Recognizing Human-Object Interactions in Videos

MUNA IBRAHIM M ALMUSHYTI

Recognizing Human-Object Interactions in Videos

Muna Almushyti

A Thesis presented for the degree of
Doctor of Philosophy

Department of Computer Science
Durham University
United Kingdom
May 2023

Abstract

Understanding human actions that involve interacting with objects is very important due to the wide range of real-world applications, such as security surveillance and healthcare. In this thesis, three different approaches are presented for addressing the problem of human-object interactions (HOIs) recognition in videos.

Firstly, we propose a hierarchical framework for analyzing human-object interactions in a video sequence. The framework comprises Long Short-Term Memory (LSTM) networks that capture human motion and temporal object information independently. These pieces of information are then combined through a bilinear layer and fed into a global deep LSTM to learn high-level information about HOIs. To concentrate on the key components of human and object temporal information, the proposed approach incorporates an attention mechanism into LSTMs.

Secondly, we aim to achieve a holistic understanding of human-object interactions (HOIs) by exploiting both their local and global contexts through knowledge distillation. The local context graphs are used to learn the relationship between humans and objects at the frame level by capturing their co-occurrence at a specific time step. On the other hand, the global relation graph is constructed based on the video-level of human and object interactions, identifying their long-term relations throughout a video sequence. We investigate how knowledge from these context graphs can be distilled to their counterparts to improve HOI recognition.

Lastly, we propose the Spatio-Temporal Interaction Transformer-based (STIT) network to reason about spatio-temporal changes of humans and objects. Specifically, the spatial transformers learn the local context of humans and objects at specific frame times. The temporal transformer then learns the relations at a higher level between spatial context representations at different time steps, capturing long-term dependencies across frames. We further investigate multiple hierarchy designs for learning human interactions.

The effectiveness of each of the proposed methods mentioned above is evaluated using various video action datasets that include human-object interactions, such as Charades [1], CAD-120 [2], and Something-Something V1 [3].

Dedicated to

The sake of Allah, all praise is due to him.

My tender mother, Aljouharah Almushyti.

My empathetic father, Ibrahim Almushyti.

My affectionate brother, Mohammad Almushyti.

My supportive supervisor, Dr. Frederick Li.

Declaration

The work in this thesis is based on research carried out at the Department of Computer Science, Durham University, United Kingdom. No part of this thesis has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Acknowledgements

First and foremost, I want to express my gratitude to Almighty God for all of His favors, including helping me to complete this doctoral thesis. My deepest gratitude goes to my main supervisor, Frederick Li, for his excellent encouragement and support throughout my PhD journey, especially during the difficult times. He made himself available for meetings all the time, and I learned a lot from him.

I owe my parents a huge debt of gratitude for helping me complete my higher education abroad. I am especially grateful to my mother for believing in me and staying emotionally close, even though I lived thousands of miles away for ten years. I would also like to express my gratitude to my sisters (Rehab, Mariah, and Randa) and younger siblings (Marwan, Tammam, and Ghassan) for the innocent conversations, funny jokes, and enjoyable adventure stories that brought me happiness. Furthermore, I want to acknowledge my brother Mohammad for his unwavering support and for tolerating my mood swings during my pursuit of higher education.

Many thanks to all of my colleagues, including Olanrewaju Tahir Aduragba, for generously sharing their knowledge and offering help whenever I needed it. I would like to extend a special thanks to my dear friend Seyma Yucer and her husband Furkan Tektas, who have been there for me whenever I needed them during my studies in Durham. My sincere gratitude also goes to my other close friends, Latifah Abduh, Laila Alrajhi, Aishah Alsehaim, and Fatimah Alghamdi, who have actively supported me and made my PhD experience unforgettable.

Finally, I want to convey my appreciation to Qassim University for offering me a complete scholarship that facilitated the funding of my doctoral studies. Additionally, I would like to extend my thanks to the Department of Computer Science at Durham University for accepting me as a student and providing me with access to their exceptional facilities.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Videos provide valuable information and cues about human behavior that can aid in recognizing specific human activities and actions [19]. These actions involve various temporally structured movements, such as running and jumping [20]. Most activities are carried out by humans who interact with objects or other people. This research focuses on analyzing the interactions between humans and objects, which are referred to as human-object interactions (HOIs). HOI recognition and action classification are terminologies used to describe similar tasks in videos, but there is a slight distinction between them. In action recognition and classification tasks, a clip's action is assigned based on a predefined list of actions, taking into account all the video's features [19]. In contrast, the HOI recognition task uses information about humans and objects, in addition to the video's features, to categorize an action.

HOI recognition generally involves two phases: detection and recognition. In the detection phase, humans and objects involved in the action are localized. In the recognition phase, information about humans and objects including their relationship are used [21]. Thus, to improve the rate of HOI recognition, researchers have used various features such as the appearance of the human or the object [22–24],

the human pose [23–26], the human gaze [25] and the relative location of the object with respect to the human [22], geometric and semantic information [27]. In this research, we use pre-trained off-the-shelf detectors to detect humans and objects and focus on using visual features for recognizing HOIs in videos. It is important to note that the HOI recognition task differs from the action detection task in that the latter's models are trained not only to identify the action but also to predict the temporal and spatial position of the action instance, such as a human [28].

To solve the problem of HOI recognition, various efforts have been made, which can be classified into two main categories: hand-crafted approaches and deep learning models. Deep learning techniques have shown more promising results than traditional methods in various computer vision tasks. Thus, in this research, we will employ state-of-the-art deep learning methods to tackle the problem of HOI recognition.

1.1 Motivation

The understanding of human behavior in videos is crucial for a broad range of real-world applications, including human-robot collaboration, human-computer interaction (HCI), video surveillance, autonomous driving (e.g., self-driving cars), sports, and retail stores.

Human-robot collaboration is utilized in a range of fields, including those that involve both human and robotic labor, such as the industrial, medical, and rehabilitation sectors [29]. The ability of robots to understand and interpret human actions and interactions with objects leads to more effective cooperation between humans and robots in a shared environment. Thus, to properly perform tasks, a thorough understanding of human behavior is necessary. Fig. 1.1 shows assembly and machine tending tasks in one of the motorbike manufacturing facilities in India, Bajaj Auto, where utilizing robots with the help of humans facilitates a faster production line [4].

Another application of using human-object interactions is in the field of human-computer interaction (HCI), which benefits from accurate modeling of human actions

Figure 1.1: Tasks including assembly and machine tending are performed with the help of robots at Bajaj Auto manufacturer, from [4].

Figure 1.2: A robot's response to a human action, from [5].

Figure 1.3: AmazonGo store, from [6].

in various applications such as augmented reality, interactive games, and social robots that respond based on understanding human actions. An example of a social robot interaction is shown in Fig. 1.2.

Furthermore, in crowded public areas like airports, hospitals, and shopping malls, maintaining security and safety is crucial, and this is achieved by closely monitoring all activities using a video surveillance system equipped with cameras. It is especially vital to identify interactions with objects and recognize various activities to ensure a prompt response if necessary [30]. Also, in healthcare settings, it is crucial to be able to recognize patient actions, especially those of the elderly, and analyze anomalous activity like falls in order to intervene promptly.

In addition, the recognition of HOIs plays a crucial role in the functioning of self-driving vehicle systems and other applications related to autonomous driving. In other words, HOI involves understanding the driving scene, including human behaviors (such as pedestrians) and interactions with other objects (automobiles, bicycles, other vehicles), and incorporating this information in making appropriate decisions, such as 'slow down' or 'stop,' in response to such interactions [31].

Action recognition applications are also becoming increasingly common in the world of sports, with uses that include analyzing and recognizing a player's actions. These actions often involve interactions with objects such as a ball and other players [32].

Furthermore, HOI recognition is important in the retail industry. For instance, at AmazonGo grocery stores, as depicted in Fig. 1.3, the need for a physical check-out process has been eliminated. Customers can conveniently enter the store by scanning the AmazonGo application at the entrance without any additional steps. As customers obtain items, their virtual cart is automatically updated, and they can simply leave the store when they are finished shopping [33]. Currently, there are 23 Amazon Go stores in the United States [34]. In such a system, HOI recognition is imperative.

Therefore, developing an accurate action recognition model is crucial for achieving the desired impacts and outcomes of all the aforementioned applications. The model should incorporate knowledge of the context of human activities and sur-

rounding objects, as well as temporal information presented in videos. For this reason, this thesis aims to explore how human-object interaction modeling in videos can improve action recognition models, by taking into account the spatial and temporal contexts of humans and objects.

1.2 Problem Statement

This thesis takes into account human-object interactions in an effort to solve the problem of human action recognition. This study focuses on single-person interactions with various objects to perform specific actions, such as brushing teeth, opening a refrigerator, and making cereal.

While the majority of prior research focuses on recognizing HOIs in still images [21,24,25,35], these approaches neglect temporal information. As a result, they are unable to identify time-related interactions, limiting their practical applicability [36]. Moreover, although a significant portion of deep learning models has been effectively employed to recognize actions in videos, including 2/3D Convolutional Neural Networks (CNN) and Long Short-Term Memory(LSTM) networks, the majority of these networks tend to treat video frames equally, without prioritizing key elements of the actions, such as human-object interactions. Moreover, how to model human-object relations in a way that boosts the performance of action recognition models while taking into account the natural way humans behave when interacting with objects has not been fully explored. As a result, this study's objective is to explore and introduce networks that incorporate long-range spatiotemporal relations between humans and objects, while also considering the hierarchical representation and contextual views of the interactions.

1.3 Research Challenges

The primary objective of studies in human action recognition in videos is to develop a model that can effectively understand human intentions and actions involving Human-Object Interactions (HOIs). The complexity of human behaviors can re-

sult in the erroneous categorization of actions and interactions. Misclassification of HOIs can arise from various challenges. Individuals can perform the same interaction differently, meaning they may demonstrate different poses while performing the same action. For example, one person may drink tea while standing, while another may drink tea while sitting. Another challenge to recognizing actions is the similarities between HOIs. For instance, different objects, such as drinking bottles and spray bottles, can have very similar appearances [37] but are used in different actions. Moreover, different interactions may involve similar motion patterns, which can confuse the recognition model and lead to incorrect results. This refers to the challenge of inter-class similarity [38]. Moreover, Various background settings, including cluttered environments, present difficulties in the recognition of Human-Object Interactions (HOIs) in videos [38].

Furthermore, other challenges include certain objects affording different interactions, like when a human cleans an oven or takes food from it, where the oven affords to different interactions such as opening, cleaning and closing. In addition, a variety types of objects afforded to same action (e.g., refrigerators and doors can be involved in the same interactions including open and close). Moreover, the presence of various objects in the scene at the same time could affect model learning.

To sum up, recognizing human-object interactions (HOIs) with the presence of the above-mentioned issues is difficult, and it is essential to identify and learn discriminative features including important contextual information such as spatial relationships and temporal dynamics,that facilitate understanding of such actions.

1.4 Thesis Scope

Although there has been a lot of research on image-based human-object interactions (HOIs) [21, 24, 39], videos have received less attention due to the presence of more challenges such as spatio-temporal changes. This thesis addresses this gap by focusing on human-object interactions in videos, utilizing deep learning models to understand high-level human-object interactions. While there is a connection between this research and the field of action recognition, this thesis concentrates on

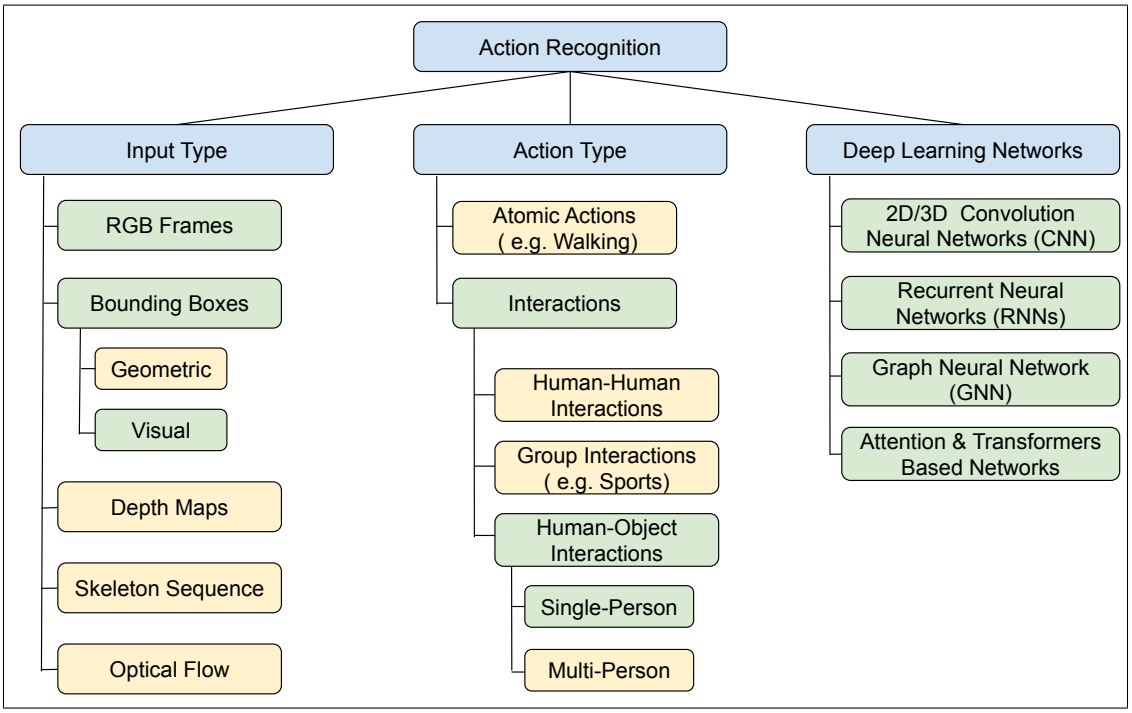


Figure 1.4: Video-based action recognition employs various inputs, actions, and deep learning models. The only inputs and models utilised by the proposed approaches in this thesis are denoted by the green rounded rectangles.

human-object interactions rather than just the overall features of the video, which are considered critical components for recognizing actions. Thus, in order to evaluate the effectiveness of the proposed methods in this thesis, we utilize action datasets that contain both human(s) and objects in the video for HOI recognition, along with annotations related to HOI. In fact, research on the action recognition problem from videos is broad and covers different aspects depending on the input data, types of actions, and network architecture [40], as shown in Fig. 1.4. The datasets utilized in this thesis generally consist of videos of a single person interacting with objects, with no human-human or group interactions present. Although other studies use various forms of information for identifying human-object interactions, such as skeleton data, this research focuses solely on the appearance features (e.g. visual information) of bounding boxes of humans and objects in addition to video frame features. Additionally, this thesis mainly considers supervised learning, where all the labels of the videos in the datasets are available and used to train the proposed models in this research.

1.5 Thesis Contributions

This thesis presents different approaches to learn human-object interaction relations in order to recognize human actions. The following is a summary of the thesis' significant contributions:

- Proposing an LSTM-based framework with attention mechanism for recognising human-object interactions (HOIs) in videos. HOIs is modeled by using hierarchical LSTMs to capture the dynamics of Human and objects in a video sequence. (**Chapter** 3)

- Introducing a novel teacher-student network based on graph neural networks to learn spatial and temporal interrelations between humans and objects in a video from two different contextual views. This approach enables the capture of long-term and non-local dependencies between humans and objects across video frames. (**Chapter** 4)

- Exploring how knowledge from the teacher contextual view of interactions can be obtained, and distilling it to the student view of interactions to improve action recognition performance. (**Chapter** 4)

- Presenting a novel transformer-based framework to learn spatio-temporal interrelations between humans and objects in videos by considering the hierarchical representation of HOIs. (**Chapter** 5)

- Investigating the effects of various hierarchical structures on HOI learning. (**Chapter** 5)

1.6 Publications

The following are the peer-reviewed publications in which the work from this thesis has been published:

- **Conference paper** (Contributing to Chapter 3):

Almushyti, M. and Li, F.W., Recognising human-object interactions using attention-based LSTMs. In Computer Graphics and Visual Computing (CGVC), September 2019, (pp. 135-139). https://diglib.eg.org/handle/10.2312/cgvc20191269.

