST-ResNet [34], 3D-ResNext [54], R(2+1)D with RGB input [53], I3D with RGB input [52], TVNet [42], MFNet [58] and T-C3D [141]

	Method	UCF101	HMDB51	FPS
Two-stream	original TS	88.0%	59.4%	14
	Hidden TS	90.3%	58.9%	<14
	LTC	91.7%	64.8%	<14
	TSN	94.2%	69.4%	5
Recurrent networks	Att-ConvLSTM	92.4%	66.4%	N/A
	ShuttleNet	95.4%	71.7%	N/A
3D CNNs	C3D	85.2%	65.5%	314
	TSN (RGB)	86.2%	-	N/A
	Res3D	85.8%	54.9%	N/A
	T-C3D	91.8%	62.8%	969
	ST-ResNet	93.5%	66.4%	N/A
	3D-ResNext	94.5%	70.2%	<314
	R(2+1)D (RGB)	93.6%	66.6%	N/A
	I3D (RGB)	95.6%	74.8%	N/A
	TVNet	95.4%	72.5%	N/A
	MFNet (Flow)	95.6%	74.2%	32
	MFNet (RGB)	96.0%	74.6%	506
Ours	C3D-ACTF	87.4%	69.2%	300
	MFNet-single-ACTF	96.1%	75.7%	486
	MFNet-ACTF	**96.3%**	**76.3%**	**478**

TABLE 3.1: Comparison of top-1 accuracy and speed with state-of-the-art methods on UCF101 and HMDB51 datasets.

Our competitive performance is achieved by instantiating MFNet, denoted as MFNet-ACTF. For this experiment, we set our batch size to 80 and conduct the experiment using four NVIDIA Tesla P100 GPUs. The results are obtained through an average of five experiments on each dataset.

The performance results in Table 3.1 show that our network achieves the best results on both benchmark datasets. More specifically, our MFNet-ACTF network achieves a 1.7% improvement on HMDB51 dataset over the networks whose input are solely RGB frames. Our method even surpasses several networks with both RGB and optical flow as input, as well as methods utilizing recurrent neural networks. For UCF101 dataset, our MFNet-ACTF also produces the best result. It is noted that the improvement is not as significant as that on HMDB51 dataset, mainly due to the fact that there is little room for improvement. In addition, we tested our extracted temporal feature separately by performing action recognition task utilizing only the ACTF extracted with MFNet. Such a network is denoted as MFNet-single-ACTF. Compared with the case where the optical flow input is utilized with MFNet, our MFNet-single-ACTF achieves a performance gain of 0.5% on UCF101 and 1.5% on HMDB51. The superior performance achieved

Method	Top-1 HMDB51
C3D	65.5%
C3D-single-ACTF	67.9%
C3D-ACTF	69.2%

TABLE 3.2: Top-1 accuracy of C3D network on HMDB51 dataset with and without our proposed framework.

by ACTF alone justifies the superiority of ACTF over optical flow.

The speed results in Table 3.1 show that our proposed method balances between high accuracy and relatively high inference speed. Despite achieving high accuracy on both dataset, two-stream methods such as TSN could not achieve real-time requirements, reaching only 5 FPS. Compared with two-stream methods, our proposed method is much faster in inference speed, reaching a speed of 478 FPS, which is well above real-time requirements. Our speed is even faster than that achieved by C3D network. Note that our speed is slower than that achieved by T-C3D network, but we achieved a much higher accuracy compared to theirs, with a 13.5% increase in top-1 accuracy on HMDB51 dataset.

We suggest in Section 3.2 that our proposed framework which includes ACTF feature can be used in combination with any CNN-based frame-level feature extraction networks, such as C3D network. To verify this, we conducted experiments on the baseline network C3D with and without ACTF. We first perform action recognition with only the temporal feature extracted through our proposed ACTF. The frame-level feature is extracted through conv5 layer of the C3D network. We denote this network as C3D-single-ACTF. We then perform action recognition by attentively combining the ACTF feature as well as the Spatial-Temporal Pooled feature which is extracted from *pool5* layer of C3D, similar to the implementation of MFNet-ACTF. We denote this modification as C3D-ACTF. The top-1 accuracy of the networks are shown in Table 3.2.

The results in Table 3.2 clearly show that applying our proposed framework in the C3D network improves the accuracy of the baseline C3D network. Even by only utilizing the extracted ACTF feature as temporal feature, we obtain an improvement of 2.4%. This shows that our ACTF feature effectively represent the temporal pattern in the video and thus lead to better results. A larger gain of 3.7% is achieved when the extracted ACTF feature is used with the Spatial-Temporal Pooled feature. The results are consistent with that shown in Table 3.1, where using ACTF feature improves the accuracy of MFNet.

This suggests that our proposed framework is generic, and can be used with other baselines.

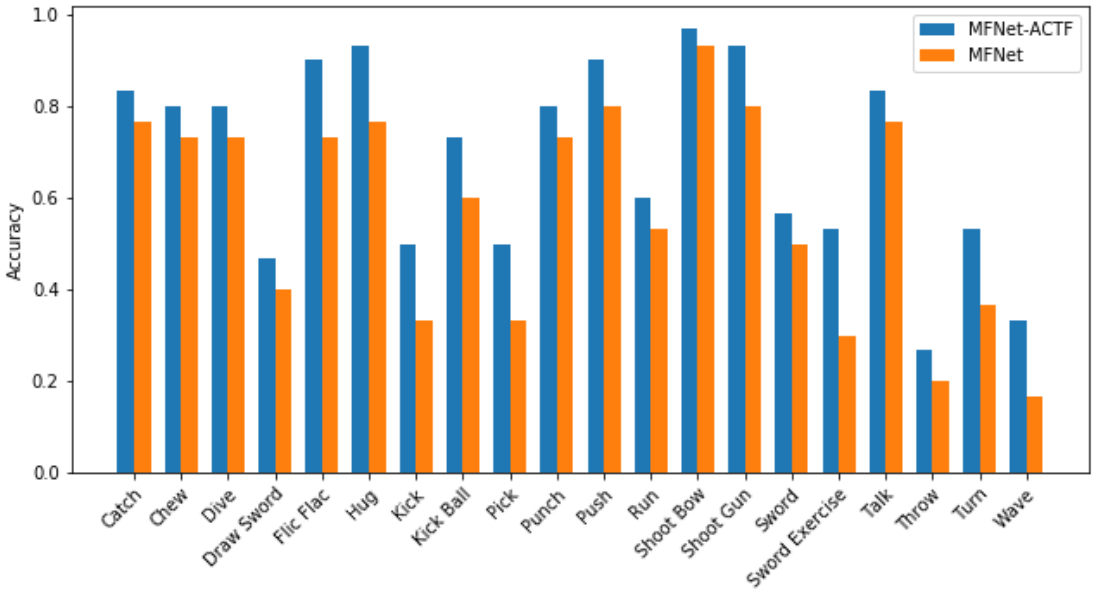
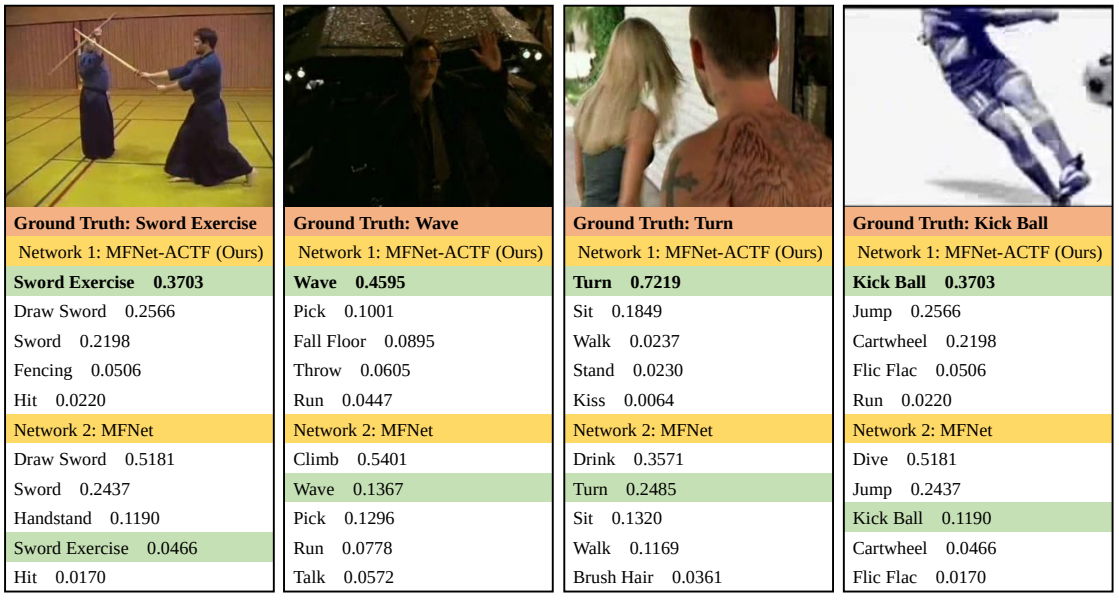


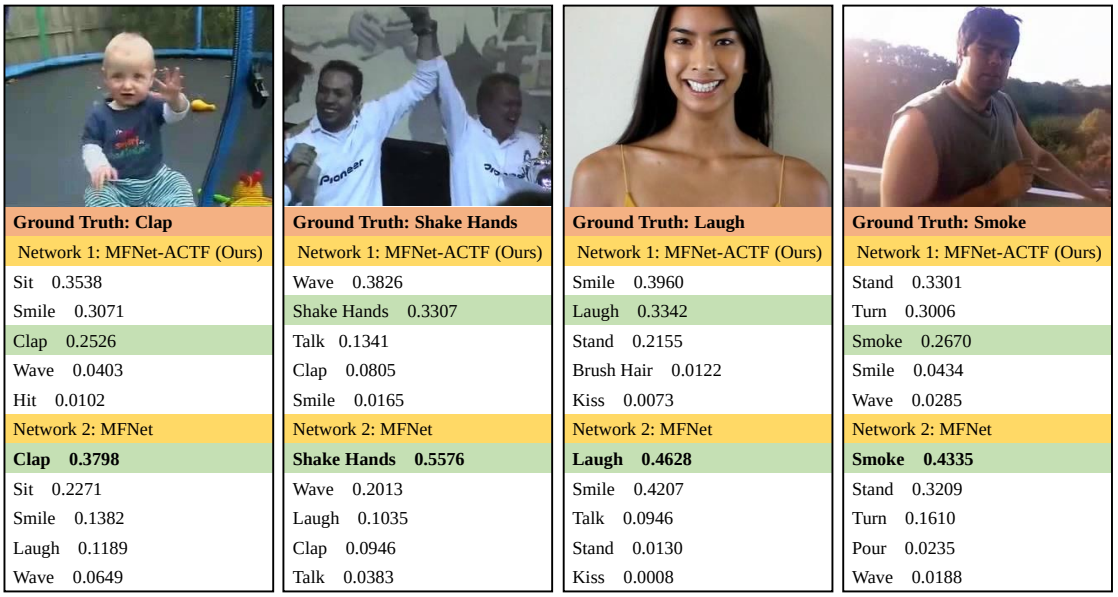
FIGURE 3.5: Accuracy comparisons of 20 classes on split 1 of the HMDB-51 between our proposed MFNet-ACTF network and the original MFNet network.



Ground Truth: Sword Exercise	**Ground Truth: Wave**	**Ground Truth: Turn**	**Ground Truth: Kick Ball**
Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)
Sword Exercise 0.3703	**Wave 0.4595**	**Turn 0.7219**	**Kick Ball 0.3703**
Draw Sword 0.2566	Pick 0.1001	Sit 0.1849	Jump 0.2566
Sword 0.2198	Fall Floor 0.0895	Walk 0.0237	Cartwheel 0.2198
Fencing 0.0506	Throw 0.0605	Stand 0.0230	Flic Flac 0.0506
Hit 0.0220	Run 0.0447	Kiss 0.0064	Run 0.0220
Network 2: MFNet	Network 2: MFNet	Network 2: MFNet	Network 2: MFNet
Draw Sword 0.5181	Climb 0.5401	Drink 0.3571	Dive 0.5181
Sword 0.2437	Wave 0.1367	Turn 0.2485	Jump 0.2437
Handstand 0.1190	Pick 0.1296	Sit 0.1320	Kick Ball 0.1190
Sword Exercise 0.0466	Run 0.0778	Walk 0.1169	Cartwheel 0.0466
Hit 0.0170	Talk 0.0572	Brush Hair 0.0361	Flic Flac 0.0170

FIGURE 3.6: Examples from HMDB51 dataset where our proposed MFNet-ACTF succeeds in recognizing the action while the original MFNet fails.

We further investigate the improvement of performance over different actions and present the comparison of performance between our proposed MFNet-ACTF network and the



Ground Truth: Clap	Ground Truth: Shake Hands	Ground Truth: Laugh	Ground Truth: Smoke
Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)	Network 1: MFNet-ACTF (Ours)
Sit 0.3538	Wave 0.3826	Smile 0.3960	Stand 0.3301
Smile 0.3071	Shake Hands 0.3307	Laugh 0.3342	Turn 0.3006
Clap 0.2526	Talk 0.1341	Stand 0.2155	Smoke 0.2670
Wave 0.0403	Clap 0.0805	Brush Hair 0.0122	Smile 0.0434
Hit 0.0102	Smile 0.0165	Kiss 0.0073	Wave 0.0285
Network 2: MFNet	Network 2: MFNet	Network 2: MFNet	Network 2: MFNet
Clap 0.3798	**Shake Hands 0.5576**	**Laugh 0.4628**	**Smoke 0.4335**
Sit 0.2271	Wave 0.2013	Smile 0.4207	Stand 0.3209
Smile 0.1382	Laugh 0.1035	Talk 0.0946	Turn 0.1610
Laugh 0.1189	Clap 0.0946	Stand 0.0130	Pour 0.0235
Wave 0.0649	Talk 0.0383	Kiss 0.0008	Wave 0.0188

FIGURE 3.7: Examples from HMDB51 dataset where our proposed MFNet-ACTF fails in recognizing the action while the original MFNet succeeds.

original MFNet network. Figure 3.5 shows the accuracy of 20 classes from split-1 of the HMDB51 dataset, where our network outperforms the original network by a noticeable margin. It is worth noticing that for actions with similar spatial appearance but different actions, e.g. ”Sword” and ”Sword Exercise”, our network performs significantly better than the original network. Our network obtains a 23.3% performance gain over the original MFNet on the action class ”Sword Exercise”. The large performance gain proves the effectiveness of the additional temporal feature extracted as the ACTF feature in improving the complete video representation.

Several examples from HMDB51 dataset is presented in Figure3.6 where our proposed MFNet-ACTF could accurately recognize the respective actions while the original MFNet network could not. It could be observed that the spatial features of the given examples, or more intuitively the appearance of the given examples, could not provide effective representation for accurate action recognition. For example, for the first video, the scenario as shown could be present in action classes ”Sword”, in which most videos present people fighting with a sword, and ”Draw Sword”, in which videos present the action of a sword drawn out. The difference between these action classes could only be determined through the temporal feature instead of the spatial feature. Thus the original network which can only extract the spatial feature of the video cannot distinguish the actions correctly while our proposed framework succeeds in recognizing the different actions.

Method	Top-1 HMDB51 split-1
MFNet	70.8%
MFNet-ICCF	72.6%
MFNet-single-ACTF	72.9%

TABLE 3.3: Comparison of the network architectures that use only temporal feature for action recognition.

Method	Top-1 HMDB51 split-1
MFNet-ACTF-no-attn	72.5%
MFNet-attn@ACTF	73.3%
MFNet-attn@final	73.0%
MFNet-ACTF	73.6%

TABLE 3.4: Comparison of the network architectures that use all or partial attentive concatenation.

Meanwhile, we also observe some cases where our proposed MFNet-ACTF may not recognize the actions as accurately as that using the original MFNet. The examples from HMDB51 dataset are presented in Figure 3.7. Generally, MFNet-ACTF may not perform well when similar actions patterns presented in different classes. One typical example is that of "Shake Hands" for the second video, where the action pattern consist of both the shaking of hands as well as hand waving. The difference in these two classes requires the network to place a much higher weight on the spatial feature rather than the temporal feature. It is noted that the probability outputs of the classes "Shake Hands" and "Wave" by our MFNet-ACTF are very close. This could suggest that though accurate temporal feature is extracted, the weight mechanism applied may weigh the temporal feature higher than required.

3.3.3 Ablation Study

In this section, we justify our proposed design of the ACTF feature through ablation study. Specifically, we examine the performance of our proposed ICCF and the ACTF feature separately. We then examine the performance of the attention mechanisms used to combine the different modules of our proposed generic action recognition framework as discussed in Section 3.2. All experiments conducted in our ablation study are performed on split 1 of the HMDB51 dataset. We set our batch size to 16 and conduct the experiment using one NVIDIA TITAN Xp GPU. The much smaller batch size is a key reason of the lower accuracy reported than that in Table 3.1.

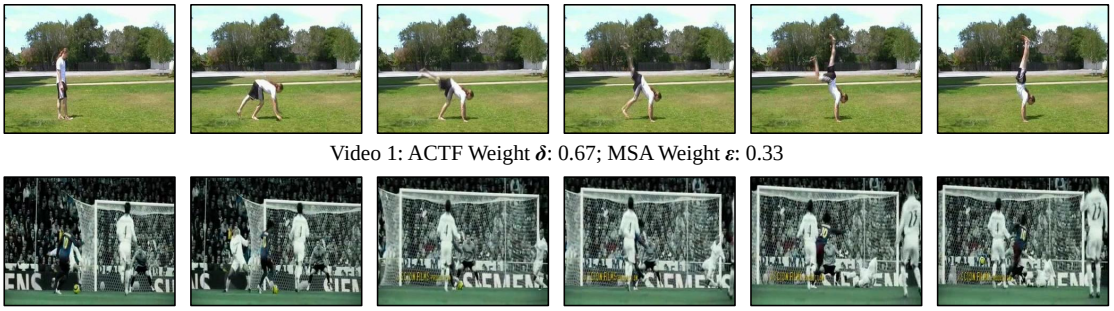
We instantiate MFNet to justify our proposed ICCF and the ACTF feature introduced in Section 3.2.2 and utilize only temporal feature for action recognition. First the proposed ICCF is extracted as our temporal feature. The network that utilizes only ICCF is denoted as MFNet-ICCF. We then employ the ACTF feature as our temporal feature. Similar to the previous denotation, the network that utilizes only the ACTF feature is denoted as MFNet-single-ACTF. The comparison of the performances of these two networks with the baseline MFNet is shown in Table 3.3.

The result in Table 3.3 shows that by utilizing only temporal feature, even with only bilinear inter-frame correlation, the performance of the network is improved by a margin of 1.8%, indicating that utilizing inter-frame correlation information helps to extract high-quality temporal feature of the video. The improvement achieved by using high quality temporal feature over feature obtained from spatial-temporal pooling coincides with findings in preceding works [31, 35, 52]. However, our temporal feature is obtained from RGB input through inter-frame correlation rather than using optical flow.

Results in Table 3.3 shows further improvement when we employ the ACTF feature. As described in Section 3.2.2, the ACTF feature is a weighted combination of ICCF, which is a bilinear inter-frame correlation feature, and IMF, which is a linear inter-frame correlation feature. This result proves that the bilinear inter-frame correlation feature and linear inter-frame feature complements each other.

To better combine the features extracted from different modules, we introduced attentive concatenation of features as mentioned in Section 3.2.3. Here we justify the need for utilizing attentive concatenation of features. Table 3.4 presents the comparison between the networks that utilize attentive concatenation at every step and the networks that partially or do not utilize attentive concatenation for feature combination. Here MFNet-ACTF-no-attn denotes the network where all feature combination utilizes direct concatenation instead of attentive concatenation. Meanwhile, MFNet-ACTF denotes the network utilizing our proposed temporal feature extraction framework with attentive concatenation at every step of feature combination. MFNet-attn@ACTF denotes the network that performs temporal-wise attentive concatenation when constructing ICCF, and feature-wise attentive concatenation of ICCF and IMF as shown in Figure 3.3. The concatenation of the temporal feature and spatial feature is by direct concatenation. Similarly, MFNet-attn@final denotes that attentive concatenation is adopted only for ACTF and Spatial-Temporal Pooled feature combination while direct concatenation is adopted at other stages.

The result given in Table 3.4 clearly shows the advantage of adopting attentive concatenation for feature combination. We note that if the network combines features with only direct concatenation, its performance would be even worse than that of MFNet-single-ACTF, whose ACTF feature is constructed with attentive concatenation of ICCF and IMF features. The performance is improved even when attentive concatenation is used in some stages of feature combination only. It can be observed that applying attentive concatenation at different stages complements each other, with over 1% improvement made when all stages adopt attentive concatenation.



Video 1: ACTF Weight $\boldsymbol{\delta}$: 0.67; MSA Weight $\boldsymbol{\varepsilon}$: 0.33

Video 2: ACTF Weight $\boldsymbol{\delta}$: 0.36; MSA Weight $\boldsymbol{\varepsilon}$: 0.64

FIGURE 3.8: The weights of ACTF feature δ and the weights of Spatial-Temporal Pooled feature ϵ for two videos. Attentive concatenation learns these weights dynamically.

We also investigate the weights δ and ϵ on the ACTF feature and the Spatial-Temporal Pooled feature for different videos. Figure 3.8 shows examples where either temporal feature or spatial feature dominates the feature combination process. Video 2 shows a video where the Spatial-Temporal Pooled feature, or the spatial feature, dominates the feature combination. These videos tend to have clear visual characteristics, such as the soccer goal that appears in most videos describing the sport soccer. The appearance of these videos are therefore sufficient for action recognition, and dominate the feature combination process. By contrast, feature combination in Video 1 is dominated by ACTF feature, which is the temporal feature. We observe that similar videos tend to have actions that would mix up with other categories. In this case, the handstand action is similar to actions that may occur in diving or in somersault, where a person would also go upside down. Also, there is no iconic background items in Video 1. For these videos, the temporal features dominate the feature combination, thus has a larger weight δ. The different weights with respect to the different videos could prove that adopting attentive concatenation could attend to the more important feature which is related to the characteristic of the video itself.

3.4 Summary

In this chapter, we propose a new method for extracting the temporal feature of a video while avoiding the use of optical flow. The new temporal feature namely Attentive Correlated Temporal Feature (ACTF) is an attentive combination of both bilinear inter-frame correlation and linear inter-frame correlation features. The bilinear inter-frame correlation feature is extracted through a bilinear operation with respect to successive frames within a certain region, while the linear inter-frame feature is extracted through inter-frame temporal pooling. For overall evaluation on UCF101 and HMDB51, our method obtains state-of-the-art results when instantiating MFNet combined with our ACTF feature. We verify our design through thorough ablation study, and then further demonstrate that the proposed feature can be introduced to other similar action recognition networks instead of using optical flow.

Chapter 4

PNL: Efficient Long-Range Dependencies Extraction with Pyramid Non-Local Module for Action Recognition

As discussed in Chapter 1, extracting spatiotemporal correlation features is another important factor towards robust and efficient vision-based action recognition, with spatiotemporal long-range dependencies as an effective form of spatiotemporal correlation feature. In this chapter, following the work in Chapter 3, a new spatiotemporal long-range dependencies extraction method is presented. Section 4.1 introduces the motivation of this chapter. Section 4.2 presents the proposed method with an introduction and comparison over the previous Non-Local Block designed for spatiotemporal long-range dependencies extraction. Extensive experiments are conducted in Section 4.3, followed by the conclusion of this chapter in Section 4.4.

4.1 Introduction

Capturing long-range spatiotemporal dependencies has proven to play an essential role in extracting effective video features for action recognition. Previously, feature extraction techniques such as SIFT [142], GLOH [143] and Dense Trajectory [144] model

such dependencies through hand-crafted features. More recently, convolutional and recurrent modules have replaced these hand-crafted features as the predominant modules for video features extraction. However, each convolution or recurrent module extracts spatiotemporal dependencies only within spatial or temporal local regions. Therefore, it requires a stack of multiple convolution or recurrent modules to model long-range spatiotemporal dependencies. Such strategy is computationally inefficient, while also causing difficulties in network optimization.

Inspired by the non-local means for the image filtering task [60, 61], the non-local neural network [62] is proposed to address the challenge of capturing long-range dependencies directly. The proposed network captures long-range dependencies through direct modeling the correlations of each single pixel at any spatiotemporal location in a single non-local block. Without bells and whistles, the insertion of non-local blocks improves the action recognition accuracies of existing networks, which proves the effectiveness of the non-local block in extracting long-range dependencies.

Despite the great improvements brought by the introduction of the non-local block, the original non-local block has its own limitations. The original non-local block significantly increases the parameter size and computational cost of the inserted networks, thanks to the fact that the long-range dependencies is captured through pixel correlations. The increases in action recognition accuracies of the inserted networks are at the cost of significant decreases in computational efficiencies of the networks.

On the other hand, when we recognize action, it is more intuitive to focus on not only the correlations between each single pixel, but also on the correlations between larger regions of each frame, as can be shown in Figure 4.1. To classify the action, we relate the boy with the backboard, which suggest a high possibility of the "playing basketball" action. This is more intuitive and efficient than extracting pixel correlations that relate the basketball across frames, as well as pixel correlations that relate the basketball with the hands and elbows.

To this end, to improve both the effectiveness and efficiency of the original non-local block, we propose a novel long-range spatiotemporal dependencies extraction module: the Pyramid Non-Local (PNL) module. The proposed PNL module extends the original non-local block, and incorporates regional feature correlations at multiple scales through a pyramid structured module. The multi-scaled correlations are combined with a self-attentive combination function. Our main contributions are summarized as follows:

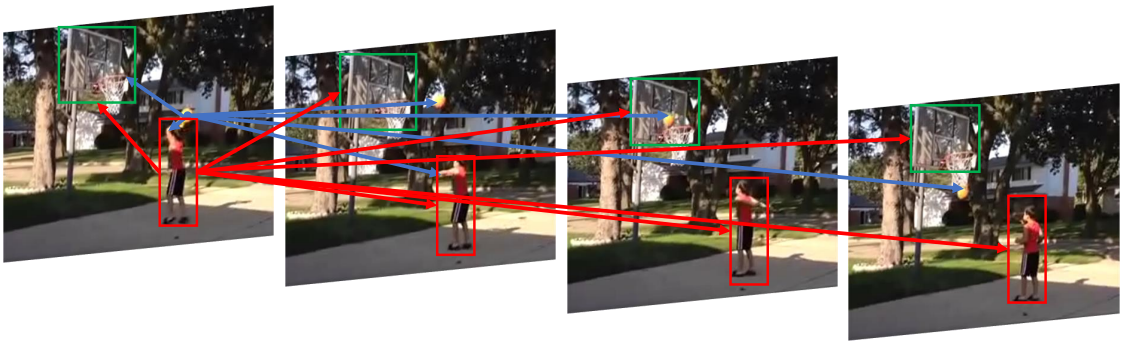


FIGURE 4.1: Illustration of utilizing regional correlations for action recognition. The original non-local block captures long-range spatiotemporal dependencies through pixel correlations, shown as blue arrows. The action "playing basketball" could alternatively be recognized through regional correlations between the boy and the backboard, shown as red arrows.

* We propose a novel long-range spatiotemporal dependencies extraction module, Pyramid Non-Local (PNL) module. The PNL module extends the original non-local block through incorporating regional feature correlations at multiple scales. This extension upscales the effectiveness of the original non-local block by attending to the interactions between different regions.

* We conduct comprehensive analysis over the computation cost required by our proposed PNL module. We further demonstrate its efficiency through comparing the computation cost of the PNL module against the original non-local block.

* We conduct extensive experiments on two action recognition benchmark datasets: Mini-Kinetics [55] and UCF101 [74] with multiple frameworks utilizing our proposed PNL module. The results demonstrate that our proposed PNL module brings noticeable improvements over baseline methods and methods utilizing the original non-local block, while requires less computation cost. Our network achieves state-of-the-art performance for the Mini-Kinetics dataset.

The rest of this chapter is organized as follows: in Section 4.2, we introduce and analyze the proposed Pyramid Non-Local module (PNL) in detail. After that, we present the experimental results of our proposed PNL module, with thorough ablation experiments on the design of PNL module and visualization of feature outputs. Finally, we conclude the chapter in Section 4.4.

4.2 Methodology

The primary goal of this chapter is to develop a more effective and efficient module to extract long-range spatiotemporal dependencies. To achieve this, we propose to extend the original non-local block [62] through incorporating regional correlations. In this section, we introduce our proposed Pyramid Non-Local (PNL) module with detailed illustration of how it is extended from the original non-local block. We then provide solid proof of its higher computation efficiency compared to the original non-local block.

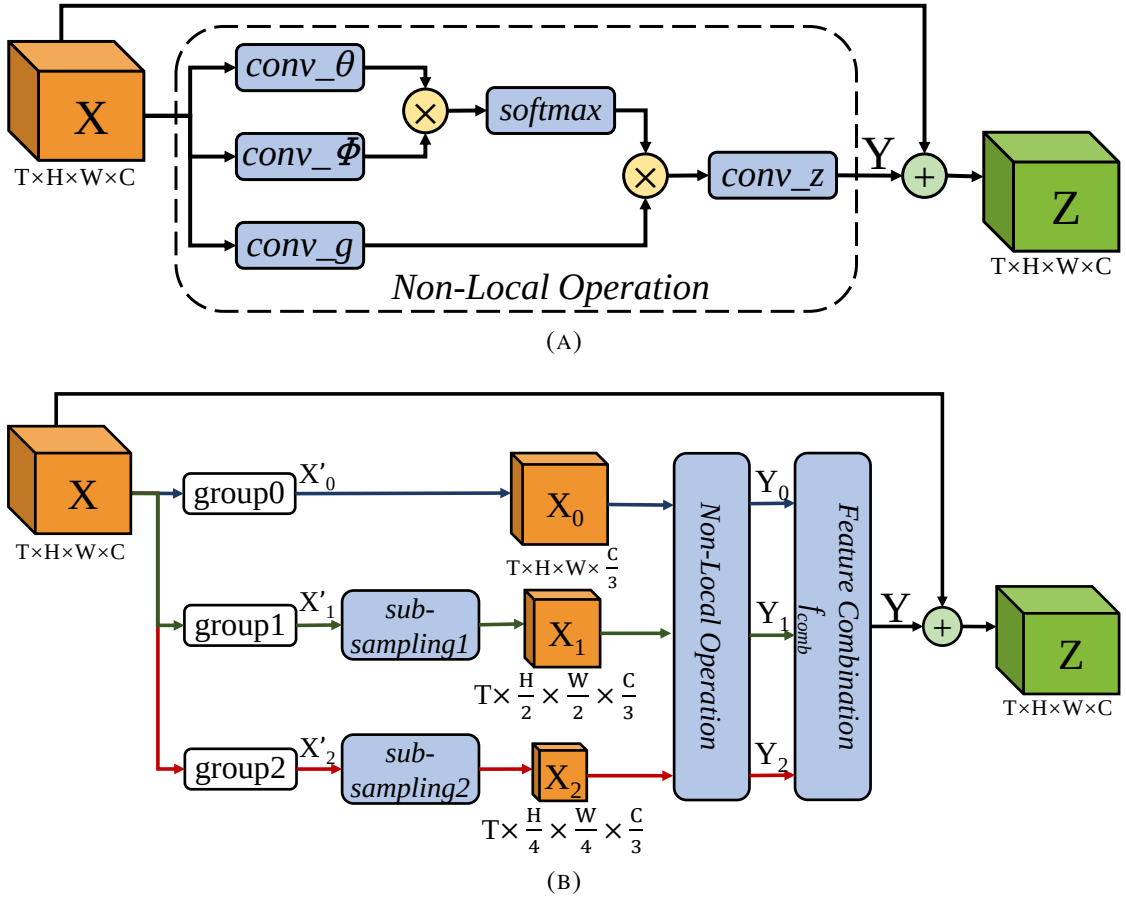


FIGURE 4.2: Comparison of the original non-local block (a) with our proposed PNL module (b). We present the case where the embedded Gaussian function is utilized for the non-local operation. For the PNL module, we present the case where $n = 3$. The dimension of the input and output features are also presented, with the "batch" dimension ignored.