- **Journal paper** (Contributing to Chapter 4):

  Almushyti, M. and Li, F.W., Distillation of human–object interaction contexts for action recognition, The Journal of computer animation and visual world, August 2022, 33(5): e2107. https://doi.org/10.1002/cav.2107.

- **Conference paper** (Contributing to Chapter 5):

  Almushyti, M. and Li, F.W., STIT: Spatio-Temporal Interaction Transformers for Human-Object Interaction Recognition in Videos, In the 26th International Conference on Pattern Recognition(ICPR), 2022. https://ieeexplore.ieee.org/document/9956030.

1.7 Thesis Structure

This thesis is organized into six chapters. Chapter 2 covers three sections of the literature review. The first section provides background information on the fundamentals of the methods used in this thesis, as well as a description of the datasets utilized in this research. Recent models, using deep learning networks, for action recognition and modelling human-object interactions are also discussed in Sections 2.2 and **??**, respectively.

Chapter 3 covers the use of LSTM networks with the help of an attention mechanism to learn high-level human-object interactions.

Chapter 4 presents a teacher-student-based network where the teacher aids the student's learning through knowledge distillation. Graph attention networks (GAT) are used to construct the views of both the teacher and student networks. Each network captures a different contextual view of human-object interactions, considering spatial or temporal relations. Comprehensive experiments are conducted to determine which view can serve as the teacher network.

Taking inspiration from the recent growth of transformer networks in the computer vision area, Chapter 5 explores the possibility of employing two-level transformers to model spatio-temporal relations between humans and objects to identify their interactions. Various network designs have been investigated in this chapter.

Chapter 6 summarizes the limitations of the methods proposed in this thesis and outlines potential future research directions.

-

CHAPTER 2

Literature Review

This chapter provides background information on the techniques and methodologies presented in the contribution chapters (Chapters 3 - 5). It also summarizes the datasets used in this thesis and reviews state-of-the-art models for action recognition. Additionally, the chapter includes a review of higher-level modeling networks, such as those using human-object interactions, that are utilized for action recognition.

2.1 Background

This section covers the basic background details for the methods that are presented in the next chapters of this thesis (Chapters 3 - 5).

2.1.1 Video Representation and Feature Extraction

There are several modalities available in videos that can serve as inputs for action recognition models, such as RGB frames [41], optical flow [12], and skeleton data [42]. Since visual information is the main focus of this research, we are primarily discussing models that utilize RGB data. Two main categories of deep learning-based models, including CNNs and transformers, are used to extract visual information

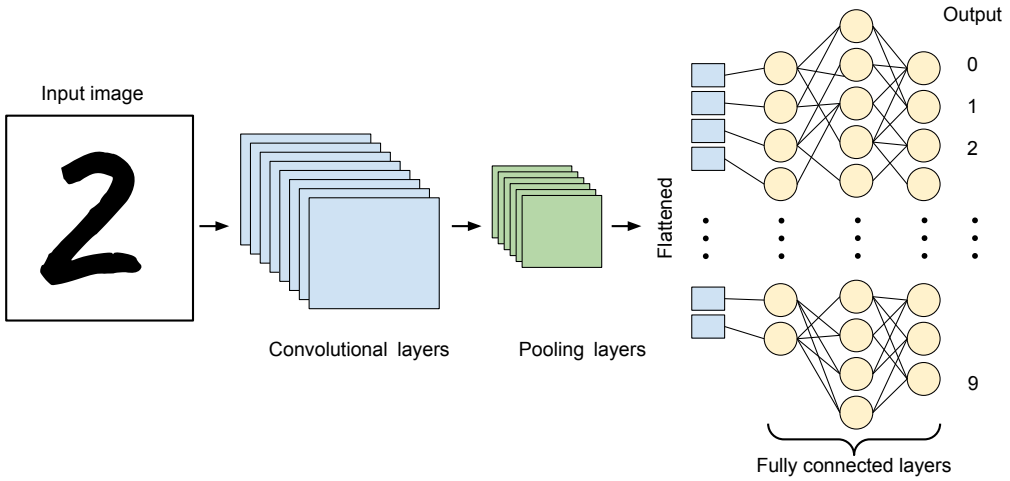


Figure 2.1: Simple Convolution Neural Network, from [7].

from video frames.

Convolution Neural Networks (CNN/ConvNet)

CNN is a form of artificial neural network architecture that consists of three main blocks: convolution layers, pooling layers, and fully connected (FC) layers. Typically, this network is initially built with N convolution layers followed by a pooling layer, and finally, one or more FC layers are stacked at the end of the network [7]. Feature extraction is performed in the convolution layers using kernels. These kernels typically take the form of small-dimensional matrices of weights and pass through the input image (2D kernels) to perform element-wise multiplication with the input at each position. The results are then summed up to obtain a single value at the current spatial position. The output of the various kernels generates the feature maps [43].

Commonly, the features of the convolution layer are extracted and sent to non-linear activation functions such as the Rectified Linear Unit (ReLU). The output is then passed to a pooling layer, which performs a down-sampling operation to reduce both the dimensionality of feature maps and the number of parameters to be learned. Examples of pooling operations include max and average pooling [43]. The feature maps of the last pooling or convolution layer are flattened as one-dimensional

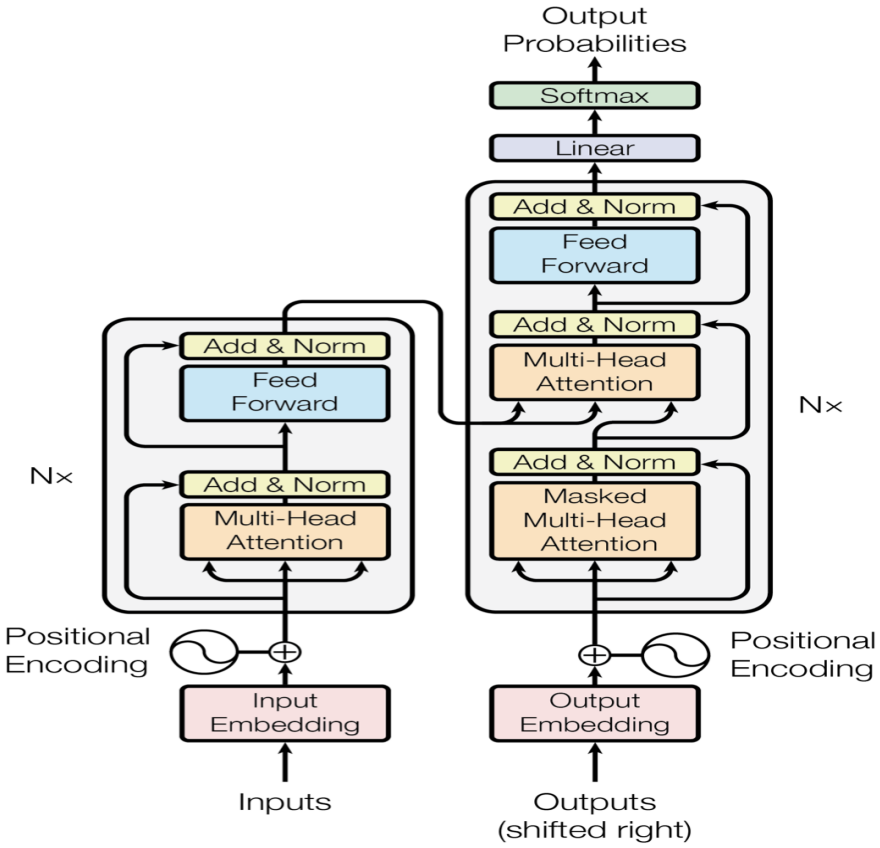


Figure 2.2: Transformer architecture, from [8].

maps and are connected to one or two FC layers, where the final FC layer performs an activation function that depends on the learning task. For instance, to classify multi-class images, Softmax activation can be used [7]. Figure 2.1 shows a simple CNN architecture for MNIST [44] classification. There are different architectures for convolution networks, such as AlexNet [45], VGG [46], Inception [47], and ResNet [48]. The CNN kernels can be expanded to support 3D dimensions, which are frequently used in videos [49]. Similarly, the I3D model [50] has been introduced by inflating pretrained 2D convolution kernels to 3D for extracting space-time features from video clips. In this research, we use the I3D model as a backbone to extract features from videos.

Transformers

Transformers are deep learning models that were first introduced in [8] for sequence-to-sequence modeling, such as for machine translation tasks. In this case, trans-

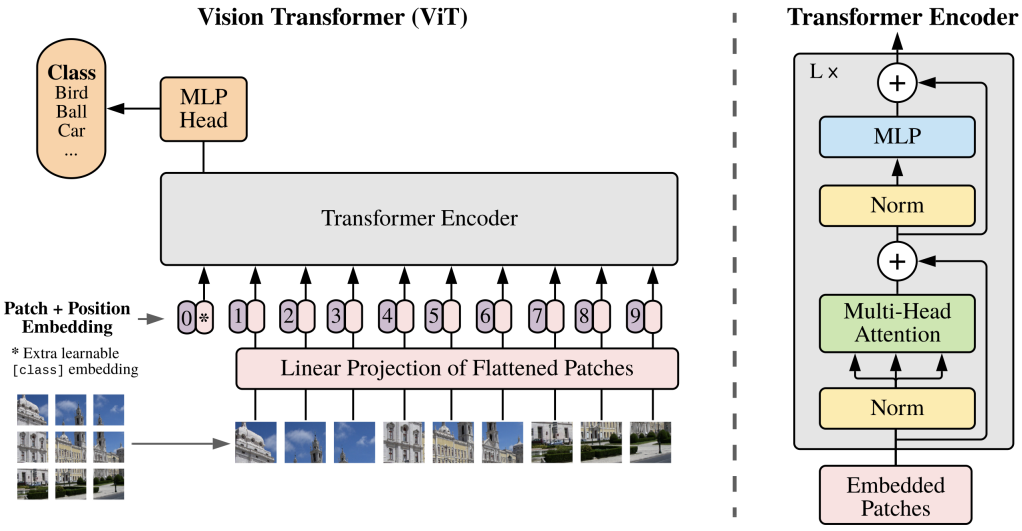


Figure 2.3: Vision transformer, from [9].

formers consist of two parts: an encoder and a decoder. Figure 2.2 presents the transformer architecture. Having both an encoder and a decoder in a network is not necessary and depends on the application. For instance, if the purpose is to obtain sequence representation for a classification task, utilizing merely the encoder can be sufficient. The transformer encoder mainly comprises multiple layers with multi-head self-attentions (MHA) in each layer and feed-forward layers (MLPs). Both layer normalization with residual connections are applied prior to and following the MLPs. The inputs (e.g., words in a sentence or image patches) are embedded with their positional information and are fed to the transformer encoder. In self-attention, the input is transformed into three components - queries (Q), keys (K), and values (V) - through linear transformations. The attention function, which is known as Scaled Dot-Product Attention [8], is written as:

$$Attention(Q, K, V) = Softmax(\frac{Q.K^T}{\sqrt{d_k}})\ V \tag{2.1}$$

where d_k is the key dimension and the multi head attention (MHA) can be computed as follows:

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_m)W^h \tag{2.2}$$

where $head_i = Attention(QW_i^Q, KW_i^k, VW_i^V)$. W^h and m represent the learnable matrix and the number of heads, respectively.

For computer vision tasks, the ViT transformer [9] has recently been used by dividing an image into patches, as shown in Fig. 2.3. In this research, we adapt the ViT transformer encoder as a relational model between humans and objects. More details are provided in Chapter 5.

2.1.2 Recurrent Neural Networks (RNN)

RNN is a powerful network architecture for sequential data of any length, such as sentences and sequences of images (e.g., frames) in videos. In RNNs, all inputs are related to one another in a way that the same task is performed on all elements of a sequence. As a result, parameters (e.g., weights) are shared throughout all time steps. Each state has the ability to store the knowledge of previous input states. Consequently, a hidden state could be thought of as the network's memory. The hidden state h at time t is defined as follows:

$$\boldsymbol{h}_t = f_H(W_{IH}x_t + W_{HH}h_{t-1} + b_h) \tag{2.3}$$

where f_H is a non-linear activation function, such as a tanh or sigmoid, and x_t is the current input. W_{IH} is the weight matrix between the input and the hidden layer. W_{HH} is the weight matrix between the current state and the previous state. b_h is the bias of the hidden layer, and h_{t-1} is the previous hidden state at time step $t-1$. Finally, y_t in the output layer can be calculated as follows:

$$\boldsymbol{y}_t = f_O(W_{HO}h_t + b_o) \tag{2.4}$$

where f_O is the activation function, such as softmax, and W_{HO} is the weight matrix between the hidden layer and the output layer. The weights W are shared throughout time steps [51].

Several RNN types can be used for various computer vision tasks, such as one-to-many RNNs for image captioning [52], many-to-one RNNs for action recognition [53, 54], and many-to-many RNNs for tasks like video descriptions [55].

Practically, issues such as vanishing and exploding gradients arise during the training of RNNs. In particular, the vanishing gradient problem occurs during backpropagation, where the network gradients become small, making it difficult to update the network weights and prolonging the time required to obtain the final results. Consequently, standard RNNs are unable to process very long sequences [56]. Several solutions have been proposed to address the vanishing gradient problem and the long-term dependency of RNNs, including Gated Recurrent Units (GRUs) [57] and Long Short-Term Memory (LSTM) [58]. As LSTM is used in Chapter 3, it will be described below.

Long short-term memory (LSTM)

The main components of LSTMs include gate networks and the cell state. Thus, two vectors are involved at each time step: the hidden vector and the cell state vector, which acts as a "memory" and carries all relevant information throughout the sequence. Moreover, each LSTM cell has three different gates, namely the input (i_t), forget (f_t), and output (o_t) gates, which are mathematically represented as follows:

$$\boldsymbol{f}_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{2.5}$$

$$\boldsymbol{i}_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{2.6}$$

$$\boldsymbol{o}_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{2.7}$$

The cell state (c_t) and hidden state (h_t) at time t can be expressed as:

$$\boldsymbol{c}_t = f_t * c_{t-1} + i_t * tanh(W_c[h_{t-1}, x_t] + b_c) \tag{2.8}$$

$$\boldsymbol{h}_t = o_t * tanh(c_t) \tag{2.9}$$

Where σ and $tanh$ are sigmoid and tangent hyperbolic activation functions, respectively, and W_f, W_i, W_o, and W_c are learnable weights. The forget gate f_t

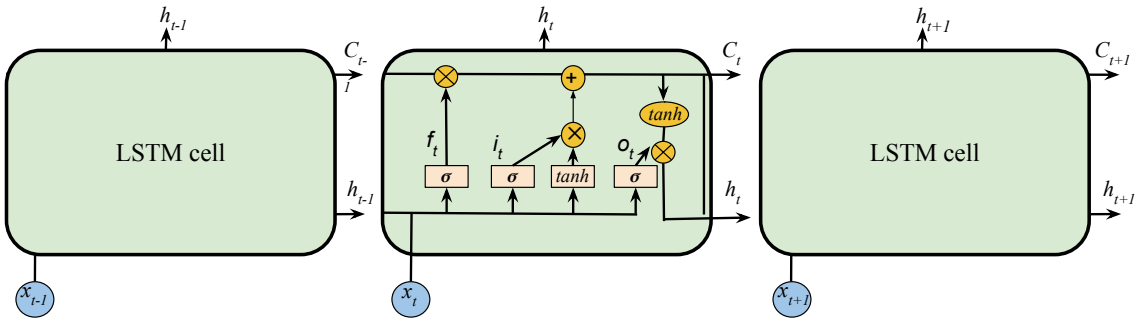


Figure 2.4: Long short-term memory (LSTM) architecture. f_t,i_t and o_t indicate forget, input and output gates, respectively.

determines what information should be ignored or kept. The input gate i_t selects what information should be added to the cell state based on the previous hidden state h_{t-1} and the current data x_t. The last gate is the output gate o_t, which controls what information from the memory cell (e.g., cell state) should be sent as the current hidden state's output h_t. Figure 2.4 shows an LSTM cell architecture. Additionally, there is an extension of LSTMs known as a bidirectional LSTM (BLSTM) network, where the input is modeled by the LSTM networks in both forward and reverse directions [59]. In this thesis, LSTMs are utilized to capture temporal changes of humans and objects in videos. More details are presented in Chapter 3.