4.2.1 Review of Non-Local Block

As our proposed module is built by extending the original non-local block, we begin by briefly reviewing the original non-local block as introduced in [62]. The structure of the original non-local block is as shown in Figure 4.2a. Let the video input be denoted as $\mathbf{X} \in \mathbb{R}^{T\times H\times W\times C}$, where T, H, W and C denote the temporal length, height, width and number of channels of the video, respectively. The original non-local block captures long-range spatiotemporal dependencies through a non-local operation, which is a weighted sum of the correlation features at all positions, formulated as:

$$\mathbf{y}_i = \frac{1}{\mathcal{M}(\mathbf{X})} \sum_{\forall j} f(\theta(\mathbf{x}_i), \phi(\mathbf{x}_j)) g(\mathbf{x}_j), \tag{4.1}$$

where $\mathbf{y}_i$ is the output response $\mathbf{Y}$ at space-time position i, while $\mathbf{x}_i$ and $\mathbf{x}_j$ are the input features at space-time positions i and j. $\mathcal{M}(\mathbf{X})$ is the normalization factor, and is defined as $\mathcal{M}(\mathbf{X}) = f(\theta(\mathbf{x}_i).\phi(\mathbf{x}_j))$. $\theta(\cdot)$, $\phi(\cdot)$ and $g(\cdot)$ are learnable transformations of the input features and are implemented as convolution layers with kernel size of $1 \times 1 \times 1$. Due to the convolutional implementation, we specify the transformations $\theta(\cdot)$, $\phi(\cdot)$ and $g(\cdot)$ as $conv\_\theta$, $conv\_\phi$ and $conv\_g$, respectively. The pairwise function $f(\cdot, \cdot)$ computes the affinity between the input features at all space-time positions. The choice of the pairwise function $f(\cdot, \cdot)$ varies. Figure 4.2a shows the case where the embedded Gaussian version of $f(\cdot, \cdot)$ is adopted. The final output of the non-local block $\mathbf{Z}$ is computed by adding the long-range dependencies $\mathbf{Y}$ from the non-local operation with the original input $\mathbf{X}$ in an element-wise manner.

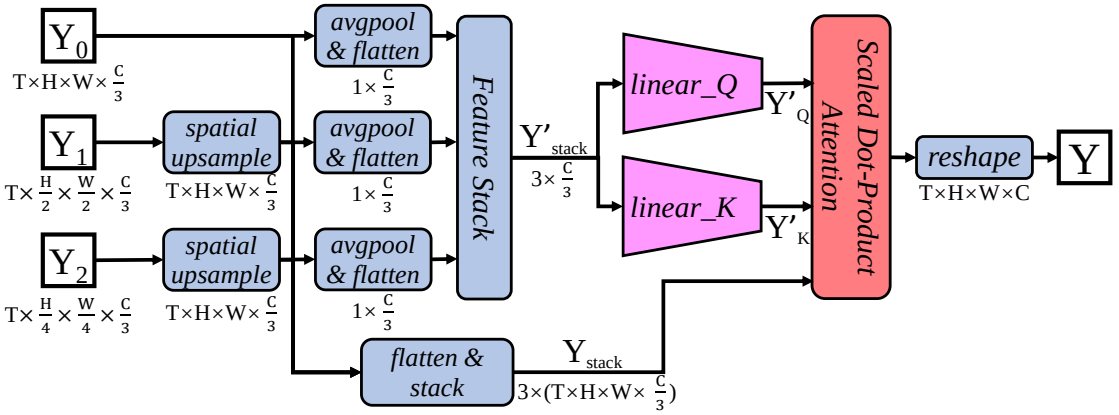


FIGURE 4.3: Structure of the combination function f_{comb}. f_{comb} is designed by adopting a self-attention mechanism, and combines the multi-scaled dependencies attentively. The dimensions of the outputs after each operation are also presented, with the "batch" dimension ignored.

4.2.2 Pyramid Non-Local Module

While the original non-local block is designed to capture long-range dependencies between any two space-time positions in the input feature, such dependencies are extracted only at the pixel level, where pixels at every space-time position are included in the computation. The use of only pixel correlations may not be effective and efficient due to the existence of trivial background pixels and the exclusion of regional correlations. On the other hand, multi-scale regional features have been proven effective in tasks such as object detection [145] and salient object detection [146]. Inspired by these works, we introduce the Pyramid Non-Local module (PNL) which incorporates multi-scale regional correlations, utilizing a pyramid structured module. The structure of the PNL module is as shown in Figure 4.2b. Formally, to extend the original non-local block to incorporate regional correlations, we first obtain n features of different scales from the original input $\mathbf{X}$. We leverage the channel grouping technique as in [29, 147], grouping the channels into n groups, each containing $C' = C/n$ channels, with n strictly larger than 1. We denote the results of channel grouping to be $\mathbf{X}'_0, \mathbf{X}'_1$,..., $\mathbf{X}'_{n-1}$. We then obtain the features of n scales through sub-sampling operations over the n groups of feature. Note that the sub-sampling operation does not apply to $\mathbf{X}'_0$, where we preserve a group of channels with the same resolution and scale as the original input. The sub-sampling operations are implemented as pooling operations across the spatial dimensions. The result of the k^{th} sub-sampling operation $\mathbf{X}_k$ is of spatial size $\frac{H}{2^k} \times \frac{W}{2^k}$. The results of the above channel grouping and sub-sampling process are therefore scaled features denoted as $\mathbf{X}_k \in \mathbb{R}^{T \times \frac{H}{2^k} \times \frac{W}{2^k} \times C'}$, where $k \in [0, (n-1)]$.

Through the pooling process, each feature point in the scaled features corresponds to a region of the original input. Therefore, the correlations of each feature point in the scaled features can be viewed as the correlations of the corresponding regions in the original input. To capture the long-range dependencies on both the pixel level and regional level, we input the scaled features of $\mathbf{X}_0, \mathbf{X}_1$,..., $\mathbf{X}_{n-1}$ to the non-local operation, as reviewed in Section 4.2.1. For all the input scaled features, we share the parameters of the non-local operations. The end results of this step are thus long-range dependencies at multiple scales, denoted as $\mathbf{Y}_0, \mathbf{Y}_1$,..., $\mathbf{Y}_{n-1}$. To obtain the overall long-range dependencies denoted as $\mathbf{Y}$, we combine the long-range dependencies of $\mathbf{Y}_0, \mathbf{Y}_1$,..., $\mathbf{Y}_{n-1}$ with a combination function f_{comb}. f_{comb} could be as simple as a vanilla concatenate function, where $\mathbf{Y} = concat(\mathbf{Y}_0, \mathbf{Y}_1$,..., $\mathbf{Y}_{n-1})$. However, the vanilla concatenate function

weighs all input features equally, which is not ideal. To combine the multi-scaled long-range dependencies dynamically, our proposed f_{comb} adopts a self-attention mechanism, utilizing the scaled dot-product attention introduced in [148]. The structure of f_{comb} is presented in Figure 4.3.

Given the multi-scaled long-range dependencies $\mathbf{Y}_0, \mathbf{Y}_1$,..., $\mathbf{Y}_{n-1}$, the scale-attended long-range dependencies $\mathbf{Y}$ is computed as:

$$\mathbf{Y} = Reshape(Attend(\mathbf{Y}'_Q, \mathbf{Y}'_K, \mathbf{Y}_{stack})), \tag{4.2}$$

where the $Attend(\cdot)$ function is implemented as the scaled dot-product attention while $Reshape(\cdot)$ reshapes the output of the $Attend(\cdot)$ function to match that of the original input feature $\mathbf{X} \in \mathbb{R}^{T\times H\times W\times C}$. The $Attend(\cdot)$ function is formulated as:

$$Attend(\mathbf{Y}'_Q, \mathbf{Y}'_K, \mathbf{Y}_{stack}) = \sigma(\frac{\mathbf{Y}'_Q {\mathbf{Y}'_K}^T}{\sqrt{C'}})\mathbf{Y}_{stack}, \tag{4.3}$$

where $\sigma(\cdot)$ is the softmax function, which ensures that the weights for all scales add up to 1. $\mathbf{Y}_{stack}$ is obtained by first flattening long-range dependencies of all scales spatiotemporally and stacked along a separate "scale dimension". Both $\mathbf{Y}'_Q$ and $\mathbf{Y}'_K$ are obtained through a three step process: first, the multi-scaled long-range dependencies are upsampled to the same spatial dimension, followed by a spatiotemporal average pooling and flattening operation to obtain a representation for the dependencies of each scale; second, the pooled dependencies are stacked along the separate "scale dimension" to form a stacked representation feature, denoted as $\mathbf{Y}'_{stack}$; third, separate trainable linear layers, $linear\_Q$ and $linear\_K$ are applied to $\mathbf{Y}'_{stack}$ to obtain $\mathbf{Y}'_Q$ and $\mathbf{Y}'_K$. The end product of Equation 4.3 and Equation 4.2 would be the overall long-range dependencies with dynamic weights applied to the dependencies at each scale.

4.2.3 Computational Efficiency Analysis for PNL Module

In this section, we prove the efficiency for extracting the long-range dependencies $\mathbf{Y}$ with our proposed PNL module against the original non-local block. In this proof, we adopt the case where $f(\cdot, \cdot)$ is the embedded Gaussian version. For notation simplicity, we denote $N = T \times H \times W$. Under this notation, the dimension for input $\mathbf{X} \in \mathbb{R}^{T\times H\times W\times C}$ could be simplified as $\mathbf{X} \in \mathbb{R}^{N\times C}$.

We first compute the computation cost for the original non-local block. Although $conv\_\theta$, $conv\_\phi$ and $conv\_g$ operations are convolutional, they are essentially linear multiplicative operations, especially given that their kernel size are of $1 \times 1 \times 1$. They are designed to project the original input to an embedding space with lower dimension. As designed in [62], the embedding space is of dimension $\mathbb{R}^{N \times \frac{C}{2}}$. Similarly, the operation $conv\_z$ as shown in Figure 4.2a is also a linear multiplicative which projects the computed dependencies back from the embedding space. The total computation cost for operations $conv\_\theta$, $conv\_\phi$, $conv\_g$ and $conv\_g$ could therefore be computed as:

$$Cost_{nl,embs} = 4 \times C^2 \times N \times \frac{C}{2} = 2C^3N. \tag{4.4}$$

The subsequent matrix multiplication of the resulting embeddings from $conv\_\theta$ and $conv\_\phi$ would be computed as:

$$Cost_{nl,matmul} = \frac{C}{2} \times N \times N = \frac{1}{2}CN^2. \tag{4.5}$$

The same computation cost also applies to the matrix multiplication between the softmax result of the previous matrix multiplication with the embedding from $conv\_g$. The computation cost of softmax function is negligible compared to the multiplicative computations as listed above. The approximate total computation cost of the original non-local block is thus computed as:

$$\begin{aligned} Cost_{nl} &= Cost_{nl,embs} + 2 \times Cost_{nl,matmul} \\ &= 2C^3N + CN^2. \end{aligned} \tag{4.6}$$

We now consider the computation cost of our proposed PNL module, which utilizes the non-local operation while incorporating regional correlations. To compute the overall computation cost for the PNL module, we first compute the computation cost for the process of obtaining $\mathbf{Y}_k$ from $\mathbf{X}_k$ as indicated in Figure 4.2b, denoted as $Cost_{nl,k}$. The computation of $Cost_{nl,k}$ follows the same procedures as that of the computation of $Cost_{nl}$. However, they differ in two perspectives: first, the channel number of $\mathbf{X}_k$ is $C' = C/n$ and second, $\mathbf{X}_k$ is of dimension $\mathbb{R}^{T \times \frac{H}{2^k} \times \frac{W}{2^k} \times C'}$. Following the notation above,

N_k is computed as:

$$\begin{aligned} N_k &= T \times \frac{H}{2^k} \times \frac{W}{2^k} \\ &= \frac{1}{2^{2k}} T \times H \times W \\ &= \frac{1}{4^k} N. \end{aligned} \tag{4.7}$$

Thus, we could compute $Cost_{nl,k}$ as:

$$\begin{aligned} Cost_{nl,k} &= 2{C'}^3 N_K + C' {N_k}^2 \\ &= \frac{1}{n^3} 2C^3 \frac{1}{4^k} N + \frac{1}{n} C \times (\frac{1}{4^k} N)^2 \\ &= \frac{1}{n^3 \times 4^k} 2C^3 N + \frac{1}{n \times 16^k} C N^2. \end{aligned} \tag{4.8}$$

Hence, the total computation cost of obtaining the multi-scaled long-range dependencies in our PNL module can be computed as:

$$\begin{aligned} Cost_{pnl,dep} &= \sum_{k=0}^{n-1} Cost_{nl,k} \\ &= (\sum_{k=0}^{n-1} \frac{1}{4^k}) \frac{1}{n^3} 2C^3 N + (\sum_{k=0}^{n-1} \frac{1}{16^k}) \frac{1}{n} C N^2. \end{aligned} \tag{4.9}$$

The scale of the scaled features must be a positive integer with the constraints of $\lceil \frac{H}{2^k} \rceil \geq 2$ and $\lceil \frac{W}{2^k} \rceil \geq 2$, where $k \in [0, (n-1)]$, such that regional correlations could be computed. Therefore, n could only take the integer values that satisfy the above constraints, subject to the spatial size of the input feature. As defined in Section 4.2.2, n is strictly larger than 1. Therefore, the largest $Cost_{pnl,dep}$ is obtained with $n = 2$ with $Cost_{pnl,dep} = \frac{5}{32} 2C^3 N + \frac{17}{32} C N^2$. Meanwhile, the computation with regards to f_{comb} is negligible compared to the computation cost of obtaining the dependencies. The above proof clearly proves that our proposed PNL is more efficient than the original non-local block in terms of requiring lower computation cost.

4.3 Experiments and Discussion

In this section, we present the evaluation results of our proposed PNL module. The evaluation is conducted through action recognition experiments on two public benchmark

datasets. We present state-of-the-art results with a competitive architecture. Further visualization results are also presented to justify the effectiveness of our proposed module.

4.3.1 Experimental Settings

Datasets and Baselines. For the action recognition task, we conduct experiments on two challenging public benchmark datasets: Mini-Kinetics [55] and UCF101 [74]. The Mini-Kinetics is a subset of the Kinetics-400 [76] dataset, with 200 of its categories. It contains a total of 80K training data and 5K validation data. To obtain the state-of-the-art result on the Mini-Kinetics dataset, we instantiate MFNet [58] as the baseline thanks to its outstanding performance on Kinetics-400 dataset.

The UCF101 [74] dataset contains 13,320 videos with 101 categories. For the UCF101 dataset, we follow the settings as in previous works [53, 58], and adopt the three train/test splits for evaluation. We report the average top-1 accuracy over the three splits. Our proposed PNL module can be used with any current CNN networks. Due to the high performance of MFNet, the effectiveness of our proposed PNL module may not be obvious. Thanks to its the steady performance, ResNet-50 [28] is adopted as the additional baseline for experiments on the UCF101 dataset, denoted here as R-50. We adopt the exact same architecture configurations as in [62], where the temporal dimension is trivially addressed through pooling operations and the convolutional kernels are of size $1 \times k \times k$.

Implementation Details. Our experiments are all implemented using PyTorch [149]. Following the implementation in [58], the input is a frame sequence with each frame of size 224×224. For the MFNet baseline, we follow the implementation in [58] and sample a sequence of 16 frames. Whereas for the R-50 baseline, we sample a sequence of 32 frames as suggested in [62]. To accelerate our training, we utilize the pretrained model of MFNet trained on Kinetics-400, and the pretrained model of R-50 trained on ImageNet [150]. The stochastic gradient descent algorithm [136] is used for optimization, with the weight decay set to 0.0001 and the momentum set to 0.9. Our initial learning rate is set to 0.01. Similar to the original non-local block [62], we ensure that the initial state of the entire PNL module to be an identity mapping. This further ensures that the proposed PNL modules can be inserted into any pretrained networks while maintaining its initial behavior.

Pairwise Function	Top-1	Top-5
MFNet baseline	78.35%	94.65%
Embedded Gaussian	82.16%	95.83%
Gaussian	81.68%	95.51%
Dot Product	81.45%	95.54%
Concatenation	81.79%	95.36%

TABLE 4.1: **Ablation 1 - Type of pairwise function:** A single PNL module with $n = 4$ with different types of pairwise function $f(\cdot, \cdot)$ is inserted into the MFNet baseline. All are inserted to the last multi-fiber unit right before the end of the $conv4$ stage.

PNL position	Top-1	Top-5
MFNet baseline	78.35%	94.65%
$conv2$	81.55%	95.48%
$conv3$	81.79%	95.63%
$conv4$	82.14%	95.74%
$conv5$	81.45%	95.46%

TABLE 4.2: **Ablation 2 - Position of PNL:** A single PNL module with $n = 3$ is inserted into the MFNet baseline. The insertion is located at the last multi-fiber unit right before the end of each stage.

Combination Function	Top-1	Top-5	# Params	FLOPs
MFNet baseline	78.35%	94.65%	7.843M	11.176G
Vanilla concatenation	81.93%	95.37%	7.911M	11.208G
Self-attention mechanism	82.16%	95.83%	7.92M	11.218G

TABLE 4.3: **Ablation 3 - Type of combination function:** A single PNL module with $n = 4$ is inserted into the MFNet baseline at the last multi-fiber unit right before the end of $conv4$. The multi-scaled long-range dependencies are combined with different types of f_{comb}.

4.3.2 Ablation Experiments

We obtain an optimal form of the PNL module, while verifying our design through ablation experiments. The ablation experiments are all conducted on the Mini-Kinetics dataset utilizing the MFNet baseline.

Pairwise Function. We first discuss the effect of the pairwise function $f(\cdot, \cdot)$ in the non-local block. Following [62], we utilize four types of pairwise functions, namely embedded Gaussian, Gaussian, dot product and concatenation. The result of utilizing each pairwise function is as shown in Table 4.1. Consistent improvements can be observed regardless of the pairwise function utilized. Among which, the embedded Gaussian

n scales	Top-1	Top-5
MFNet baseline	78.35%	94.65%
2	81.98%	95.59%
3	82.14%	95.74%
4	82.16%	95.83%

(A) Insert at the $conv4$ stage

n scales	Top-1	Top-5
MFNet baseline	78.35%	94.65%
2	81.41%	95.33%
3	81.55%	95.48%
4	81.64%	95.55%
5	81.68%	95.62%
6	81.70%	95.65%

(B) Insert at the $conv2$ stage

TABLE 4.4: **Ablation 4 - Number of scales:** A single PNL module with different scales of dependencies is inserted into the MFNet baseline at $con4$ stage and at the $conv2$ stage.

function as depicted in Figure 4.2a achieves the best performance. Therefore, the pairwise function $f(\cdot, \cdot)$ would be the embedded Gaussian version by default for the rest of the experiments.

Position of PNL Module. Table 4.2 compares the result where a single PNL module is inserted to the different stages of the MFNet baseline. Note that the number of scales n is constrained by the size of the input feature, as mentioned in Section 4.2.3. The higher level of the feature map, the smaller the spatial size of the input feature is, which results in the smaller number of scales n. For fair comparison, the number of scales for the inserted PNL module should be the same across all positions. We therefore adopt the number of scales for the feature map from the $conv5$ stage, which is of spatial size 7×7. This results in the number of scales to be set at $n = 3$. The improvement of adding the single PNL module gradually increases with the PNL module inserted into deeper stages until the $conv4$ stage. However, the improvement by adding PNL module decreases sharply when then PNL module is inserted at the $conv5$ stage. The fact that the spatial dimension of feature map at $conv5$ stage is too small (7×7) such that precise spatial dependencies could not be obtained even at the original feature map scale could be a reason of the sudden drop in improvement. This phenomena is inline with that observed in [62], where inserting the original non-local block at the last convolution stage also results in the smallest improvement. Thanks to the best performance obtained by

inserting at the $conv4$ stage, we insert PNL module right before the last multi-fiber unit of the $conv4$ stage by default. The multi-fiber unit in the MFNet baseline is equivalent to a residual block in the ResNet baseline.

Combining Multi-scaled Dependencies with f_{comb}. As mentioned in Section 4.2.2, the multi-scaled dependencies obtained from the multi-scaled features are combined with a combination function f_{comb}. Here we compare the results utilizing two different types of combination function: a vanilla concatenation function, and the function utilizing self-attention mechanism as proposed in Section 4.2.2. The results are presented in Table 4.3. In addition to the Top-1 and Top-5 accuracies, we also compare the number of parameters and required computation FLOPs with respect to the different combination functions. It can be observed that our proposed self-attended f_{comb} outperforms the vanilla concatenation combination by 0.26%. This is at a cost of only $0.09M$ extra parameters, which is less than 0.12% increase in parameter size. This indicates that our proposed f_{comb} is both effective and efficient, with a negligible computation cost.

Number of Scales. Table 4.4 shows the result of utilizing different numbers of scales in the PNL module. We first insert the single PNL module to the $conv4$ stage of MFNet, where the input feature is of spatial size 14×14. Due to the constraint of n as defined in Section 4.2.3, the number of scales is limited to a maximum number of $n = 4$ when the PNL module is inserted right before the last multi-fiber unit of the $conv4$ stage. Note that when $n = 1$, the PNL module would be exactly same as the original non-local block. Hence we would not discuss the case where $n = 1$. The results in Table 4.4a shows that with the increase in number of scales, the improvement brought by the PNL module would slightly increase. This indicates that the region correlations of different scales complement each other. The additional region correlations at different scales enhance the ability of the network to model long-range dependencies, which results in the improvement of the network's action recognition accuracy. From Section 4.2.3, it is also clear that with the increase in n, the computation cost of PNL module decreases. Hence, when $n = 4$ scales are utilized, our proposed PNL module is both effective and efficient.

To further prove the improvement of PNL module with the increase in number of scales, we insert the PNL module before the last multi-fiber unit of the $conv2$ stage. With a larger spatial size of 56×56 for the feature at the $conv2$ stage, the maximum number of scales is now limited to $n = 6$. The results as shown in Table 4.4b coincide with that in Table 4.4a, where the improvement brought by the PNL module would also increase

	Method	Mini-Kinetics Top-1	# Params	FLOPs
Two-stream CNNs	MARS [151]	73.5%	-	-
	ResFrame TS [152]	73.9%	-	-
	I3D (TS) [52]	78.7%	25.0M	$>$107.9G
3D CNNs	C3D [51]	66.2%	33.3M	-
	I3D (RGB) [52]	74.1%	12.06M	107.9G
	(2+C1)D [153]	75.74%	**7.3M**	31.9G
	S3D [55]	78.0%	8.77M	43.47G
	MFNet [58]	78.35%	7.84M	**11.17G**
CNN with long-range dependencies	Res50-NL [62]	77.53%	27.66M	19.67G
	Res50-CGD [154]	77.56%	25.58M	17.88G
	Res50-CGNL [63]	77.76%	27.2M	19.16G
	MFNet-NL	79.74%	8.15M	11.66G
Ours	**MFNet-PNL**($\times$1)	82.16%	7.92M	11.22G
	MFNet-PNL($\times$5)	**83.09%**	8.12M	11.38G

TABLE 4.5: Comparison of top-1 and top-5 accuracy, number of parameters and computation cost in FLOPs with state-of-the-art methods on the Mini-Kinetics datasets.

Method	Top-1	Top-5	# Params	Flops
R-50	81.62%	94.62%	23.92M	10.29G
MFNet	96.00%	-	7.77M	10.78G
R-50 + NL	82.88%	95.74%	26.38M	18.72G
R-50 + CGNL	83.38%	95.42%	26.22M	18.23G
R-50 + PNL($\times$1)	85.22%	95.82%	24.46M	13.31G
R-50 + PNL($\times$5)	86.75%	96.43%	25.98M	16.56G
MFNet-NL	96.91%	-	8.07M	11.58G
MFNet-PNL($\times$1)	97.53%	-	7.79M	11.14G
MFNet-PNL($\times$5)	97.92%	-	8.02M	11.25G

TABLE 4.6: Comparison of top-1 and top-5 accuracy, number of parameters and computation cost in FLOPs of R-50 and MFNet, as well as their variants on the UCF101 dataset. The parameter size and computation FLOPs are lower for the same network than that tested on Mini-Kinetics due to the fewer number of classes. We do not report the top-5 accuracies for networks with MFNet baseline due to its saturation towards 100%.

with an increase in number of scales. This further proves the fact that the long-range dependencies captured at different scales complement each other. As the best result is obtained through inserting the PNL module to the $conv4$ stage, for the rest of the experiments, n would be set to 4 by default.

4.3.3 Results and Comparison

Table 4.5 shows the comparison of top-1 accuracy on the Mini-Kinetics dataset with other current state-of-the-art methods which include:

1. *Two-stream CNN methods:* MARS [151], Residual Frame with two-stream input (ResFrame TS) [152] and I3D with two-stream input [52].
2. *3D CNN methods:* C3D [51], I3D with RGB input [52], (2+C1)D [153], MFNet [58] and S3D [55].
3. *CNN with long-range dependencies:* Res50-NL [62], Res50-CGD [154], Res50-CGNL [63] and MFNet with non-local block inserted (MFNet-NL).

The above methods are compared with MFNet-PNL($\times 1$) which includes only a single PNL module with the MFNet baseline, and MFNet-PNL($\times 5$) which includes five PNL modules. For the single PNL module case, the PNL module is inserted right before the last multi-fiber unit of the $conv4$ stage of the MFNet baseline. For the five PNL modules case, the PNL modules are inserted to every other multi-fiber unit of the $conv3$ and the $conv4$ stages of the MFNet baseline. For this experiment, we set our batch size to 64 for the Mini-Kinetics dataset and conduct the experiments using two NVIDIA GP100 GPUs.

The results in Table 4.5 clearly show that with the addition of our proposed PNL module, the network achieves the best result on the Mini-Kinetics dataset with limited increase in the number of parameters and computation cost compared to the original MFNet baseline. By inserting a single PNL module, the network achieves a 3.81% increase over the baseline model. MFNet-PNL($\times 1$) which utilizes a single PNL module also outperforms the network with the same MFNet baseline but utilizes the original non-local block, denoted as MFNet-NL. In contrast, a single PNL module has 0.22M less parameters and requires 0.42G less FLOPs compared to the original non-local block. The optimal network performance on Mini-Kinetics is obtained by adding five PNL modules, which increases the classification accuracy by 3.35% compared to the baseline. It can be noted that even with five PNL modules added, the total number of parameters and required computation FLOPs are both lower than the network with the original non-local block. This further proves the effectiveness and efficiency of our proposed PNL module.

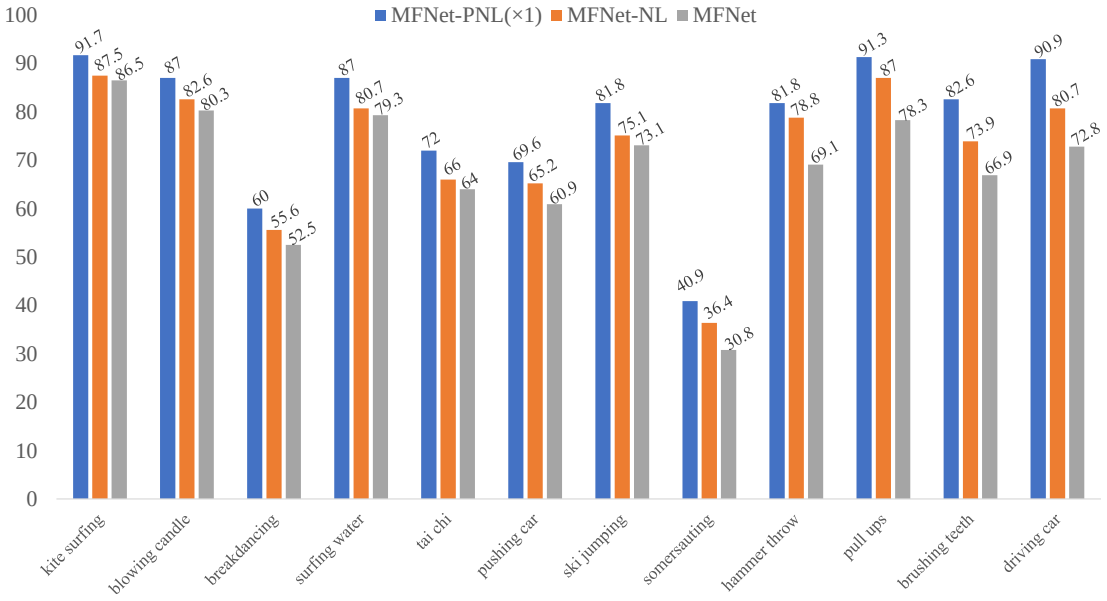


FIGURE 4.4: Detailed comparison of accuracy per class on the Mini-Kinetics between the original MFNet baseline with networks resulting from insertion of a single PNL module (MFNet-PNL($\times 1$)) or a single non-local block (MFNet-NL). Here we present the accuracies of 12 classes where MFNet-PNL($\times 1$) outperforms the original MFNet baseline by a margin of at least 5%. In all classes presented, the MFNet-PNL($\times 1$) also outperforms MFNet-NL.

Besides testing on the Mini-Kinetics dataset, we also conduct experiments on the UCF101 dataset. Here we further utilize a simpler ResNet-50 baseline in addition to the MFNet baseline to showcase the effectiveness of the PNL module. The result is as presented in Table 4.6. Here a single non-local block or PNL module is inserted at the exact same location, which is right before the last residual block of $res4$ stage. We also insert five PNL modules to every other residual block of $res3$ and $res4$ stage to the ResNet-50 baseline. By comparison, inserting a single PNL module brings an extra 2.34% increase in top-1 accuracy compared to inserting a single non-local block. At the same time, our PNL module has 1.92M less parameters and requires 5.41G less FLOPs compared to the original non-local block. A further increase of 1.53% in accuracy is achieved by inserting five PNL modules, which still contains less parameters and requires less FLOPs compared to the network that inserts a single non-local block. Consistent improvements over classification accuracy and computation efficiency can also be observed when comparing MFNet-PNL($\times 1$), MFNet-PNL($\times 5$) with MFNet-NL. However, we note that the improvement in classification accuracy utilizing PNL modules is limited, mainly due to the fact that there is little room for improvement for the MFNet baseline on UCF101.

			
Kitesurfing	**Breakdancing**	**Surfing Water**	**Tai Chi**
MFNet-PNL (x1)	**MFNet-PNL (x1)**	**MFNet-PNL (x1)**	**MFNet-PNL (x1)**
Kitesurfing 77.79	**Breakdancing 31.17**	**Surfing Water 81.26**	**Tai Chi 66.04**
Windsurfing 11.01	Somersaulting 29.06	Water Skiing 7.70	Lunge 11.47
Surfing Water 6.87	High Kick 17.53	Crossing River 5.91	Dancing Ballet 6.94
MFNet-NL	**MFNet-NL**	**MFNet-NL**	**MFNet-NL**
Surfing Water 44.19	Somersaulting 85.58	Water Skiing 74.91	Lunge 44.28
Kitesurfing 40.68	Breakdancing 11.48	Slacklining 6.06	Tai Chi 21.91
Windsurfing 7.86	Capoeira 0.91	Surfing Water 4.25	Dancing Ballet 15.68
MFNet	**MFNet**	**MFNet**	**MFNet**
Surfing Water 44.19	Capoeira 95.58	Water Skiing 65.78	Dancing Ballet 41.11
Kitesurfing 30.18	Somersaulting 1.48	Crossing River 6.24	Lunge 21.91
Windsurfing 17.33	Breakdancing 0.66	Slacklining 3.48	Tai Chi 15.26

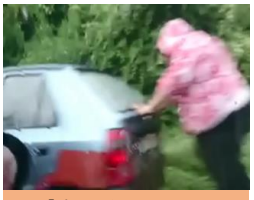
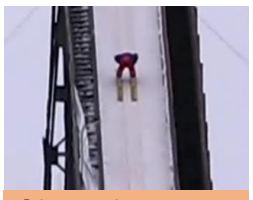
			
Pushing Car	**Ski Jumping**	**Hammer Throw**	**Driving Car**
MFNet-PNL (x1)	**MFNet-PNL (x1)**	**MFNet-PNL (x1)**	**MFNet-PNL (x1)**
Pushing Car 54.92	**Ski Jumping 72.78**	**Hammer Throw 41.18**	**Driving Car 69.10**
Driving Car 12.18	Abseiling 22.59	Golf Driving 14.34	Air Drumming 28.59
Crossing River 10.81	Water Skiing 1.39	Chopping Wood 10.45	Headbanging 1.91
MFNet-NL	**MFNet-NL**	**MFNet-NL**	**MFNet-NL**
Driving Car 66.18	Abseiling 88.01	Golf Driving 19.54	Air Drumming 67.18
Pushing Car 15.37	Ski Jumping 11.78	Chopping Wood 16.92	Headbanging 14.29
Crossing River 9.28	Ice Climbing 0.02	Hitting Baseball 14.52	Driving Car 11.82
MFNet	**MFNet**	**MFNet**	**MFNet**
Crossing River 60.87	Water Skiing 83.24	Chopping Wood 29.45	Air Drumming 92.18
Driving Car 11.57	Abseiling 12.59	Hitting Baseball 11.35	Headbanging 4.20
Pushing Car 9.28	Ski Jumping 1.53	Shot Put 9.23	Finger Snapping 1.82

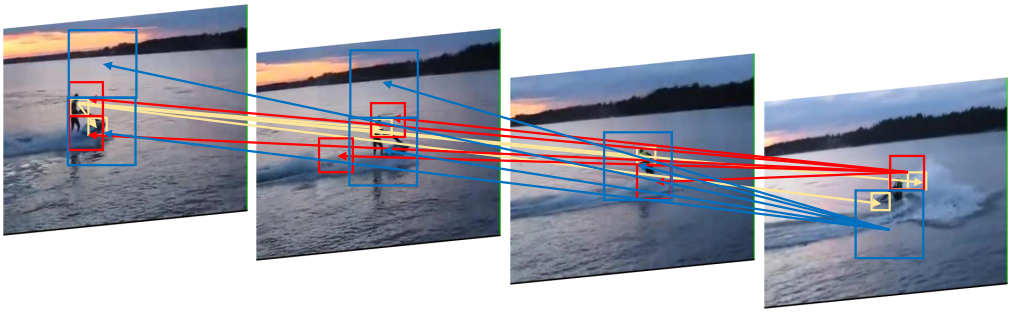
FIGURE 4.5: Eight examples taken from the 12 classes presented in Figure 4.4. The numbers on the right of each class shows the probability of the class from the classifier in percentages. We show three classes with highest probability. The class with the highest probability is the result of the top-1 classification, highlighted in green.

The above results further justifies the effectiveness and efficiency of our PNL module compared to the original non-local block.

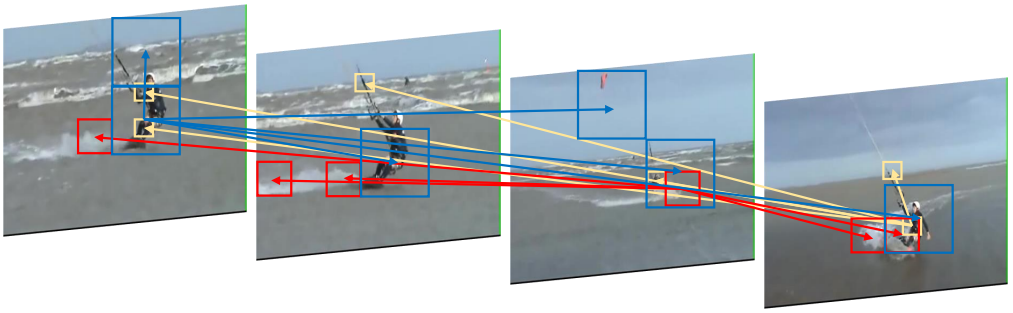
We further investigate the improvements over different actions and present the comparisons of performance between the baseline MFNet network with that of inserted a single non-local block or a single PNL module. Figure 4.4 shows the accuracy of 12 classes from the Mini-Kinetics dataset, where inserting a single PNL module outperforms the original baseline network by a noticeable margin of over 5%. Inserting the single PNL module also outperforms that of inserting a single original non-local block in all of the 12 classes presented. To further illustrate the effectiveness of our PNL module, we present several examples in Figure 4.5 where inserting a single PNL module to the original baseline outperforms the baseline network with or without non-local block inserted. The superior performance over inserting the non-local block in these examples illustrates that modeling regional correlations in long-range dependencies could bring additional information to the network, thus resulting in more effective video features.

4.3.4 Visualization

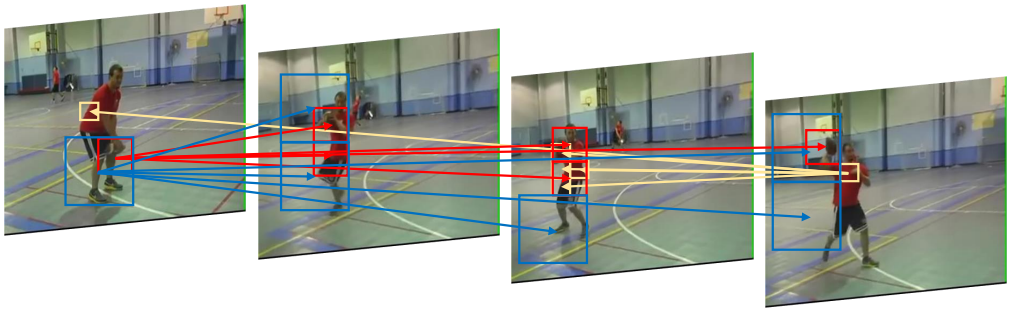
To justify the effectiveness of our proposed module in capturing regional long-range dependencies at multiple scales, we visualize the interactions of the different regions in sample videos. Here for simplicity, we utilize the MFNet-PNL($\times 1$) network. The visualization of the behaviour of our PNL module is as shown in Figure 4.6. It could be observed that the multi-scaled long-range dependencies complements each other, providing effective information towards the final classification. For example, for the action "Kitesurfing" in Figure 4.6b, the smallest scale long-range dependencies, obtained through the original feature map, captures the correlations between the person, the board underneath and the rope above. These correlations imply the involvement of a person and a board within the action. Similar correlations between the person, board and rope may also be found in actions such as "Windsurfing" or "Surfing Water". Hence, if only the correlations at the smallest scale are captured, the action may be mis-classified. A similar example is presented at the top right corner of Figure 4.5, where MFNet-NL and the original MFNet mis-classified a "Kitesurfing" action with the "Surfing Water" action. Whereas the largest scale long-range dependencies, obtained through the sub-sampled scaled feature map, captures the correlation between the person and the kite above. Such region correlation between the person and the kite is unique for the "Kitesurfing" action.



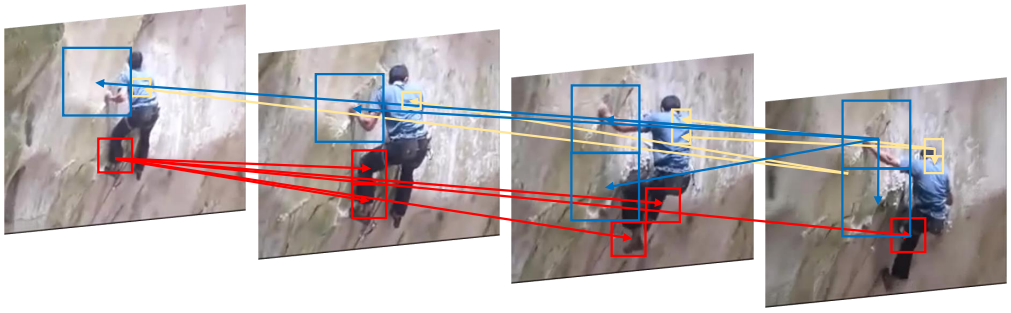
(A) Visualizing Action "Jetskiing"



(B) Visualizing Action "Kitesurfing"



(C) Visualizing Action "Passing Football"



(D) Visualizing Action "Rock Climbing"

FIGURE 4.6: Visualization of the behaviour of our PNL module. From a reference region, we visualize the five of the most correlated regions computed from PNL module at three different scales, shown in different colors. We observe that these correlations complements each other, thus capturing more effective long-range spatiotemporal dependencies. Figure best viewed in color and zoomed in.

Hence, capturing such correlation provides additional information about the action, and enables the network with the PNL modules to classify the action as "Kitesurfing" correctly. The action presented in Figure 4.5 also demonstrates the correct classification with MFNet-PNL($\times 1$).

4.4 Summary

In this chapter, we propose a novel module for effective capturing of long-range spatiotemporal dependencies. The proposed PNL module extends the original non-local block by incorporating regional correlations at multiple scales through a pyramid structural design. Our method obtains state-of-the-art result on the Mini-Kinetics dataset when instantiating MFNet, while bringing significantly less computation cost than the original non-local block. We further justify the design of our PNL module through detailed ablation studies. We further demonstrate the effectiveness of the PNL module by visualizing the captured dependencies in sampled videos.

Chapter 5

ARID: A Comprehensive Study on Recognizing Actions in the Dark and A New Benchmark Dataset

The robustness of deep learning methods towards videos under adverse environments (e.g., dark videos) is an important aspect of method robustness. While current vision-based action recognition deep learning methods, including ACTF and PNL introduced in Chapter 3 and Chapter 4 have achieved remarkable performance in current action recognition datasets, these methods may not generalize well to dark videos, partly due to the lack of relevant datasets. In this chapter, we pioneered the investigation of deep learning methods in dark videos by introducing a novel dataset: the Action Recognition In the Dark (ARID) dataset. The motivation and introduction of the proposed dataset is presented in Section 5.1, with the basic details of the ARID dataset illustrated in Section 5.2. Further, detailed experiments and analysis over the ARID dataset are demonstrated in Section 5.3. Lastly, the paper is summarized in Section 5.4.

5.1 Introduction

Although much progress has been made in vision-based action recognition, current research still mostly focuses on videos shot under normal illumination. This is partly due to the fact that current benchmark datasets for action recognition [52, 72, 74] are normally collected from web videos, which are shot mostly under normal illumination. Yet

videos under normal illumination conditions are not available in many cases, such as night surveillance [155, 156], and self-driving at night [157]. It is true that additional sensors, such as infrared or thermal imaging sensors, could be utilized for recognizing actions in the dark. However, such sensors are of high cost and the deployment of such sensors on a large scale may not be practical. Hence we focus on action recognition in the dark without the need for additional sensors. To this end, we collected a new dataset: Action Recognition In the Dark (ARID) dataset, dedicated to the task of action recognition in dark videos. To the best of our knowledge, it is the first dataset focused on human actions in the dark.

Currently, there already exists a large number of videos in various datasets, shot under normal illumination. Intuitively, we may make use of these videos to create synthetic dark videos. In this chapter, we will show the distinct characteristics of real dark videos that cannot be replicated by synthetic dark videos through detailed analysis and comparison. This implies that a real dark video dataset is necessary for the task of action recognition in the dark.

Recently, neural networks, especially convolutional neural network (CNN) based solutions have proven to be effective for various computer vision tasks, including the action recognition task. For action recognition, state-of-the-art results on previous action recognition datasets are mostly achieved through either two-stream networks [2, 158] or 3D-CNN-based networks [51, 159]. To gain further understanding of the challenges faced with action recognition in dark videos, we analyze how dark videos affect current action recognition models. Additionally, we explore potential methods for substantial improvements in action recognition accuracy utilizing current action recognition models.

In summary, we explored the task of action recognition in dark videos. The contribution of this chapter is three-fold and is summarized as follows:

* We develop a new dataset, the Action Recognition in the Dark (ARID) dataset, dedicated to the task of recognizing actions in dark videos, which, to the best of our knowledge, is the first dataset focused on human actions in dark videos.
* We discover the distinct characteristics of real dark videos through statistical and visual analysis and comparison with synthetic dark videos.

* We benchmark the performance of various action recognition models on our dataset while exploring potential methods to improve action recognition accuracy with current models, and reveals challenges in the task of action recognition in dark videos.

The rest of this chapter is organized as follows: in Section 5.2, we introduce our proposed Action Recognition In the Dark (ARID) dataset in detail. After that, we benchmark current action recognition models on our dataset, with a thorough analysis of our dataset and its challenges in Section 5.3. Finally, we conclude the chapter in Section 5.4.

5.2 Action Recognition In the Dark (ARID) Dataset

FIGURE 5.1: Sample frames for each of the 11 action classes of the ARID dataset. All samples are manually tuned brighter for display purposes.

Although a small number of videos taken in the dark do exist in current action recognition benchmark datasets, such as Kinetics [52] and HMDB51 [72], the task of human action recognition in dark environments has rarely been studied. This is partly due to the very low proportion of dark videos in current benchmark datasets, and a lack of datasets dedicated to action analysis in the dark. To bridge the gap in the lack of dark video data, we introduce a new Action Recognition In the Dark (ARID) dataset. In this session, we take an overview of the dataset from three perspectives: the action classes, the process of data collection as well as some basic statistics of our ARID dataset.

5.2.1 Action Classes

The ARID dataset includes a total of 11 common human action classes. The list of action classes can be categorized into two types: *Singular Person Actions*, which includes jumping, running, turning, walking, and waving; and *Person Actions with Objects*, which includes drinking, picking, pouring, pushing, sitting, and standing. Figure 5.1 shows the sample frames for each of the 11 action classes in the ARID dataset.

5.2.2 Data Collection

The video clips in the ARID dataset are collected using 3 different commercial cameras available in the market. The clips are shot strictly during night hours. All clips are collected from a total of 11 volunteers, among which 8 males and 3 females. We collected the clips in 9 outdoor scenes and 9 indoor scenes, such as carparks, corridors and playing fields for outdoor scenes, and classrooms and laboratories for indoor scenes. The lighting condition of each scene is different, with no direct light shot on the actor in almost all videos. In many cases, it is challenging even for the naked eye to recognize the human action without tuning the raw video clips.

5.2.3 Basic Statistics

The ARID dataset contains a total of 3,784 video clips, with each class containing at least 110 clips. The clips of a single action class are divided into 12-18 groups with each group containing no less than 7 clips. The clips in the same group share some similar

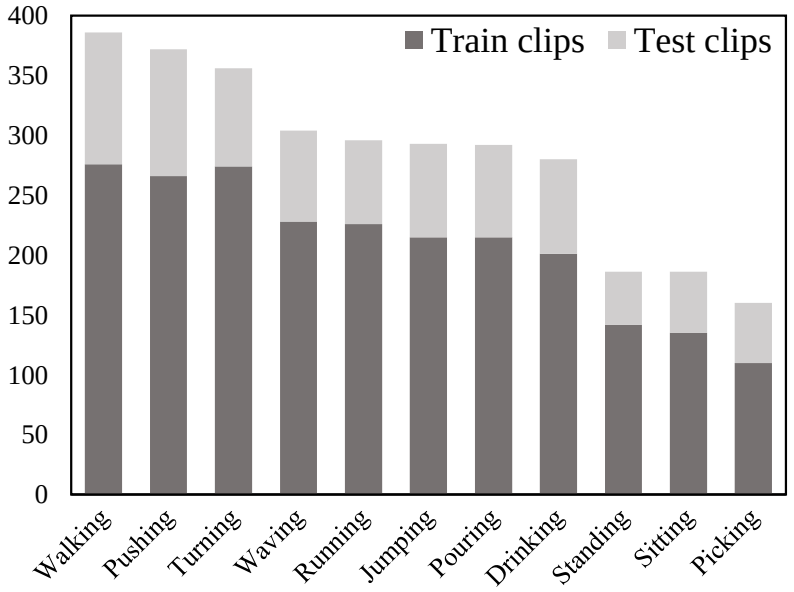


FIGURE 5.2: The distribution of clips among all action classes in ARID. The dark grey and light grey bars indicate the number of clips in the train and test partitions.

features, such as being shot under similar lighting conditions or shot with the same actor. Figure 5.2 shows the distribution of clips among all the classes.

The training and testing sets are partitioned by splitting the clip groups, with 70% of the groups in the training partition, and the remaining 30% of the groups in the testing partition. We selected three training/testing splits, such that each group would have an equal chance to be present in either the training partition or the testing partition.

The video clips are fixed to a frame rate of 30 FPS with a resolution of 320×240. The minimum clip length is 1.2 seconds with 36 frames, and the duration of the whole dataset is 8,721 seconds. The videos are saved in `.avi` format and are compressed using the *DivX* codec.

5.3 Experiments and Discussions

In this section, we gain further understandings of our proposed ARID dataset through a detailed analysis of the dataset. The main objectives are two-fold: 1) validate the necessity of a video dataset collected in the real dark environment and 2) provide a benchmark for current action recognition models and methods while revealing the challenges with regards to the task of action recognition in dark videos. In the following, we first introduce the experiment settings along with the construction of a synthetic dark video

dataset. We then introduce methods used to enhance dark video frames in ARID in an effort to improve action recognition accuracy. We then analyze our introduced ARID dataset in detail through three perspectives: statistical and visual analysis of ARID, analysis of the ARID classification results, and visualization of extracted features from videos in ARID.

5.3.1 Experimental Settings

To obtain the action recognition results on our ARID dataset, we utilize both two-stream models and 3D-CNN-based models with PyTorch[149]. For our experiments, the inputs to all 3D-CNN-based models are sequences of 16 sampled frames, with each frame resized to 224×224. The inputs to the spatial stream of our two-stream models are RGB sampled frames, resized to 224×224. Whereas the inputs to the temporal stream are stacks of optical flow pre-computed on both x and y axis and resized to the same size as the input for the spatial stream, i.e. 224×224. To accelerate training, we utilize the pretrained models pretrained on the Kinetics [52] or ImageNet [30] dataset when available. Due to the constraints in computation power, a unified batch size of 16 is applied to all 3D-CNN model experiments, while a unified batch size of 32 is applied to all two-stream model experiments. The action recognition results are reported as the average top-1 and average top-5 accuracies over the three splits.

Compared to collecting a new dataset for the dark environment, it is more intuitive to obtain "dark" videos through synthesizing dark videos from current publicly available datasets, which mainly consist of videos shot under normal illumination. To showcase the necessity of a real dark video dataset, we compare a synthetic dark video dataset with our introduced ARID. More specifically, the synthetic dark video dataset is constructed based on the HMDB51 [72] dataset, denoted as HMDB51-dark. We synthesize dark videos by gamma intensity correction formulated as:

$$D(t, x, y) = I(t, x, y)^{(1/\gamma)} \tag{5.1}$$

where $D(t, x, y)$ is the value of the pixel in the synthetic dark video, located at spatial location (x, y) at the t^{th} frame, and $I(t, x, y)$ is the pixel value at the corresponding location in the original video. Both $D(t, x, y)$ and $I(t, x, y)$ are normalized to the range of $[0, 1]$. γ is the parameter that controls the degree of "darkness" in the synthetic dark

video, typically in the range of $[0.1, 10]$, where a smaller number would result in lower pixel values, producing darker synthetic videos.

FIGURE 5.3: Comparison of a sample frame of normal illumination taken from the video in the HMDB51 dataset (left) and the corresponding frame taken from the synthetic dark video from our HMDB51-dark dataset (right). The frame in the original HMDB51 video has more details, including the background and a clearer contour of the actor. Best viewed in color.

We note that the dark videos collected in our ARID are shot under different illumination conditions. To simulate the differences in illumination within dark videos, we apply different γ values when synthesizing dark videos. More specifically, the γ value is obtained randomly from a normal distribution $\mathcal{N}(\mu, \sigma^2)$ with the constraint of $\gamma \geq 0.1$. Here the mean μ is set to 0.2 and the standard deviation σ is set to 0.07. Figure 5.3 shows the comparison of sample frames of videos from the original HMDB51 dataset with that from the corresponding synthetic dark videos.

5.3.2 Frame Enhancement Methods

For humans to better recognize actions in dark videos, an intuitive method is to enhance each dark video frame, such that objects and actions are visually clearer. In this chapter, to better understand the effect of dark videos on current action recognition models, we investigate the effect of applying different frame enhancement methods on ARID. Specifically, we applied five common frame enhancement methods: Histogram Equalization (**HE**) [160], Gamma Intensity Correction (**GIC**), **LIME** [161], **BIMEF** [162] and **KinD** [163]. Among them, **HE** and **GIC** are traditional image enhancement methods. In particular, **HE** produces higher contrast images, whereas **GIC** is often used to adjust the luminance of images. Whereas both **LIME** and **BIMEF** are based on the Retinex theory [164], which assumes that images are composed of reflection and illumination. **LIME** estimates the illumination map of dark images while imposing a structure prior to the initial illumination map, while **BIMEF** proposes a multi-exposure fusion

algorithm. **KinD** is a more recent deep neural network-based method utilizing a two-stream structure for simultaneous reflectance restoration and illumination adjustment. KinD is implemented with weights pretrained on the LOL Dataset [80]. The results of applying the above methods to the ARID dataset are denoted as ARID-HE, ARID-GIC, ARID-LIME, ARID-BIMEF, and ARID-KinD respectively. The **GIC**is also applied to the synthetic dark dataset HMDB51-dark, whose result is denoted as HMDB51-dark-GIC.

5.3.3 Statistical and Visual Analysis of ARID

To better understand the necessity of real dark videos, we compute and compare the statistics of the ARID dataset with the HMDB51 dataset as well as the synthetic HMDB51-dark dataset. Figure 5.5 displays the RGB values and Y value histograms of datasets ARID, ARID-GIC, HMDB51, HMDB51-dark and HMDB51-dark-GIC. We also provide the bar charts of the mean value and standard deviation value of the above datasets as shown in Figure 5.4. The gamma values for obtaining both ARID-GIC and HMDB51-dark-GIC are both set to $\gamma = 5$.

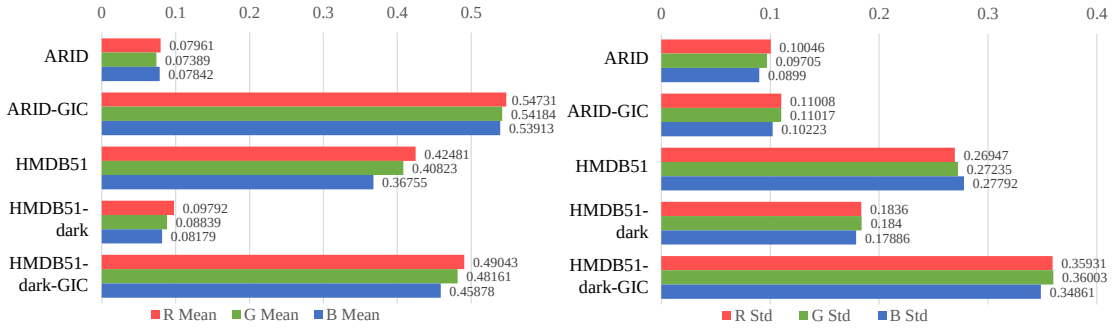


FIGURE 5.4: Bar charts of the RGB mean (left) and standard deviation (right) values for various datasets, including ARID and its **GIC** enhanced output ARID-GIC, HMDB51 and the synthetic dark dataset HMDB51-dark, as well as the **GIC** enhanced output of the synthetic dart dataset, HMDB51-dark-GIC. All values are normalized to the range of [0.0. 1.0]. Best viewed in color.

The histograms of ARID, as shown in Figure 5.5(a), depict the characteristics of videos in our ARID dataset. Compared to the original HMDB51, the distribution of RGB values in the ARID dataset is much more concentrated towards the region of lower values. Though background light with higher values can be seen in a small portion of videos, they consist of only a small part of the whole frame, thus have few effects on the overall

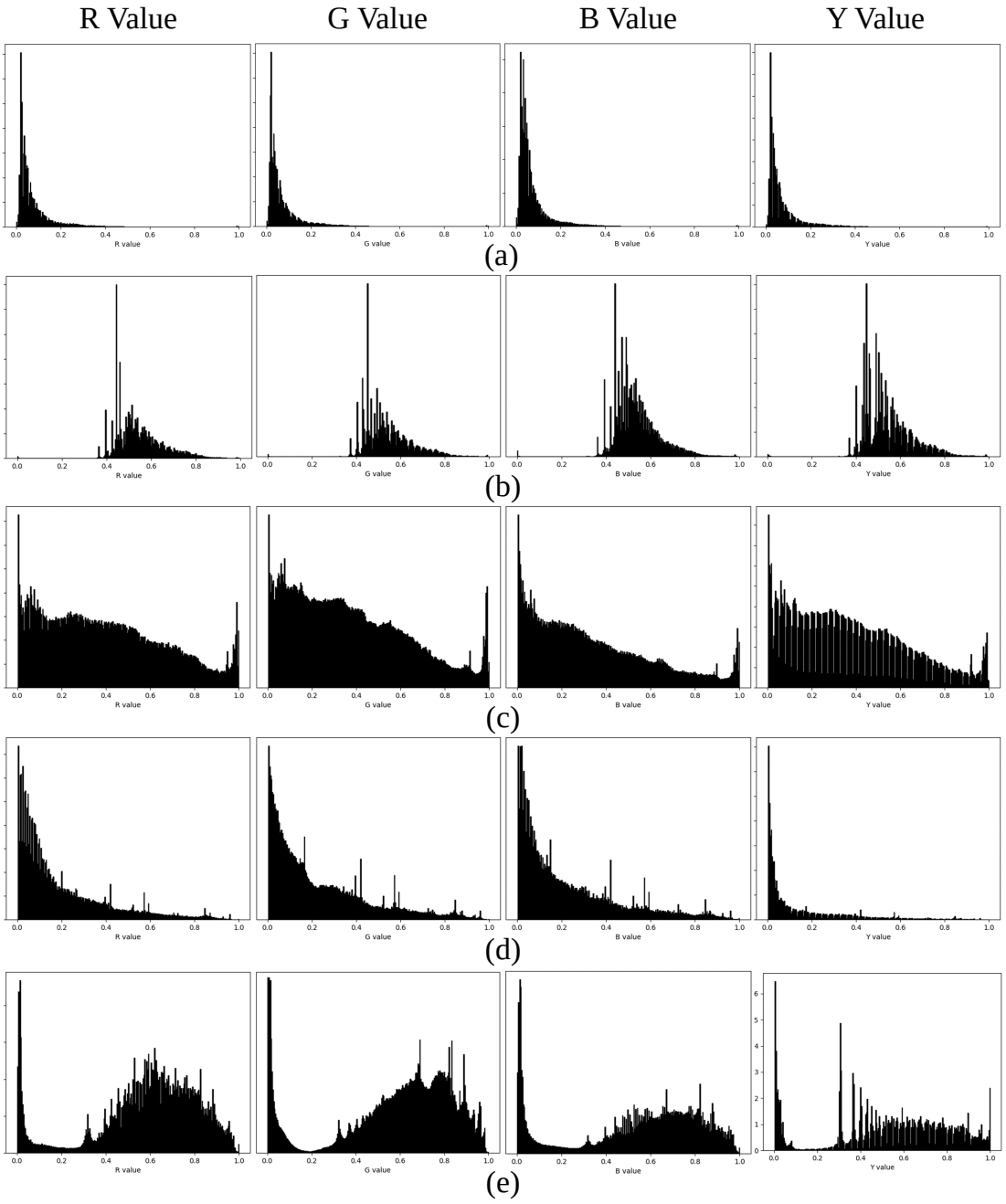


FIGURE 5.5: Histograms for RGB and Y values of (from top to bottom): (a) ARID, (b) ARID-GIC, (c) HMDB51, (d) HMDB51-dark and (e) HMDB51-dark-GIC. All values are normalized to the range of [0.0. 1.0]. Best viewed zoomed in.

histogram. The fact that pixels in ARID possess lower RGB mean and standard deviation values as shown in Figure 5.4 implies that that video frames in ARID are lower in brightness and contrast compare to video frames in HMDB51. This is further justified by the sampled frames and their RGB and Y histograms comparison between ARID and HMDB51 datasets, presented in Figure 5.6. The lower brightness and lower contrast for video frames in ARID make it challenging even for the human naked eye to identify actors and the actions in each frame.

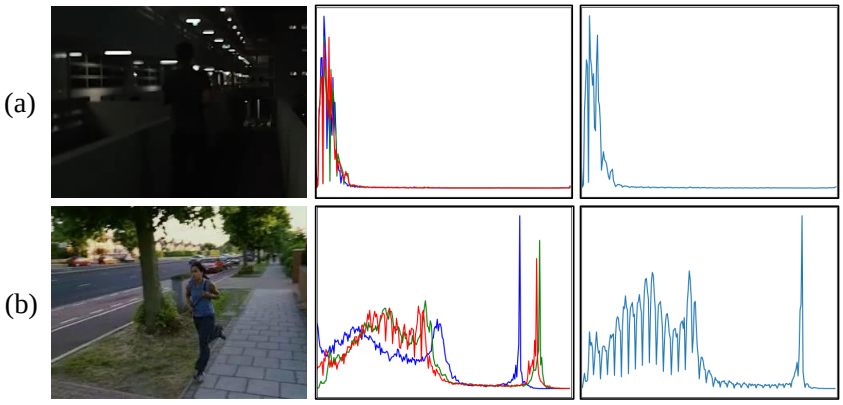


FIGURE 5.6: Comparison of sampled frames and the RGB and Y value histograms of their corresponding videos from (a) ARID dataset and (b) HMDB51 dataset. The RGB (middle) and Y value (right) histograms of the video from the ARID dataset are more concentrated at the lower value. Best viewed in color and zoomed in.

We observe that our real dark dataset ARID and the synthetic dark dataset HMDB51-dark are very similar in terms of the Y value, which reflects the luminance of the video frames. This in part shows that our synthesized operation simulates the real dark environment relatively well in terms of video brightness. However, further comparison in terms of RGB values indicates that the real dark dataset ARID is still lower in both RGB mean and standard deviation values. We notice that although the Y values of both ARID and HMDB51-dark are concentrated in the lower values, the histogram of Y values for HMDB51-dark is more spread out. This matches the observation of the RGB histograms where we notice that there are quite a number of pixels in the HMDB51-dark dataset with relatively high pixel values. This is due to the fact that bright pixels exist in the original HMDB51 dataset, which corresponds to the bright background of the frames. For extreme bright pixels with a normalized value of 1.0, the output synthetic dark pixel value $D_{1.0}(t_{1.0}, x_{1.0}, y_{1.0})$ corresponding to these pixels is still 1.0 according to Equation 5.1. Pixels with high values make up a considerable portion of the overall pixels, as shown in Figure 5.5(c), reaching a small peak near the normalized value of 1.0. Hence the corresponding output pixels in the synthetic dark videos have higher pixel values, which

raises both the mean value and standard deviation of HMDB51-dark, which in terms is reflected as frames with higher brightness and contrast. This indicates that videos from HMDB51-dark would visually be more distinguishable. The sampled frames and their corresponding RGB and Y histograms comparison between ARID and HMDB51-dark datasets, as shown in Figure 5.7 justifies the observation of lower brightness and contrast for video frames in ARID.

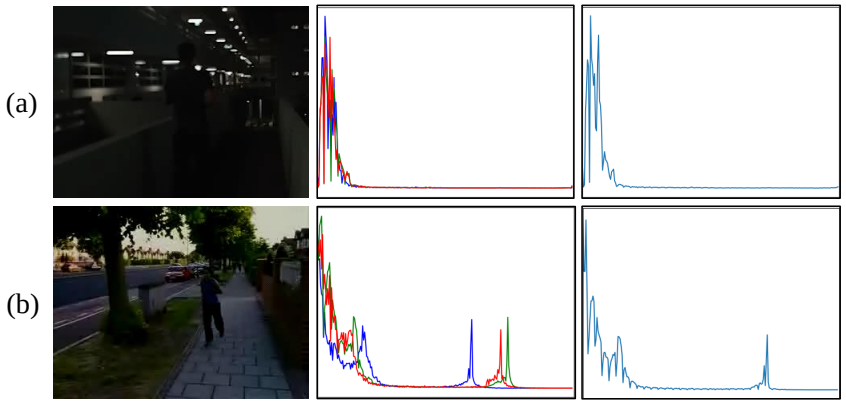


FIGURE 5.7: Comparison of sampled frames and the RGB and Y value histograms of their corresponding videos from (a) ARID dataset and (b) HMDB51-dark dataset. The Y value histogram (right) of the HMDB51-dark video is similar to that of the Y value histogram of the ARID video. However, the RGB histogram (middle) of the video from the ARID dataset is still more concentrated. The peaks of the RGB histogram of the HMDB51-dark video comes from the bright background. Best viewed in color and zoomed in.

As mentioned in Section 5.3.2, the **GIC** method could enhance frames by adjusting the luminance of the frames. By setting $\gamma \geq 1.0$, the resulting pixel value after applying the **GIC** method should be larger than the input pixel value. This is justified through the comparison between Figure 5.5(a) and (b), as well as the comparison between Figure 5.5(d) and (e). In both cases, the RGB histogram of the dataset after applying the **GIC** method shows that pixel values would shift towards regions of larger values quite significantly. This is supported by the fact that RGB mean values for both cases increase. Sampled frames as shown in Figure 5.8 also justifies that **GIC** enhancement greatly increases the visibility of each video frame. Note that Figure 5.8(b) is the direct output after the **GIC** enhancement of Figure 5.8(a). The person seen running can not be clearly observed by the naked eye in Figure 5.8(a), whereas the person becomes more visible in Figure 5.8(b). The brighter and clearer video frames also further justify the comparison of the Y histograms, where the Y histograms for videos after applying **GIC** method would also shift to larger values, indicating a rise in luminance for video frames.