2.1.3 Graph Neural Networks (GNNs)

Graphs are a type of data structure that are represented by nodes and edges that connect them [60]. Neural networks applied to graph data are called Graph Neural Networks (GNNs). There are many applications for GNNs, including scene graph generation [61], action recognition [62], text classification [63], recommender systems [64], and health record modeling [65]. Moreover, different forms of GNNs have been developed, such as the Graph Convolution Network (GCN) [63] and the Graph Attention Network (GAT) [66]. Since Chapter 4 proposes a network based on GAT, GAT is explained here in more detail.

GAT

Graph attention network (GAT) implies using attention mechanism in order to update node states in a graph. The learnable attention score between two nodes x and y, which reveals the importance of node y to node x, is computed as:

$$\boldsymbol{\alpha}_{xy} = \frac{\exp\left(\text{LeakyReLU}(a^T[Wh_x||Wh_y])\right)}{\sum_{k\in N_x}\exp\left(\text{LeakyReLU}(a^T[Wh_x||Wh_k])\right)} \tag{2.10}$$

where N_x represents node x's neighbors, W is a trainable weight matrix related to the linear transformation of graph nodes, and a^T is a trainable weight of a multilayer perceptron (MLP) with one layer. Finally, in order to obtain the final hidden state of node x, it can be written as:

$$h_x = \sigma(\sum_{y\in N_x}\alpha_{xy}Wh_y) \tag{2.11}$$

where $\sigma$is a non-linear activation function, such as LeakyReLU [66]. The authors of GAT [66] used multi-head attention to stabilize the learning process, where M independent heads are applied to calculate the hidden state of graph nodes, and afterward concatenate or average these features to obtain the final hidden state of each node. In this thesis, the GAT network is utilized to learn the relations between humans and objects nodes from various contextual views, as proposed in Chapter 4.

2.1.4 Transfer Learning

The process of transfer learning involves applying knowledge from previously learned tasks to a new or related task to increase model performance, especially when the source model is trained on large datasets. Pre-trained models' knowledge can be transferred in various ways, including reusing their weights as initial weights to train a model on different datasets and for related tasks (known as fine-tuning). Another way is fixing the pre-trained model weights and using the model as a feature extractor without its final layer for different tasks, such as classification. Since the earliest layers of the model learn coarse features, whereas the later deep layers capture more fine-grained information, only the later layers need to be fine-tuned, and the first

layers can be fixed [67].

Examples of models commonly used for transfer learning are 2D ResNet-50 [48] and VGG-16 [46], which are pre-trained on the ImageNet dataset [68] for image classification tasks. In the experiments of the methods proposed in this thesis, we initialize the I3D backbone with pre-trained parameters from the Kinetics-400 dataset [69].

2.1.5 Knowledge Distillation (KD)

Distilling knowledge has been proposed as a way to transfer knowledge from an ensemble of classifiers or a large network into a small network [70]. This approach involves compressing complex networks without losing their performance [71], which is accomplished by minimizing the loss between the small network's predictions (student) and the large network's softened labels (teacher). Recently, KD has been extended and combined with privileged information [72], which is only available during training, to form a generalized distillation approach [73]. In the action recognition task, knowledge is distilled between multiple modalities (e.g., skeleton, RGB, optical flow), which can be considered privileged information because not all of them are available during inference [74–77]. KD is also used in other directions, such as defending against adversarial attacks [78] and classifying unlabeled data by unifying diverse classifiers [79]. To improve segmentation accuracy, KD is used in semantic segmentation, for example, by distilling intra-class feature variation or inter-class distance from the teacher network to the student [80, 81]. Additionally, KD is used to enhance object detectors by selecting valuable areas (e.g., foreground) to distill [82–84].

In this research, Knowledge Distillation (KD) is used to transfer knowledge between two contextual views, including global and local views. More details are presented in Chapter 4.

Figure 2.5: Twenty actions from UCF101 dataset [10].

2.1.6 Datasets and Loss Functions

There are numerous datasets that can be utilized for action recognition tasks, including UCF101 [10], HMDB-51 [85], NTU RGB+D [86], Kinetics [69], Something-Something v1 (SSv1) [3], Sports-1M [87], Charades [1], and CAD-120 [2]. However, each dataset is utilized for a specific field of study. For instance, the NTU RGB+D dataset [86] is used for action recognition models that concentrate on exploiting human skeleton sequences. Additionally, Sports-1M [87] is more closely associated with sporting activities, which are not the main focus of this study. Therefore, we select datasets that have been used for high-level modeling of actions involving humans and objects, in order to make fair comparisons with earlier works. In this thesis, four different datasets are selected that contain videos with different actions, most of which involve interacting with objects. The datasets and loss functions used to train the proposed models in this research are explained below.

- The UCF101 dataset [10] includes a variety of human actions, such as playing tennis and applying eye makeup. Since the scope of this research is recognizing

HOIs, 20 classes related to HOIs from the UCF101 dataset (split 1) are used. Figure 2.5 shows these 20 actions. The cost function used during training the proposed network in Chapter 3 is Cross-Entropy, which for a training example can be formulated as:

$$H(y_i, y_i') = -\sum_{i=1}^{c} y_i log y_i' \tag{2.12}$$

where y_i and y_i' are the actual and predicted probability distributions for action classes, respectively. c indicates the number of action classes used in training (e.g., 20). The loss over the entire dataset is formulated as:

$$loss = \frac{1}{m}\sum_{i=1}^{m} H(y_i, y_i') \tag{2.13}$$

where m is the number of training examples. The goal during training is to minimize this loss by gradient descent algorithm [88]. For evaluation, we use the accuracy metric to measure the performance of the proposed model, which can be calculated by dividing the number of correctly recognized action videos by the total number of videos in the dataset. This dataset is used in Chapter 3.

- The Charades dataset [1] consists of 9,848 multi-label videos showcasing indoor daily activities that involve humans interacting with various types of objects. The number of videos in the training phase is about 8K, and 1.8K for validation. In total, there are 157 action classes. Since the Charades dataset is a multi-label video dataset, the proposed model is trained using binary Cross-Entropy loss, and the final results are reported using mean average precision (mAP). This dataset is used in Chapters 4 and 5.

- The CAD-120 dataset [2] contains 120 videos featuring 10 different daily life interactions performed by 4 different subjects, as shown in Fig. 2.6. Depth images, bounding boxes, and skeleton information are also available, but only RGB images are used. Each video in CAD-120 [2] has only one high-level activity label. Similar to the UFC101 dataset [10] and as in [89], the proposed models are trained and evaluated using accuracy and cross-entropy loss,

(a) Picking objects (b) Arranging objects (c) Stacking objects (d) Making cereal

(e) Taking medicine (f) Taking food (g) Cleaning objects (h) Unstacking objects

(i) Microwaving food (j) Having meal

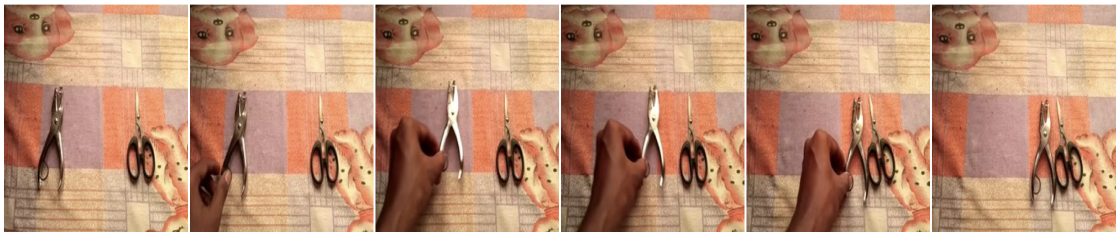
Figure 2.6: Examples of human-object interactions in CAD-120 dataset [2].

respectively. This dataset is used in Chapters 3, 4 and 5.

- Something-Something v1 (SSv1) [3] contains 174 classes and 108,499 videos, each with a single label. Unlike Charades [1], most videos in SSv1 have a clear background, and actions involve hands interacting with objects rather than the whole human body. Since it is a multi-class dataset, we use the standard procedure for training (e.g. Cross-Entropy loss) and evaluating the proposed model in Chapter 5. Two examples of videos can be seen in Fig. 2.7.

Moreover, in order to train the proposed model in Chapter 4 using knowledge distillation, we have also incorporated a distillation loss between the student and teacher networks. Further details can be found in Chapter 4.

(a) Putting a white remote into a cardboard box



(b) Moving puncher closer to scissor

Figure 2.7: Interaction examples from Something-Something v1 (SSv1) dataset [3].

2.2 Human-object interactions (HOIs) in Videos

Recognizing human-object interactions requires an understanding of human actions and how humans interact with objects in a video. Before the rise of deep learning, a range of crafted methods were utilized to identify actions and interactions. These included strategies such as space-time volume techniques [90], space-time interest points approaches [91], and methods centered around trajectories [92,93]. For additional information, readers are directed to explore this survey [94].

In this research, our focus lies in the utilization of deep learning models. We employ action recognition models as global descriptors and to extract human and object features, thereby facilitating the modeling of their relations to recognize such interactions. Accordingly, we have divided this section into three subsections that focus on the primary relevant literature concerning the human-object interaction approaches proposed in this thesis: temporal modelling, contextual understanding, and attention, along with the long-range dependencies of HOIs. It's important to note that certain works may fall into multiple categories simultaneously. However, we have assigned them to the most fitting category and have refrained from redundant repetitions.

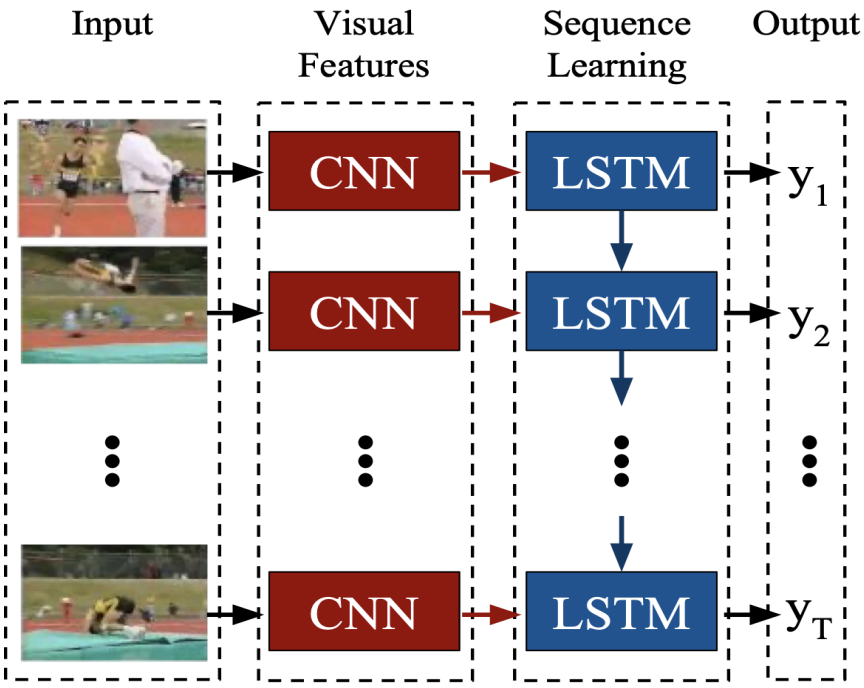


Figure 2.8: Long-term Recurrent Convolutional Networks (LRCNs), from [11].

2.2.1 Temporal Modelling

For capturing temporal changes in sequences of frames, prior works have used either 2D or 3D CNN networks for recognizing actions in videos. Simple models for action recognition can be created by first extracting spatial information from frames using 2D CNNs and then using various pooling procedures, such as max pooling [95]. In fact, it is necessary to model the temporal relations between the CNN features of various frames because a single 2D CNN cannot handle the temporal information in videos. Therefore, in order to focus on temporal modeling, the Temporal Segment Networks (TSN) [41] are presented. As input, each video is divided into K segments, each containing an equal number of frames. Then, a frame or 'snippet' is chosen at random from each segment, and a 2D CNN is used as the basis for feature extraction, producing the segment's class scores. Finally, various aggregation algorithms such as average pooling are used to combine the class scores from all video segments to produce the video-level predictions. TSN is often trained with different modes such as RGB and optical flow and fused via late-fusion. Although TSN can capture long-term information, it cannot capture the temporal ordering of frames or more complex

temporal correlations. Thus, improved variants of TSN are proposed, including TRN [96] and TSM [97] networks, where relation and shift modules are utilized, respectively, in order to extract and learn more information along the temporal dimension.

Moreover, 3D convolutional neural networks have been proposed, where an extra time dimension is added to kernels to extract spatio-temporal features from videos [49, 98–100]. Likewise, the I3D model has been introduced by inflating pretrained 2D convolution kernels to 3D to extract space-time features from video clips [50]. In addition, the X3D network [101] expands the 2D architecture across other axes, including depth, spatial, width, and frame rate, which enables training the network with fewer parameters than other 3D networks such as Slowfast [100], yielding comparable results.

To improve training efficiency, the R(2+1)D [102] model extends 3D CNNs by splitting the spatial and temporal kernels into two separate operations, utilizing a 2D CNN for the spatial dimension and a 1D convolution for the temporal dimension. The R(2+1)D achieves better performance than 3D CNN convolutions on the Sports-1M video dataset [87]. The studies previously mentioned aid in capturing coarse features, including background information, when information from the entire video is extracted and modeled equally. However, fine features like human-object interactions that offer crucial clues regarding actions are not taken into account. Thus, in this research, we use the I3D model as backbone and propose different modules for learning human-object relations, which helps to boost the action recognition models.

Alternatively, Recurrent Neural Networks (RNNs) are utilized for capturing sequential frame changes and predicting final actions in videos [11, 95]. Long-Short Term Memory (LSTM), a type of RNN, is also used to represent temporal changes between frames in long sequences. LRCN [11], presented in Fig. 2.8, is one of the proposed RNNs for action recognition. The design of LRCN involves the use of Convolutional Neural Networks (CNNs) to extract features of each frame at a specific time t and subsequently feed them to an LSTM cell. The output of each cell at time t is used to predict the action at time t, implying that it has information

from the previous time step $t-1$. The final video classification score is calculated by averaging the predictions made at each time step. LRCN is also used for other tasks, such as image and video descriptions.