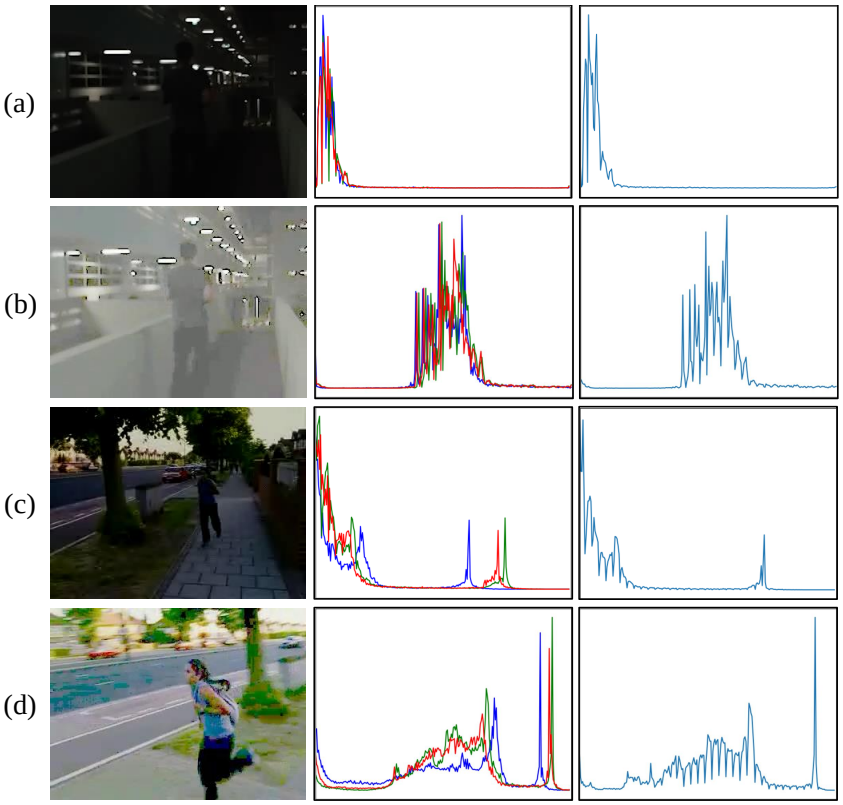


FIGURE 5.8: Comparison of sampled frames and the RGB (middle column) and Y (right column) value histograms of their corresponding videos from (a) ARID dataset, (b) ARID-GIC dataset, (c) HMDB51-dark dataset and (d) HMDB51-dark-GIC dataset. **GIC** enhancement shifts the RGB and Y value histograms towards the larger values, indicating brighter video frames. The RGB and Y values of ARID-GIC are both more concentrated than that of HMDB51-dark-GIC, which matches the low contrast and pale image as shown in the left column. Best viewed in color and zoomed in.

Though both the histograms and sampled frames of ARID-GIC and those of HMDB51-dark-GIC show the effectiveness of **GIC** enhancement in increasing the luminance of dark videos, there are still significant differences between the histograms of either dataset. The most significant difference is that pixel values of ARID-GIC are much more concentrated compared with pixel values of HMDB51-dark-GIC. This is in line with the fact that pixel values of ARID are concentrated compared with pixel values of HMDB51-dark. Unlike the **HE** enhancement where the distribution of pixel values could be altered and creates higher contrast video frames, the **GIC** enhancement focuses on adjusting the illumination of video frames. Hence the characteristic of pixel value distribution would not change drastically after the **GIC** enhancement. The concentration of pixel values indicates that video frames in ARID-GIC are still low in contrast. This is further justified by comparing the sampled frames as shown in Figure 5.8(b) and (d). The sampled frame from ARID-GIC looks pale as compared to the sampled frame from HMDB51-dark-GIC

as shown in Figure 5.8(b).

From the above observation, we can summarize the main characteristic of the real dark videos collected in our ARID dataset: low brightness and low contrast. Though the character of low brightness could be simulated by the synthetic dark videos, the characteristic of low contrast cannot be easily replicated by synthetic dark videos. This is partly due to the bright backgrounds and pixels that commonly exist in videos shot under normal illumination. The fact that synthetic dark videos are higher in contrast compared to real dark videos confirms that real dark videos are necessary and irreplaceable for the task of action recognition in a dark environment.

5.3.4 Classification Results on ARID

In this section, we illustrate how current action recognition models perform in the task of action recognition in the dark on our ARID dataset. We further explore potential ways to improve the performance of action recognition in real dark videos and reveal some challenges faced with action recognition in dark videos.

Model Benchmarking. The performances of current action recognition models are presented in Table 5.1, which includes:

1. **Two-stream models:** the original two-stream method with VGG [26] backbone (VGG-TS) [2], Temporal Segment Network (TSN) [35] and I3D with two-stream inputs (I3D-TS) [52]
2. **3D-CNN based models:** C3D [51], Separable-3D [55], 3D-ShuffleNet [165], 3D-SqueezeNet [166], 3D-ResNet-18 [53], 3D-ResNet-50 [54], 3D-ResNet-101 [54], I3D with RGB input (I3D-RGB) [52], Pseudo-3D-199 [56] and 3D-ResNext-101 [54].

The performance results in Table 5.1 show that among the current two-stream and 3D-CNN-based action recognition models examined, 3D-ResNext-101 achieves the best performance with a top-1 accuracy of 74.73%. We notice that the top-5 accuracies are relatively high for all methods, partly because of the limited number of classes in our dataset. By comparing the performance of I3D with only RGB input and with two-stream inputs, a noticeable increase of 4.49% in accuracy is observed by adding optical

	Models	Top-1 Accuracy	Top-5 Accuracy
Two-stream	VGG-TS	32.08%	90.76%
	TSN	57.96%	94.17%
	I3D-TS	72.78%	99.39%
3D-CNN	C3D	40.34%	94.17%
	Separable-3D	42.16%	93.44%
	3D-ShuffleNet	44.35%	93.44%
	3D-SqueezeNet	50.18%	94.17%
	3D-ResNet-18	54.68%	96.60%
	I3D-RGB	68.29%	97.69%
	3D-ResNet-50	71.08%	99.39%
	3D-ResNet-101	71.57%	99.03%
	Pseudo-3D-199	71.93%	98.66%
	3D-ResNext-101	74.73%	98.54%

TABLE 5.1: Performance of current two-stream and 3D-CNN based action recognition models on ARID dataset.

flow as the temporal input. This increase highly suggests that optical flow could provide a more accurate and effective representation of videos in our ARID dataset. However, such an increase comes at a cost of pre-computing optical flow, which is expensive in both computation power and storage. Hence we focus on 3D-CNN-based models for the rest of this chapter.

We also notice that though our dataset is of relatively small size and has fewer classes than current normal illumination video datasets, there is plenty of room for improvement in accuracy. To explore potential ways for further improving accuracy for dark videos, we selected 3D-CNN-based models, i.e. C3D, I3D-RGB, 3D-ResNet-101, and 3D-ResNext-101, as the baselines for further experiments.

Improvement Exploration with Synthetic Dark Dataset. An intuitive method for improving accuracy is by using frame enhancement methods as introduced in Section 5.3.2. To test whether frame enhancement methods could improve accuracy, we employ the **GIC** method on the synthetic HMDB51-dark dataset due to its larger data size and ease of obtaining dark data from the current datasets. The performance of the chosen 3D-CNN-based models on the synthetic dataset HMDB51-dark and its corresponding **GIC** enhanced HMDB51-dark-GIC is illustrated in Table 5.2.

The results in Table 5.2 exhibit sharp decreases in classification accuracies when the same networks are utilized for the dark data. The decreases in accuracies are expected,

Models	HMDB51-dark	HMDB51-dark-GIC	HMDB51
C3D	21.13%	21.64%	50.13%
I3D-RGB	27.90%	41.79%	54.64%
3D-ResNet-101	42.48%	50.78%	61.70%
3D-ResNext-101	44.90%	58.62%	63.80%

TABLE 5.2: Performance of various 3D-CNN based action recognition models on the synthetic HMDB51-dark and its **GIC** enhanced HMDB51-dark-GIC. The performance of the respective models on the original HMDB51 is presented for reference.

given that dark videos contain fewer details as displayed in Figure 5.3, hence less information could be extracted for recognizing actions. Besides, we also notice consistent increases in accuracies when the **GIC** method is applied to enhance the dark video frames. The degrees of increases vary across different models. However, besides the C3D model, the degrees of increases for the other three models all exceed 8%, which is rather significant. The largest degree of increase of 13.89% is achieved with the I3D-RGB model. As the synthetic data is darkened with random gamma values while the **GIC** enhancement utilizes a fixed gamma value, it is nearly impossible to recover the original videos. Despite this, our results show that the **GIC** enhancement still brings a relatively consistent and significant amount of accuracy improvements for most models through enhancing each video frame.

Applying Frame Enhancement Methods to ARID. The success in applying the straightforward frame enhancement method of **GIC** in increasing classification accuracies for synthetic dark videos gives us a hint on potential ways to improve accuracy for action recognition in real dark videos. To justify if the same **GIC** method could also improve action recognition accuracy on our ARID dataset, we perform experiments on the **GIC** enhanced ARID dataset: ARID-GIC, utilizing the four 3D-CNN based models aforementioned. The results are as presented in Table 5.3 with the Top-1 accuracy and the relative improvements compared to their respective performances on the original ARID dataset.

The results in Table 5.3 illustrate that the action recognition accuracies on the ARID dataset could be improved consistently through **GIC** enhancement with all models, thanks to the increase in the illumination of each video frame as presented in Figure 5.8. The increase in accuracy is consistent with findings in the synthetic dark dataset HMDB51-dark. However, we also notice that the improvements of performances by using **GIC** on ARID are rather limited compared to the improvements in the synthetic

Datasets	Accuracy	C3D	I3D-RGB	3D-ResNet-101	3D-ResNext-101
ARID-GIC	Top-1	44.09%	69.14%	75.15%	78.06%
	Improv.	3.75%	0.85%	3.58%	3.33%
ARID-HE	Top-1	39.49%	63.67%	65.49%	75.82%
	Improv.	-0.85%	-4.62%	-6.08%	1.09%
ARID-LIME	Top-1	39.61%	73.02%	75.45%	77.40%
	Improv.	-0.73%	4.73%	3.88%	2.67%
ARID-BIMEF	Top-1	45.23%	68.89%	68.28%	73.39%
	Improv.	4.89%	0.60%	-3.29%	-1.34%
ARID-KinD	Top-1	46.64%	67.55%	70.59%	69.62%
	Improv.	6.30%	-0.74%	-0.98%	-5.11%
ARID	Top-1	40.34%	68.29%	71.57%	74.73%

TABLE 5.3: Performance of various 3D-CNN based action recognition models on variants of ARID enhanced by **GIC**, **HE**, **LIME**, **BIMEF** and **KinD**. The Improvements (Improv.) are compared with the performances of the respective models on the original ARID dataset, which is also presented for reference.

dark dataset. The degrees of improvements are capped at 3.75%, while three out of the four models experience less improvement than that in the synthetic HMDB51-dark. As **GIC** method is a straightforward method based on simple exponential calculation, we further examine if more sophisticated frame enhancement methods could further improve action recognition accuracy. We thus examine the accuracy on datasets ARID-HE, ARID-LIME, ARID-BIMEF, and ARID-KinD, which are results of the output by frame enhancement methods **HE**, **LIME**, **BIMEF** and **KinD** respectively using the same models. The results are presented in Table 5.3.

Interestingly, Table 5.3 demonstrates that not all frame enhancement methods result in improvements in action recognition accuracies for dark videos. Of all the frame enhancement methods, the most consistent improvement is achieved by the rather simple **GIC** method. Whereas the accuracy drops for most networks when utilizing the recent deep learning-based method **KinD** and the **HE** method. We also observe that none of the improvements matches that achieved in the **GIC**-enhanced HMDB51-dark-GIC dataset.

Effects of Frame Enhancement Methods on ARID. To gain a better understanding of the difference between the outcome of utilizing the different enhancement methods, we visualize the frame output of each enhancement method. Figure 5.9 presents the sampled frames and their respective RGB histograms of the output of the above enhancement methods with the same input ARID video frame.

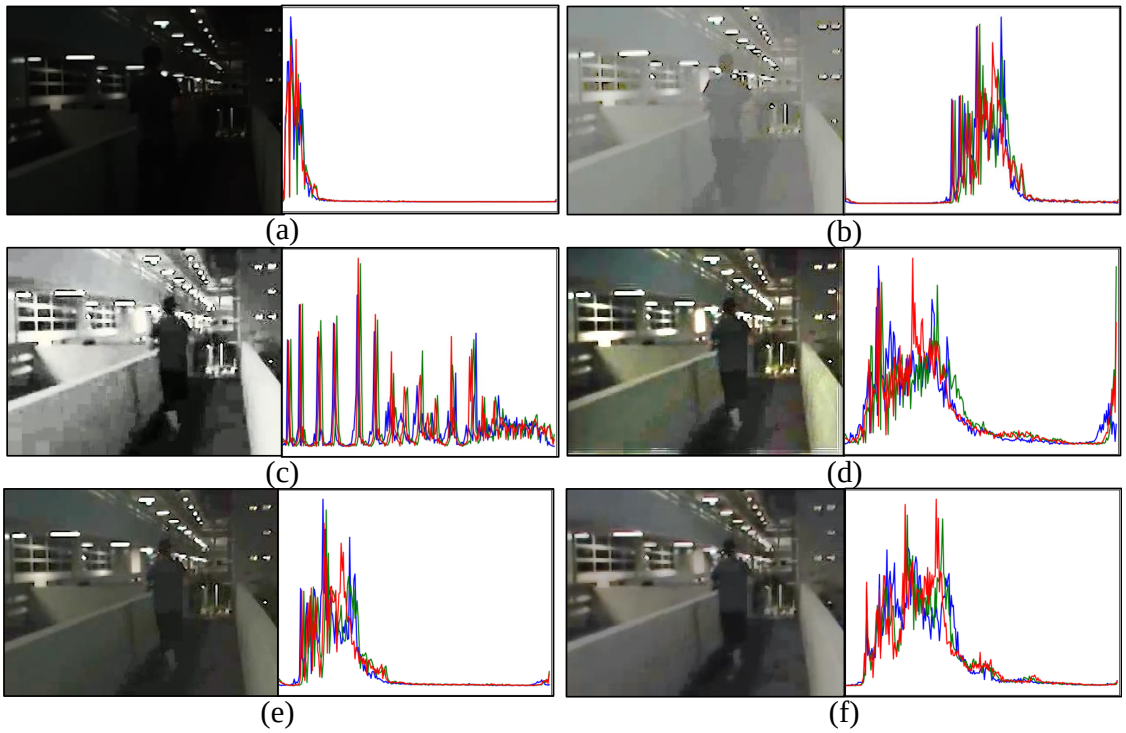


FIGURE 5.9: Comparison of the sampled frames and their respective RGB histograms from (a) ARID, (b) ARID-GIC, (c) ARID-HE, (d) ARID-LIME, (e) ARID-BIMEF and (f) ARID-KinD. Best viewed in color.

Figure 5.9 clearly shows that visually, the outputs of all frame enhancement methods improve the visibility of the video. The actor who is running can be seen clearly in all sampled frames except the sample frame from the original video in ARID substantially. In fact, the sampled frame of ARID-GIC does not appear to be the best enhancement visually, as it is still low in contrast. In comparison, all other methods produce higher contrast images, as justified by the RGB histograms in Figure 5.9. This indicates that current frame enhancement which clearly improves dark video frames visually may not bring improvement in action recognition accuracy for dark videos. We argue that some enhancements can be regarded as artifact or adversarial attacks for videos. Though enhanced frames are clearer visually, some enhancements break the original distribution of videos and introduce noise. The change in distribution and introduction of noise could lead to a decrease in performance for action recognition models.

Accuracy for Each Action Class. To further understand how the action recognition performs for each individual class, we present the confusion matrices for the ARID dataset and all its variants. In particular, the confusion matrices in are constructed using the worst performing model, C3D model (Figure 5.10(a)), and the best performing model, 3D-ResNext-101 model (Figure 5.10(b)) respectively.

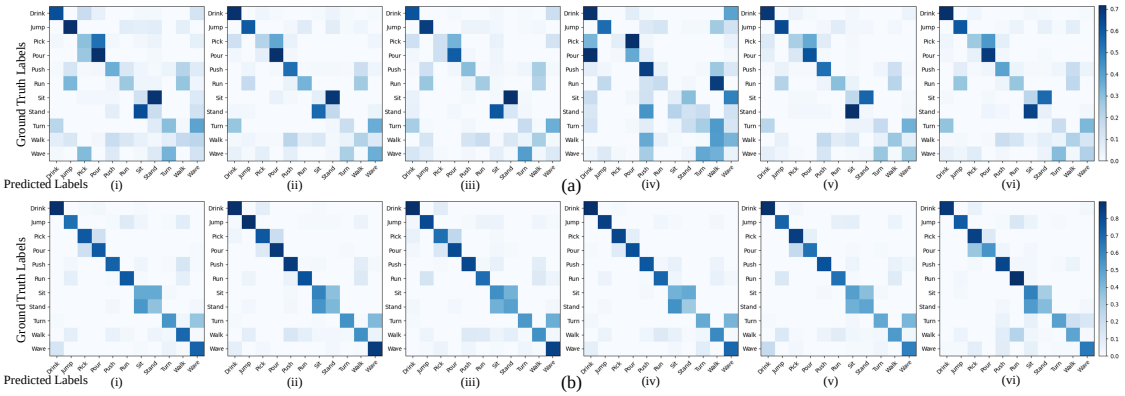


FIGURE 5.10: Comparison of the normalized confusion matrices for (a) C3D and (b) 3D-ResNext-101 models. The normalized confusion matrices show the accuracies for each class as the values at the diagonal corresponding to the ground truth labels at the vertical axis. The normalized confusion matrices are constructed with respect to (i) ARID, (ii) ARID-GIC, (iii) ARID-HE, (iv) ARID-LIME, (v) ARID-BIMEF and (vi) ARID-KinD. Best viewed in color and zoomed in.

The pattern in Figure 5.10 clearly presents the performance of the models for the different classes with different frame enhancement methods. In general, the patterns of the confusion matrices for the same model remain largely the same for the original ARID dataset and all of its variants. This is in line with the results shown in Table 5.3 where none of the frame enhancement methods would bring a significant increase in accuracy. In most cases, four of the eleven classes could reach an accuracy of greater than 50%: Drinking, Jumping, Pouring, and Pushing. We observe that for these classes, the visible background would change more obviously due to the action. For example, the action "Jumping" could cause light in the background to be seen flashing as the actor jumps up and down.

It could further be observed that a large improvement is achieved by utilizing a much deeper network as 3D-ResNext-101. However, there are still several classes in which most videos are misclassified. One noticeable case is for classes "Standing" and "Sitting", where relatively large percentages of either class being misclassified as the other class. The two classes in our dataset share similar action patterns and differ in the order sequence of frames. The other noticeable case is for classes "Turning" and "Waving", where they differ mostly in the movement of the hand.

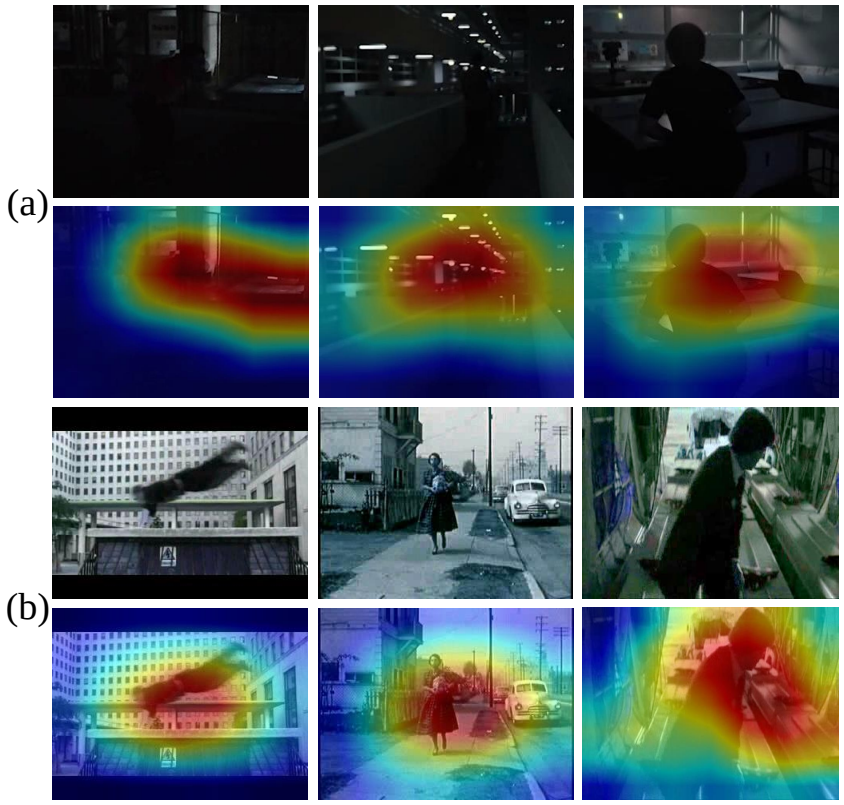


FIGURE 5.11: Comparison of sampled frames and their corresponding *CAM*s from (a) ARID and (b) HMDB51 dataset, extracted by utilizing 3D-ResNext-101 model. We present sampled frames from three common classes: Jumping (left), Running (mid) and Standing (right). Best viewed in color and zoomed in.

5.3.5 Feature Visualization with ARID

To further understand the performance of current action recognition models on ARID and analyze the effect of dark videos on current models, we extract and visualize features at the last convolution layer of the action recognition models. The visualizations of features are presented as *Class Activation Maps (CAM)* [167], which depicts the focus of the model with respect to the given prediction. Specifically, our *CAM*s are extracted by utilizing the 3D-ResNext-101 model first, due to the best performance achieved by the 3D-ResNext-101 model on both HMDB51 and ARID datasets. Figure 5.11 compares the sampled frames from the ARID and HMDB51 datasets, with the corresponding *CAM*s extracted by 3D-ResNext-101 trained on either datasets. We observe that for the frames in HMDB51 with normal illumination, the 3D-ResNext-101 model is able to focus on the actors. The model is able to explicitly exclude most of the information from the background. Whereas for the dark video, instead of focusing on the actor, the model focuses more on the surroundings. For example, for the action in Figure 5.11(a)(left),

the network classifies the action as "Jumping" not by focusing on the person. Rather it focuses on the background whose details are uncovered due to the person jumping backward. Therefore the *CAM* shows the network focusing on a narrow beam in the background. The focus on the background instead of the actor could be partly due to the fact that clear outlines of actors rarely exist in dark videos.

From Table 5.3, we have concluded that for each model, certain frame enhancement methods could positively affect the final classification accuracy. To gain further understanding of how the different frame enhancement methods actually affect the 3D-ResNext-101 model, we compare the *CAM*s with respect to the same sampled frame from the original ARID dataset and the five frame enhanced ARID datasets as shown in Figure 5.12. Compared to the original video frame, the outline of the actor is much clearer in all enhanced frames. The focus area of the network is significantly more concentrated towards the actor compared with *CAM* of the original frame, where it includes large portions of the background. However, noticeable offsets do exist between the focus of the network of the frame enhanced sample frames and the actual actor. The offsets show that the focus area of networks would still include portions of the background, which is rather irrelevant to the actor or the action. In contrast, the *CAM*s of HMDB51 video frames show the areas of network focus center explicitly around the actors. This may partly explain the inability of frame enhancement methods to improve action recognition accuracy while being able to focus on a more concentrated area of each video frame.

We notice that the effect of the same frame enhancement methods varies across different action recognition models. Among the five frame enhancement methods, four of them produce opposite effects on C3D and 3D-ResNext-101 models. To further understand how the frame enhancement methods affect the different models, the *CAM*s extract by C3D is presented as shown in Figure 5.13 for analysis and comparison.

The comparison of *CAM*s extracted by the C3D model shows that the concentration effect by applying frame enhancement is unapparent compared to the case where the 3D-ResNext-101 model is utilized. We observe that the focus area for C3D is significantly more spread out, indicating that C3D focuses more on unrelated backgrounds. This explains the lower accuracy of C3D compared to the 3D-ResNext-101 model. By comparison, the dark red region where the network weights the most is relatively more concentrated when **KinD** or **BIMEF** methods are used compared to that when **HE** or **LIME** methods are used. This coincides with the results as presented in Table 5.3, which

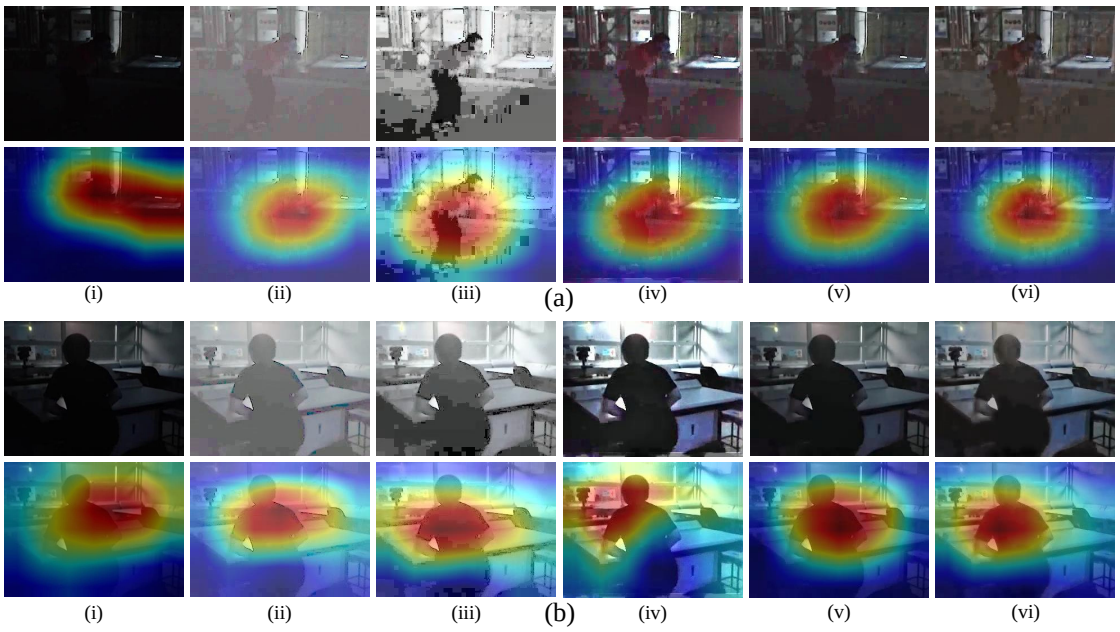


FIGURE 5.12: Comparison of sampled frames and their corresponding *CAM*s of classes: (a) Jumping and (b) Standing, extracted by utilizing 3D-ResNext-101 model. The sampled frames and their *CAM*s are from (i) ARID, (ii) ARID-GIC, (iii) ARID-HE, (iv) ARID-LIME, (v) ARID-BIMEF and (vi) ARID-KinD.

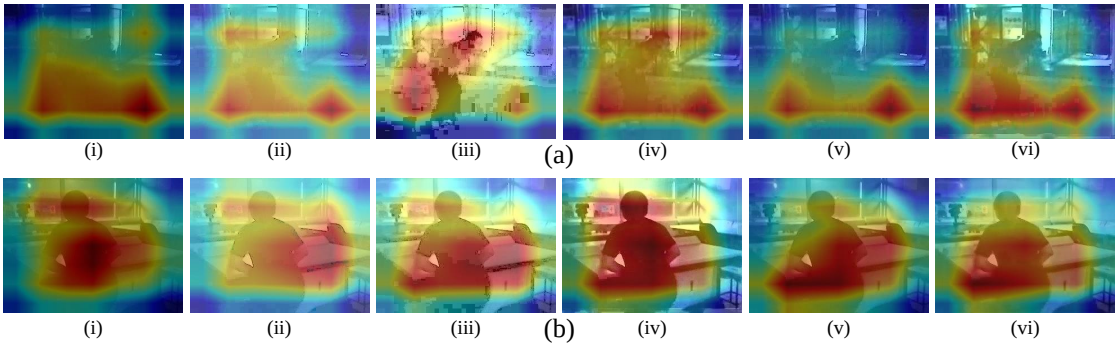


FIGURE 5.13: Comparison of the *CAM*s of classes: (a) Jumping and (b) Standing, extracted by utilizing C3D model, corresponding to the same sampled frames in Figure 5.12. The *CAM*s are extracted from (i) ARID, (ii) ARID-GIC, (iii) ARID-HE, (iv) ARID-LIME, (v) ARID-BIMEF and (vi) ARID-KinD datasets.

suggest that **KinD** and **BIMEF** methods produce positive improvements while **HE** or **LIME** methods result in negative improvements.

5.3.6 Discussion

From the results and analysis presented above, we can draw three major conclusions about our ARID dataset and the task of action recognition in the dark from statistics and classification result aspects:

1. Videos taken in a dark environment are characterized by their low brightness and low contrast. Among which the characteristic of low contrast cannot be fully synthesized by synthetic dark videos, therefore synthetic dark videos cannot be directly applied to the task of action recognition in the dark.

2. Though current frame enhancement methods could produce visually clearer video frames, the accuracy improvements made for current action recognition models after frame enhancing dark videos are rather limited. Some frame enhancement methods even deteriorate classification accuracy, since some enhancements can be regarded as artifacts or adversarial attacks for videos. Breaking the original distribution of videos might decrease the performance of a statistical model. Better frame enhancement methods developed may be helpful in further improving action recognition accuracy in dark videos.

3. Current action recognition models fail to focus on the actor for classification in many dark videos. This might be caused by unclear outlines of actors and shows that action recognition models could tend to focus on the actors for frame-enhanced dark videos. However, the focuses in frame-enhanced dark videos contain offsets. We believe that better action recognition models with a better ability to focus on actors, especially with unclear outlines, could be a critical part of improving action recognition accuracy in dark videos.

The above dataset pioneered by us provides fundamental data for the task of action recognition in the dark. The relevant conclusions made through benchmarking current action recognition models provide basic statistical and characteristic analysis of ARID that could contribute to exploring more effective solutions for ARID.

5.4 Summary

In this chapter, we introduced the Action Recognition In the Dark (ARID) dataset, which is, as far as we are aware, the first dataset dedicated to the task of action recognition in the dark. The ARID includes 4k video clips with 11 action categories. The ARID dataset is still under active development, and is the benchmark dataset for a current workshop: The 4th Workshop and Prize Challenge: Bridging the Gap between Computational Photography and Visual Recognition (UG2+) in conjunction with IEEE CVPR 2021. To understand the challenge behind real dark videos, we analyze our ARID dataset with three perspectives: statistical, classification result, and feature visualization. We discover distinct characteristics of real dark videos, proving their necessity over synthetic dark videos. Empirical analysis shows that current deep learning action recognition models and frame enhancement methods may not be effective in recognizing action in dark videos.

Despite great efforts are being made to further enhance the ARID dataset through the introduction of more video clips and more complex actions, it should be noted that the scale of the ARID dataset is still much smaller than current public datasets. The limitation in dataset scale further limits the complexity of the method trained as methods with higher complexity tend to overfit on smaller datasets. While videos shot in the dark presents distinct characteristics, given the availability of a large number of videos with normal illumination in current public video datasets, one intuitive method to further improve the performances of current deep learning action recognition models on ARID without expanding the dataset scale would be to make full use of these publicly available videos. It would be best if models trained on public datasets with normal illuminated videos could adapt automatically to dark videos, enhancing the generalizability of current models to dark videos while lessening the need for annotating dark videos which is very costly.

Chapter 6

Aligning Correlation Information for Domain Adaptation in Action Recognition

While the introduction of the ARID dataset in Chapter 5 pioneers in vision-based action recognition in dark videos, empirical results show great room for improvements for current action recognition models on ARID. In addition, the limited scale of ARID also prohibits complex deep learning methods to be trained due to the high risk of overfitting. Thanks to the high cost of video collection and annotation, simply increasing the dataset scale may not be a feasible solution. Alternatively, videos shot in normal illumination could be fully utilized to train transferable models that could be generalized to dark videos. However, the distinct characteristics of dark videos suggest significant domain shifts across dark and normal videos, therefore the direct transfer of models would be unavailable.

To address the domain shift, domain adaptation greatly enhances the generalization of deep learning models when confronting label-scarce domains, such as dark videos. In this chapter, we focus our investigation on Video Unsupervised Domain Adaptation (VUDA), where existing VUDA methods tend to align video features by aligning spatial and temporal features while ignoring the alignment of spatiotemporal correlation features. Therefore, we propose a novel VUDA method that aligns spatiotemporal correlation features across video domains. The motivation of the proposed work is introduced

in Section 6.1, while the detailed architecture of ACAN is presented in Section 6.2. Further, a novel cross-domain dataset: the HMDB-ARID dataset is presented in Section 6.3. Extensive experiments are conducted in this novel dataset and previous cross-domain datasets with the results illustrated in Section 6.4. Finally, Section 6.5 concludes this chapter.

6.1 Introduction

Action recognition has long been studied thanks to its applications in various fields. Despite achieving promising results, most research assumes that the distribution of the test data is in line with that of the train data. Meanwhile, due to the high cost of annotating videos, it is desirable if networks trained in one domain could be directly applied to another. However, significant decrease in performances are observed when networks are applied to cross-domain scenarios. To alleviate the impact of domain shift, studies have been conducted on unsupervised domain adaptation (UDA), which aims to leverage data from the labeled source domain to boost performance on the unlabeled target domain. Previously, UDA has been explored on image-based tasks, such as image recognition [89, 105, 107], object detection [108, 110, 168] and person re-identification [169].

Comparatively, there is limited research towards applying DA methods to videos for tasks such as action recognition. This is mainly due to the fact that videos contain data with more modalities, which complicates the adaptation process. In this chapter we focus on the case where the target domain is unlabeled, and the task of applying DA to cross-domain videos is termed the *Video Unsupervised Domain Adaptation* (VUDA). Earlier works use the same adaptation strategies as that for image DA while utilizing 3D Convolutional Neural Networks (3D-CNNs) instead of 2D Convolutional Neural Networks (2D-CNNs) for feature extraction. Current improvements in DA methods for video tasks focus on improving alignment along the temporal direction. Such improvements are in line with the additional temporal information provided in videos compare to images. They are achieved mainly through applying attention mechanisms to features of video segments sampled across the temporal direction [115, 116]. Alternatively, auxiliary tasks such as clip order prediction [118] are utilized to extract robust temporal representation [117].

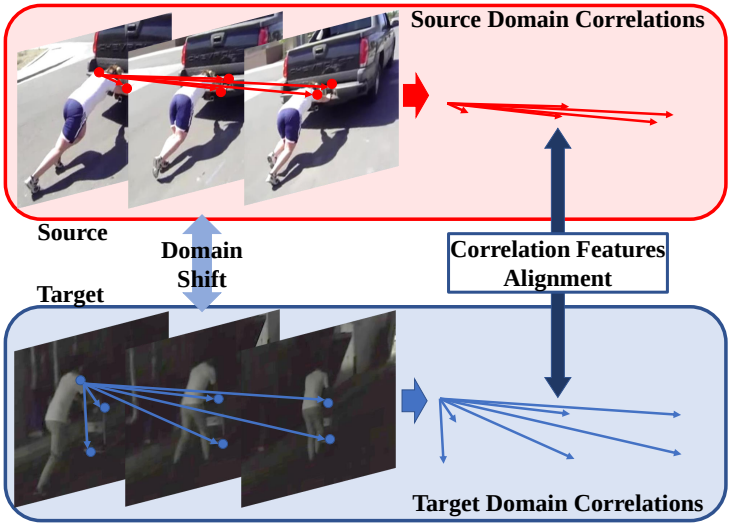


FIGURE 6.1: Illustration of our proposed correlation features alignment. The correlation features are extracted as long-term dependencies of pixels across spatiotemporal dimensions. For the same action in the source and target domains, their corresponding correlation features are are distinct due to the different postures of the actors. While correlation features are highly associated with the action, alignment of video features should include the alignment of correlation features. Here we show two samples with the action "Push" from HMDB51 (top) and ARID (bottom).

Intuitively, the correlations between pixels are highly associated with an action. In supervised action recognition, such correlations have been recently exploited to aid the extraction of accurate video features. One significant example is the non-local neural network [62]. The correlation features are extracted as long-term dependencies of pixels across spatiotemporal dimensions and construct spatiotemporal features by self-attention [148, 170]. The correlation features have brought significant increase in network performance [62–64, 171]. However, correlation features of the same action could be very different, as depicted in Figure 6.1. The same action "Push" sampled from two different datasets results in distinct correlations information. Given the close relation between correlation features and the action, it is therefore reasonable to not only align spatial and temporal features alone but also to align correlation features. We therefore propose an Adversarial Correlation Adaptation Network (ACAN) that aligns correlation features in an adversarial manner.

For an action within a domain, its correlation features, and the embedded correlation information, would be similar, thanks to the similar appearance and postures of the actors. Yet outliers may be presented in each domain, which may impact the transferability of the network. To cope with this impact, we propose that the joint distribution of

correlation information should be aligned. We believe that such a joint distribution of correlation information could be computed as the covariance of the correlation information [172], implemented as its corresponding Gram matrix [173]. Therefore, aligning the correlation features of two domains is interpreted as minimizing the difference between the Gram matrices of the correlation information. While direct minimization of the Gram matrix difference could come at a price of decreasing network discriminability and high computation cost, we propose to minimize the *pixel correlation discrepancy* (PCD).

Besides the complexity of the process of video data, the lack of research in DA methods for action recognition and other video-based tasks are also partly due to the lack of sufficient and meaningful cross-domain datasets. Apart from current cross-domain VUDA datasets, we proposed a new *HMDB-ARID* dataset from HMDB51 [72] and a recent dark video dataset, ARID [174]. The different illumination conditions of videos in HMDB51 and ARID causes larger domain shift, making the *HMDB-ARID* dataset more challenging.

Our main contributions are summarized as follows:

* We proposed a novel ACAN network for domain adaptation in action recognition by aligning correlation features across domains.
* We further improve the effectiveness of correlation alignment by aligning the joint distribution of correlation information of different domains through minimizing *pixel correlation discrepancy* (PCD).
* We introduce a more challenging VUDA dataset: the *HMDB-ARID* dataset. To our knowledge, this is the first cross-domain VUDA dataset that includes videos shot under different illumination.
* We perform extensive experiments, whose results demonstrate the effectiveness of our proposed method, achieving state-of-the-art performance across multiple current and novel VUDA datasets.

The rest of this chapter is organized as follows: in Section 6.2, we introduce our proposed Adversarial Correlation Adaptation Network (ACAN) with the process of minimizing *pixel correlation discrepancy* (PCD) thoroughly. Further, in Section 6.3, we

introduce our proposed *HMDB-ARID* dataset in detail. After that, we present and analyze the experimental results of our proposed ACAN on previous and our novel VUDA datasets, with a thorough ablation study on the design of ACAN in Section 6.4. Finally, we conclude the chapter in Section 6.5.

6.2 Method

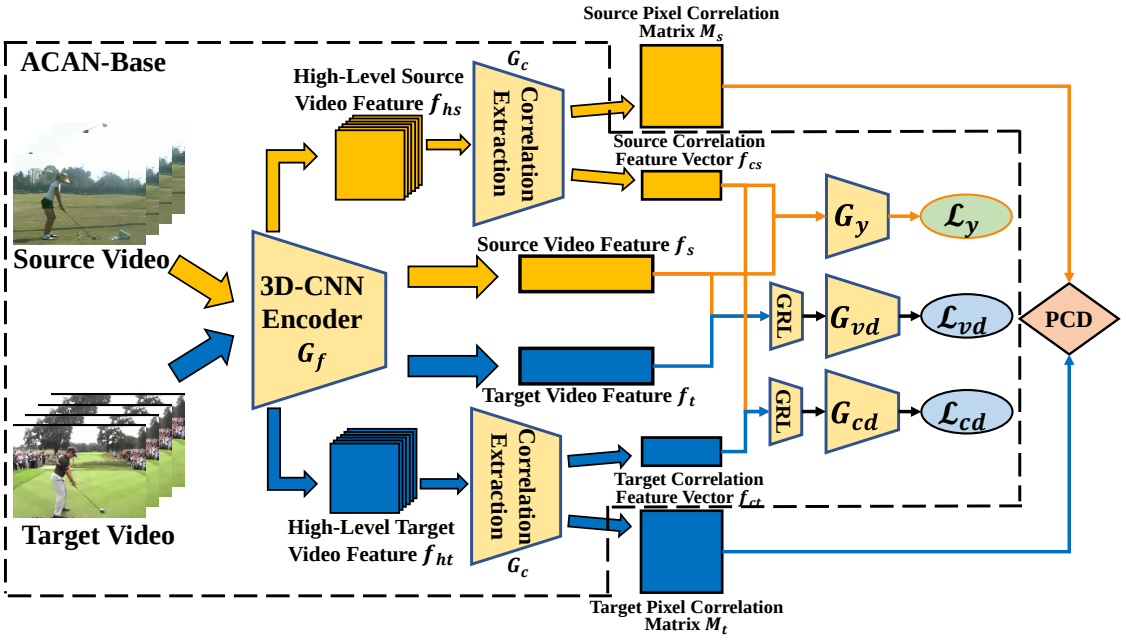


FIGURE 6.2: Overview of the structure of ACAN. We first generate video features with a shared 3D-CNN encoder for both source and target domain videos. The source and target correlation feature vectors are obtained through high-level video features, extracted from a deeper layer of the encoder. An adversarial domain loss is applied to both the video features and the correlation feature vectors for aligning the video features and correlation feature vectors. Further, aligning the joint correlation information distribution requires the alignment of the Gram matrices constructed from the pixel correlation matrices (PCM). To achieve this, we further introduce the pixel correlation discrepancy. Figure best viewed in color and zoomed in.

In video UDA, we are given a source domain with N_s labeled videos $\mathcal{D}_s = \{(V_s^i, y_s^i)\}_{i=1}^{N_s}$, and a target domain with N_t unlabeled videos $\mathcal{D}_t = \{V_t^j\}_{j=1}^{N_t}$. The source and target domains are characterized by two underlying probability distributions p_s and p_t respectively. The goal of video UDA is to construct a network capable of learning transferable features and minimizing a target classification risk.

Current VUDA approaches still rely on aligning only spatial and/or temporal features and fail to align correlation features which correlate long-term pixel dependencies. To cope with this challenge, we propose an Adversarial Correlation Alignment Network (ACAN) to align cross-domain correlation features in an adversarial manner. We further introduce the *pixel correlation discrepancy* (PCD), motivated by the theoretical results in style transfer. We begin by presenting the base architecture of ACAN, followed by an illustration on the minimization of PCD.

6.2.1 Base Architecture

Figure 6.2 presents the base architecture of our proposed ACAN, illustrated as ACAN-Base. During training, given a source and target video pair (V_s^i, V_t^j), the source and target video features f_s^i, f_t^j are obtained through a shared 3D-CNN encoder $G_f(.;\theta_f)$. Meanwhile, the high-level source and target video feature f_{hs}^i, f_{ht}^j are extracted from a deeper layer of the 3D-CNN encoder (e.g. conv4 layer). The high-level video features are processed by a shared correlation extraction module G_c where the correlation features of the input videos are extracted. The results are the source and target correlation matrices $\mathbf{M}_s^i, \mathbf{M}_t^j$ as well as the source and target correlation feature vectors f_{cs}^i, f_{ct}^j. $G_c(.;\theta_c)$ are built based on the non-local operation [62], which extracts the correlation features as long-range dependencies between spatiotemporal pixels. The source correlation feature vector and video feature f_{cs}^i, f_s^i are concatenated to form the overall feature representation of source video V_s^i, which would be input to a classifier G_y for action predictions. The action class prediction loss $\mathcal{L}_y$ is computed with respect to the predictions from G_y, formulated as:

$$\mathcal{L}_y = \frac{1}{N_s}\sum_{i=1}^{N_s} L_y(G_y(f_{cs}^i \oplus f_s^i), y_i), \tag{6.1}$$

where L_y is the cross entropy loss function, and $\oplus$ denotes the concatenation operation.

To accommodate the domain shift between source and target domains, adversarial-based UDA approaches are proved to perform well on image data [89, 92, 95, 96] and language data [175]. Therefore we also leverage this technique for VUDA, which aims to align the global distributions with additional domain discriminators that are trained with the feature generators in a min-max fashion. Domain discriminators are designed to discriminate the video features while the feature generators are trained to deceive the domain

discriminators. Here the feature generators are referred to as a combination of G_f and G_c. We adopted separate domain discriminators for the source/target video features $f_*^\star$ $(* \in (s,t), \star \in (i,j))$ and the source/target correlation features $f_{c*}^\star$. The two domain discriminators are denoted as the video domain discriminator $G_{vd}(.;\theta_{vd})$ and the correlation domain discriminator $G_{cd}(.;\theta_{cd})$. During the adversarial training process, the parameters θ_{vd} and θ_{cd} are learned by minimizing the video domain loss $\mathcal{L}_{vd}$ and the correlation domain loss $\mathcal{L}_{cd}$, respectively, which are formulated as:

$$\mathcal{L}_{vd} = \frac{1}{N_s}\sum_{i=1}^{N_s} L_b(G_{vd}(f_s^i), d_i) + \frac{1}{N_t}\sum_{j=1}^{N_t} L_b(G_{vd}(f_t^j), d_j), \tag{6.2}$$

$$\mathcal{L}_{cd} = \frac{1}{N_s}\sum_{i=1}^{N_s} L_b(G_{cd}(f_{cs}^i), d_i) + \frac{1}{N_t}\sum_{j=1}^{N_t} L_b(G_{vd}(f_{ct}^j), d_j), \tag{6.3}$$

where L_b is the binary cross-entropy loss of the domain discriminators, while d_i and d_j are the domain label for the source and target domains respectively. Meanwhile, the parameters of the feature extractors θ_f and θ_c are learned to maximize the domain losses. To achieve this, a Gradient Reverse Layer (GRL) is inserted before each domain discriminator as in Figure 6.2.

The overall loss function to be optimized can therefore be formulated as:

$$\mathcal{L} = \mathcal{L}_y - (\lambda_v \mathcal{L}_{vd} + \lambda_r \mathcal{L}_{cd}), \tag{6.4}$$

where λ_v and λ_r are the trade-off weights for the video domain loss and correlation domain loss respectively.

6.2.2 Minimizing Pixel Correlation Discrepancy

In the base ACAN network, the same DA approach is applied to both video and correlation features. However, it remains a question *whether such an approach is the most effective way for aligning correlation features across different domains?* Aligning correlation features can be further achieved through aligning the joint distribution of correlation information. The joint distribution could be computed as the covariance of correlation information, implemented as its corresponding Gram matrix. The key to the above question therefore lies in the expression of the correlation information. As illustrated in Figure 6.2, correlation features are extracted from G_c, whose structure is shown in

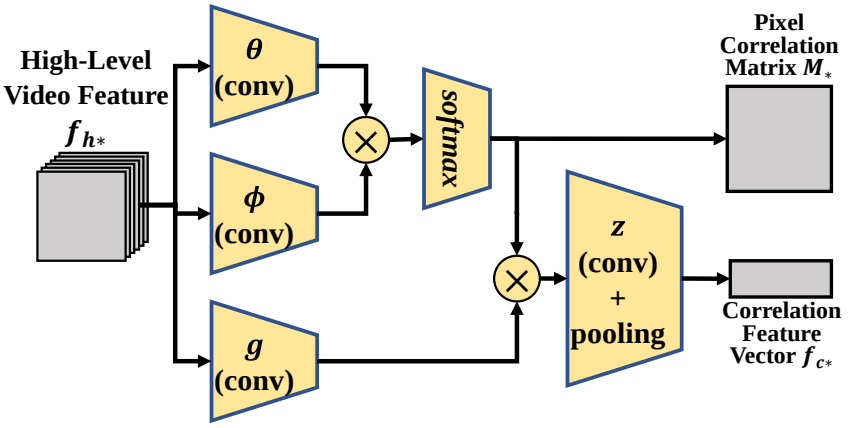


FIGURE 6.3: Structure of the correlation extraction module G_c. G_c extract correlation features (pixel correlation matrix $\mathbf{M}_*$ and correlation feature vector f_{c*}) through the high-level video feature f_{h*}. It is built upon the non-local operation. $\mathbf{M}_*$ is obtained through multiplication of f_{h*} projected on latent spaces, and represents the correlation between each spatiotemporal pixel feature. f_{c*} is further obtained by multiplying the $\mathbf{M}_*$ f_{h*} projected on the latent space, followed by pooling operation over spatiotemporal dimensions. The projection functions are implemented with convolution layers of $1 \times 1 \times 1$ kernel.

Figure 6.3. For the i^{th} input video, we define the pixel correlation matrix (PCM) $\mathbf{M}_*^i$ as:

$$\mathbf{M}_*^i = \varphi(\theta(f_{h*}^i)^T \phi(f_{h*}^i)), \tag{6.5}$$

where φ is the softmax operation. Both $\theta(\cdot)$ and $\phi(\cdot)$ are linear functions projecting the high-level video features to latent spaces. In practice, they are implemented as convolution layers with a kernel size of $1 \times 1 \times 1$. The value $\mathbf{M}_{*,pq}^i$ at the (p, q) position of PCM represents the correlation between the video feature at spatiotemporal point p, $f_{h*,p}^i$, and the video feature at spatiotemporal point q, $f_{h*,q}^i$. We argue that PCM could be viewed as the correlation information of the video. Therefore the joint correlation information distribution is constructed as the Gram matrix of the PCM, denoted as $\mathcal{G}^i \in \mathbb{R}^{N_M \times N_M}$, where N_M is the number of spatiotemporal points in the feature map $\theta(f_{h*}^i)$. $\mathcal{G}^i$ is computed by:

$$\mathcal{G}^i = {\mathbf{M}_*^i}^T \mathbf{M}_*^i. \tag{6.6}$$

The alignment of correlation features thus requires the minimization of the distance between the Gram matrices $\mathcal{G}$, termed as the video covariance loss $\mathcal{L}_{vs}$, formulated by:

$$\mathcal{L}_{vs} = \| \mathbf{E}(\mathcal{G}_s) - \mathbf{E}(\mathcal{G}_t) \|^2, \tag{6.7}$$

where the subscripts s and t denotes the Gram matrices for source and target videos respectively. However, such computation is inefficient, requiring a cost of $O({N_M}^2)$. Furthermore, improving network transferability through minimizing $\mathcal{L}_{vs}$ comes at the price of decreasing network discriminability. To minimize $\mathcal{L}_{vs}$ more efficiently while causing less impact on the network's discriminability, we simplify according to the theory in [176].

Theorem 6.2.1. Given the Gram matrices $\mathcal{G}_s, \mathcal{G}_y$ constructed from source and target features $\mathbf{M}_s$, $\mathbf{M}_t$, the minimization of distance between the Gram matrices $\mathcal{L}_{vs}$ can be seen as a distribution alignment process from $\mathbf{M}_t$ to $\mathbf{M}_s$.

As proven in [176], the theorem indicates that minimizing $\mathcal{L}_{vs}$ could be reformulated as minimizing the distribution discrepancy of $\mathbf{M}_t$ and $\mathbf{M}_s$. Set the underlying distributions of $\mathbf{M}_s$ be p_{Ms} and that of $\mathbf{M}_t$ be p_{Mt}. Here we propose the *pixel correlation discrepancy* (PCD), denote as $d_M(p_{Ms}, p_{Mt})$. Computing and minimizing this discrepancy is achieved by representing the distributions p_{Ms} and p_{Mt} as elements on the reproducing kernel Hilbert space (RKHS). As such, the distribution discrepancy could be defined as distance of distribution embedded elements on the RKHS.

Further, to align the distributions of p_{Ms} and p_{Mt} in a more fine-grained manner, it is important to align the distributions taking the relations between relevant classes into consideration. That is to align p_{Ms} and p_{Mt} within the same action classes in source and target domains, instead of aligning it only in by the global distributions. The overall PCD is therefore formulated as:

$$d_M(p_{Ms}, p_{Mt}) \triangleq \mathbf{E}_c \parallel \mathbf{E}_{p_{Ms}(c)}[\zeta(\mathbf{M}_s)] - \mathbf{E}_{p_{Mt}(c)}[\zeta(\mathbf{M}_t)] \parallel^2_{\mathcal{H}}, \tag{6.8}$$

where $\mathbf{E}_{p_{M*}(c)}$ is the mean embedding of distribution p_{M*} for action class c on the RKHS $\mathcal{H}$. The feature map ζ is closely related to the RKHS characteristic kernel k by $k(\mathbf{M}_s, \mathbf{M}_t) = \langle \zeta(\mathbf{M}_s), \zeta(\mathbf{M}_t) \rangle$. The use of mean embedding for each class enables our PCD to align distributions of correlation information within each action class instead of only focusing on the global correlation information distribution. In practice, we may further assume that each video belongs to a certain action class with a class-related weight w_c. We therefore could estimate PCD in Equation 6.8 as:

$$d_M(p_{Ms}, p_{Mt}) = \frac{1}{C} \sum_{c=1}^{C} \parallel \sum_{i=1}^{N_s} w_{sc}^i \zeta(\mathbf{M}_s^i) - \sum_{j=1}^{N_t} w_{tc}^j \zeta(\mathbf{M}_t^j) \parallel^2_{\mathcal{H}}, \tag{6.9}$$

where C is the number of action classes. When computing the weight of a source video for a certain action class, given that the labels are provided, the weight w_{sc}^{i} is computed by:

$$w_{sc}^{i} = \frac{y_s^i}{\sum_{k=1}^{N_s} y_s^k}. \tag{6.10}$$

Meanwhile, since the labels are not available for the target videos, we cannot compute the weight w_{tc}^{j} directly. Instead, we utilize the output from the action classifier G_y which characterizes the probability of assigning a given video to an action class. This is denoted as the pseudo-label for a target video and is computed by:

$$y_t^j = G_y(f_{ct}^i \oplus f_t^i). \tag{6.11}$$

The resulting pseudo-labels of the target videos could be used as in Equation 6.10 for computing the weight of a target video for an action class. Finally, since the feature map ζ cannot be computed directly in most cases, we expand Equation 6.9 while utilizing the characteristic kernel k. The PCD could therefore be reformulated as:

$$\begin{aligned} d_M(p_{Ms}, p_{Mt}) = \frac{1}{C}\sum_{c=1}^{C}\Big(&\sum_{i=1}^{N_s}\sum_{i'=1}^{N_s} w_{sc}^{i} w_{sc}^{i'} k(\mathbf{M}_s^i, \mathbf{M}_s^{i'}) \\ &+ \sum_{j=1}^{N_t}\sum_{j'=1}^{N_t} w_{tc}^{i} w_{tc}^{j'} k(\mathbf{M}_t^j, \mathbf{M}_t^{j'}) \\ &- 2\sum_{i=1}^{N_s}\sum_{j=1}^{N_t} w_{sc}^{i} w_{tc}^{j} k(\mathbf{M}_s^i, \mathbf{M}_t^j)\Big), \end{aligned} \tag{6.12}$$

where the kernel k would typically be of Gaussian form, hence $k(\mathbf{M}_s^i, \mathbf{M}_t^j) = -exp(\frac{\|\mathbf{M}_s^i - \mathbf{M}_t^j\|^2}{2\sigma^2})$. The overall optimization objective is thus formulated as:

$$\mathcal{L} = \mathcal{L}_y - (\lambda_v \mathcal{L}_{vd} + \lambda_r \mathcal{L}_{cd}) + \lambda_d d_M, \tag{6.13}$$

where λ_d is the trade-off weight for the PCD. Minimizing our proposed PCD is superior in effective alignment of cross-domain correlation features thanks to its relatively solid theoretical motivation. We find that aligning correlation features with other approaches (e.g. MMD [101], CORAL [104]) all produce inferior performances than our proposed approach.

Dataset	RGB Mean	RGB Std
HMDB51	[0.424,0.364,0.319]	[0.268,0.255,0.260]
UCF101	[0.409,0.397,0.358]	[0.266,0.265,0.270]
Kinetics	[0.432,0.395,0.377]	[0.228,0.222,0.217]
ARID	[0.079,0.074,0.073]	[0.101,0.098,0.090]

TABLE 6.1: Comparison of RGB mean and standard deviation (std) over common action recognition datasets and the ARID dataset.

Statistics	*UCF-HMDB$_{small}$*	*UCF-Olympic*	*UCF-HMDB$_{full}$*	*HMDB-ARID*
Video Length (seconds)	1-21	1-39	1-33	1-30
Video Classes #	5	6	12	11
Training Video #	UCF:482/HMDB:350	UCF:601/Olympic:250	UCF:1438/HMDB:840	HMDB:770/ARID:2288
Validation Video #	UCF:189/HMDB:150	UCF:240/Olympic:54	UCF:571/HMDB:360	HMDB:330/ARID:823

TABLE 6.2: Comparison of current and our novel VUDA datasets.

6.3 The HMDB-ARID Dataset

There are very limited cross-domain benchmark datasets for VUDA, therefore hindering its research. Previous cross-domain datasets introduced for VUDA [114, 119, 177] are of very small-scale, with not more than 6 classes, and typically less than 1,000 videos. The lack of classes and data over these cross-domain datasets introduces limited domain discrepancy, and therefore the performances of DA approaches are saturated. More recently, larger cross-domain video datasets, such as *UCF-HMDB$_{full}$* have been introduced with larger domain discrepancies.

Though larger cross-domain datasets are introduced, both domains included in these datasets are still based on current well-established action recognition datasets. These action recognition datasets may include different classes with different videos, yet most of them are collected on public video platforms. This would lead to similar video statistics among these datasets, as compared in Table 6.1. Similar video statistics suggest high probability of similar scenarios exist among current action recognition datasets, thus the domain shift between these datasets may not be significant. Consequently, the difficulty of adapting the same model across the different domains with similar video statistics or similar scenarios may be trivial. VUDA approaches that perform well in these cross-domain video datasets may not be well applicable in real-world applications where the gap between domains may be much larger than current cross-domain datasets. We argue that VUDA approaches would be more useful for bridging with video domains with large distribution shifts, such as dark videos (adverse illumination) or hazy videos (adverse contrast).

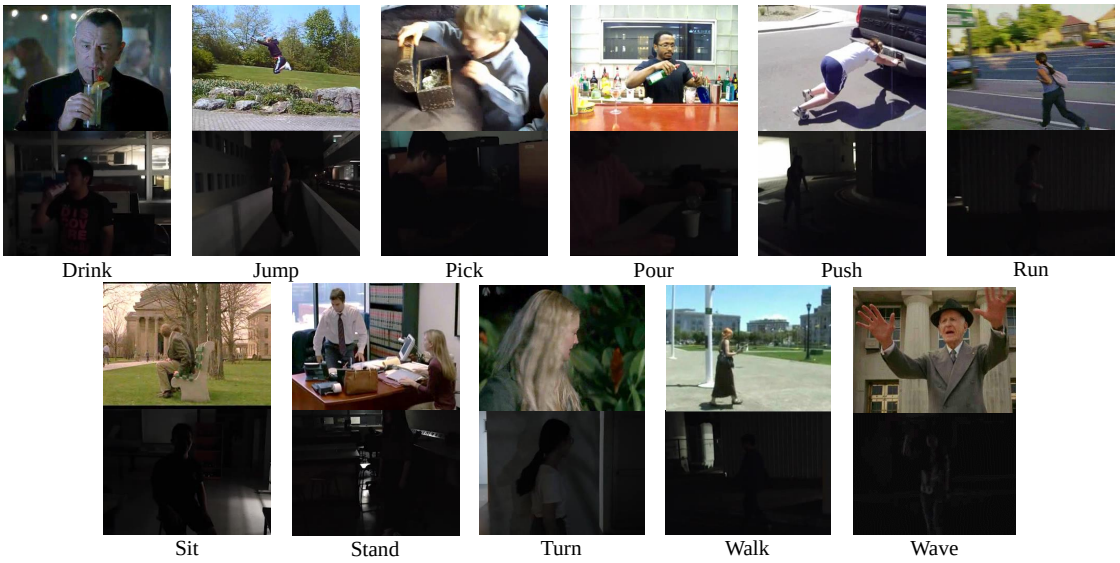


FIGURE 6.4: Sampled frames for each action class from the videos in *HMDB-ARID*. Note that the sampled frames from HMDB51 are shown in the upper row, whereas the sampled frame from ARID are shown in the lower row. Best viewed zoomed in.

To explore how to leverage current datasets to boost performance on videos shot in adverse environments, we propose a novel cross-domain dataset. It incorporates both the current action recognition dataset and a more recent dark dataset, ARID [174], whose videos are shot under adverse illumination conditions. Compared with current action recognition datasets, videos in ARID are characterized by low brightness and low contrast. Statistically, videos in ARID possess much lower RGB mean value and standard deviation (std), as presented in Table 6.1. The larger statistical differences between ARID and current action recognition datasets, such as HMDB51 [72], would strongly suggest a larger domain shift between the different datasets.

The ARID dataset includes a total of 11 human action classes. These includes *drink, jump, pick, pour, push, run, sit, stand, turn, walk and wave*. When proposing the cross-domain *HMDB-ARID* dataset, we include all 11 action classes in ARID and HMDB51. For both datasets, we follow the official split method to separate the train and validation sets. The *HMDB-ARID* dataset thus includes 770 training videos and 330 validation videos from HMDB51, and 2288 training videos and 823 validation videos from ARID. Figure 6.4 shows the comparison of sampled frames from *HMDB-ARID* dataset. Compared to previous cross-domain VUDA datasets, besides containing larger domain shift, our dataset also contains a larger number of total videos for both training and validation, as illustrated in Table 6.2.

6.4 Experiments

In this section, we evaluate our proposed ACAN performing cross-domain action recognition on two VUDA datasets: *UCF-HMDB$_{full}$* and our new *HMDB-ARID*. We present state-of-the-art results on both datasets. We also present detailed ablation studies and qualitative analysis of our proposed ACAN to verify our design.

6.4.1 Experimental Settings and Details

We perform action recognition tasks on both the *UCF-HMDB$_{full}$* dataset and our new *HMDB-ARID* dataset. The *UCF-HMDB$_{full}$* dataset [115] is introduced as an expansion of the original *UCF-HMDB$_{small}$* dataset [119], with more classes and larger domain discrepancy. The *UCF-HMDB$_{full}$* contains a total of 3,209 videos with 12 action classes, all from the original UCF101 [74] and HMDB51 [72] datasets. It includes two settings: UCF→HMDB and HMDB→UCF, where the direction of the arrow symbol is set from the source domain towards the target domain. We use the same splits as provided in the original paper [115]. The novel *HMDB-ARID* dataset is as introduced in Section 6.3, and also consist of two settings: HMDB→ARID and ARID→HMDB. For all four settings, we report the top-1 accuracy on the target dataset.

Our experiments are implemented using PyTorch [149] library. To obtain video features, we instantiate a lightweight 3D-CNN, MFNet [58], as G_f for both source domain videos and target domain videos. The MFNet is utilized thanks to its performance on current action recognition benchmarks (namely UCF101 [74], HMDB51 [72] and Kinetics [76]), while requiring a fraction of the parameters needed in networks such as I3D [52].

The source and target feature extractors share parameters. Following the implementation in [58], the input is a frame sequence of 16 frames with each frame of size 224×224. The correlation extraction module takes the high-level video feature from the output of $conv4$ layer in MFNet as input, which is a feature map of size 14×14. The stochastic gradient descent algorithm [136] is used for optimization, with the weight decay set to 0.0001 and the momentum to 0.9. The batch size is set to 8 per GPU. Our initial learning rate is set to 0.005 and is divided by 10 after 20 and 35 epochs. All experiments are conducted using two NVIDIA GP100 GPUs.