Furthermore, the ConvLSTM network is proposed as an extension of LSTM, where spatiotemporal correlations can be captured by incorporating a convolutional structure [103]. Other architectures that integrate convolutional and other recurrent networks, such as the Gated recurrent unit (GRU), are also presented for recognizing actions in videos. ConvGRU [104] outperforms 3D CNN in some action recognition datasets, such as the 20BN Something-Something dataset [3], where temporal reasoning is crucial for identifying fine-grained actions present in the dataset.

Moreover, the authors of [105] argue that LSTMs have a higher capability to capture the temporal features of a video sequence at a higher level, while 3D CNNs are more effective at capturing the temporal relations between nearby frames within a sequence. As a result, the I3D-LSTM model is proposed [105] as having superior performance to I3D alone, where top-1 accuracy is improved from 94.3% to 95.1% on the UCF-101 dataset [10].

Inspired by the success of recurrent neural networks (RNNs) in modeling sequence data, such as LSTM or GRU, they have also been used for spatio-temporal reasoning with objects in videos [106]. In [106], an object-relational network (ORN) is presented where each object in video frames is represented by its appearance, shape (e.g. using a mask), and class. Pairwise relations h_{o_t} between objects at time t and a prior time step (e.g. $t-2$) are learned via MLPs. Then, at time t, object representations of a frame are learned by aggregating the relations at time t via summation of h_{o_t}. To learn the long-term dependencies between frames, a GRU is applied, and the output is fed to a classifier. Along with ORN, a global representation of the video is also included, allowing for the capturing of the global context. The inclusion of ORN in baseline models, such as Inflated ResNet-18, increased accuracy from 38.3% to 40.9% on the EPIC Kitchens dataset [107]. Moreover, to capture high-order object interactions, an attention mechanism (e.g., α-attention) is applied over K groups of objects at each frame, followed by an LSTM process [108].

Another relation network called the actor-centric relational network (ACRN)

[109] is proposed to focus on the relation between actors and video-level features for identifying actions and avoiding explicit object detection. In ACRN, pairwise relationships between each spatial location in the 3D feature maps, which have global video information, and each actor feature (including appearance and bounding boxes) are computed via convolution operations. These relations, along with actor features, are fed to a classifier to identify actions. On the basis of average precision on the AVA dataset, ACRN outperforms the I3D model by 2.3% [110].

Furthermore, in [111], the Structural-RNN (SRNN) design was proposed, in which HOIs are modeled using a spatial-temporal graph with an RNN. This method can capture high-level information and perceive the sequence of human and object interactions. Truong et al. [112] extended the SRNN by modeling object-object relations, where spatial and temporal information between objects was observed to recognize HOIs. The results illustrate that accuracy improved from 83.2%, as reported in [111], to 90.4% on the CAD-120 dataset [2] for sub-activity recognition [112]. Skeleton and geometric features are used in [111, 112]. In contrast, in the proposed network in Chapter 3, we exclusively employ visual features of humans and objects. Additionally, we propose utilizing a hierarchical design to learn human-object interactions, where the temporal changes of humans and objects are individually captured using LSTMs, and their H-O relationship is subsequently learned using a bilinear layer. Later, through a deep global LSTM, high-level interactions are learned. Chapter 3 provides more information. Therefore, in this study, we propose utilizing LSTMs for the temporal modeling of humans and objects in videos, which helps recognize actions. More details can be found in Chapter 3.

2.2.2 Contextual Understanding

Learning the context of human actions and interactions is crucial for understanding actions within various forms of sources and features. For instance, context can be captured using multi-stream networks, where each network is dedicated to extracting a specific type of information, such as spatial, semantic, or temporal details from a video. The most common multi-stream networks were proposed in an earlier study by Simonyan and Zisserman [12], where two streams extract visual and temporal

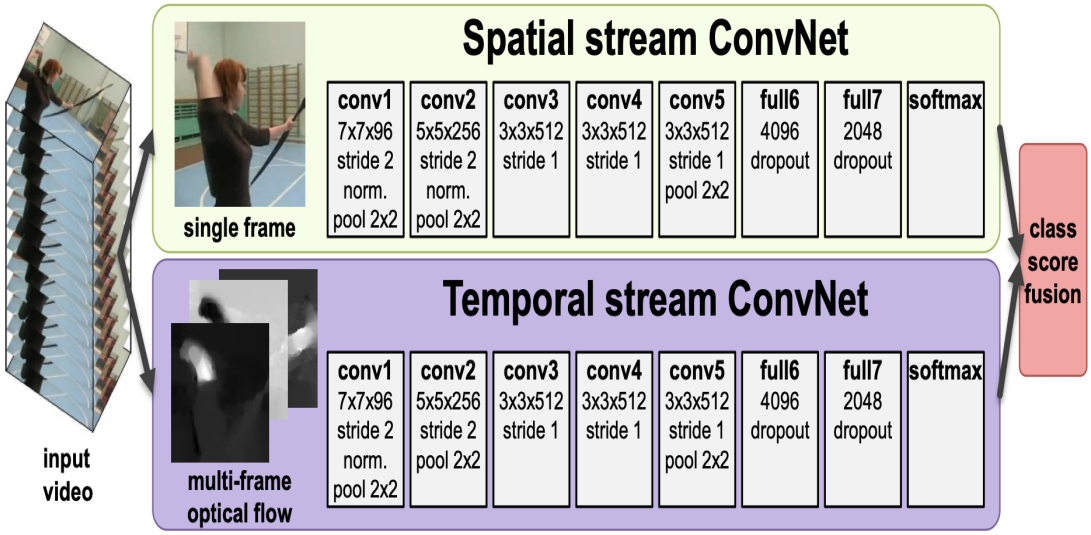


Figure 2.9: Two-stream Networks for action recognition, from [12].

(e.g., motion) features separately. As shown in Fig. 2.9, the RGB frame of a video is fed to the first stream to capture spatial information. The second stream takes a stack of optical flow as input to encode the temporal information of a video. Both of these streams implement a deep ConvNet, and the Softmax scores of these networks are combined via late fusion to obtain the final action label. When the second stream is added, the spatial and temporal networks are improved by about 15% and 4%, respectively, on the UCF-101 dataset [10]. This study has two principal limitations. Firstly, the frames were sampled from videos and thus some of the temporal information may have been missed. Secondly, the optical flow vectors must be computed before being fed to the temporal stream. Numerous studies within the same line of work have focused on exploring the most effective approaches for integrating spatial and temporal stream networks for action recognition [113–115]. One such approach is the ST-ResNet network [114], which utilizes residual connections from the temporal stream to the spatial stream, and experimental results have demonstrated that these connections significantly enhance the network's performance. Specifically, using residual connections improves accuracy by 3.3% and 5.0% on the UCF-101 [10] and HMDB-51 [85] datasets, respectively.

Other two-stream networks utilized for action recognition, such as the Slowfast network [100], use the same input mode, such as RGB, for both streams without

incorporating optical flow as an additional input. Slowfast captures the spatio-temporal features necessary for action recognition by employing distinct temporal rates in each path. In the same video clip, the slow path employs fewer frames (T), while the fast path uses a greater number of frames (αT) with a smaller channel size. Additionally, connections from the fast pathway to the slow pathway are considered after various residual blocks to merge the slow path's features with the learned representation from the fast path, which captures more motion information about an action. Experiments demonstrate that Slowfast outperforms other models that employ optical flows by solely utilizing RGB mode frames on the Kinetics-400 [69] dataset with a 5.9% accuracy increase.

Moreover, a multi-stream network design is employed to detect human-object interactions in images. This approach incorporates three distinct streams: human, object, and interaction streams. The human and object streams specialize in learning specific features related to humans and objects, respectively. Simultaneously, the interaction stream captures pairwise relations between them by integrating different pieces of information concerning human-object relationships, such as spatial configurations [21,116]. Furthermore, contextual cues, including whole image features, are integrated into both the human and object streams to enhance the network's performance [116]. Recently, human poses are utilized to enhance the human stream, while semantic priors are applied to refine the object stream. These refinements have led to promising results on two widely recognized human-object interaction datasets for images, namely HICO-DET [21] and V-COCO [117]. However, image-based human-object interactions are not suitable for application in videos. This is because they do not consider the temporal relations and long-range dependencies of humans and objects, which are necessary to maintain the context of human-object interactions over time. For further insights into human-object interactions in images, readers are encouraged to refer to this survey [118].

Furthermore, context of Human-object interactions (HOIs) in videos can be learned using Graph neural networks, such as, graph convolution network (GCN) [119] and graph attention networks (GAT) [66], where spatio-temporal relations between visual nodes, including humans and objects can be captured [111, 120–123].

Space-time graphs have been proposed in [120], where object context relations during time are captured via GCN and objects in adjacent frames are connected based on their intersection over unions (IOU). The experiments in [120] on Charades [1] and Something-Something v1 (SSv1) [3] datasets show better results than I3D and Non-local models, which do not consider human-object relations. Additionally, in [124], graph attention is used to model the relations between humans and objects, considering their spatial distance in each clip.

Besides that, Herzig et al. [125] propose to learn the hierarchical context of actions by combining the union box of both objects with object features to represent the object and to have more spatial appearance information. Also, a non-local operation is employed to learn the relationship between objects at the frame level. The temporal context of these relations is learned at a deeper level, and after aggregating the relations at time t, the actions can be recognized. The proposed model in [125] outperforms the model in [120] on the Charades dataset [1] by 1% mAP. In addition, object features are represented in [126] by using its annotations (e.g. bounding boxes and categories) and used as graph nodes for spatial and temporal modeling, along with global video appearance features, for recognizing actions. This model outperforms I3D on Something-Something V2 [3] by about 5% top-1 accuracy.

Although earlier studies have focused on modeling various object features and relations to improve the performance of action recognition models, the contexts of humans and objects have not been sufficiently studied. Inspired by the Slowfast design, we propose to use two different contextual views of human-object interactions where different patterns of interactions can be learned. Also, in contrast to conventional procedures for fusing multi-stream networks such as early and late fusion, we investigate incorporating knowledge distillation in the design of the proposed network, which is covered in more detail in Chapter 4.

2.2.3 Attention and Long-Range Dependencies of HOIs

Attention mechanisms help the network focus on specific important information in videos from either a spatial or temporal dimension [127–130]. Attentions with recurrent networks are utilized in numerous studies on action recognition to enhance

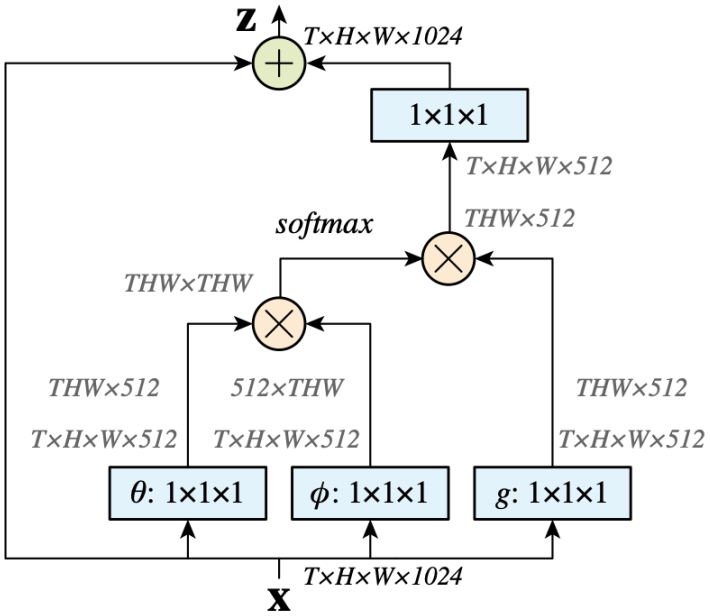


Figure 2.10: Non-Local (NL) block, from [13]. T indicates the temporal dimension. H×W is the spatial size while 1024 denotes the number of channels. ϕ,θ and g are different embeddings (e.g. $1 \times 1 \times 1$ convolutions) for the same input X. $\otimes$ and $\oplus$ are matrix multiplication and element-wise summation, respectively.

network performance [128, 131–134]. The first proposed network is in [128], where a soft attention mechanism is incorporated with an LSTM network, and the attention score is learned between the current frame features and the hidden state of the previous time step h_{t-1}. The output of the LSTM at time t is also used to predict the action class. Thus, predictions of 30 frames, which are sampled from each video, are averaged to get the final action class. The findings from experiments on three datasets, including UCF-11 [135], HMDB-51 [85], and Hollywood2 [136], demonstrate that incorporating the attention mechanism improves model performance compared to baseline models such as LSTMs or CNNs, which do not use attention at all. Since the attention in [128] focuses on spatial attention at time t, Du et al. [131] propose to consider spatio-temporal attention for LSTMs to capture more contextual cues about actions. Furthermore, to focus on different aspects of video information, two-stream LSTM networks are proposed [134], where temporal attention is applied to optical flow images in the first stream, and the second stream generates spatio-temporal attended features where RGB frames are used as input.

Moreover, since 3D CNN networks deal with whole video information equally, attention over temporal (e.g. frame level) and channel dimensions for 3D models is proposed in [129] to capture the most discriminative information from videos. This experimentally improves the performance of 3D models, with an accuracy gain of roughly 4% on the HMDB-51 dataset [85]. Other forms of attention have been proposed for action recognition, such as second-order pooling [130] and self-attention [8, 13, 137]. The self-attention mechanism implies intra-relations where it uses the same input to compute attention scores. In [13], the Non-local block (NL) is proposed to be added after selective blocks of CNNs. Figure 2.10 shows a type of non-local block operation (e.g. self-attention) for space-time input video tensor. The results on Kinetics-400 [69] and Charades [1] show that adding the Non-local block to either 2D CNN or 3D CNN (e.g. I3D) backbones improves the performance of the network, indicating that it aids in learning long-range dependencies in videos.

Guo et al. in [137] argue that the NL block learns spatio-temporal correlations of pixels concurrently, which may lead to capturing unnecessary information, especially for complex actions, and eventually result in incorrect classification of actions. Thus, a separable self-attention (SSA) is proposed [137] to model spatial and temporal correlations in a sequential manner, where first spatial attention, including both position and channel attention, at a frame level, is learned, then the temporal correlations are processed. SSA outperforms the NL 3D CNN in terms of performance, improving top-1 accuracy by 1.7% and 2% on the Something-Something-V2 [3] and Kinetics-400 [69] datasets, respectively.

Considering the effectiveness of attention mechanisms in various networks, we have incorporated attention mechanisms, including self-attention, into all of the networks presented in this thesis, such as LSTMs, graph neural networks.

More recently, the success of transformer-based networks in natural language processing (NLP) [8] has demonstrated their efficacy in learning long-range dependencies. This success has extended to computer vision tasks, such as image classification [9, 138], tracking multiple objects [139], and action recognition [140, 140–147]. The Vision Transformer (ViT) [9] achieved state-of-the-art performance in image classification without applying convolution layers and has been extended to video

action recognition, where spatio-temporal tokens are extracted from videos and fed to transformer encoders [14,147]. In [142], the VTN network is proposed, where spatial features of each frame can be extracted from any backbone (e.g., 2D ResNet-50 or ViT) and then fed to a transformer with Longformer attention for temporal modeling.

Furthermore, Arnab et al. [14] proposed ViViT with various network designs to model the space and time correlations between video tokens using different attention mechanisms, including self-attention and dot-product attention. The video tokens in [14] are mainly 3D tokens that are embedded from "Tubelets" and can be seen as spatio-temporal patches by considering the temporal dimension. Figure 2.11 shows the tubelet embeddings. In [14], two main designs were proposed for modeling video tokens, each with distinct approaches. In one of the proposed network architectures, the transformer encoder receives spatio-temporal tokens as input. Pairwise interactions between tokens from various temporal indices are learned in this design, resulting in learning long-range dependencies between tokens. Alternatively, the second model (FE: Factorized Encoder) learns the interactions between tokens from the same temporal indices but different spatial indices via a spatial transformer. Then, the spatial class token representations (e.g., class token) are passed to the temporal transformer, and the output is used to classify actions. Among other alternative models and prior methods that used deep 3D convolutional networks, the FE model achieves the best results on three action datasets, including Epic Kitchens (EK) [148], Something-Something-v2 [3], and Moments in Time [149].

Moreover, MVit [140] incorporates multiscale feature learning into transformer-based networks by applying an attention pooling layer to reduce spatiotemporal resolution while maintaining a lower computation cost compared to ViViT [14] or VTN [142]. Recently, multi-view transformers were proposed in [146] to encode the input video into N different views, with each view fed to a separate transformer encoder. These views are then fed via a global transformer to output the classification score. This design learns tokens (e.g., tubelets) reasoning at different temporal indices in each view, which helps to understand actions. On various datasets, including Kinetics-400 [69] and Something-Something v2 [3], MVT performs better

than other transformer networks like ViViT [14], MVit [140], and other counterparts works such as Slowfast [100], which use various temporal frame rates within each view (e.g., stream).

As the design of vision transformer (ViT) [9] includes dividing an image into patches of tokens and learning the relation between these tokens, it is also used as a relation module between different types of features. For instance, transformers are used to learn visual relations between the features of humans located in the center clip, which is considered as a query, and the features from the entire clip to learn the context of the action by utilizing the properties of self-attention in the transformer [150]. Similarly, long-term contextual information in videos is captured using the Long-Term Feature Bank (LFB) Operator [151], such as non-local, which learns relationships between object features derived from short-term clips and other objects within a wide temporal window (e.g., long-term features). On the Charades dataset [1], the LFB NL model increases accuracy by 2.8% compared to the model proposed in [120].

Furthermore, transformers can capture the spatio-temporal context of objects by considering spatial information such as location, as well as the category of the object [152,153]. Additionally, human and object coordinates of each frame are used as input tokens that are embedded and sent to spatial transformers for a few-shot HOI recognition task, rather than using patches of a frame [154].

Motivated by the success of the Factorized Encoder (FE) design in [14] and other transformers in modeling long-range dependencies, we propose using the ViT transformer as a relation module for modeling human-object interactions. This is done after extracting tokens from a CNN network. Our proposed network design achieves better results compared to earlier works and other proposed networks in this research. For more information about our network design, please refer to Chapter 5. In contrast to [154], which required precise object annotations, we use human and object visual features as input tokens in the hierarchical transformers network proposed in Chapter 5.

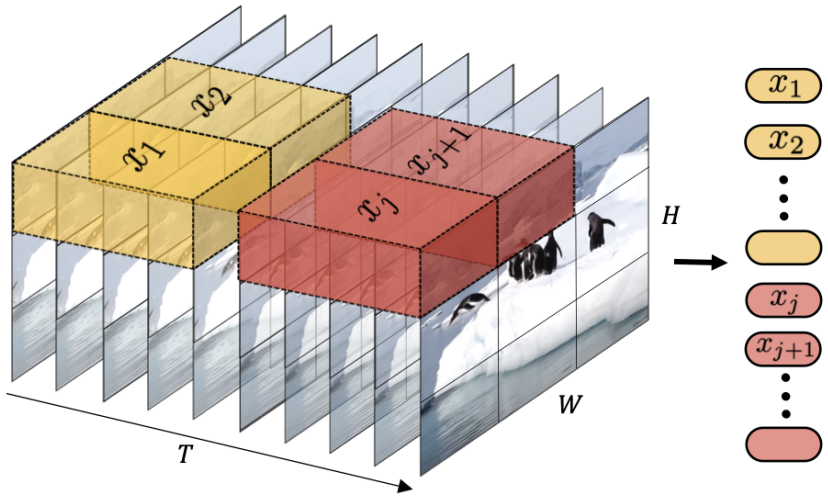


Figure 2.11: Tubelet embeddings, from [14].

2.3 Conclusion

Convolutional neural networks (CNNs), recurrent neural networks (RNNs), and long short-term memory (LSTM) networks are commonly used for action recognition tasks, typically by utilizing the features of entire frames. These networks have also been employed in previous studies to model the temporal aspects of humans and objects. In Chapter 3, we introduce a novel approach that utilizes LSTMs with a hierarchical design for modeling HOIs, with the goal of enhancing HOI recognition. Our proposed method involves training two-level LSTM networks with an attention mechanism and a bilinear layer, enabling effective learning of the relationships between humans and objects.

Moreover, recent methods for video action recognition, as reviewed in Section 2.2.2, have utilized two streams to capture better spatial and temporal information about actions. These methods consider additional input modes, such as optical view, or using the same mode but with different frame rates in each, as in SlowFast. However, key discriminative indicators of an action, such as inter-object or inter-human relations, have not been considered. Therefore, in Chapter 4, we propose a network that leverages two different contextual views of human and object relations, providing different cues for interaction that help recognize actions. The contextual views help to capture human-object interactions both locally (spatially) and globally

(temporally) throughout a video. We also propose context knowledge distillation to transfer knowledge from the teacher's contextual view of HOIs to the student network, which incorporates information from different contexts of such interactions.

Unlike the methods reviewed in Section 2.2.3, which proposed various transformer-based architectures for action recognition, we investigate the use of hierarchical structures to model human and object interactions via transformers. In Chapter 5, we propose hierarchical transformer encoders and examine their effectiveness in learning the relationships between human and object tokens in space and time, based solely on visual appearance features.

CHAPTER 3

Recognising Human-Object Interactions Using Attention-based LSTMs

3.1 Introduction

Distinguishing human actions is a major challenge in computer vision. The ability of deep learning algorithms to recognize human-object interactions (HOIs) aids in addressing this challenge [23]. HOI identification generally involves localizing the human and corresponding object for interaction. In the case of videos, both the human and the object need to be tracked over time, and their relationships must also be modeled. This phase is essential for recognizing HOIs [21]. To improve the accuracy of HOI recognition, researchers have utilized various forms of information, such as the physical appearance of the human or object [23,24,155], the posture of the human body [26, 156], where the human is looking [156], and the positioning of the object relative to the human [155]. Although HOIs have been extensively studied in terms of images, there have been limited studies examining HOIs from video streams. This chapter investigates HOIs in videos by proposing a deep learning framework that uses hierarchical LSTMs to capture spatio-temporal information of interactions. HOIs are recognized based only on RGB frames from videos without using skeleton

or depth information. Training a detector from scratch to find the human and object in each video frame requires extensive human and object annotations, such as their bounding boxes, which is time-consuming. To avoid this, we use a pre-trained detection model to localize humans and objects in videos. We also use LSTMs and an attention mechanism to highlight important parts of human and object temporal information. Each human and object in videos is represented by LSTMs, and the second-level global LSTM captures high-level information of object and human interaction. This chapter's contributions can be summarised as follows:

- Proposing an LSTM-based framework with attention mechanism for recognizing human-object interactions (HOIs) in videos. The HOIs are modeled by using hierarchical LSTMs to capture the dynamics of the human and objects in a video sequence.

- Investigating the use of a bilinear layer which can handle the features of human and object, generating a discriminative feature representation from human and object information.

- performing experiments on two datasets, including a subset of the UCF101 dataset (UCF101-20) related to HOIs [10] and the CAD-120 dataset [2]. Using a bilinear layer on the UCF101-20 dataset can generate a more discriminative feature representation for recognizing HOIs, resulting in a 5% improvement over the conventional method of feature fusion, such as concatenation. Furthermore, the proposed network design yields promising results.

3.2 Methodology

3.2.1 Preliminary

Recurrent neural networks (RNNs) are a powerful network architecture for recognizing sequential data of different lengths, such as sentences and sequences of images in a video. In traditional neural networks, layers behave independently. However, in RNNs, all inputs are related to one another in a way that the same task is performed for all elements of a sequence. The type of RNN used in this chapter is

Long Short-Term Memory (LSTM) [58]. In LSTM, two vectors are involved at each time step: the hidden vector and the cell state vector. LSTM can add or remove information to the cell state by using different gates, namely input, forget, and output gates. This design helps maintain the long dependency of a sequence in LSTM. More details about the LSTM network are provided earlier in Subsection 2.1.2. In the proposed model, two different modes of LSTMs are used, including local and global LSTMs. At a lower level, two local LSTMs are employed to independently model the temporal information of humans and objects, one for each category (e.g., LSTM H for human and LSTM O for objects). At a higher level, the aggregated human-object features are then fed to the global LSTM

3.2.2 Human-object interaction model

As mentioned earlier, limited research has addressed the problem of HOIs in videos. Learning the global description of a video's temporal information is important for accurately classifying actions. Inspired by the success of employing a hierarchical architecture in modeling the temporal dynamics of group activities [157], we propose a hierarchical design for handling HOIs. Firstly, the inputs to our framework are human and object tracklets, which are sequences of bounding boxes of humans and objects in a video. These tracklet features are fed to LSTM layers. Specifically, each tracklet, including human and object tracklets, will be fed into a LSTM layer to capture intensive temporal information in a sequence. The proposed framework can be divided into three parts: input pipeline, modeling H-O interactions, and classification procedure.

- **Input pipeline**: To model human-object interactions, it is essential to have information about the parts involved in an interaction. Therefore, spatial information (e.g., appearance) of humans and objects in videos is crucial. This information includes the shape and texture of humans and objects in video frames. The input of the model consists of object and human tracklets in a video sequence. The spatial features of these tracklets are extracted via convolutional neural networks (CNNs). Here, pre-trained models including ResNet [48] and VGG-19 [46] for feature extraction are used, which include a

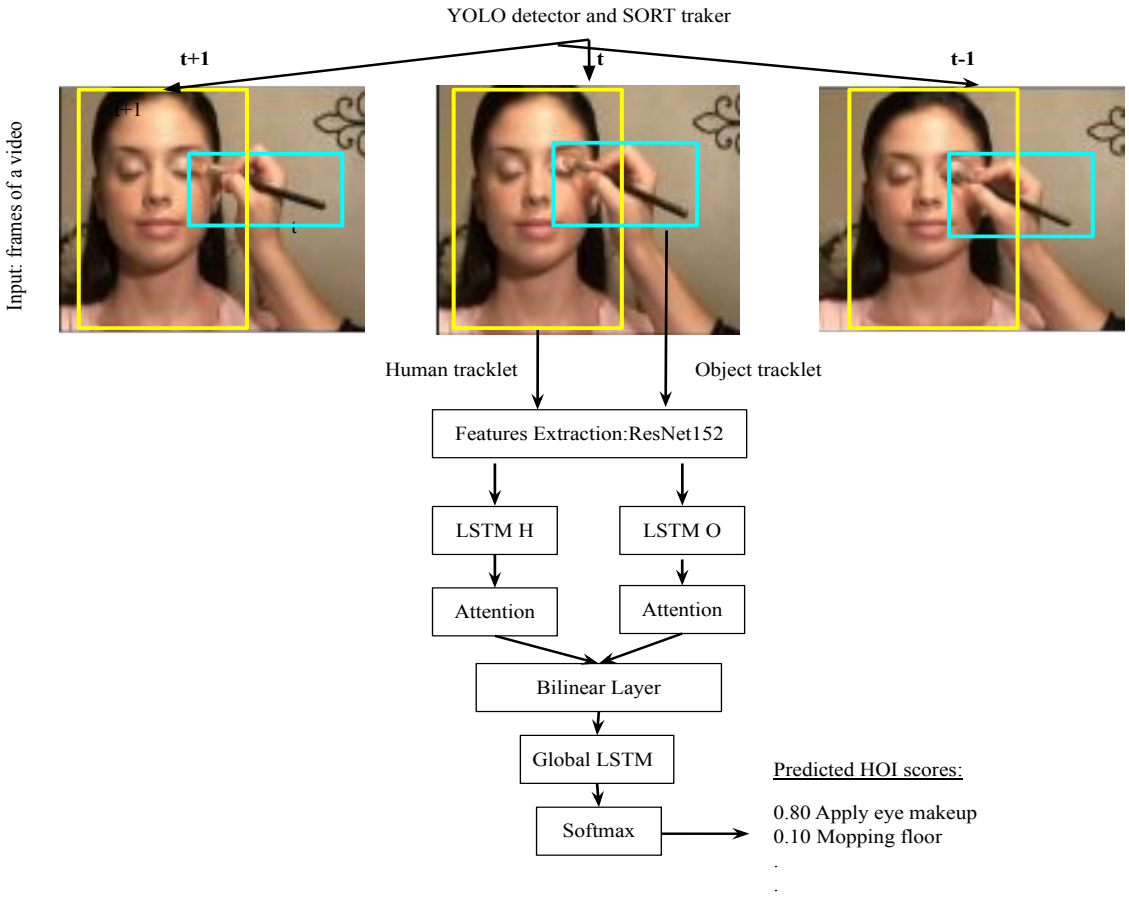


Figure 3.1: The proposed hierarchical LSTM framework.

series of convolutional layers with kernels that extract different features, such as edges, color, gradient orientation, etc., from the bounding boxes around the person and object in each video frame.

- **Modeling H-O interactions:** As shown in Figure 3.1, both the human and object tracklet features are fed into LSTMs. Specifically, the human tracklet is fed into an LSTM to capture the temporal information in human movement (e.g., motion), while the object's tracklet is fed into another LSTM to learn the object's motion during the video. A soft attention mechanism [158] is applied to the output of both LSTMs related to each of the human and object. In the soft attention mechanism, a soft alignment score is computed between the last hidden state and each hidden state in the LSTM layer through multiplication. This score is then fed into a softmax layer where the output represents the

attention distribution. This output is considered as attention weights with the size being equal to the number of time steps in the LSTM layer. Finally, the context vector is computed by multiplying the attention weights (e.g., scores) and LSTM hidden states. The vectors generated after applying the attention mechanism over LSTMs are fused using a bilinear layer. The purpose of this layer is to aggregate features from humans and objects, which can imply the pairwise interactions between these H-O features [159]. The bilinear layer operation can be formulated by:

$$Y = WA_h \otimes A_o \tag{3.1}$$

where A_h and A_o are the human and object features after the attention layer is applied, respectively. W is the learnable weights and $\otimes$ indicates the outer product. This can produce a representation of human and object interactions. Y is then fed to a deep LSTM to learn high-level information about HOIs. This is followed by a classifier with the softmax activation function. Figure **??** illustrates the proposed framework.

- **Classification procedure**: To predict the action label of each video, a one-layer dense classifier with a softmax activation function is utilized, which is commonly used in multi-class classification problems to compute the probability for each class. The propsed model is trained using the cross-entropy cost function, and the loss over the entire dataset is formulated as:

$$loss = \frac{1}{m} \sum_{i=1}^{m} H(y_i, \hat{y}_i) \tag{3.2}$$

Where m is the number of training examples and H is the cross-entropy loss. y_i and $\hat{y}_i$ are the actual and predicted probability distributions for action classes, respectively. The goal during training is to minimize this loss using the gradient descent algorithm.

3.3 Experiments

3.3.1 Datasets and evaluation metrics

- **Datasets:** As initial experiments, we validated our proposed model on the UCF101 dataset [10]. UCF101 contains a variety of human actions that are performed either indoors or outdoors, such as applying eye makeup and skateboarding. Since the scope of this study is recognizing HOIs, we used 20 classes from the UCF101 dataset (split 1) that are related to HOIs. The set of 20 action videos present several challenges such as differences in lighting conditions, the presence of cluttered backgrounds, diverse camera viewpoints, and variations in object appearances, as well as varying video quality. We further evaluated our model on the CAD-120 dataset [2], which contains 120 videos of ten human actions involving interactions with objects, performed by four subjects. The presented dataset poses several challenges that are indicative of real-life situations encountered during object interactions. An example of an intra-variation problem arises when four individuals execute identical actions, yet adopt varying postures or hands. Furthermore, the existence of occlusion among objects poses an additional difficulty in the dataset. Human and objects annotations are provided with the dataset, and we used only the RGB features of humans and objects.