Method	Encoder	UCF→ HMDB	HMDB→ UCF
Source Only	TRN-Res101	73.1%	73.9%
TA^{3}N	TRN-Res101	75.3%	79.3%
TCoN	TRN-Res101	**87.2%**	89.1%
Target Only	TRN-Res101	90.8%	95.6%
Source Only	I3D	80.3%	88.8%
SAVA	I3D	82.2%	91.2%
Target Only	I3D	95.0%	96.8%
Source Only	MFNet	78.6%	88.4%
ACAN(Ours)	MFNet	**85.8%**	**93.2%**
Target Only	MFNet	96.0%	97.1%

TABLE 6.3: Results on the two settings for *UCF-HMDB$_{full}$*

6.4.2 Overall Results

There are limited studies focusing on applying DA approaches to the action recognition task. Here we first compare previous methods that utilizes the *UCF-HMDB$_{full}$* benchmark. This includes TA^{3}N [115], TCoN [116] and SAVA [117]. Due to the different encoders used in experiments, we report both (a) "Source only" results, where the network is trained with supervised source data only and validated on the target data, and is the lower bound performance for the adaptation process; and (b) "Target only" results, where the network is directly trained and validated with supervised target data and is the upper bound performance for the adaptation process. The comparison of performance should focus on the networks' improvement with respect to the performance with the "Source only" setting. The comparison should also focus on the distance between the network's performance and the performance with the "Target only" setting. For the performance of TA^{3}N. we follow the works in [117] and obtain the results by running the publicly available code. Table 6.3 shows the comparison of performances between our proposed ACAN and the methods as mentioned on *UCF-HMDB$_{full}$*.

The results in Table 6.3 show that our proposed ACAN achieves the best result under the HMDB→UCF setting and very competitive performance under the UCF→HMDB setting. Our ACAN with MFNet encoder shows 85.8% top-1 accuracy for UCF→HMDB setting, indicating the improvement brought by ACAN towards the lower bound of the UCF→HMDB setting is 7.2%. This is significantly higher than that brought by SAVA (1.9%) and TA^{3}N (2.2%). The large improvement brought by ACAN enables our network to perform better on UCF→HMDB setting despite the lower bound of MFNet is lower than that of I3D [52]. Under this setting, our ACAN is also closer to the upper

Method	Encoder	HMDB→ ARID	ARID→ HMDB
Source Only	TRN-Res101	17.8%	15.7%
TA^{3}N	TRN-Res101	22.4%	19.8%
Target Only	TRN-Res101	52.8%	50.9%
Source Only	MFNet	48.3%	37.9%
DANN	MFNet	50.7%	40.6%
MK-MMD	MFNet	50.2%	40.1%
MCD	MFNet	47.6%	36.8%
ACAN(Ours)	MFNet	**58.0%**	**46.4%**
Target Only	MFNet	76.1%	67.6%

TABLE 6.4: Results on the two settings for *HMDB-ARID*.

bound of the encoder, with a gap of 10.2%. Comparatively, the gap to the upper bound performance is 15.5% for TA^{3}N and 12.8% for SAVA.

Under the HMDB→UCF setting, our proposed ACAN gains a 4.8% improvement towards the lower bound performance, which is greater than that brought by SAVA (2.4%). The larger increase built upon the strong MFNet encoder enables our ACAN to achieve the best result under this setting. The gap towards the upper bound performance is also the smallest for ACAN, with 3.9% compared to 16.3% for TA^{3}N, 6.5% for TCoN, and 5.6% for SAVA.

We further compare performance of several methods on our novel *HMDB-ARID* dataset, with both HMDB→ARID and ARID→HMDB settings, as shown in Table 6.4. Note that both settings are more challenging, given that the gap between the lower bound performance (trained with supervised source data) and the upper bound performance (trained with supervised target data) is larger compared to the settings for *UCF-HMDB$_{full}$*. In addition to comparing with the TA^{3}N with TRN-Res101 [37] encoder, we also compare with performances with other typical DA approaches, e.g. DANN [89], MK-MMD [101] and MCD [178], with MFNet as the encoder.

The performance results in Table 6.4 indicate that our proposed ACAN achieves the best results in either setting related to our novel *HMDB-ARID* dataset. Our ACAN achieves a top-1 accuracy of 58.0% for the HMDB→ARID setting and 46.4% for the ARID→HMDB setting. Our ACAN also brings the most significant improvement with respect to the lower bound performance, with 9.8% and 8.5% for the two settings respectively. Comparatively, TA^{3}N which does not utilize correlation alignment only brings 4.6% and 4.1% increase with respect to the lower bound performance. This shows that previous methods that fail to align correlations would not be able to handle the larger

domain shift caused by a more significant difference in video statistics. Note that the gap to the upper bound performance obtained by training with supervised target data is still relatively large, suggesting further improvements could be made on this novel *HMDB-ARID* dataset.

6.4.3 Ablation Studies

We justify our proposed design of ACAN through thorough ablation studies. Specifically, we first examine the performance of our ACAN in four scenarios and justify the need for introducing correlation features in the extraction process, the use of two separate domain losses, and the introduction of PCD. We also introduce an alternative form of the joint correlation information distribution difference minimization to compare and justify our current design of PCD. All ablation studies are conducted under the UCF→HMDB and HMDB→ARID settings, with the batch size and other training parameters as mentioned in Section 6.4.1. The MFNet [58] is instantiated as the encoder for all ablation studies.

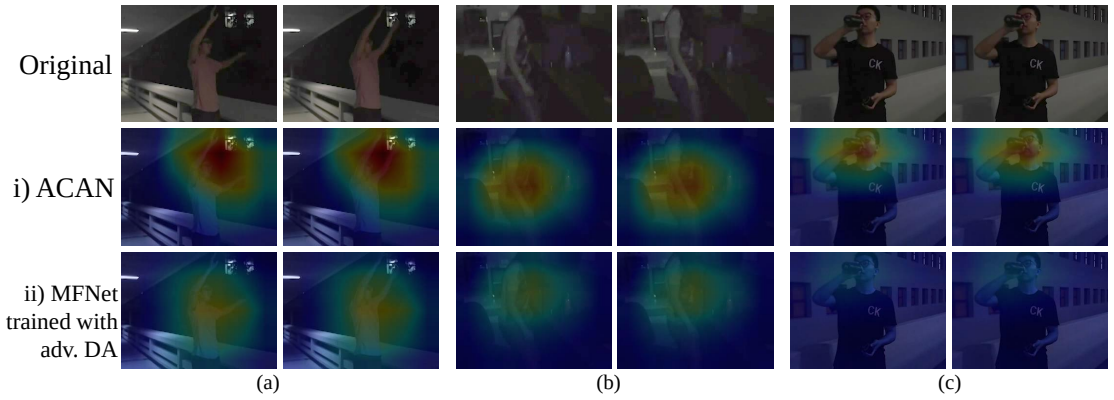
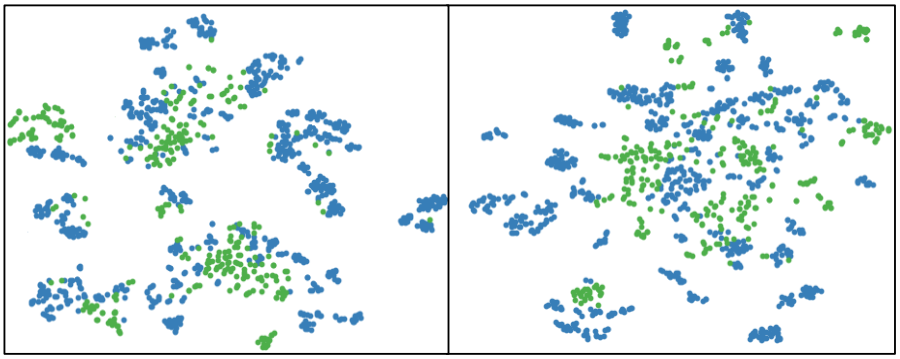


FIGURE 6.5: Class activation maps (CAMs) on ARID, utilizing i) ACAN and ii) MFNet trained with adversarial DA approach. The CAMs are obtained from three actions: (a) "Wave"; (b) "Stand"; and (c) "Drink". We also show the original frames at the top row from which the CAMs are computed. The original frames are tuned brighter for visualization.

The necessity of correlation feature alignment. We first justify the need for correlation features for alignment, which is achieved by (a) comparing the "Source only" results with and without the introduction of correlation features, and (b) comparing the use of adversarial DA approaches with and without correlation features. Results in Table 6.5 justifies the use of correlation features, where the use of correlation features consistently



(a) ACAN (b) MFNet with adv. DA

FIGURE 6.6: Comparison of t-SNE visualization of video features of both source and target domains under HMDB→ARID. The video features are obtained from (a) ACAN and (b) MFNet trained with the adversarial DA approach. The green dots represent the data from the source domain while the blue dots represent the data from the target domain.

Method	UCF→HMDB	HMDB→ARID
Source only w/o. correlation	76.1%	48.1%
Source only w. correlation	78.6%	48.3%
Adv. DA w/o. correlation	80.2%	50.7%
Adv. DA w. correlation	84.2%	52.6%

TABLE 6.5: Ablation experiments on including correlation features, on UCF→HMDB and HMDB→ARID settings.

Method	UCF→HMDB	HMDB→ARID
ACAN	85.8%	58.0%
ACAN $-\mathcal{L}_{cd}$	84.9%	56.9%
ACAN $-\mathcal{L}_{vd}$	84.5%	56.7%
MFNet + PCD	83.8%	56.1%

TABLE 6.6: Ablation experiments on the domain loss $\mathcal{L}_d$ on UCF→HMDB and HMDB→ARID settings.

improves the performance of the network under both "Source only" training and when DANN method is used for DA. It could also be observed that the use of correlation features brings more improvement when the DANN method is applied. This is consistent with our argument of improving video feature alignment by using correlation alignment.

The effectiveness of domain loss $\mathcal{L}_d$. We then justify our design of the domain loss $\mathcal{L}_d$, which is the weighted sum of $\mathcal{L}_{vd}$ and $\mathcal{L}_{cd}$. We compare with the variants of ACAN where either $\mathcal{L}_{vd}$ or $\mathcal{L}_{cd}$ alone is used for domain loss, denoted as ACAN $-\mathcal{L}_{cd}$

Method	UCF→HMDB	HMDB→ARID
ACAN	85.8%	58.0%
ACAN-Base	84.2%	52.6%
ACAN (L2-norm)	85.0%	54.2%

TABLE 6.7: Ablation on PCD and alternative way of minimizing joint correlation information distribution difference, on UCF→HMDB and HMDB→ARID settings.

and ACAN $-\mathcal{L}_{vd}$. We also tested on the case where the domain loss is not applied (hence aligning correlation features by minimizing PCD alone), such a case is denoted as MFNet+PCD. As indicated in Table 6.6, both losses contribute to the effective alignment of video features. The removal of either loss brings a decrease in network performance for both dataset settings. Further decrease is observed when no domain loss is applied. Meanwhile, the domain discriminators corresponding to either domain loss bring only a negligible growth in computation cost. Hence it is worthwhile to include two separate domain discriminators, with two domain losses for the overall domain loss $\mathcal{L}_d$.

The effectiveness of PCD. PCD is introduced for improving the effectiveness of correlation alignment by matching the joint correlation information distribution of video domains. We examine the effect of PCD through comparison with the ACAN variant without PCD, which is ACAN-Base as shown in Figure 6.2. The results in Table 6.7 demonstrates the effectiveness of PCD, whose absence caused a noticeable 1.6% decrease for UCF→HMDB setting, and a significant 5.4% decrease for HMDB→ARID setting. Though the introduced PCD improves the effectiveness of correlation alignment greatly, minimizing PCD involves kernel estimation which brings an increase in computation cost. Inspired by the hypothesis presented in [179], minimizing the joint distribution difference, and hence the distance between distributions p_{Ms} and p_{Mt} could also be achieved through matching the norm of p_{Ms} and p_{Mt} towards a shared restrictive scalar R. The computation of distribution distance with this method is simpler given that no kernel estimation is required. In this case, the equation for the overall loss Equation 6.13 is reformulated as:

$$\begin{aligned}\mathcal{L} =&\mathcal{L}_y - (\lambda_v \mathcal{L}_{vd} + \lambda_r \mathcal{L}_{cd}) + \\ &\lambda_{dist}(L_{dist}(\frac{1}{N_s}\sum_{i=1}^{N_s} n(\mathbf{M}_s^i), R) + \\ &L_{dist}(\frac{1}{N_t}\sum_{j=1}^{N_t} n(\mathbf{M}_t^j), R)).\end{aligned} \tag{6.14}$$

Here L_{dist} is the distance loss between the norm of PCMs and the restrictive scalar R, and is implemented as L_2-distance, while $n(\cdot)$ denotes the norm function. R is set to 25 during the experiments. We denote the variant of ACAN with loss function in Equation 6.14 as ACAN (L2-norm) and compare with the original ACAN. The results shown in Table 6.7 shows that the variant formulated by Equation 6.14 could still bring noticeable improvement than that when the distributions of p_{Ms} and p_{Mt} are not aligned. However, compared to PCD, the improvement is relatively minor. This further justifies the effectiveness of PCD.

6.4.4 Qualitative Analysis

To better understand the effect of ACAN, we perform qualitative analysis on trained networks. We first show the class activation maps (CAM) [167] of the target ARID videos with ACAN and with MFNet (encoder) trained with adversarial DA approach in Figure 6.5. The dark videos in ARID make it difficult for accurate video features to be extracted. Therefore if correlation alignment is not utilized, the network may fail to focus on the actual action in the target domain. It may rather briefly focus on the whole actor (Figure 6.5(ii-a)), or on unrelated background (Figure 6.5(ii-b)). With the involvement of correlation features and its alignment, ACAN is able to focus on the waving hand for the "Wave" action, or the person standing for the "Stand" action, thus showing much stronger performance on the HMDB→ARID setting. Further, we visualize the distribution of the source and target domains under the HMDB→ARID setting with t-SNE, as shown in Figure 6.6. It could be observed from Figure 6.6 that our proposed ACAN can group both the data from the source domain (green dots) and data from the target domain (blue dots) into denser clusters. Our ACAN could also match the target domain data with source domain data more accurately.

6.5 Summary

In this chapter, we propose a novel domain adaptation method for action recognition across different domains. The new ACAN aligns correlation features in an adversarial manner while minimizing joint correlation information distribution differences by minimizing PCD. We further introduce a novel VUDA dataset, *HMDB-ARID*, with a larger domain shift, and is the first VUDA dataset that includes videos shot in adverse

conditions. Our method obtains state-of-the-art result on both the *UCF-HMDB$_{full}$* and *HMDB-ARID* datasets. We further justify our design through an ablation study and validate the effectiveness of ACAN with qualitative results.

Chapter 7

Partial Video Domain Adaptation with Partial Adversarial Temporal Attentive Network

In real-world applications, the constraint of sharing label spaces between source and target domain for vanilla VUDA may not hold. More practically, the source label space subsumes the target label space. In this chapter, such a scenario is investigated for Video Domain Adaptation, and the task of Partial Video Domain Adaptation (PVDA) is formulated with a novel method proposed to tackle PVDA. Section 7.1 introduces the motivation of formulating PVDA and the proposed method. Section 7.2 then introduces the proposed method in detail. This work pioneers in PVDA, and to our knowledge there are no current PVDA benchmark datasets available. Section 7.3 proposes several benchmark datasets to facilitate PVDA research, with our proposed method evaluated and the results presented and analyzed in Section 7.4. Section 7.5 concludes this chapter.

7.1 Introduction

Video-based problems have long been studied thanks to their wide applications in various fields. Neural networks have made notable advances in these problems with the availability of large-scale labeled video data. However, sufficiently large-scale training

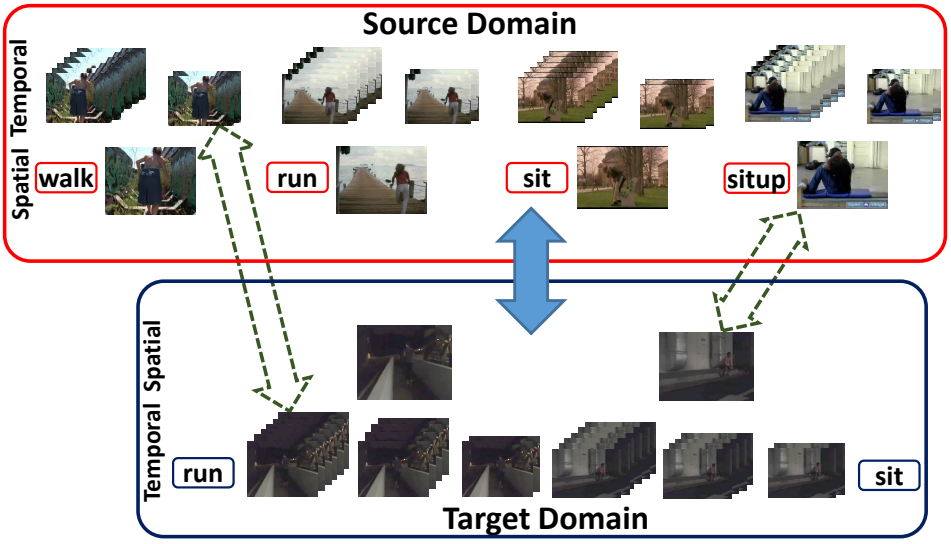


FIGURE 7.1: PVDA is a more general setting where the source label space subsumes the target label space. The key challenge of PVDA is the negative transfer caused by outlier source-only classes ('walk' and 'situp'), with extra probability triggered by the incorrect alignment of target temporal features to the source temporal features of the outlier classes, depicted as the left dashed arrow between videos from classes 'run' and 'walk'.

video data is sometimes unavailable, as annotations of video data are costly. It is desirable if networks trained in one video domain could be directly applied to another to reduce video labeling consumption. Various *Video-based Unsupervised Domain Adaptation* (VUDA) methods have been proposed to enable networks transfer knowledge from a labeled source domain to an unlabeled target domain by learning domain-invariant feature representations in the absence of target labels.

Though existing VUDA methods enable the learning of transferable features across domains, they generally assume that the video source and target domains share an identical label space, which may not hold in real-world applications. As it is often impractical to find a relevant video source domain with label space identical to that of the target domain of interest, commonly unlabeled. With the presence of large-scale labeled public video datasets, (e.g. HMDB and UCF), it is more feasible to transfer representations learned in these datasets to unlabeled small-scale datasets. Such a scenario is defined as *Partial Domain Adaptation* (PDA), which relaxes the constraint of identical source and target label spaces by assuming that the target label space is a subspace of the source one. This assumption is more practical since large-scale public video datasets can subsume categories of the small-scale target datasets. However, the PDA problem is more challenging, since source-only classes may negatively influence the distribution alignment

of target data, causing *negative transfer*, the key challenge of the PDA problem.

Compared to images that only contain spatial features, videos contain additional temporal features. This leads to a novel *Partial Video Domain Adaptation* (PVDA) problem, with trained networks transferred from video source domain to target domain, with the label space of video target domain being the subspace of the video source domain. When transferring networks for PVDA, negative transfer would be triggered due to the possible spatial-temporal domain shift as depicted in Figure 7.1, where the appearances of videos in class 'walk' are different from that of videos in class 'run', i.e. spatial features are different among videos in the two classes. However, videos from both classes share similar motion patterns where the actor moves further away from the camera in an upright position, indicating similar temporal features among the videos. When performing data distribution alignment, the similarities in temporal features would lead to videos in class 'run' of the target domain to incorrectly align with videos in class 'walk' of the source domain, triggering negative transfer.

A crucial step for tackling negative transfer in PVDA is the filtration of source-only outlier classes. Different from images, temporal features should be leveraged for PVDA from two perspectives: on one hand, effective temporal features should be constructed such that temporal features in outlier source-only classes discriminates those in target classes, alleviating the possibility of triggering negative transfer by temporal features; on the other hand, the temporal features should also contribute towards the filtration of source-only classes while eliminating possible mistakes caused by mis-classification of spatial features. To this end, we propose a **Partial Adversarial Temporal Attentive Network (PATAN)** to address the two challenges uniformly. **PATAN** first constructs robust overall temporal features by attentive combination of local temporal features which contain different aspects of the whole motion. The attentive combination builds upon the contribution of the local temporal feature towards the class filtration process where source-only classes are filtered. The constructed temporal feature would therefore have higher discriminability over source-only and target classes. Further, **PATAN** mitigates negative transfer in PDA through a class filtration process by utilizing local and overall temporal features jointly, alleviating possible mistakes during the class filtration process brought by the spatial features.

To further facilitate PVDA research, we propose three sets of benchmark datasets, built from widely used public datasets and a recent video dataset dedicated to low-illumination

videos. The benchmark datasets proposed are: (a) *UCF-HMDB$_{partial}$*, (b) *MiniKinetics-UCF*, and (c) *HMDB-ARID$_{partial}$*. The proposed datasets cover a wide range of PVDA scenarios, providing adequate baseline environment with distinct domain shift.

Our main contributions are summarized as follows:

* We formulated a novel and challenging *Partial Video Domain Adaptation* (PVDA) problem. To the best of our knowledge, this is the first research that explores partial transfer in videos.
* We analyze the challenges underlying PVDA and introduce PATAN to address the challenges. PATAN constructs robust temporal features by attending to filtration-contributing local temporal features, while utilizing both spatial and temporal features for accurate class filtration.
* Finally, we pioneer several PVDA benchmark datasets, and demonstrate the effectiveness of our proposed method, achieving state-of-the-art performance across the multiple PVDA benchmark datasets proposed.

7.2 Proposed Method

In the scenario of *Partial Video Domain Adaptation* (PVDA), we are given a source domain $\mathcal{D}_S = \{(V_{iS}, y_{iS})\}_{i=1}^{n_S}$ with n_S labeled videos associated with $|\mathcal{C}_S|$ classes, and a target domain $\mathcal{D}_T = \{V_{iT}\}_{i=1}^{n_T}$ with n_T unlabeled videos associated with $|\mathcal{C}_T|$ classes. The PVDA scenario is more general than VUDA by assuming that the source label space $\mathcal{C}_S$ is a superset of the target label space $\mathcal{C}_T$, i.e. $\mathcal{C}_T \subset \mathcal{C}_S$. The source and target domains of PVDA are characterized by two underlying probability distributions p and q respectively, where $p \neq q$. We also have $p_{\mathcal{C}_T} \neq q$, where $p_{\mathcal{C}_T}$ denotes the distribution of the source domain data in label space $\mathcal{C}_T$ of target domain.

To tackle the PVDA problem, we aim to construct a network capable of learning transferable features across source and target domains and minimizing the target classification risk. Compared to VUDA, PVDA poses more challenges to the network due to the existence of outlier label space in the source domain $\mathcal{C}_S \backslash \mathcal{C}_T$, which causes negative transfer effect to the network's performance. Meanwhile, during the training of the network, only unlabeled target domain data are accessible. Hence the part of which $\mathcal{C}_S$ shares

with $\mathcal{C}_T$ is unknown. Therefore the key towards mitigating negative effects lies in the class filtration process which filters out the outlier source-only classes.

Current PDA approaches are built for image-based PDA problems, where the negative transfer could only be triggered by the alignment of the distributions of spatial features. Whereas for videos, negative transfer could be additionally triggered by the alignment of the distributions of temporal features (e.g. the alignment of target videos in 'run' to source videos in 'walk' as depicted in Figure 7.1). Thanks to the fact that current feature extractors would pay more attention across the spatial dimension, current PDA approaches may not be sensitive to negative transfer caused by the incorrect alignment of temporal features. Therefore, we propose a novel Partial Adversarial Temporal Attentive Network (PATAN), to enable partial domain adaptation in an adversarial manner while mitigating negative transfer utilizing attentive temporal features. We begin by reviewing adversarial-based partial domain adaptation approaches, followed by a detailed illustration of PATAN.

7.2.1 Adversarial-based Partial Domain Adaptation

Domain adaptation (DA) is achieved by matching the feature distributions of the source and target domains. One major line of approaches learns the domain-invariant features $\mathbf{f}$ in an adversarial manner where additional domain discriminators are trained with the feature generators in a min-max fashion. More specifically, the parameters θ_f of the feature extractor G_f are learned by maximizing the losses of the domain discriminator G_d, while the parameters θ_d of the domain discriminator G_d are trained by minimizing the losses of the domain discriminators G_d. Additionally, the loss of the source classifier G_y is also minimized. The overall objective of adversarial-based DA networks can be formulated as in [90]:

$$\begin{aligned}\mathcal{L}_0 = &\frac{1}{n_S}\sum_{z_i\in\mathcal{D}_S} L_y(G_y(G_f(z_i)), y_i) \\ &- \frac{\lambda}{n_A}\sum_{z_i\in\mathcal{D}_A} L_d(G_d(G_f(z_i)), d_i),\end{aligned} \tag{7.1}$$

where z_i is an input data point, $\mathcal{D}_A = \mathcal{D}_S \cup \mathcal{D}_T$ is the union of source and target domains with $n_A = |\mathcal{D}_A|$, d_i is the domain label of input z_i, and λ is the trade-off for the domain loss L_d with respect to the source classification loss L_y. Both losses are implemented as

cross-entropy losses. The min-max optimization process is achieved by the connecting a Gradient Reverse Layer (GRL) to G_d.

While the aforementioned adversarial-based networks can be applied to standard DA tasks, yielding reliable results, their performance deteriorates for PDA tasks due to negative transfer caused by outlier source-only classes within the label space of $\mathcal{C}_S \backslash \mathcal{C}_T$. Hence a class filtration process is applied to filter out these outlier classes.

The end result of this class filtration process are class weights γ_l for each source domain label $l \in \mathcal{C}_S$, indicating the probability of each class of label space $\mathcal{C}_S$ overlapping with label space $\mathcal{C}_T$. To obtain the class weights γ, it is observed that the output of the source classifier G_y for data z_i well represents the probability distribution of z_i over the source label space $\mathcal{C}_S$. The probability of the target data assigned to labels with space overlapped between $\mathcal{C}_S$ and $\mathcal{C}_T$ should be significantly larger than the probability of the target data assigned to outlier classes with label space $\mathcal{C}_S \backslash \mathcal{C}_T$. Therefore, the class weights γ are generally obtained by the label predictions of the target data through the source classifier G_y, which indicates the probability of assigning the target data to each source class, and is formulated as:

$$\gamma = \frac{1}{n_T} \sum_{z_i \in \mathcal{D}_T} G_y(G_f(z_i)) = \frac{1}{n_T} \sum_{i=1}^{n_T} \hat{y}_i, \tag{7.2}$$

where $\hat{y}_i$ is the label prediction of target input data z_i.

To down-weigh the contributions of the data in outlier classes for PDA tasks, the class weights γ are applied to the adversarial-based networks, yielding the objective of PDA networks formulated as in [180]:

$$\begin{aligned} \mathcal{L}_p = & \frac{1}{n_S} \sum_{z_i \in \mathcal{D}_S} \gamma_{y_i} [L_y(G_y(G_f(z_i)), y_i) + \lambda L_d(G_d(G_f(z_i)), d_i)] \\ & - \frac{\lambda}{n_T} \sum_{z_i \in \mathcal{D}_T} L_d(G_d(G_f(z_i)), d_i), \end{aligned} \tag{7.3}$$

where y_i is the ground truth of input z_i in the source domain, while γ_{y_i} is the corresponding class weight.

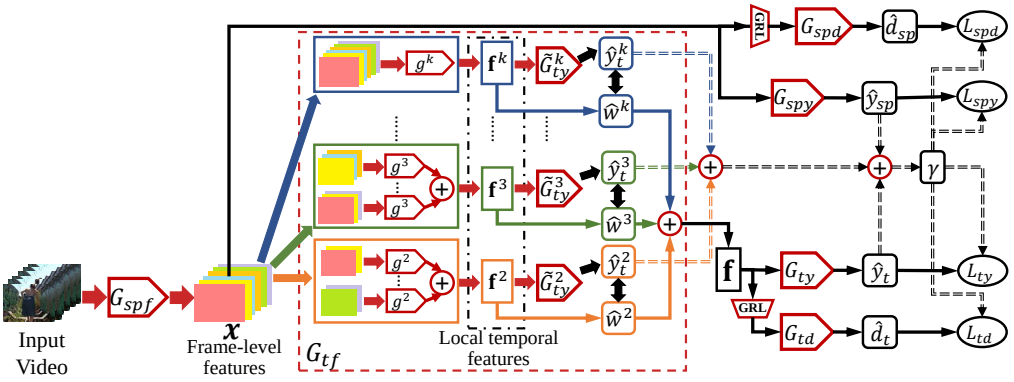


FIGURE 7.2: Architecture of the proposed PATAN. To mitigate negative transfer for PVDA effectively, robust overall feature $\mathbf{f}$ is constructed by weighted combination of local temporal features $\mathbf{f}^r$. The local temporal features $\mathbf{f}^r$ are built by fusing the time ordered frame-level features. The class weights of source domain classes γ averages over the label predictions of the spatial feature, weighted local temporal features and the overall temporal feature of target data. γ is applied to both the source domain label classifier and spatial/temporal domain discriminators. *Best viewed in color and zoomed in.*

7.2.2 Partial Adversarial Temporal Attentive Network

Intuitively, when tackling the PVDA problem, the approach in Section 7.2.1 could be directly integrated into videos, i.e. $z_i = V_i$. Given that videos contain both spatial and temporal features, one typical method for obtaining transferable video features is by separating the feature extractor G_f into a spatial feature G_{spf} and temporal feature extractor G_{tf}. The network constructed for PVDA could be formulated by simply substituting $G_f(z_i)$ in Equation 7.2 and Equation 7.3 with $G_{tf}(G_{spf}(V_i))$.

One major drawback of direct integration of the above PDA approach into videos lies in the fact that video representations obtained through conventional video feature extractors (e.g. convolution-based networks) are mainly from the spatial features. The overall temporal information is generally encoded implicitly, usually implemented as a temporal pooling mechanism. Without explicit temporal features, the class filtration process in Section 7.2.1 would depend mainly on the spatial features. Therefore the negative transfer may only be alleviated along the spatial dimension, while negative transfer can still be triggered by the matching of temporal features of data from outlier classes.

In view of such drawback, we propose **Partial Adversarial Temporal Attentive Network (PATAN)** to mitigate negative transfer by utilizing spatial and temporal features jointly, as shown in Figure 7.2. To utilize the temporal features of videos for negative

transfer mitigation, the temporal features should be explicitly extracted first. Given the fact that humans can recognize actions by reasoning the observations across time, temporal features could be extracted utilizing Temporal Relation Module [37]. We denote an input video with k frames as $V_i = \{x_i^{(1)}, x_i^{(2)}, ..., x_i^{(k)}\}$, where $x_i^{(j)}$ is the jth frame-level feature representation of the ith video obtained from the spatial feature extractor G_{spf}. The temporal feature of V_i is constructed by a combination of multiple local temporal features, each built upon clips with r temporal-ordered sampled frames where $r \in [2, k]$. Formally, a local temporal feature $\mathbf{f}_i^r$ is defined by:

$$\mathbf{f}_i^r = \sum_m g^r((V_i^r)_m), \tag{7.4}$$

where $(V_i^r)_m = \{x_i^{(a)}, x_i^{(b)}, ...\}_m$ is the mth clip with r temporal-ordered frames, with a and b denoting the frame indices. The local temporal feature of $\mathbf{f}_i^r$ is computed by fusing the time ordered frame-level features through function g^r, implemented as an Multi-Layer Perceptron (MLP).

Further, the key for mitigating negative transfer along the temporal dimension lies in the design of an effective class filtration process to down-weigh the effects of outlier classes with temporal features. The class filtration process is built upon the observation that the probability of the target data assigned to outlier classes with label space $\mathcal{C}_S \backslash \mathcal{C}_T$ should be significantly small. To make full use of the local temporal features and to eliminate possible mis-assignment of target classes to spatial features, we apply the above observation to all local temporal features of the target data. The label prediction of local temporal feature $\mathbf{f}_i^r$ is obtained as $\hat{y}_{ti}^r = \tilde{G}_{ty}^r(\mathbf{f}_i^r)$, which gives the probability distribution of $\mathbf{f}_i^r$ across the source label space $\mathcal{C}_S$. Here $\tilde{G}_{ty}^r$ is the auxiliary source classifier for $\mathbf{f}_i^r$, and is trained as cross-entropy loss with source local temporal features, i.e. $\mathbf{f}_{i'}^r$ where $V_{i'} \in \mathcal{D}_S$.