- **Evaluation metrics:** For both datasets, we measure the performance of our model using accuracy, which is calculated by dividing the number of correctly recognized action videos by the total number of videos in the test set.

3.3.2 Implementation details

UFC101:

PyTorch is used to implement the framework. In order to detect and track humans and objects in video frames, the You only look once (YOLO) [160] object detection model pre-trained on COCO dataset [161] and Simple online and real-time tracking (SORT) tracker [162] are employed. The detected object and human bounding boxes

are then fed to ResNet152 [48] to extract spatial features.

In the UCF101-20 experiments, the training set consisted of 1,093 videos, with 25% of those videos reserved for validation, and the testing set comprised 486 videos. From each video, 28 frames are sampled uniformly, and the detected humans and objects are cropped and resized to 224x224 to meet the required size for the feature extraction models, including ResNet-152 [48] and VGG-19 [46]. For training, Adam optimizer [163], which has been empirically shown to be better than others in terms of convergence speed, is chosen with the learning rate set to 10^{-4}. To reduce overfitting, batch normalization and regularization techniques such as dropout are employed. Since pre-trained models are used for detection and feature extraction, the network does not follow an end-to-end training approach. All hyper-parameters are only trained after the pre-trained model has extracted features. The training is carried out using a batch size of 32 videos and for 40 epochs. The training and validation accuracy against 40 epochs are shown in Figure 3.3. The model is trained at different epochs, such as 20, 30, and 40 epochs, and the highest validation accuracy is achieved when training the model with 40 epochs. All experiments in this study are conducted on a single Nvidia GeForce RTX 2080 Ti GPU.

CAD-120:

In order to extract human and object features from each frame, the method described in [164] is followed, where ROI crop is employed and human and objects are resized to prepare them for use as input to Resnet-50. Thus, each human and object has a feature dimension of 2048. The model is trained for 100 epochs using Adam optimizer with a learning rate of 2e-6. The learning rate is decayed by 0.8 after 50 epochs. Additionally, Leave-One-Out cross-validation is used, where one subject out of four subjects is used for testing the model in each fold.

3.3.3 Results and discussion

UFC101: As can be seen in Table 3.1, the results indicate that the encoding of human-object interaction can be improved by including a bilinear layer, rather than simply concatenating human and object features. Additionally, the importance of

Table 3.1: Results of the proposed framework.

Architecture	**Acc.(%)**
ResNet-152 + Our model with attention and concatenation	59.77
ResNet-152 + Our model with attention and bilinear layer	**64.50**
VGG-19 + Our model with attention and bilinear layer	55.66
ResNet-152 + Our model with attention and bilinear layer(w/o a global LSTM)	63.05
ResNet-152 + Our model with bilinear layer(w/o attention)	46.14

different parts of our model was evaluated. The model was trained without the final global LSTM layer, resulting in a drop of accuracy to 63%, highlighting the important role of the global LSTM layer in modeling HOIs. The model was also run without applying the attention mechanism, resulting in a very low accuracy of 46.14%. This indicates that giving more attention to the significant part of the video sequence improves the learning process. Our model was also evaluated using VGG-19 model [46] as feature extraction instead of ResNet-152, and the results confirmed that using residual mapping in ResNet-152 leads to extracting more complex features than stacking convolutions in VGG-19. This implies that using better models for feature extraction and detection leads to better results in terms of accuracy. Fig. 3.2 shows the confusion matrix for the performance of our proposed model using the ResNet-152 backbone with respect to 20 human actions that appear in the UCF101-20 dataset, achieving an accuracy of 64.50%.

In fact, Comparing our results with existing methods is challenging because the reported results on UCF101 are based on using all 101 categories in the dataset, whereas our study focuses on only 20 classes, which are related to the scope of this research. Thus, as we use a subset of the UCF101 dataset mostly related to human-object interactions, it would not be fair to compare our results with other methods that have not reported results on the same subset.

Moreover, the proposed method assumes that the main and most important

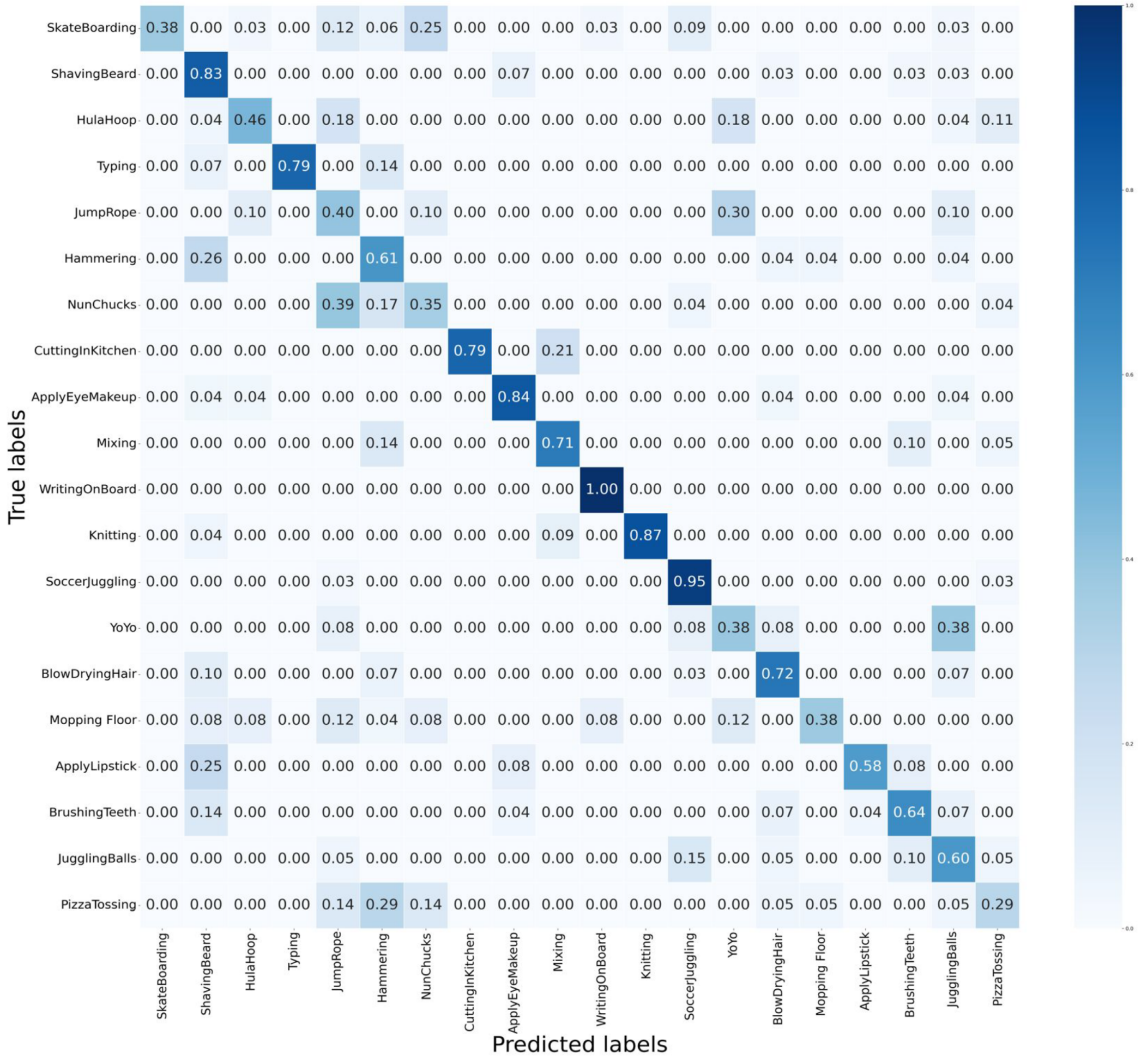


Figure 3.2: Visualization of confusion matrix for UCF101-20 dataset [10] after applying our proposed model.

object for an interaction has already been detected. However, in cases where a human is interacting with objects, it is not easy to detect the target object for the interaction using pre-trained detectors. Figure 3.4 illustrates some false detections. Furthermore, in real-world cases for interactions, more than one object can be considered important for interactions. Thus, in the next experiments, we use the ground truth of objects that are available in the CAD-120 dataset. For the object's LSTM, we concatenate the features of objects in a frame at time t to form the object representations for LSTM input at time t.

CAD-120: The ablation studies on CAD-120 [2] are presented in Table 3.2. It is observed that the proposed model helps to capture more discriminative cues

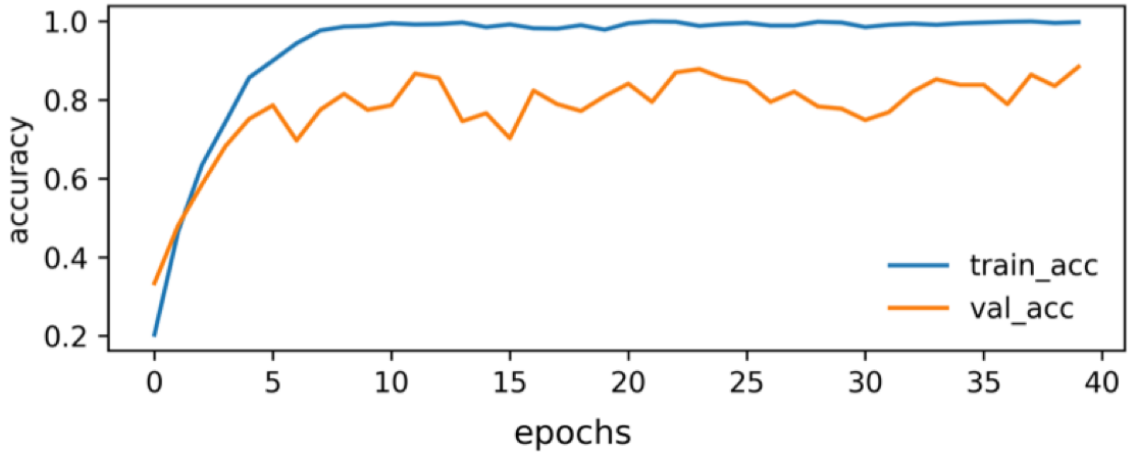


Figure 3.3: Training our model with 40 epochs

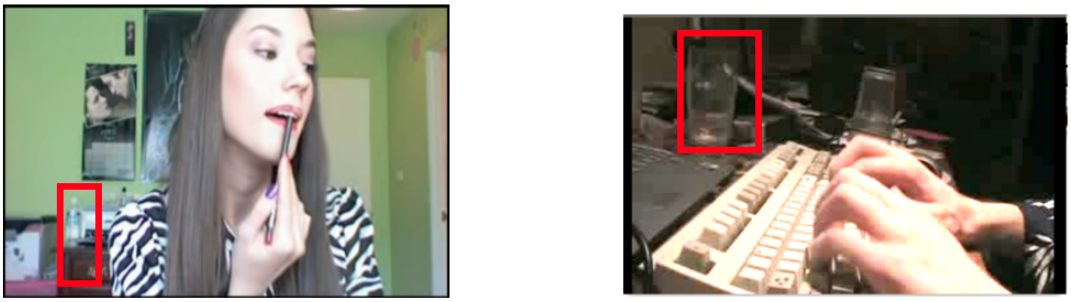


Figure 3.4: Some of false detections cases: (Left) Drinking bottle is detected as the main object for interaction instead of lip brush. (Right) A Cup is detected as the main interacting object instead of the keyboard.

about human interactions, with the accuracy reaching 94.35%. Without considering temporal learning, where only human and object features are used and pooled over time, the accuracy drops to 86.69%. Comparing the bilinear layer with just combining features from human and object LSTMs, we can observe the importance of the bilinear layer for learning the relation between humans and objects, where the accuracy increased by around 3%. Finally, by adding attention and global LSTM, the model achieves the best results. Fig. 3.5 shows the confusion matrix for the performance of our model towards human interactions present in the dataset. Also, since recent transformer networks boost the performance of models in different computer vision tasks, we replaced the LSTMs and attention layers with transformers. The results show improvement in the model's performance, which is expected since transformers have the power to model long-range dependencies of humans and ob-

Table 3.2: Results of the proposed framework on CAD-120 [2].

Architecture	**Acc.(%)**
baseline	86.69
Our model with summation (w/o a biliner/global LSTM/attention)	89.98
Our model with bilinear layer only(w/o a global LSTM/attention)	93.37
Our model with attention, global LSTM and bilinear layer	**94.35**
Our model with transformer and bilinear layer	**95.90**

jects in videos. More details about transformer design are provided in Subsection 2.1.1.

Comparison to prior works: Table 3.3 reports the accuracy achieved by prior works using the CAD-120 dataset [2]. As we can see, our model outperforms the previous ones even when only using visual features, compared to [2, 16], where additional depth information is included. Moreover, our model demonstrates the importance of temporal modeling of human and objects for recognizing actions and achieving better results compared to [89], which uses information from the entire video frames.

Table 3.3: Results with CAD-120 [2]. Note that in [2, 15–17] additional skeleton or depth information has been employed.

Model	**Acc.(%)**
Wang et al. [15]	81.2
*Liu et al. [16]	93.3
*koppula et al. [2]	80.6
*Tayyub et al. [17]	95.2
Sanou et al. [89]	93.6
our model	**94.35**

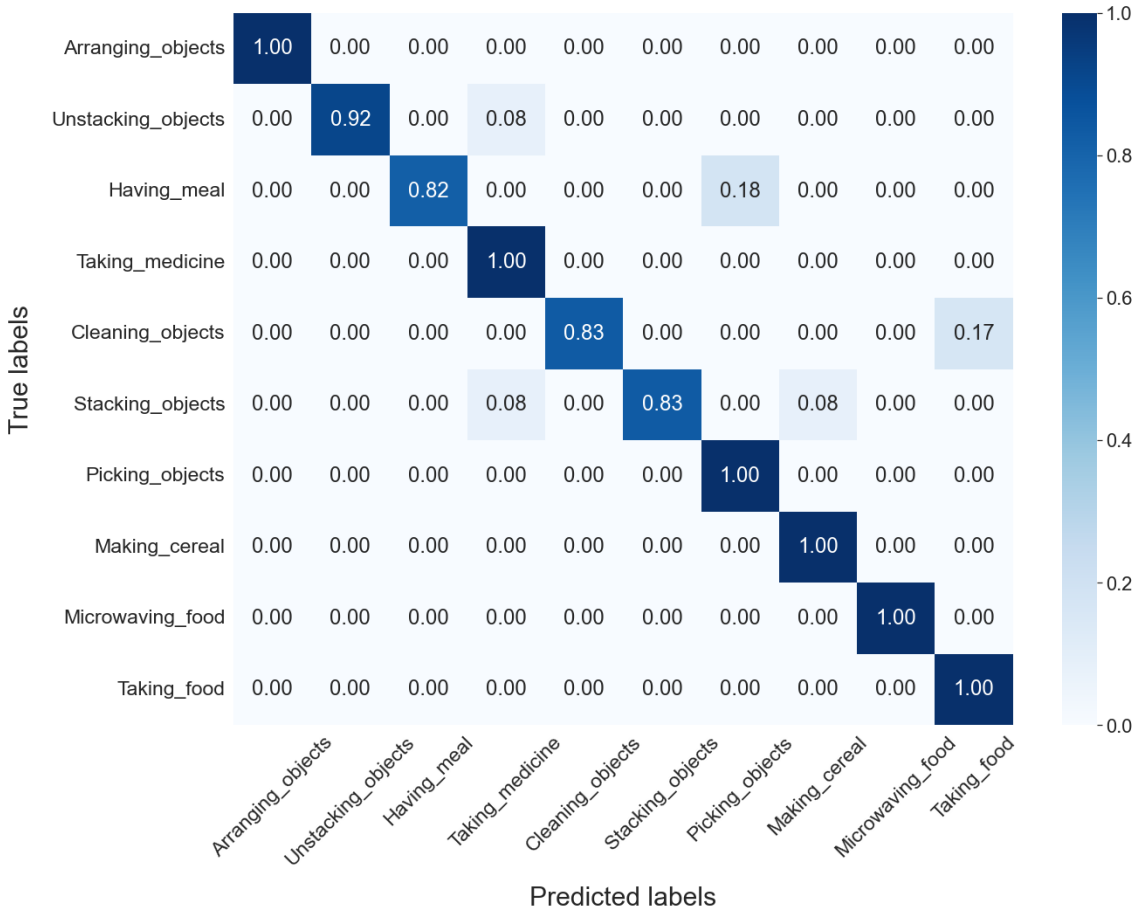


Figure 3.5: Visualization of confusion matrix for CAD-120 [2] after applying our proposed model.

3.4 Conclusion

This chapter introduced a new framework designed to solve the problem of human-object interaction recognition using LSTMs with attention and a bilinear layer to model human and object temporal information. The results show the importance of a hierarchical design that directs the network to learn high-level information about an interaction.

One of the drawbacks of the proposed method is that it does not consider the spatial relations between objects within the same frame, which provide important cues about the local context of an interaction. Moreover, in real-world interactions, not all objects appear at all times as they may disappear at one time step and reappear in another time step. Hence, using an LSTM for objects may not be practical in such cases because, in some time steps, the object may disappear. Therefore,

in the next chapters, Chapter 4 and Chapter 5, we propose to use RPN networks to produce object proposals at each frame, which can eventually capture all the objects that appear at different time steps This can be a solution for not having the ground truth bounding boxes. Additionally, in the upcoming chapter, we consider all humans and objects as nodes in graphs and study the global and local context of human-object interactions for recognizing actions. Also, as more than one object can be involved in an interaction, we consider all the objects that appear in a scene to learn the most important objects for recognizing an interaction. Thus, not only human-object relations but also object-object relations are included in the next chapter

CHAPTER 4

Distillation of Human-Object Interaction Contexts for Action Recognition

4.1 Introduction

Chapter 3 presents a novel model for recognizing human-object interactions, which emphasizes on temporal learning of each human and object individually before utilizing a bilinear layer to learn their higher-level relationship. However, this approach fails to consider the context of human-object interactions, such as the local relationships between objects at a particular time t. Therefore, in this chapter, we investigate methods to capture the interaction context by incorporating diverse relational views of humans and objects. Also, this chapter focuses on indoor and daily human interactions because they represent some of the most common situations humans encounter on a daily basis. This makes them particularly pertinent for practical applications, including surveillance and human-computer interaction (HCI).

Human action recognition tasks that typically involve interaction with objects are challenging even for deep learning methods especially under complex scenarios. A human can interact with the same object but performing different actions. For example, a human can hold a laptop and can put it somewhere. These two actions,

"hold" and "put", are different but they involve the same object. In addition, a variety types of objects afforded to same action (e.g., refrigerators and doors can be involved in the same interactions including open and close) needs to be considered [165]. Moreover, the existence of different objects around a human could confuse model predictions. For example, if a human is drinking a coffee and there is a book nearby, a model may inaccurately predict that the human is both reading and drinking. Furthermore, during a video sequence, the states of humans and objects change over time, such as a human can hold an object and release it at any time step, followed by interacting with another object which makes identifying correct interactions very challenging. Hence, identifying humans and objects at each time step and learning their relations can help understand a scene. This implies learning objects that are closely located for identifying interactions. The transition of human and object states over time also offers crucial cues for understanding what a human is performing. Consequently, it is important to capture contextual information about interactions at a specific time and throughout a video, making action recognition success. Although modelling HOIs has been broadly studied in images [21, 24, 25, 35], it has received less consideration in videos. Even deep learning methods have been developed for recognizing human actions in videos, most of them, including Covnet [12], recurrent neural networks (RNNs) [11, 166] and 3D convolution models [50, 102], only take individual frame-wise information as inputs without explicitly modeling human-object relations across a video sequence. Hence, such methods failed to capture useful global context cues, i.e., long-term human object dependencies, for assisting action recognition. Recent works [106, 120, 125, 126, 167] have proposed to model human-object relations by performing spatio-temporal reasoning through multi-head attention mechanism for recognizing actions in videos. As they capture more context cues to reason HOIs, they have achieved promising results over baselines that do not consider human-object relations.

In this chapter of our research, we propose to capture human-object relations from their local and global contextual views as well as transferring knowledge between these views. The local contextual view captures human-object relations at a specific time, e.g., spatial relations. The global contextual view encodes human-

object relations over time, e.g., temporal relations, to capture long-term human-object relations. The design of the network for global and local contextual views is flexible. Motivated by the success of graph attention networks (GAT) [66] in different tasks including person re-identification [168], action recognition [62, 120, 169] and video question answering [170], our method exploits GAT to construct our two contextual views modules. Since the global context of an interaction offers complementary information to the local contexts of such interaction and vice versa, previous works combined different types of context features via concatenation [126] or summation [120], or even considered the global features as an extra node in the graph [171]. Inspired by [172] and instead of learning these contexts via features level which are prone to noise, we propose to apply knowledge distillation, transferring knowledge about interactions from global to local contextual views, and vice versa. We, therefore, exploit teacher-student network design, investigating which of the proposed contextual views can form a better teacher, offering richer HOI information to guide the student network for improving action recognition performance.

To the best of our knowledge, we are the first to investigate knowledge distillations between two HOI contextual views for action recognition in videos. The main contributions of this chapter are:

- Proposing a novel teacher-student network based on graphs neural networks to learn spatial and temporal interrelations between humans and objects in a video from two different contextual views. Hence, the long-term and non-local dependency between humans and objects across video frames can be captured.

- Investigating how knowledge from the teacher's contextual view of interactions can be obtained, and distilling it to the student's contextual view of interactions to improve action recognition performance.

- Evaluating our model on Charades and CAD-120 [2] datasets [1] and conducting comprehensive experiments in transferring knowledge between local (e.g., Spatial) and global (e.g., Temporal) contextual views of human-object interactions. Our teacher-student design is effective to distill knowledge between global context and local context graphs. We also observe that the student

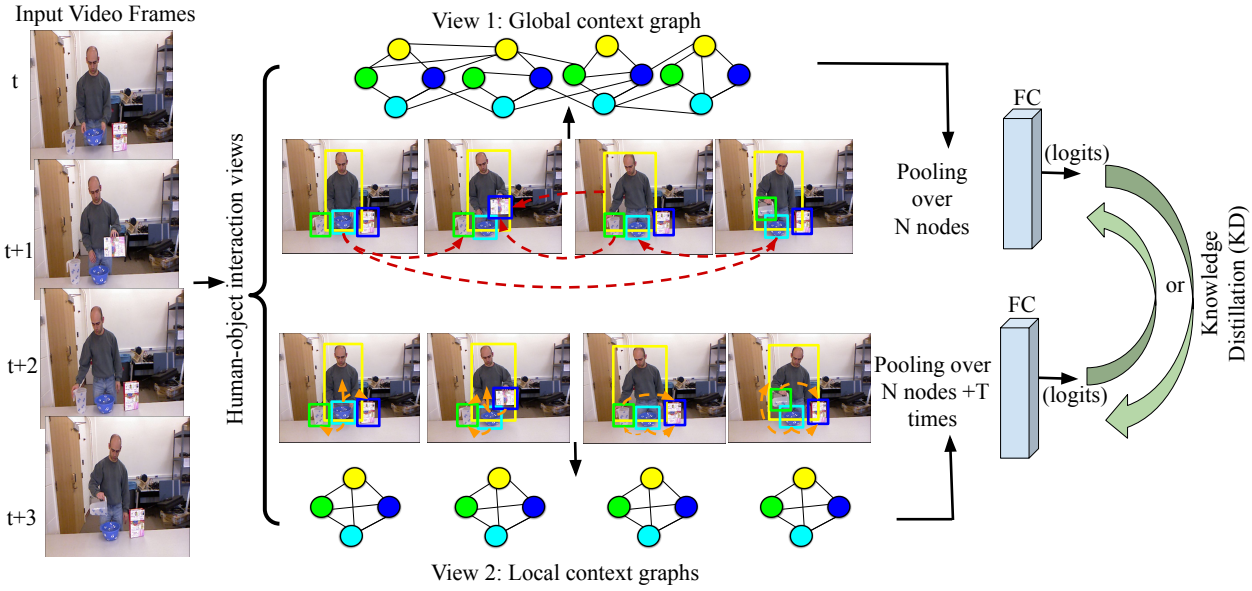


Figure 4.1: Overview of our proposed GLIDN network.

network outperforms its teacher by exploiting both global and local contexts of an interaction.

4.2 Methodology

4.2.1 Network overview

Figure 5.1 shows the architecture of our GLIDN. It takes video frames and the bounding boxes of human and objects at each frame as inputs. Frame features (e.g., appearance features) are then extracted by a convolutional neural network, such as ResNet [48]. RoIAlign [173] is then applied to extract features of each human and object boxes from the backbone feature map. The bounding boxes are generated via Region Proposal Network [174] if they are not available in the dataset. These extracted region features are used as the initial features of graph nodes in both the global and local contextual views. The human-objects relations from the teacher contextual view are distilled into the student context representation by aligning logits from the two contextual views.

4.2.2 Global and Local Context Graphs

As mentioned earlier, we utilize graph attention network (GAT) [66] as our graph networks to learn the relations between human and objects from different contextual views.

The global context graph is constructed to learn the relation between each entity (e.g., human or object) and all other entities in a video. The graph is constructed based on the learned adjacency matrix between humans and objects over time in a video as in [120].Hence, the interaction score between two nodes in GAT is:

$$\alpha_{i,j} = \sigma(a[W_o(x_i)|W_o(x_j)]) \tag{4.1}$$

where W_o is a learnable transformation which is shared between object nodes in a video. a is a weight matrix projecting the concatenated features to a scalar that reflects attention coefficient between two nodes (e.g., humans or objects). "|" indicates concatenation. In this global context graph, coefficients represent the learned interaction scores between humans and objects. In other words, $\alpha_{i,j}$ is a scalar that represents the relation between two nodes i and j (e.g. edge) in the adjacency matrix A, which is of the size $N \times N$ where N is the number of humans and objects that appeared in the video. σ is a nonlinearity function such as LeakyReLU. Later, $\alpha_{i,j}$ is normalized across all other nodes within the video with respect to node i via softmax. Thus, the updated node features via GAT can be formulated as:

$$x_i = \sum_{j \in N} \alpha_{i,j} W_o x_j \tag{4.2}$$

Through this graph, long-term dependency of HOIs in a video can be captured since each object is attended to all other objects over the video at different time frames.

On the other hand, in the local context, there are T number of graphs, where T indicates the number of frames in the video. Through these local graphs, besides relations induced by closely located humans and objects, non-local dependency relations between human and objects in a video frame can also be captured. Non-local means when objects and humans are distant from each other within a frame. Hence,

each node captures local contextual information via learning relation with other nodes (e.g., human or objects) within the same frame regardless they are spatially close to or distant from each other. Local context is therefore learned from various interactions in which humans / objects attend to others in the same frame.

In short, the way of updating graph nodes is the same in both global and local graphs using Eq. 2, yet the nodes relation scope is different. In global graph, each graph node attends (learns relation) to all other nodes in the video. In contrast, in local graph, only relations between nodes at the same frame is learned. Hence, the local and global contexts use the same operation (e.g., GAT) but consider different structures. Through these graphs, the relations between humans and objects can be learned even though they are not nearby in space and time. Hence, various human-object, object-object and human-human relations within individual frames and throughout a video can be extensively learned.

4.2.3 Global and Local Context Distillation

In order to have an informative representation of HOIs, features from both global and local contextual views should be fully utilized. This may not be simply done by combining features from the two contexts, despite it is a standard way for gathering information from different sources or views. In contrast, we adapt a teacher-student framework to utilize global and local context of HOIs through knowledge distillation. To implement such a knowledge transfer, we incorporate soft labels from the teacher context graph network to guide the student context graph network during training. These soft targets are probability distributions from the logits in the teacher network.

In our experiments, different distillation losses are utilized, depending on the nature of a dataset. For CAD-120 dataset, we minimize the KL divergence between soften labels of teacher and student as in [172, 175]. For Charades, we use l_2 loss as distillation loss to meet the property of training multi-label videos. Hence, the l_2

distillation loss can be formulated as [176]:

$$L_{Distill} = \frac{1}{n}\sum_{i=1}^{n}(P(t)_i - P(s)_i)$$
$$P(s)_i = \frac{1}{1 + e^{\frac{l_c}{T}}} \tag{4.3}$$

where $P(t)_i$ and $P(s)_i$ are softened sigmoid predictions from teacher and student networks, respectively. l_c is the logit from the last fully connected layer in the network, and T is a hyper-parameter that represents the temperature for class c [176].

4.2.4 Training

We first train teacher network, which captures one contextual view (e.g., global context) of HOIs along with hard labels, using cross-entropy loss. We then fix the teacher network and train the student network which is another contextual view of HOIs (e.g., local context). Hence, the objective function for training the student network can be:

$$L_{student} = \lambda_1 L_{CE} + \lambda_2 L_{Distill} \tag{4.4}$$

where L_{CE} is cross-entropy loss between student predictions and hard labels (e.g., ground truth). λ_1 and λ_2 are hyper-parameters for balancing the two losses and are set empirically (see Section 4.3.4). For testing, the results is reported using only the student network.

4.3 Experiments

4.3.1 Datasets and Settings

Datasets. We conduct extensive experiments on two public datasets, including Charades [1] and CAD-120 [2]. We particularly choose these datasets not only because they are used for evaluating action recognition models but also because they have a variety of human object interactions where this research focuses on. We demonstrate the flexibility and capability of modelling human interactions via our

Figure 4.2: Examples of HOIs from Charades dataset [1].

proposed model by considering large-scale and small datasets as well as diverse 2D and 3D backbones using only the RGB mode. In this chapter, we focus on indoor and daily human interactions that are presented in Charades and CAD-120 datasets. As such, we have not used UCF101-20 dataset [10] as we did in Chapter 3, since that dataset includes some outdoor and sporty interactions, such as soccer juggling.

Evaluation Metric. Since Charades dataset is a multi-label video dataset, we use mean average precision to report the final results. In contrast, each video in CAD-120 [2] has only one activity label. Thus, accuracy is adopted as the evaluation metric as in [89].

4.3.2 Implementation Details

For implementation, we use Pytorch deep learning framework. Below are implementation details for the Charades and CAD-120 datasets.

Charades: For extracting the bounding boxes for video frames in Charades, we use RPN which is pre-trained on MSCOCO dataset [161] with ResNet-50-FPN backbone. We use the top 15 proposals at each frame. These proposals are class-agnostic where the top proposals mean that the abjectness score is high (e.g., the confidence score that a bounding box contains an object). Since person is a class in MSCOCO dataset [161], the RPN can generate proposals that contain humans. These proposal features (bounding boxes) represent human and object nodes in the graphs.