To obtain the overall temporal feature and the class weights of each source class, one straight-forward strategy is to aggregate all local temporal features and their corresponding label predictions. However, not all local temporal features are equally important towards the mitigation of negative transfer. We introduce a *label attention* mechanism to attend to local temporal features that contribute more toward the class filtration process. Specifically, the temporal features would be robust and the class filtration process would be effective only if temporal features in outlier source-only classes discriminates from those in target classes. If the features are of low discriminability and therefore the predictions are uncertain, the class weights of source classes which correlates with the

predictions would be similar across all source classes. The network would be unable to filter out outlier source-only classes. Therefore, the proposed network should construct effective overall temporal features which attend to discriminative features that better distinguish if the label of the input data lies within the target label space $\mathcal{C}_T$ or the outlier label space $\mathcal{C}_S \backslash \mathcal{C}_T$. The certainty of the label prediction $\hat{y}_{ti}^r$ which corresponds to $\mathbf{f}_i^r$ is quantified by the additive inverse of the entropy of the label prediction as:

$$\mathbb{C}(\hat{y}_{ti}^r) = \sum_{c=1}^{|\mathcal{C}_S|} \hat{y}_{ti,c}^r log(\hat{y}_{ti,c}^r). \tag{7.5}$$

For more stable optimization, a residual connection is added towards the formulation of the local temporal feature weight. The weight $\hat{w}_i^r$ of the local temporal feature $\mathbf{f}_i^r$ could therefore be generated as:

$$\hat{w}_i^r = \tanh(1 + \mathbb{C}(\hat{y}_i^r)), \tag{7.6}$$

where the $\tanh$ function is applied to ensure that weight $\hat{w}_i^r$ is constraint within a range of $[0, 1]$.

The weight $\hat{w}_i^r$ computed represents the contribution of the corresponding local temporal feature towards the class filtration process, which ultimately computes the class weights γ_l for each source domain label $l \in \mathcal{C}_S$. The above *label attention* weight is applied to both the generation of temporal attentive class weights utilizing the local temporal features and also the construction of the overall temporal feature. Formally, the overall temporal feature of input video V_i with k frames is constructed by:

$$\mathbf{f}_i = G_{tf}(V_i) = \sum_{r=2}^{k} \hat{w}_i^r \mathbf{f}_i^r, \tag{7.7}$$

where G_{tf} denotes the overall temporal feature extractor as shown in Figure 7.2. Meanwhile, the temporal attentive class weights generated for filtering out outlier source-only classes is formulated as:

$$\gamma = \frac{1}{n_T(k+1)} \sum_{i=1}^{n_T} (\hat{y}_{ti} + \hat{y}_{spi} + \sum_{r=2}^{k} \hat{w}_i^r \hat{y}_i^r), \tag{7.8}$$

where the $\hat{y}_{ti}$ and $\hat{y}_{spi}$ are the label predictions of the ith input target video with temporal feature $\mathbf{f}_i$ and spatial feature $\mathbf{x}_i$, computed as $\hat{y}_{ti} = G_{ty}(\mathbf{f}_i)$ and $\hat{y}_{spi} = G_{spy}(\mathbf{x}_i)$.

Finally, PATAN enables partial domain adaptation for videos by down-weighing the contributions of all source data belonging to the outlier label space $\mathcal{C}_S \backslash \mathcal{C}_T$. This is achieved by applying the temporal attentive class weight γ to the source label classifier as well as the spatial and temporal domain discriminators over the source domain data. The overall optimization objective of the proposed PATAN is formulated as:

$$\begin{aligned}
\mathcal{L} = & \frac{1}{n_S} \sum_{V_i \in \mathcal{D}_S} \gamma_{y_i} L_{ty}(G_{ty}(G_{tf}(G_{spf}(V_i))), y_i) \\
& + \frac{1}{n_S} \sum_{V_i \in \mathcal{D}_S} \gamma_{y_i} L_{spy}(G_{spy}(G_{spf}(V_i)), y_i) \\
& - \frac{\lambda_{sp}}{n_S} \sum_{V_i \in \mathcal{D}_S} \gamma_{y_i} L_{spd}(G_{spd}(G_{spf}(V_i)), d_i) \\
& - \frac{\lambda_{sp}}{n_S} \sum_{V_i \in \mathcal{D}_S} \gamma_{y_i} L_{td}(G_{td}(G_{tf}(G_{spf}(V_i))), d_i) \\
& - \frac{\lambda_t}{n_T} \sum_{V_i \in \mathcal{D}_T} L_{spd}(G_{spd}(G_{spf}(V_i)), d_i) \\
& - \frac{\lambda_t}{n_T} \sum_{V_i \in \mathcal{D}_T} L_{td}(G_{td}(G_{tf}(G_{spf}(V_i))), d_i),
\end{aligned} \tag{7.9}$$

where y_i is the ground truth of input V_i in the source domain, while γ_{y_i} is the corresponding class weight, and λ_{sp} and λ_t are the trade-offs for the domain loss L_{spd} and L_{td} with respect to the source classification losses L_{ty} and L_{spy}.

7.3 PVDA Benchmark Datasets

There are very limited cross-domain benchmark datasets for VUDA. Current cross-domain VUDA datasets are designed for the standard VUDA tasks, with the source label space constraint to be the same as target label space. To further facilitate PVDA research, we propose three sets of benchmark datasets, UCF-HMDB$_{partial}$, MiniKinetics-UCF, and HMDB-ARID$_{partial}$, which cover a wide range of PVDA scenarios and provide adequate baseline environment with distinct domain shift to facilitate PVDA research.

UCF-HMDB$_{partial}$. UCF-HMDB$_{partial}$ is built from two widely used video datasets: UCF101 (**U**) [74] and HMDB51 (**H**) [72]. The overlapping classes between the two datasets are collected, resulting in 14 classes with 2,780 videos. Among which are 980 training videos and 210 testing videos from HMDB51, 1,324 training videos and 266

UCF101 Class	HMDB51 Class
RockClimbingIndoor	climb
Diving	dive
Fencing	fencing
GolfSwing	golf
HandstandWalking	handstand
SoccerPenalty	kick_ball
PullUps	pullup
Punch	punch
PushUps	pushup
Biking	ride_bike
HorseRiding	ride_horse
Basketball	shoot_ball
Archery	shoot_bow
WalkingWithDog	walk

TABLE 7.1: List of overlapping classes between UCF101 and HMDB51.

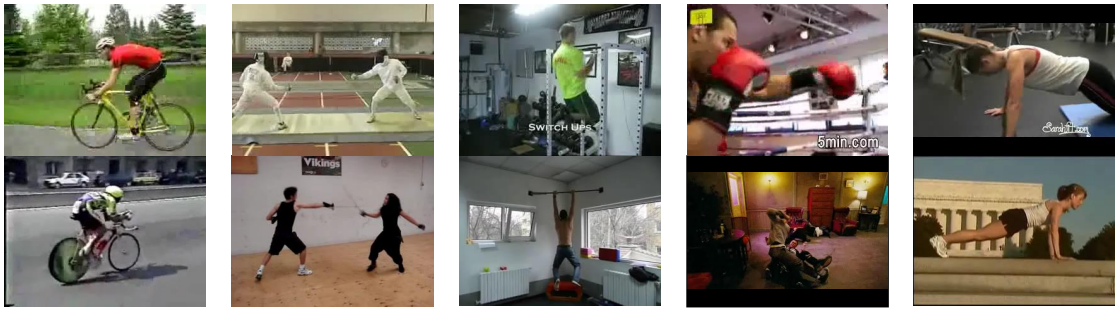


FIGURE 7.3: Sampled frames of videos from classes in UCF-HMDB$_{partial}$. Sampled frames from UCF101 are shown in the upper row, and those from HMDB51 are shown in the lower row.

testing videos from UCF101. The list of the 14 overlapping classes are listed in Table 7.1. The first 7 categories in alphabetic order of the target domain are chosen as target categories, and we construct two PVDA tasks: **U-14→H-7** and **H-14→U-7**. We follow the official split for the training and validation sets. Figure 7.3 shows the comparison of sampled frames from UCF-HMDB$_{partial}$.

MiniKinetics-UCF. MiniKinetics-UCF is built from two large-scale video datasets: MiniKinetics-200 (**M**) [55] and UCF101 (**U**) [74]. MiniKinetics-200 is a subset of the Kinetics [76] dataset, with 200 of its categories. There are 45 overlapping classes between MiniKinetics-200 and UCF101, as shown in Table 7.2. Similar to the construction of UCF-HMDB$_{partial}$, the first 18 categories in alphabetic order of the target domain are

MiniKinetics-200 Class	UCF101 Class	MiniKinetics-200 Class	UCF101 Class	MiniKinetics-200 Class	UCF101 Class
archery	Archery	high_jump	HighJump	pole_vault	PoleVault
bench_pressing	BenchPress	hula_hooping	HulaHoop	pull_ups	PullUps
biking_through_snow	Biking	javelin_throw	JavelinThrow	riding_or_walking_with_horse	HorseRiding
blowing_out_candles	BlowingCandles	jetskiing	Skijet	rock_climbing	RockClimbingIndoor
bowling	Bowling	juggling_balls	JugglingBalls	salsa_dancing	SalsaSpin
brushing_teeth	BrushingTeeth	long_jump	LongJump	shaving_head	ShavingBeard
canoeing_or_kayaking	Kayaking	lunge	Lunges	shot_put	Shotput
catching_or_throwing_baseball	BaseballPitch	making_pizza	PizzaTossing	skateboarding	SkateBoarding
catching_or_throwing_frisbee	FrisbeeCatch	marching	BandMarching	skiing	Skiing
clean_and_jerk	CleanAndJerk	playing_basketball	Basketball	squat	BodyWeightSquats
crawling_baby	BabyCrawling	playing_cello	PlayingCello	surfing_water	Surfing
diving_cliff	CliffDiving	playing_guitar	PlayingGuitar	swimming_breast_stroke	BreastStroke
dunking_basketball	BasketballDunk	playing_tennis	TennisSwing	tai_chi	Taichi
golf_driving	GolfSwing	playing_violin	PlayingViolin	throwing_discus	ThrowDiscus
hammer_throw	HammerThrow	playing_volleyball	VolleyballSpiking	walking_the_dog	WalkingWithDog

TABLE 7.2: List of overlapping classes between MiniKinetics-200 and UCF101.

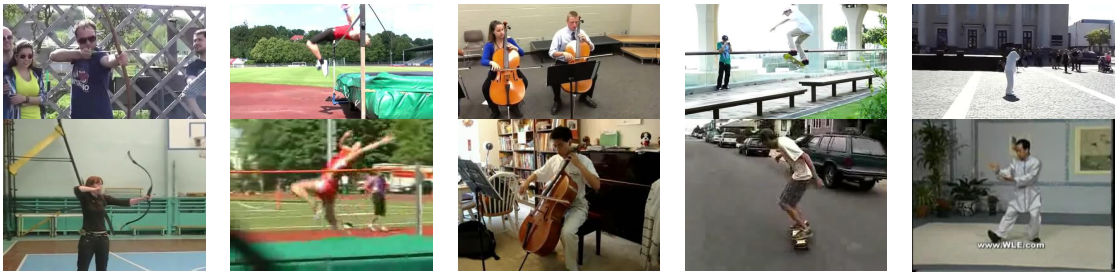


FIGURE 7.4: Sampled frames of videos from classes in MiniKinetics-UCF. Sampled frames from MiniKinetics-200 are shown in the upper row, while those from UCF101 are shown in the lower row.

chosen as target categories, resulting in two PVDA tasks: **M-45→U-18** and **U-45→M-18**. In this dataset, there are a total of 22,102 videos, with 4,253 training videos and 683 testing videos from UCF101, along with 16,743 training videos and 423 testing videos from MiniKinetics-200. The number of videos is nearly 8 times larger than that of UCF-HMDB$_{partial}$. Thus this dataset could validate the effectiveness of PVDA approaches on large-scale datasets. Figure 7.4 shows the comparison of sampled frames from MiniKinetics-UCF.

HMDB-ARID$_{partial}$. HMDB-ARID$_{partial}$ is built with the goal of leveraging current video datasets to boost performance on videos shot in adverse environments. It incorporates both HMDB51 (**H**) [72] and a more recent dark dataset, ARID (**A**) [174], with videos shot under adverse illumination conditions. Compared with current action recognition datasets (e.g. UCF101, HMDB51, MiniKinetics-200), videos in ARID are characterized by low brightness and low contrast. Statistically, videos in ARID possess much lower RGB mean value and standard deviation (std) as presented in Table 7.4. This leads

HMDB51 Class	ARID Class
RockClimbingIndoor	climb
Diving	dive
Fencing	fencing
GolfSwing	golf
HandstandWalking	handstand
SoccerPenalty	kick_ball
PullUps	pullup
Punch	punch
PushUps	pushup
Biking	ride_bike

TABLE 7.3: List of overlapping classes between HMDB51 and ARID.

Dataset	RGB Mean	RGB Std
HMDB51	[0.424,0.364,0.319]	[0.268,0.255,0.260]
UCF101	[0.409,0.397,0.358]	[0.266,0.265,0.270]
MiniKinetics-200	[0.435,0.394,0.381]	[0.225,0.225,0.214]
ARID	[0.079,0.074,0.073]	[0.101,0.098,0.090]

TABLE 7.4: Comparison of RGB mean and standard deviation (std) over common action recognition datasets and the ARID dataset.

to larger domain shift between ARID and HMDB51 compared to other cross-domain datasets. The overlapping classes between the two datasets are collected, resulting in 10 classes with 3,252 videos, which includes 2,012 training videos and 390 testing videos from ARID, and 700 training videos and 150 testing videos from HMDB51. The list of the 10 overlapping classes is listed in Table 7.3. Similar to the other two PVDA benchmarks, the first 5 categories in alphabetic order of the target domain are chosen as target categories, resulting in two PVDA tasks: **H-10**→**A-5** and **A-10**→**H-5**. For all the aforementioned benchmarks, the training and validation sets are separated following the official split methods. Figure 7.5 shows the comparison of sampled frames from HMDB-ARID$_{partial}$.n sets are separated following the official split methods.

7.4 Experiments

In this section, we evaluate our proposed PATAN by performing cross-domain action recognition on PVDA benchmark datasets introduced in Section 7.3. We present state-of-the-art results on all proposed benchmark datasets. We also present ablation studies

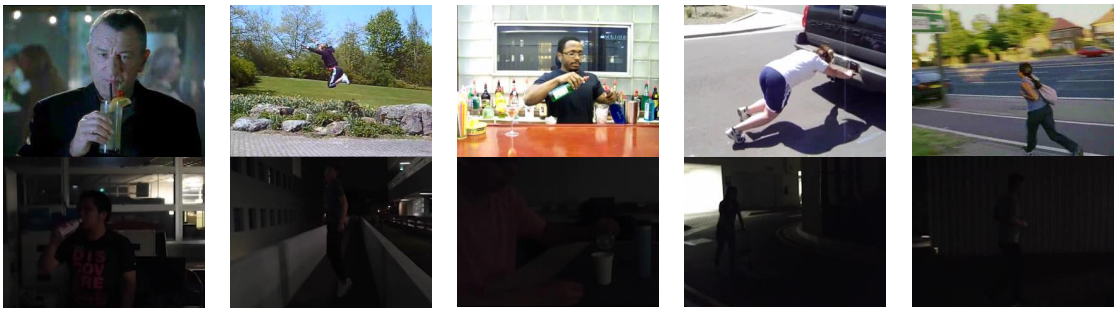


FIGURE 7.5: Sampled frames of videos from classes in HMDB-ARID$_{partial}$. Sampled frames from HMDB51 are shown in the upper row, while those from ARID are shown in the lower row.

and empirical analysis of our proposed network to verify our design.

7.4.1 Experimental Settings

We perform action recognition tasks on all three benchmark datasets: UCF-HMDB$_{partial}$, MiniKinetics-UCF and HMDB-ARID$_{partial}$, with a total of six cross-domain settings as suggested in Section 7.3. For all six settings, we use all labeled source videos and all unlabeled target videos for PVDA following standard evaluation protocols [101, 181]. We report the top-1 accuracy on the target datasets. Our networks and experiments are implemented using the PyTorch [149] library. To obtain video features, we instantiate Temporal Relation Network [37] as the backbone for video feature extraction for both source domain videos and target domain videos, with the model pretrained on ImageNet [30]. The source and target feature extractors share parameters. New layers are trained from scratch, and their learning rates are set to be 10 times that of the pretrained-loaded layers.

The stochastic gradient descent algorithm [136] is used for optimization, with the weight decay set to 0.0001 and the momentum to 0.9. The batch size is set to 8 per GPU. Our initial learning rate is set to 0.005 and is divided by 10 for two times during the training process. We train our networks with a total of 50 epochs for UCF-HMDB$_{partial}$ and HMDB-ARID$_{partial}$, while for MiniKinetics-UCF we train for 30 epochs. The flip-coefficient of the Gradient Reverse Layer (GRL) is increased gradually from 0 to 1 as in DANN [89]. All experiments are conducted using two NVIDIA RTX 2080 GPUs.

7.4.2 Overall Results and Comparisons

We compare the performance of PATAN with competitive and state-of-the-art UDA/VUDA approaches and state-of-the-art PDA approaches. These include: (a) adversarial-based methods: DANN [89], TA^{3}N [115], PADA [180] and ETN [182]; and (b) discrepancy-based methods: MK-MMD [101], MCD [178] and MDD [93]. Additionally, we report the results of the backbone feature extractor TRN, where TRN is trained with supervised source data only and validated on the target data. Table 7.5 shows the comparison of performances between our proposed PATAN and the methods as mentioned in all six PVDA settings.

Methods		UCF-HMDB$_{partial}$		MiniKinetics-UCF		HMDB-ARID$_{partial}$	
		U-14→H-7	**H-14→U-7**	**M-45→U-18**	**U-45→M-18**	**H-10→A-5**	**A-10→H-5**
Source-only	TRN [37]	62.85%	78.95%	78.77%	54.14%	14.10%	26.00%
Adversarial-based	DANN [89]	60.95%	74.44%	79.21%	52.25%	20.77%	12.00%
	TA^{3}N [115]	50.49%	70.68%	75.70%	48.23%	18.30%	24.00%
	PADA [180]	65.71%	82.33%	82.43%	61.23%	21.79%	30.67%
	ETN [182]	67.88%	82.89%	83.33%	62.51%	21.40%	28.82%
Discrepancy-based	MK-MMD [101]	58.57%	82.71%	79.79%	55.79%	21.28%	14.00%
	MCD [178]	55.71%	73.31%	75.13%	52.48%	12.56%	14.67%
	MDD [93]	62.58%	80.45%	80.12%	50.35%	15.13%	9.33%
Ours	**PATAN**	**73.81%**	**89.85%**	**86.82%**	**65.25%**	**26.41%**	**34.67%**

TABLE 7.5: Results for Partial Video Domain Adaptation on UCF-HMDB$_{partial}$, MiniKinetics-UCF and HMDB-ARID$_{partial}$.

The results in Table 7.5 show that our proposed PATAN achieves the best results on all six settings, and substantially outperforms previous approaches by noticeable margins. It can be observed that for all UDA/VUDA approaches, i.e. DANN, TA^{3}N, MK-MMD, MCD, and MDD, there exists at least three settings where their performances are inferior to that of TRN trained without any domain adaptation methods. This suggests that these methods suffer from the negative transfer issue of PVDA.

Compared to previous PDA approaches PADA and ETN, our proposed PATAN exceeds both approaches consistently, with an average 11.27% relative improvement towards PADA, and an average 11.57% relative improvement towards ETN. These large improvements imply the effectiveness of building temporal attentive features and incorporating local and overall temporal features for class filtration. In particular, the improvement of PATAN with respect to PADA and ETN is most significant for HMDB-ARID$_{partial}$, with a relatively average improvement of 17.12% and 21.85% towards PADA and ETN respectively. HMDB-ARID$_{partial}$ possesses the largest domain shift across the source and target domains, with the lowest source-only accuracies. This suggests that the class filtration process may not be accurate by utilizing any single feature, which explains the relatively

Methods	**U-14→H-7**	**H-14→U-7**
PATAN	**73.81%**	**89.85%**
PATAN w/o attentive	71.43%	85.34%
PATAN w/o local weights	70.47%	84.21%
PATAN w/o classifier	69.52%	82.71%
PATAN w/o adversarial	67.14%	81.58%

TABLE 7.6: Ablation studies of PATAN on UCF-HMDB$_{partial}$.

small improvement in performances of PADA and ETN compared to UDA/VUDA approaches. On the contrary, by constructing temporal features by *label attention*, the effectiveness of the class filtration process is improved with features of higher certainty attended, thus explains the large improvements brought by PATAN.

7.4.3 Ablation Studies

To go deeper with the efficacy of the proposed PATAN network, we perform ablation studies by evaluating PATAN against its variants: (a) **PATAN w/o attentive** is the variant where the *label attention* weight is not computed, therefore the overall temporal feature and class weights of source classes are the result of the aggregation of all local temporal features and the label predictions of all local and overall temporal features with that of the spatial feature; (b) **PATAN w/o local weights** is the variant where the class weights γ do not incorporate all the local weights; (c) **PATAN w/o classifier** is the variant without the class weights γ applied on the source spatial and temporal classifiers, and (d) **PATAN w/o adversarial** is the variant without the class weights applied on the spatial and temporal domain discriminators. The results of the variants are presented in Table 7.6.

Specifically, PATAN outperforms PATAN w/o attentive by a noticeable margin proves the necessity of combining local temporal features and class weights with *label attention*, which constructs more discriminative overall temporal features. Similarly, PATAN's superior performance over PATAN w/o local weights proves that the class filtration process could be improved by utilizing label prediction of local temporal features. We note that the results of both PATAN w/o attentive and PATAN w/o local weights outperform that of PADA and ETN. This further justifies the effectiveness of both utilizing local temporal features for class filtration and constructing attentive overall temporal features with *label attention*. Further, PATAN outperforms both PATAN w/o classifier and PATAN

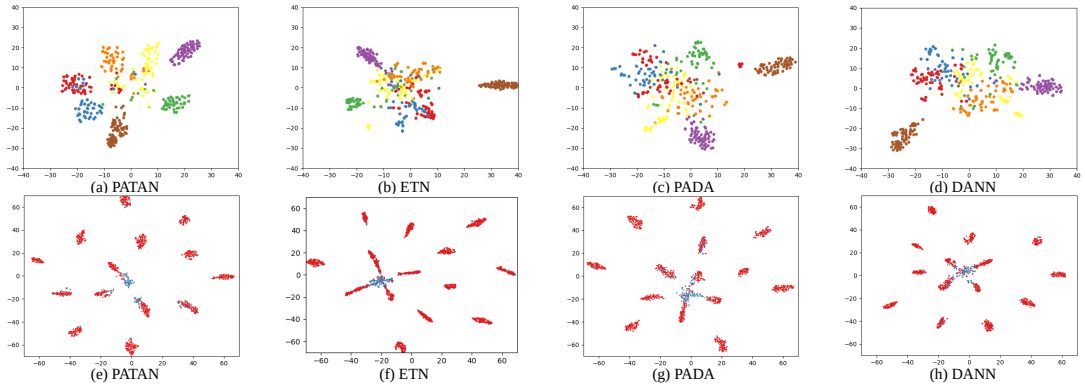


FIGURE 7.6: Visualization of features learned by PATAN, ETN, PADA, and DANN, with class information ((a)-(d)) and domain information ((e)-(h)). Different classes are denoted by different colors. The red dots represent data from the source domain while the blue dots represent data from the target domain.

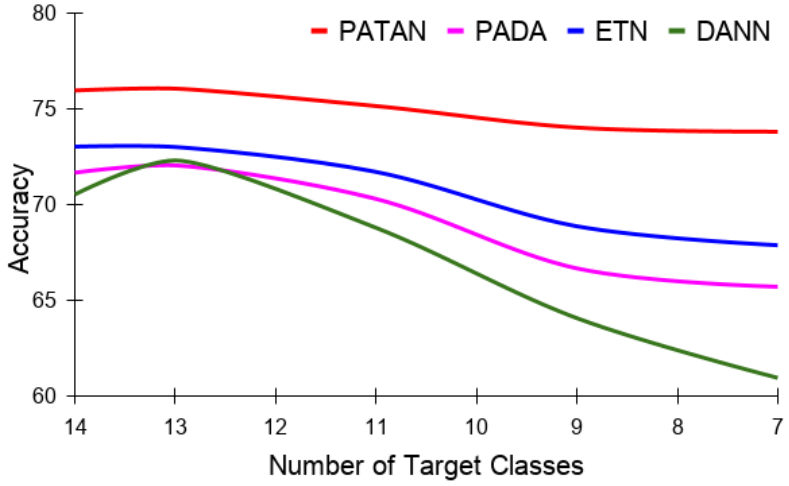
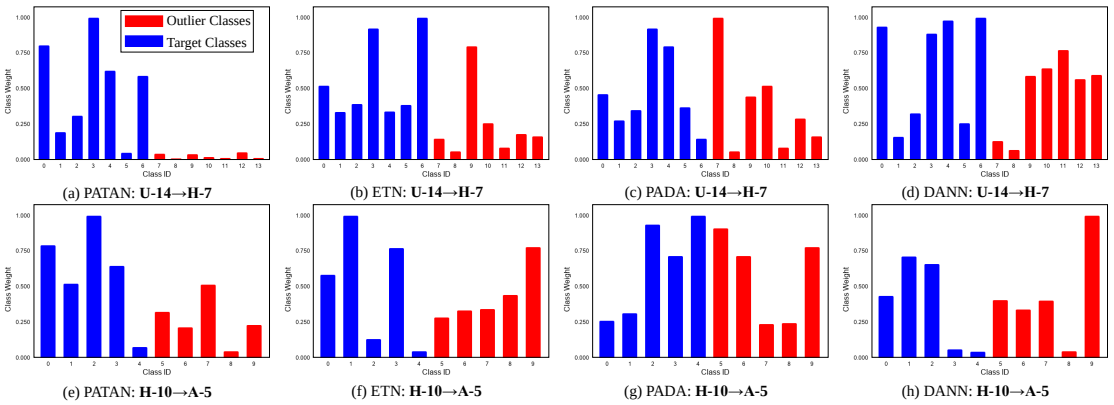


FIGURE 7.7: Accuracy with different number of target classes.

w/o adversarial by huge margins. This strongly suggests that the class weights applied can assign small weights on outlier classes and down-weigh the source data of the outlier classes effectively, the class weights applied thus mitigates negative transfer and boost the performance for PVDA.

7.4.4 Empirical Analysis

To further understand our proposed PATAN, we perform empirical analysis focusing on four areas of interest: class weights visualization, affect of number of target classes and feature visualization.



(a) PATAN: **U-14→H-7** (b) ETN: **U-14→H-7** (c) PADA: **U-14→H-7** (d) DANN: **U-14→H-7**

(e) PATAN: **H-10→A-5** (f) ETN: **H-10→A-5** (g) PADA: **H-10→A-5** (h) DANN: **H-10→A-5**

FIGURE 7.8: Histograms of class weights learned by PATAN, ETN, PADA and DANN on settings **U-14→H-7** and **H-10→A-5**.

Class weights visualization. We first illustrate and compare the learned class weights γ generated by methods PATAN, ETN, PADA and DANN for settings **U-14→H-7** and **H-10→A-5** in Figure 7.8. It could be observed that our proposed PATAN assigns much smaller weights to the outlier source only classes than to the shared target classes, which shows that PATAN could effectively filter out the outlier classes. It is noted that the difference of class weights between target and outlier classes is less significant for **H-10→A-5**, given the much larger cross-domain shift for dataset HMDB-ARID$_{partial}$. Despite the difficulty brought by the large domain shift, our proposed PATAN still assigns significantly larger weights for target classes compared to other methods. The much larger weights assigned to target classes show that our network could effectively filter out outlier classes, therefore explains the strong performance of PATAN on both datasets. Though both PADA and ETN incorporate class filtration processes to filter outlier classes, the effectiveness of such a process is hindered by failing to incorporate temporal features. This results in much poorer performances.

Affect of number of target classes. We investigate a wider spectrum of PVDA by varying the number of target classes, conducted with the UCF-HMDB$_{partial}$ dataset. The result of the accuracy of the target dataset against the different numbers of target classes is shown in Figure 7.7. It is observed that the performance of DANN, PADA, and ETN degrades noticeably with fewer target classes. This is a clear indication of negative transfer brought by the increasing outlier classes. Comparatively, the performance of PATAN is more stable and is consistently better than all compared methods. The stability of performance suggests that PATAN effectively alleviates the influence of outlier classes. It could also be observed that when the number of target classes is equivalent to that of

source classes (in this case 14), the PVDA task is turned into a standard VUDA task. Under this condition, our PATAN also performs better than DANN. This shows that the class filtration process will not degrade performance when there are no outlier classes.

Feature visualization. We further plot the t-SNE embeddings [183] of the features learned by PATAN, ETN, PADA, and DANN for the **U-14**→**H-7** with class information in the target domain as shown in Figure 7.6 (a)-(d), and with domain information as shown in Figure 7.6 (e)-(h). From Figure 7.6 (a), it is observed that the features learned by PATAN are more clustered. This proves that features extracted by PATAN with *label attention* have higher discriminability. Meanwhile, Figure 7.6 (f)-(h) shows that other methods align target data to all source classes, which includes outlier ones, triggering negative transfer. We note that though ETN and PADA include class filtration processes, the negative transfer is still triggered due to misalignment of temporal features not utilized in their class filtration processes. Comparatively, PATAN only aligns target data to the shared classes (7 classes), alleviating the effects of the outlier classes.

7.5 Summary

In this chapter, we propose a novel approach for partial video domain adaptation (PVDA). Unlike previous approaches where only spatial features are utilized for mitigating negative transfer in partial domain adaptation, the new PATAN tackles PVDA with full utilization of both spatial and temporal features, filtering out outlier source-only classes effectively. The proposed PATAN also attends to local temporal features that contribute more towards the class filtration process. We further introduce novel PVDA benchmark datasets to facilitate PVDA research, which are the first PVDA benchmark datasets introduced. Our proposed PATAN addresses the PVDA problem well, justified by extensive experiments across the proposed PVDA benchmark datasets.

Chapter 8

Conclusion and Future Works

8.1 Conclusions

In this thesis, we present our work on robust and efficient vision-based action recognition. Vision-based action recognition aims to explore both temporal features and spatiotemporal correlation features to improve recognition accuracy in videos. Two deep learning methods were proposed to tackle the challenge of extracting robust and efficient temporal features and spatiotemporal correlation features. Meanwhile, existing vision-based action recognition methods lack robustness towards videos shot in adverse environments, a typical example being dark videos. We, therefore, introduced and analyzed a novel vision-based action recognition dataset comprising dark videos. Since such dataset with dark videos is of small-scale compared to existing benchmark datasets while increasing dataset size is impractical given the high cost of video annotation, methods that allow models transfer from existing dataset to dark videos are investigated in this thesis. We developed and proposed two VUDA methods to tackle the domain shift across different domains (scenarios) in real-world adaptation problems.

In Chapter 3, we propose a novel temporal feature method: ACTF, exploiting inter-frame corresponding-regional correlation for capturing temporal information. Existing methods for temporal feature extraction either rely on costly optical flow computation and estimation, or suffer from partial temporal information loss due to pooling operations along the temporal dimension. Different from previous methods, ACTF is an attentive combination of both bilinear inter-frame correlation and linear inter-frame correlation features, and is able to extract temporal information explicitly without the need for optical flow.

Our ACTF can be combined with any spatial feature extraction network, yielding comparable performance to state-of-the-art methods in vision-based action recognition.

In Chapter 4, we proposed to capture spatiotemporal correlation features in videos in the form of long-range spatiotemporal dependencies efficiently by proposing a Pyramid Non-Local (PNL) block. The existing Non-Local Block along with its variants is proposed to extract long-range spatiotemporal dependencies in videos. However, they bring a significant increase in computation cost to the inserted networks. Furthermore, existing methods ignore the spatiotemporal correlations at the regional level. To solve the above issues, our proposed PNL extends the original NLBlock by incorporating regional feature correlations at multiple scales through a pyramid structural design. This extension not only achieves state-of-the-art results on benchmark datasets but also requires significantly less computation cost, improving network efficiency by a remarkable margin.

While the introduction of large-scale video datasets has pushed forward the development of vision-based action recognition, there are still scenarios where current datasets fell short. Dark videos are one such scenario as current datasets are commonly constructed from web videos shot in normal illumination. The shortage of dark videos in current datasets implies that existing methods for vision-based action recognition may not be robust to such videos. To overcome this issue, we introduced a novel dataset: ARID, dedicated to vision-based action recognition in dark videos in Chapter 5. ARID is thoroughly analyzed and distinct characteristics of real dark videos are presented. With the introduction of ARID, further improvements in models and frame enhancement methods that better suit dark videos could be facilitated.

Given the high cost of video annotation required to expand ARID to a larger scale, in Chapter 6 we proposed a novel ACAN network for VUDA. The limited existing VUDA methods align cross-domain video data along the temporal dimension. Whereas ACAN tackles VUDA by aligning correlation features across domains. ACAN is able to further improve the effectiveness of correlation feature alignment by aligning the joint distribution of correlation information by minimizing PCD. The development of VUDA is hindered by the lack of cross-domain datasets with large domain shifts. To this end, we further introduced a more challenging VUDA dataset in Chapter 6: the *HMDB-ARID* dataset which leverages current datasets to boost performance on dark videos. Our proposed ACAN achieves state-of-the-art results on both current and novel VUDA datasets.

Existing VUDA approaches assume that the source and target domain share a common label space, which may not be applicable in real-world scenarios. Chapter 7 therefore formulates the PVDA problem where the label space of the source video domain subsumes that of the target video domain. To this end, we proposed PATAN in Chapter 7 to tackle PVDA by constructing robust temporal features through attention on filtration-contributing local temporal features. Additionally, PATAN utilizes both spatial and temporal features for precise class filtration that could mitigate negative transfer. To facilitate further PVDA research, several PVDA benchmark datasets are also introduced in Chapter 7, while PATAN performs best across all benchmark datasets proposed.

8.2 Future Work

In total, five works are presented in this thesis for robust and efficient vision-based action recognition. Effective improvements are achieved in all these works, and the successes of these methods inspire us to conduct further in-depth research that addresses their current limitations.

For the ACTF presented in Chapter 3, given that ACTF could be inserted in a plug-and-play manner, application of ACTF onto more complex networks, such as LSTM-based networks would also be explored. Further, ACTF is built by combining the bilinear and linear inter-frame correlation features, where the bilinear inter-frame correlation feature is computed for all regions across the spatial feature, further improvements may be achieved if the corresponding regions are selected based on the frame content. Meanwhile, the application of our ACTF could be explored on more challenging higher-level tasks, e.g. action prediction or action recognition in untrimmed videos. Such explorations would also be conducted for the PNL module proposed in Chapter 4, as extracting the spatiotemporal correlation features plays an essential role in many high-level tasks such as object-detection or video description. Further, while PNL upscales the effectiveness of the original NL incorporating multi-scale regional correlations, the multi-scale features are obtained from a single convolution layer. In fact, output features from different convolution layers are of different scale and could contain effective scale representation. Therefore, PNL could be also extended to obtain multi-scaled regional correlation with features from different convolution layers, which may further improve its effectiveness.

Meanwhile, as stated in Chapter 5, the ARID dataset proposed is currently under further development. One essential development is the further expansion of the ARID dataset to include more action classes and data, to provide a more solid data foundation for the task of action recognition in the dark. Further, while ACAN and PATAN are introduced in Chapter 6 and Chapter 7 in an effort to leverage current datasets to boost network performance in dark videos, we have not explored on adapting models trained on synthetic dark datasets to real dark videos. Though analysis in Chapter 5 show distinct characteristics for real dark videos (i.e. videos in ARID) compared to synthetic dark videos, they still share some common characteristics (e.g. low luminance values). These common characteristics may further boost performance for models in dark videos. Meanwhile, while current frame enhancement methods may not be effective in recognizing action in dark videos, frame enhancement methods that could enhance visibility while introducing less adversarial attacks on the original dark video frames may likely bring solid improvements to the task of action recognition in the dark. These methods would be explored in the future.

Meanwhile, with regards to research on VUDA and PVDA as investigated in Chapter 6 and Chapter 7, though both ACAN and PATAN achieve state-of-the-art performances, the gap to the upper bound performance for all target domains are still relatively large, especially on the novel HMDB-ARID and HMDB-ARID$_{partial}$ datasets for vanilla VUDA and PVDA, suggesting further improvements could be made by better alignment of temporal features or correlation features. While spatiotemporal correlation features have not been utilized for the task of PVDA, only partial spatiotemporal correlation features extracted from a single layer of the video feature encoder have been utilized for ACAN. In the future, the alignment of cross-domain videos could be further improved by jointly aligning spatiotemporal correlation features extracted from multiple levels across the video feature encoder. Meanwhile, since dark videos are characterized by its low illumination and low contrast as stated in Chapter 5, one intuitive method for better adaptation from normal videos is to utilize adequate frame enhancement methods on dark videos before or during the adaptation process. Alternatively, other features that may be more apparent in dark videos such as edges or streaks may also be utilized for better adaptation. These possible improvements could be further explored. Additionally, cross-domain video datasets that involve a variety of large domain shift scenarios, such as blurry or hazy videos may also be explored.

Author's Publications

Journal Articles

(A) Journal papers that are included in the thesis

1. **Yuecong Xu**, Jianfei Yang, Kezhi Mao, Jianxiong Yin and Simon See. "Exploiting Inter-Frame Regional Correlation for Efficient Action Recognition." *Expert Systems with Applications* 178 (2021): 114829.
2. **Yuecong Xu**, Haozhi Cao, Jianfei Yang, Kezhi Mao, Jianxiong Yin and Simon See. "PNL: Efficient long-range dependencies extraction with pyramid non-local module for action recognition." *Neurocomputing* 447 (2021): 282-293.
3. **Yuecong Xu**, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "ARID: A Comprehensive Study on Recognizing Actions in the Dark and A New Benchmark Dataset." Submitted to *Applied Soft Computing*.
4. **Yuecong Xu**, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "Aligning Correlation Information for Domain Adaptation in Action Recognition." Submitted to *IEEE Transactions on Neural Networks and Learning Systems*.

(B) Journal papers that are not included in the thesis

1. **Yuecong Xu**, Jianfei Yang, Kezhi Mao."Semantic-filtered Soft-Split-Aware video captioning with audio-augmented feature." *Neurocomputing* 357 (2019): 24-35.
2. Pengfei Li, Kezhi Mao, **Yuecong Xu**, Qi Li, and Jiaheng Zhang. "Bag-of-concepts representation for document classification based on automatic knowledge acquisition from probabilistic knowledge base." Knowledge-Based Systems 193 (2020): 105436.

3. Haozhi Cao, **Yuecong Xu**, Jianfei Yang, Kezhi Mao, Jianxiong Yin and Simon See. "Effective Action Recognition with Embedded Key Point Shifts."*Pattern Recognition* 120(2021): 108172.

4. Qi Li, Kezhi Mao, Pengfei Li, **Yuecong Xu**, and Y.M. Edmond Lo. "A Novel End-to-end Neural Network for Named Entity Typing with Task-unrelated Entities." Submitted to *Expert Systems with Applications*.

Conference Proceedings

(A) Conference papers that are included in the thesis

1. **Yuecong Xu**, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin and Simon See. "ARID: A New Dataset for Recognizing Action in the Dark." In *Deep Learning for Human Activity Recognition, DL-HAR 2021*, Communications in Computer and Information Science 1370, 2021.

2. **Yuecong Xu**, Jianfei Yang, Haozhi Cao, Zhenghua Chen, Qi Li and Kezhi Mao. "Partial Video Domain Adaptation with Partial Adversarial Temporal Attentive Network." Accepted by *International Conference on Computer Vision (ICCV) 2021*.

(B) Conference papers that are not included in the thesis

1. Jianfei Yang, Hailin Chen, **Yuecong Xu**, Ziji Shi, Ruikang Luo, Lihua Xie, and Rong Su."Domain Adaptation for Degraded Remote Scene Classification." In *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 111-117. IEEE, 2020.

2. Chenxi Liao, and **Yuecong Xu**. "Extracting Temporal Features by Key Points Transfer for Effective Action Recognition." In *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pp. 741-746. IEEE, 2020.

3. Haozhi Cao, **Yuecong Xu**, Jianfei Yang, Kezhi Mao, Jianxiong Yin and Simon See. "Self-Supervised Video Representation Learning by Video Incoherence Detection." Submitted to *International Conference on Computer Vision (ICCV) 2021*.

Bibliography

[1] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. xix, 12, 13

[2] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014. xix, 13, 14, 28, 31, 32, 38, 68, 79

[3] J.W. Davis and A.F. Bobick. The representation and recognition of human movement using temporal templates. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 928–934, 1997. 10

[4] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, volume 2, pages 1395–1402, 2005. 10

[5] Daniel Weinland, Remi Ronfard, and Edmond Boyer. Free viewpoint action recognition using motion history volumes. *Computer Vision and Image Understanding*, 104(2):249–257, 2006. 10

[6] Ziming Zhang, Yiqun Hu, Syin Chan, and Liang-Tien Chia. Motion context: A new representation for human action recognition. In *Motion Context: A New Representation for Human Action Recognition*, pages 817–829, 2008. 11

[7] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *19th British Machine Vision Conference, September 2008*, pages 1–10, 2008. 11

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893, 2005. 11

[9] Laptev and Lindeberg. Space-time interest points. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 432–439, 2003. 11

[10] Debapratim Das Dawn and Soharab Hossain Shaikh. A comprehensive survey of human action recognition with spatio-temporal interest point (stip) detector. *The Visual Computer*, 32(3):289–306, 2016. 11

[11] Christopher G. Harris and Mike Stephens. A combined corner and edge detector. In *Proc. 4th Alvey Vision Conf., Manchester, U.K., Aug. 1988*, pages 147–151, 1988. 11

[12] Ivan Laptev. On space-time interest points. *international conference on computer vision*, 64(2):107–123, 2005. 11

[13] Bhaskar Chakraborty, Michael B. Holte, Thomas B. Moeslund, and Jordi Gonzílez. Selective spatio-temporal interest points. *Computer Vision and Image Understanding*, 116(3):396–410, 2012. 11

[14] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. In *ECCV '02 Proceedings of the 7th European Conference on Computer Vision-Part I*, volume 2350, pages 128–142, 2002. 11

[15] Geert Willems, Tinne Tuytelaars, and Luc Gool. An efficient dense and scale-invariant spatio-temporal interest point detector. In *ECCV '08 Proceedings of the 10th European Conference on Computer Vision: Part II*, pages 650–663, 2008. 11

[16] Xiaojiang Peng, Limin Wang, Xingxing Wang, and Yu Qiao. Bag of visual words and fusion methods for action recognition. *Computer Vision and Image Understanding*, 150:109–125, 2016. 11

[17] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV'10 Proceedings of the 11th European conference on Computer vision: Part IV*, volume 6314, pages 143–156, 2010. 11

[18] Xi Zhou, Kai Yu, Tong Zhang, and Thomas S. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV'10 Proceedings of the 11th European conference on Computer vision: Part V*, pages 141–154, 2010. 11

[19] Saima Nazir, Muhammad Haroon Yousaf, and Sergio A Velastin. Evaluating a bag-of-visual features approach using spatio-temporal features for action recognition. *Computers & Electrical Engineering*, 72:660–669, 2018. 11

[20] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *2013 IEEE International Conference on Computer Vision*, pages 3551–3558, 2013. 12

[21] Navneet Dalal, Bill Triggs, and Cordelia Schmid. Human detection using oriented histograms of flow and appearance. *Lecture Notes in Computer Science*, pages 428–441, 2006. 12

[22] Adrien Gaidon, Zaid Harchaoui, and Cordelia Schmid. Activity representation with motion hierarchies. *International Journal of Computer Vision*, 107(3):219–238, 2014. 12

[23] Heng Wang, Dan Oneata, Jakob Verbeek, and Cordelia Schmid. A robust and efficient video representation for action recognition. *International Journal of Computer Vision*, 119(3):219–238, 2016. 12

[24] Xiaojiang Peng, Changqing Zou, Yu Qiao, and Qiang Peng. Action recognition with stacked fisher vectors. In *European Conference on Computer Vision*, pages 581–595. Springer, 2014. 12

[25] Adam H Marblestone, Greg Wayne, and Konrad P Kording. Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10: 94, 2016. 12

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 12, 15, 79

[27] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[28] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 13, 15, 56

[29] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017. 12, 15, 52

[30] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 12, 72, 124

[31] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016. 13, 44

[32] Melvyn A Goodale and A David Milner. Separate visual pathways for perception and action. *Trends in neurosciences*, 15(1):20–25, 1992. 13

[33] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint pattern recognition symposium*, pages 214–223. Springer, 2007. 13, 14

[34] R Christoph and Feichtenhofer Axel Pinz. Spatiotemporal residual networks for video action recognition. *Advances in Neural Information Processing Systems*, pages 3468–3476, 2016. 13, 38

[35] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks for action recognition in videos. *IEEE transactions on pattern analysis and machine intelligence*, 41(11):2740–2755, 2018. 14, 32, 38, 44, 79

[36] Zhenzhong Lan, Yi Zhu, Alexander G Hauptmann, and Shawn Newsam. Deep local video feature for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 1–7, 2017. 14

[37] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018. 14, 24, 105, 118, 124, 125

[38] Limin Wang, Yu Qiao, and Xiaoou Tang. Action recognition with trajectory-pooled deep-convolutional descriptors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4305–4314, 2015. 14

[39] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 14

[40] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018. 14

[41] Wenbin Li, Da Chen, Zhihan Lv, Yan Yan, and Darren Cosker. Learn to model blurry motion via directional similarity and filtering. *Pattern Recognition*, 75: 327–338, 2018. 14

[42] Lijie Fan, Wenbing Huang, Chuang Gan, Stefano Ermon, Boqing Gong, and Junzhou Huang. End-to-end learning of motion representation for video understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6016–6025, 2018. 14, 38

[43] AJ Piergiovanni and Michael S Ryoo. Representation flow for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9945–9953, 2019. 14

[44] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634, 2015. 15

[45] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 15

[46] Joe Yue-Hei Ng, Matthew Hausknecht, Sudheendra Vijayanarasimhan, Oriol Vinyals, Rajat Monga, and George Toderici. Beyond short snippets: Deep networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4694–4702, 2015. 15

[47] Yemin Shi, Yonghong Tian, Yaowei Wang, Wei Zeng, and Tiejun Huang. Learning long-term dependencies for action recognition with a biologically-inspired deep network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 716–725, 2017. 15, 38

[48] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 15

[49] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):221–231, 2012. 15

[50] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 15

[51] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 15, 28, 37, 38, 60, 61, 68, 79

[52] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 15, 16, 38, 44, 60, 61, 67, 70, 72, 79, 103, 104

[53] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6450–6459, 2018. 15, 16, 37, 38, 56, 79

[54] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018. 15, 38, 79

[55] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 305–321, 2018. 16, 19, 49, 56, 60, 61, 79, 121

[56] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *proceedings of the IEEE International Conference on Computer Vision*, pages 5533–5541, 2017. 16, 79

[57] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5552–5561, 2019. 16

[58] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Multi-fiber networks for video recognition. In *Proceedings of the european conference on computer vision (ECCV)*, pages 352–367, 2018. 16, 37, 38, 56, 60, 61, 103, 106

[59] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6202–6211, 2019. 16

[60] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005. 16, 48

[61] Hongjun Li and Ching Y Suen. A novel non-local means image denoising method based on grey theory. *Pattern Recognition*, 49:237–248, 2016. 16, 48

[62] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7794–7803, 2018. 16, 32, 48, 50, 51, 54, 56, 57, 58, 60, 61, 93, 96

[63] Kaiyu Yue, Ming Sun, Yuchen Yuan, Feng Zhou, Errui Ding, and Fuxin Xu. Compact generalized non-local network. In *Advances in Neural Information Processing Systems*, pages 6510–6519, 2018. 17, 60, 61

[64] Yunpeng Chen, Yannis Kalantidis, Jianshu Li, Shuicheng Yan, and Jiashi Feng. Aˆ 2-nets: Double attention networks. In *Advances in Neural Information Processing Systems*, pages 352–361, 2018. 17, 93

[65] Yue Cao, Jiarui Xu, Stephen Lin, Fangyun Wei, and Han Hu. Gcnet: Non-local networks meet squeeze-excitation networks and beyond. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019. 17

[66] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 17

[67] Christian Schuldt, Ivan Laptev, and Barbara Caputo. Recognizing human actions: a local svm approach. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 32–36. IEEE, 2004. 17

[68] Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. *IEEE transactions on pattern analysis and machine intelligence*, 29(12):2247–2253, 2007. 17, 18

[69] Daniel Weinland, Edmond Boyer, and Remi Ronfard. Action recognition from arbitrary views using 3d exemplars. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–7. IEEE, 2007. 17, 18

[70] Marcin Marszalek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2929–2936. IEEE, 2009. 18

[71] Juan Carlos Niebles, Chih-Wei Chen, and Li Fei-Fei. Modeling temporal structure of decomposable motion segments for activity classification. In *European conference on computer vision*, pages 392–405. Springer, 2010. 18, 25

[72] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*, pages 2556–2563. IEEE, 2011. 18, 25, 30, 37, 38, 67, 70, 72, 94, 102, 103, 120, 122

[73] Kishore K Reddy and Mubarak Shah. Recognizing 50 human action categories of web videos. *Machine vision and applications*, 24(5):971–981, 2013. 18, 25

[74] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. 18, 30, 37, 38, 49, 56, 67, 103, 120, 121

[75] Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5842–5850, 2017. 18

[76] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 19, 38, 56, 103, 121

[77] Terrence Chen, Wotao Yin, Xiang Sean Zhou, Dorin Comaniciu, and Thomas S Huang. Total variation models for variable lighting face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 28(9):1519–1524, 2006. 19

[78] Hyunjung Shim, Jiebo Luo, and Tsuhan Chen. A subspace model-based approach to face relighting under unknown lighting and poses. *IEEE Transactions on Image Processing*, 17(8):1331–1341, 2008.

[79] Hu Han, Shiguang Shan, Xilin Chen, and Wen Gao. A comparative study on illumination preprocessing in face recognition. *Pattern Recognition*, 46(6):1691–1699, 2013. 19

[80] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *British Machine Vision Conference*. British Machine Vision Association, 2018. 19, 74

[81] Josue Anaya and Adrian Barbu. Renoir–a dataset for real low-light image noise reduction. *Journal of Visual Communication and Image Representation*, 51:144–154, 2018. 19

[82] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018. 19, 20

[83] Yuen Peng Loh and Chee Seng Chan. Getting to know low-light images with the exclusively dark dataset. *Computer Vision and Image Understanding*, 178:30–42, 2019. 19, 20

[84] Chen Chen, Qifeng Chen, Minh N Do, and Vladlen Koltun. Seeing motion in the dark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3185–3194, 2019. 20

[85] Haiyang Jiang and Yinqiang Zheng. Learning to see moving objects in the dark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7324–7333, 2019. 20

[86] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010. 21

[87] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 22

[88] Ling Huang, Anthony D Joseph, Blaine Nelson, Benjamin IP Rubinstein, and J Doug Tygar. Adversarial machine learning. In *Proceedings of the 4th ACM workshop on Security and artificial intelligence*, pages 43–58, 2011. 22

[89] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015. 22, 24, 92, 96, 105, 124, 125

[90] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016. 22, 24, 115

[91] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017. 22

[92] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR, 2018. 22, 96

[93] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR, 2019. 22, 125

[94] Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11293–11302, 2019. 23

[95] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7167–7176, 2017. 23, 96

[96] Han Zou, Yuxun Zhou, Jianfei Yang, Huihan Liu, Hari Prasanna Das, and Costas J Spanos. Consensus adversarial domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5997–6004, 2019. 23, 96

[97] Jianfei Yang, Han Zou, Yuxun Zhou, Zhaoyang Zeng, and Lihua Xie. Mind the discriminability: Asymmetric adversarial domain adaptation. In *European Conference on Computer Vision*, pages 589–606. Springer, 2020. 23

[98] Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19:513–520, 2006. 23

[99] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010. 23

[100] Arthur Gretton, Dino Sejdinovic, Heiko Strathmann, Sivaraman Balakrishnan, Massimiliano Pontil, Kenji Fukumizu, and Bharath K Sriperumbudur. Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems*, pages 1205–1213. Citeseer, 2012. 23

[101] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *International conference on machine learning*, pages 97–105. PMLR, 2015. 23, 100, 105, 124, 125

[102] Mahsa Baktashmotlagh, Mehrtash T Harandi, Brian C Lovell, and Mathieu Salzmann. Domain adaptation on the statistical manifold. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2481–2488, 2014. 23

[103] Baochen Sun, Jiashi Feng, and Kate Saenko. Return of frustratingly easy domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016. 24

[104] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016. 24, 100

[105] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015. 24, 92

[106] Jing Zhang, Wanqing Li, and Philip Ogunbona. Joint geometrical and statistical alignment for visual domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1859–1867, 2017.

[107] Xinhong Ma, Tianzhu Zhang, and Changsheng Xu. Deep multi-modality adversarial networks for unsupervised domain adaptation. *IEEE Transactions on Multimedia*, 21(9):2419–2431, 2019. 24, 92

[108] Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive faster r-cnn for object detection in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3339–3348, 2018. 24, 92

[109] Xinge Zhu, Jiangmiao Pang, Ceyuan Yang, Jianping Shi, and Dahua Lin. Adapting object detectors via selective cross-domain alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 687–696, 2019.

[110] Shaoyue Song, Zhenjiang Miao, Hongkai Yu, Jianwu Fang, Kang Zheng, Cong Ma, and Song Wang. Deep domain adaptation based multi-spectral salient object detection. *IEEE Transactions on Multimedia*, 2020. 24, 92

[111] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European conference on computer vision (ECCV)*, pages 289–305, 2018. 24

[112] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7364–7373, 2019.

[113] Yun-Chun Chen, Yen-Yu Lin, Ming-Hsuan Yang, and Jia-Bin Huang. Crdoco: Pixel-level domain transfer with cross-domain consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1791–1800, 2019. 24

[114] Arshad Jamal, Vinay P Namboodiri, Dipti Deodhare, and KS Venkatesh. Deep domain adaptation in action space. In *BMVC*, volume 2, page 4, 2018. 24, 101

[115] Min-Hung Chen, Zsolt Kira, Ghassan AlRegib, Jaekwon Yoo, Ruxin Chen, and Jian Zheng. Temporal attentive alignment for large-scale video domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6321–6330, 2019. 24, 25, 92, 103, 104, 125

[116] Boxiao Pan, Zhangjie Cao, Ehsan Adeli, and Juan Carlos Niebles. Adversarial cross-domain action recognition with co-attention. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11815–11822, 2020. 24, 92, 104

[117] Jinwoo Choi, Gaurav Sharma, Samuel Schulter, and Jia-Bin Huang. Shuffle and attend: Video domain adaptation. In *European Conference on Computer Vision*, pages 678–695. Springer, 2020. 24, 92, 104

[118] Dejing Xu, Jun Xiao, Zhou Zhao, Jian Shao, Di Xie, and Yueting Zhuang. Self-supervised spatiotemporal learning via video clip order prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10334–10343, 2019. 24, 92

[119] Waqas Sultani and Imran Saleemi. Human action recognition across datasets by foreground-weighted histogram decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 764–771, 2014. 25, 101, 103

[120] Samitha Herath, Mehrtash Harandi, and Fatih Porikli. Going deeper into action recognition: A survey. *Image and vision computing*, 60:4–21, 2017. 27

[121] Hao Yang, Chunfeng Yuan, Bing Li, Yang Du, Junliang Xing, Weiming Hu, and Stephen J Maybank. Asymmetric 3d convolutional neural networks for action recognition. *Pattern Recognition*, 85:1–12, 2019.

[122] Josep Maria Carmona and Joan Climent. Human action recognition by means of subtensor projections and dense trajectories. *Pattern Recognition*, 81:443–455, 2018.

[123] Hongsong Wang and Liang Wang. Learning content and style: Joint action recognition and person identification from human skeletons. *Pattern Recognition*, 81: 23–35, 2018.

[124] Suraj Prakash Sahoo and Samit Ari. On an algorithm for human action recognition. *Expert Systems with Applications*, 115:524–534, 2019. 27

[125] Somayeh Danafar and Niloofar Gheissari. Action recognition for surveillance applications using optic flow and svm. In *Asian Conference on Computer Vision*, pages 457–466. Springer, 2007. 27

[126] Tao Xiang and Shaogang Gong. Activity based surveillance video content modelling. *Pattern Recognition*, 41(7):2309–2326, 2008.

[127] Jianfei Yang, Han Zou, Hao Jiang, and Lihua Xie. Carefi: Sedentary behavior monitoring system via commodity wifi infrastructures. *IEEE Transactions on Vehicular Technology*, 67(8):7620–7629, 2018. 27

[128] Alessandro Ortis, Giovanni M Farinella, Valeria D'Amico, Luca Addesso, Giovanni Torrisi, and Sebastiano Battiato. Organizing egocentric videos of daily living activities. *Pattern Recognition*, 72:207–218, 2017. 27

[129] Jianfei Yang, Han Zou, Hao Jiang, and Lihua Xie. Device-free occupant activity sensing using wifi-enabled iot devices for smart homes. *IEEE Internet of Things Journal*, 5(5):3991–4002, 2018. 27

[130] Yi Zhu, Zhenzhong Lan, Shawn Newsam, and Alexander G Hauptmann. Hidden two-stream convolutional networks for action recognition. *arXiv preprint arXiv:1704.00389*, 2017. 32, 38

[131] Tsung-Yu Lin, Aruni RoyChowdhury, and Subhransu Maji. Bilinear cnn models for fine-grained visual recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 1449–1457, 2015. 32

[132] Yang Gao, Oscar Beijbom, Ning Zhang, and Trevor Darrell. Compact bilinear pooling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 317–326, 2016. 32, 33

[133] Ninh Pham and Rasmus Pagh. Fast and scalable polynomial kernels via explicit feature maps. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 239–247, 2013. 34

[134] Kai Wang, Xiaoxing Zeng, Jianfei Yang, Debin Meng, Kaipeng Zhang, Xiaojiang Peng, and Yu Qiao. Cascade attention networks for group emotion recognition with face, body and image cues. In *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, pages 640–645, 2018. 36

[135] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017. 37

[136] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010. 38, 56, 103, 124

[137] Gül Varol, Ivan Laptev, and Cordelia Schmid. Long-term temporal convolutions for action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1510–1517, 2017. 38

[138] Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 971–980, 2017. 38

[139] Hongwei Ge, Zehang Yan, Wenhao Yu, and Liang Sun. An attention mechanism based convolutional lstm network for video action recognition. *Multimedia Tools and Applications*, 78(14):20533–20556, 2019. 38

[140] Du Tran, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri. Convnet architecture search for spatiotemporal feature learning. *arXiv preprint arXiv:1708.05038*, 2017. 38

[141] Kun Liu, Wu Liu, Chuang Gan, Mingkui Tan, and Huadong Ma. T-c3d: Temporal convolutional 3d network for real-time action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018. 38

[142] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999. 47

[143] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. *IEEE transactions on pattern analysis and machine intelligence*, 27 (10):1615–1630, 2005. 47

[144] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR 2011*, pages 3169–3176. IEEE, 2011. 47

[145] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 52

[146] Wenguan Wang, Shuyang Zhao, Jianbing Shen, Steven CH Hoi, and Ali Borji. Salient object detection with pyramid attention and salient edges. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1448–1457, 2019. 52

[147] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 52

[148] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 53, 93

[149] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, pages 8026–8037, 2019. 56, 72, 103, 124

[150] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 56

[151] Nieves Crasto, Philippe Weinzaepfel, Karteek Alahari, and Cordelia Schmid. Mars: Motion-augmented rgb stream for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7882–7891, 2019. 60, 61

[152] Li Tao, Xueting Wang, and Toshihiko Yamasaki. Rethinking motion representation: Residual frames with 3d convnets for better action recognition. *arXiv preprint arXiv:2001.05661*, 2020. 60, 61

[153] Changmao Cheng, Chi Zhang, Yichen Wei, and Yu-Gang Jiang. Sparse temporal causal convolution for efficient action modeling. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 592–600, 2019. 60, 61

[154] Xiangyu He, Ke Cheng, Qiang Chen, Qinghao Hu, Peisong Wang, and Jian Cheng. Compact global descriptor for neural networks. *arXiv preprint arXiv:1907.09665*, 2019. 60, 61

[155] Yutaka Hatakeyama, Akimichi Mitsuta, and Kaoru Hirota. Detection algorithm for color dynamic images by multiple surveillance cameras under low luminance conditions based on fuzzy corresponding map. *Applied soft computing*, 8(4): 1344–1353, 2008. 68

[156] T Soumya and Sabu M Thampi. Recolorizing dark regions to enhance night surveillance video. *Multimedia Tools and Applications*, 76(22):24477–24493, 2017. 68

[157] Ana I Maqueda, Antonio Loquercio, Guillermo Gallego, Narciso García, and Davide Scaramuzza. Event-based vision meets deep learning on steering prediction for self-driving cars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5419–5427, 2018. 68

[158] Cheng Dai, Xingang Liu, and Jinfeng Lai. Human action recognition using two-stream attention based lstm networks. *Applied soft computing*, 86:105820, 2020. 68

[159] Amin Ullah, Khan Muhammad, Weiping Ding, Vasile Palade, Ijaz Ul Haq, and Sung Wook Baik. Efficient activity recognition using lightweight cnn and ds-gru network for surveillance applications. *Applied Soft Computing*, 103:107102, 2021. 68

[160] PE Trahanias and AN Venetsanopoulos. Color image enhancement through 3-d histogram equalization. In *Proceedings., 11th IAPR International Conference on Pattern Recognition. Vol. III. Conference C: Image, Speech and Signal Analysis,*, pages 545–548. IEEE, 1992. 73

[161] Xiaojie Guo, Yu Li, and Haibin Ling. Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on Image Processing*, 26(2): 982–993, 2016. 73

[162] Zhenqiang Ying, Ge Li, Yurui Ren, Ronggang Wang, and Wenmin Wang. A new image contrast enhancement algorithm using exposure fusion framework. In *International Conference on Computer Analysis of Images and Patterns*, pages 36–46. Springer, 2017. 73

[163] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, pages 1632–1640, New York, NY, USA, 2019. ACM. ISBN 978-1-4503-6889-6. doi: 10.1145/3343031.3350926. URL `http://doi.acm.org/10.1145/3343031.3350926`. 73

[164] Edwin H Land. The retinex theory of color vision. *Scientific american*, 237(6): 108–129, 1977. 73

[165] Okan Köpüklü, Neslihan Kose, Ahmet Gunduz, and Gerhard Rigoll. Resource efficient 3d convolutional neural networks. *arXiv preprint arXiv:1904.02422*, 2019. 79

[166] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016. 79

[167] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2921–2929, 2016. 85, 109

[168] Qi Cai, Yingwei Pan, Chong-Wah Ngo, Xinmei Tian, Lingyu Duan, and Ting Yao. Exploring object relation in mean teacher for cross-domain detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11457–11466, 2019. 92

[169] Fan Yang, Ke Yan, Shijian Lu, Huizhu Jia, Don Xie, Zongqiao Yu, Xiaowei Guo, Feiyue Huang, and Wen Gao. Part-aware progressive unsupervised domain adaptation for person re-identification. *IEEE Transactions on Multimedia*, 2020. 92

[170] Haifeng Chen, Dongmei Jiang, and Hichem Sahli. Transformer encoder with multi-modal multi-head attention for continuous affect recognition. *IEEE Transactions on Multimedia*, 2020. 93

[171] Limin Wang, Wei Li, Wen Li, and Luc Van Gool. Appearance-and-relation networks for video classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1430–1439, 2018. 93

[172] John A Rice. *Mathematical statistics and data analysis*. Cengage Learning, 2006. 94

[173] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge university press, 2012. 94

[174] Yuecong Xu, Jianfei Yang, Haozhi Cao, Kezhi Mao, Jianxiong Yin, and Simon See. Arid: A new dataset for recognizing action in the dark. *arXiv preprint arXiv:2006.03876*, 2020. 94, 102, 122

[175] Lisheng Fu, Thien Huu Nguyen, Bonan Min, and Ralph Grishman. Domain adaptation for relation extraction with domain adversarial neural network. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–429, 2017. 96

[176] Yanghao Li, Naiyan Wang, Jiaying Liu, and Xiaodi Hou. Demystifying neural style transfer. *arXiv preprint arXiv:1701.01036*, 2017. 99

[177] Tiantian Xu, Fan Zhu, Edward K Wong, and Yi Fang. Dual many-to-one-encoder-based transfer learning for cross-dataset human action recognition. *Image and Vision Computing*, 55:127–137, 2016. 101

[178] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018. 105, 125

[179] Ruijia Xu, Guanbin Li, Jihan Yang, and Liang Lin. Larger norm more transferable: An adaptive feature norm approach for unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1426–1435, 2019. 108

[180] Zhangjie Cao, Lijia Ma, Mingsheng Long, and Jianmin Wang. Partial adversarial domain adaptation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 135–150, 2018. 116, 125

[181] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010. 124

[182] Zhangjie Cao, Kaichao You, Mingsheng Long, Jianmin Wang, and Qiang Yang. Learning to transfer examples for partial domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2985–2994, 2019. 125

[183] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 129