For training our GLIDN, we follow training procedure in [120] and we use Inflated 3D ConvNet (I3D) model [50] with Resnet-50 and Slowfast-R50 [100] as our backbone networks. We sample 32 and 64 frames as in [100] and [120]) from each video as input with 224×224 pixels for I3D and Slowfast-R50, respectively. The inputs are randomly cropped such that the shorter side is sampled in [256, 320] pixels.for both I3D and Slowfast backbones, humans and objects are projected from 16 frames (the temporal dimension) where in the I3D one frame is sampled from two frames as in [120]. In Slowfast, the temporal stide is 4 where from 64 frames we sample 16 frames.

For I3D backbone, we initialize it with pretrained parameters on Kinetics-400 dataset [69] from [177]. Table 4.1 depicts the backbone configuration which is adapted from [120]. The output of the backbone, which is used for extracting the features of human and objects, is of size of 16×14×14×2048 dimensions, where the first dimension is the temporal dimension and 14×14 are the spatial dimension. The last dimension indicates channels. As in [120], we add 1×1 convolution layer on the top of the backbone to reduce channel dimension to 512.

For Slowfast-R50 backbone, we adopt it from [177] where it is already trained on Charades dataset.For Slowfast backbone, we extract human and objects features from the Slow path where the feature map size is 16×16×16×2048. Similar to I3D, we add 1×1 convolution layer to have 512 channels.

Layer	Configuration	Output size
input frames	-	32×224×224
$conv_1$	5×7×7, 64, stride 1, 2, 2	32×112×112
$pool_1$	$1 \times 3 \times 3$, max, stride 1, 2, 2	32×56×56
res_2	$\begin{bmatrix} 3 \times 1 \times 1, 64 \\ 1 \times 3 \times 3, 64 \\ 1 \times 1 \times 1, 256 \end{bmatrix} \times 3$	$32 \times 56 \times 56$
$pool_2$	$3 \times 1 \times 1$, max, stride 2, 1, 1	16×56×56
res_3	$\begin{bmatrix} 3 \times 1 \times 1, 128 \\ 1 \times 3 \times 3, 128 \\ 1 \times 1 \times 1, 512 \end{bmatrix} \times 4$	$16 \times 28 \times 28$
res_4	$\begin{bmatrix} 3 \times 1 \times 1, 256 \\ 1 \times 3 \times 3, 256 \\ 1 \times 1 \times 1, 1024 \end{bmatrix} \times 6$	$16 \times 14 \times 14$
res_5	$\begin{bmatrix} 3 \times 1 \times 1, 512 \\ 1 \times 3 \times 3, 512 \\ 1 \times 1 \times 1, 2048 \end{bmatrix} \times 3$	$16 \times 14 \times 14$
	Global Avg , FC	1×1×1

Table 4.1: ResNet-50 I3D Backbone that is used in our model.

Table 4.2: Configuration of our global and local contextual views for CAD-120 Dataset [2]. B indicates batch size. 30 is the number of frames that we use to extract human objects features and 6 is the maximum number of human and objects at each frame.

Model	**Output size**
Global context (GC)	B×30*6 × 2048
Average over nodes (GC)	B× 2048
Local context (LC)	B*30×6 × 2048
Reshape (LC)	B×30×6 × 2048
Average over nodes (LC)	B×30× 2048
Average over T (LC)	B× 2048

As in [120], we apply RoIAlign [173] on the output feature maps of the backbones (before the FC) and each node in the graph is with a fixed dimension of 7×7×512 (1×1×512 via max pooling).

Moreover, similar to [120], in I3D backbone, we concatenate the backbone features after performing average pooling with a contextual view features (e.g., after pooling over N nodes). The concatenated features are then fed to a fully connected layer (FC). Similarly, in Slowfast we concatenate both Fast and Slow paths with the pooled nodes features. Thus, the final feature to represent a contextual view is a concatenation of 256 (Fast), 512 (Slow) and 512 (a contextual view's nodes) features after pooling. A single Graph attention (GAT) layer is used in student and teacher networks. Table 4.3 shows the details of graphs in teacher and student contextual views.

We train I3D backbone for 60 epochs with a batch size of 8 videos, where the learning rate is set to 0.018 for the first 40 epochs and is reduced by a factor of 10 for the last 20 epochs. Following the previous works including [120, 125, 167], we use stage-wise training strategy where the model is trained end-to-end in the second stage for 30 epochs.

We adapt binary cross-entropy with sigmoid activation as a loss function for multi-label video classification in addition to the distillation loss.

For inference, we perform multi-crop-view inference on each video. In other word, we sample 10 clips from each videos and perform multi-crop testing as in [125]. Later, the result is reported based on fusing scores from 30 views via max pooling.

Table 4.3: Configuration of our global and local contextual views for charades Dataset [1]. B indicates batch size. 16 is the number of frames that we use to extract human objects features and 15 is the number of human object proposals at each frame.

Model	**Output size**
Global context (GC)	B×16*15 × 512
Average over nodes (GC)	B× 512
Local context (LC)	B*16×15 × 512
Reshape (LC)	B×16×15 × 512
Average over nodes (LC)	B×16× 512
Average over T (LC)	B× 512

CAD-120: We sample 30 frames uniformally from each video and we used the bounding box annotations that are provided within the dataset. Also, as in [164], we use the ResNet-50 [48] pre-trained model on ImagNet [178] to extract humans and objects features without fine-tuning because CAD-120 dataset has a small number of videos. For each bounding box in a frame, we apply RoI cropping and then reshape it to meet the input size of 224×224×3 for 2D ResNet backbone. Therefore, human and object node features are with the size of 2048 dimension that are produced by ResNet-50.

Moreover, we only use the contextual view features (e.g., without concatenating with the ResNet-50 features) for recognizing actions. For teacher and student networks, we used three graph attention layers in each of the networks. Details of the last Graph layer is presented in Table 4.2. For the first and second layer, the output size is of B×30*6 × 2048 and B*30×6 × 2048 for global (GC) and local contextual (LC) views, respectively.

Table 4.4: A summary of training settings in our experiments on CAD-120 [2] and Charades [1].

	CAD-120 [2]	**Charades [1]**
Optimizer	Adam	SGD
LR	2.e-5	0.018
Epochs	100	60,30
Decay	each 50 steps	each 40 steps
# of GAT Layers	3	1
Training procedure	LOSO-CV	Stage-Wise

Besides distillation loss, we train our model with cross-entropy loss with an initial learning rate of 2.e-5. We train our model for 100 epochs in total using Adam optimizer [163]. Also, Leave-One-Subject Out Cross-Validation (LOSOCV) is used for training the proposed network. Hyper-parameters for our training are summarized in Table 4.4.

4.3.3 Comparison with State-of-the-Arts

As shown in Table 4.6 and Table 4.5, we compare our GLIDN with all prior methods that applied on CAD-120 and Charades datasets, respectively. Our approach achieves the best performance.

Charades: It is noted that on Charades, our network outperforms the baselines including I3D and Slowfast, which do not consider spatio-temporal contextual views of objects. Our network also performs better than STRG [120], which has used spatio-temporal object relations. Although our global context graph is the same as in STRG [120] in term of the temporal range of objects and human, there are three main differences. First, we use graph attention instead of graph convolution network that used in STRG. Second, in our model, we consider this graph as a teacher or a student network whereas in STRG it is just a graph that is combined with another non-learnable "spatio-tempral graph". Third, we explore knowledge distillation for capturing more HOI contextual cues, while the work in STRG follows the common method for training their model (e.g., binary cross-entropy loss only). This implies that our approach of using different contextual views of object relations via distillation can help the model generalize better in identifying different types of interactions. Thus, our method has achieved better results even with much fewer number of proposals, as shown in Table 4.8.

Notably, our approach of utilizing two different contextual views of HOIs and their knowledge transfer can offer more informative cues about interaction even without any human-object abstract information (e.g., the union of both objects) as in [125]. This indicates the importance of context modeling of humans and objects without the need of additional information (e.g., visual phrases).

Moreover, our choice of graph attention network for learning human-object rela-

Table 4.5: Classification mAP (%) results on the Charades dataset [1].

Model	**Backbone**	**mAP(%)**
2-Stream [179]	VGG-16	18.6
2-Stream +LSTM [179]	VGG-16	17.8
Async-TF [179]	VGG-16	22.4
a Multiscale TRN [96]	Inception	25.2
I3D [50]	Inception	32.9
I3D [120]	R50-I3D	31.8
STRG [120]	R50-I3D	36.2
STAG [125]	R50-I3D	37.2
Pose and Joint-Aware [180]	R50-I3D	32.81
GLIDN (ours)	R50-I3D	**37.51**
LFB Max [151]	R50-I3D-NL	38.6
Slowfast 16 x 8 [100]	R50-3D	38.9
Slowfast 16 x 8+GLIDN **(ours)**	R50-3D	**41.00**

tions in both global and local contextual views is important since we have achieved 35.35 comparing to 34.2 in [120] for the global context with fewer number of nodes. Consequently, we have achieved the best results on Charades comparing to prior works that use the same backbone networks.

CAD-120: We have also achieved better results on the CAD-120 [2] than other works that use temporal sampling and 3D CNN [15,89] without fine tuning and with the use of object features extracted from 2D backbone. This implies our KD from different contextual views can remarkably contribute to HOIs reasoning, as it can better capture long-term temporal structure of interactions. Although GLIDN captures contextual information about HOIs, it does not include learning the temporal ordering of HOIs. This aspect is presented in the proposed approach in Chapter 3, which utilizes LSTMs in a hierarchical design, resulting in better results with an accuracy of 94.35%.

The confusion matrix in Figure 5.3 studies how well our method can predict actions correctly based on CAD-120. It can be observed that most false predicted actions relate to stacking and unstacking objects or some actions alike. Such actions usually involve the same object but being different in human movement directions. This may be resolved by capturing more temporal information, such as increasing the number of sampled frames.

Table 4.6: Accuracy (%) results on the CAD-120 dataset [2]. '*' indicates that prior works make use of additional skeleton or depth information and thus are not directly comparable to our approach.

Model	**Acc.(%)**
Wang et al. [15]	81.2
*Liu et al. [16]	93.3
*koppula et al. [2]	80.6
*Tayyub et al. [17]	95.2
Sanou et al. [89]	86.4
Ch3's proposed method	**94.35**
GLIDN **(ours)**	**92.85**

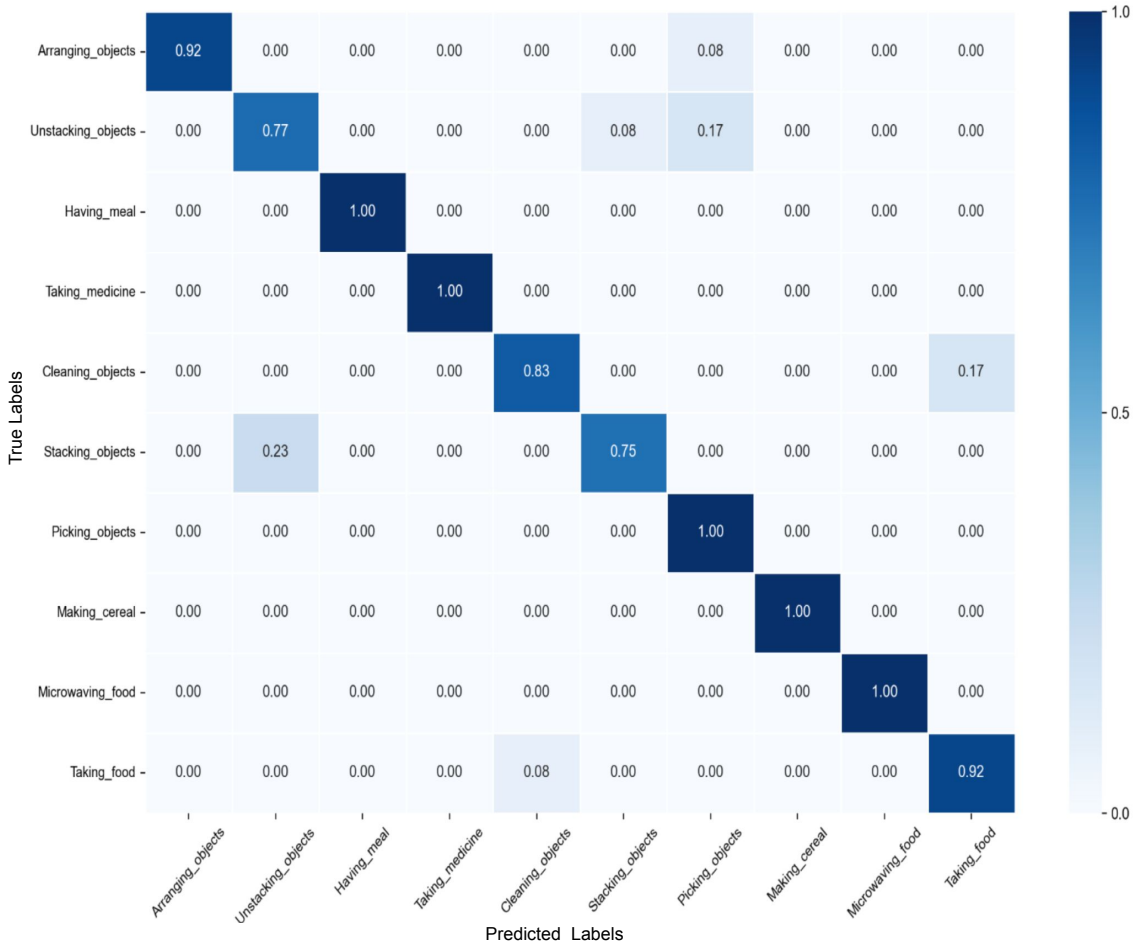


Figure 4.3: Confusion matrix for the CAD-120 dataset [2] when using our proposed GLIDN.

Table 4.7: Ablation results the CAD-120 [2] and Charades [1] datasets. Results from two different backbones are reported on Charades [1].

Model	**Charades [1] (Slowfast)**	**Charades [1] (I3D)**	**CAD-120 [2] (2D R-50)**
Baseline	38.9	34.23	74.17
Local-context (spatial)	40.73	36.45	84.97
Global-context (temporal)	39.95	35.39	84.75
Context views fusion (e.g. Concat)	40.43	36.81	85.22
Late Fusion	40.95	37.23	85.97
Local-teacher	39.89	**37.51**	87.76
Global-teacher	**41.00**	36.99	**92.85**

Table 4.8: Comparison of graph node settings with prior works on Charades [1]. 'Edges' means the union box of two object nodes.

Model	**# of nodes**	**Nodes info.**	**mAP(%)**
STRG [120]	50	objects	36.20
STRG [120]	25	objects	35.9
STAG [125]	15	objects and edges*	37.20
GLIDN **(ours)**	15	objects	**37.51**

4.3.4 Ablation Studies

To evaluate our proposed GLIDN, we conduct ablation studies to demonstrate the impact of each part of our GLIDN on learning HOIs. We first evaluate the baseline without any of interaction contextual views. We then evaluate our network by using each of the contextual views independently. Finally, we report the performance of our complete network. The ablation study results are shown in Table 4.7 for Charades [1] and CAD-120 datasets [2].

Are contextual views of humans and objects important? As shown in Table 4.7, running our network without any human-object relations or with only a single contextual view (either local or global view) degrades the network performance. Clearly, when we consider only human and object information (e.g., via concatenation) without learning their relation, the performance of the network decreases significantly by 14% in CAD-120 [2].

Also, when considering only human-object temporal relations on Charades [1],

the performance drops more than 1% mAP, which reflects the importance of local relations between human and objects at a specific time as they can provide useful context information. This indicates that some of the interactions can be recognized by focusing on the spatial relation, especially with the existence of multiple objects around a human. Finally, capturing both the global and local human-object relations via distillation can help transfer the complementary information from the teacher contextual view to the student contextual view. Hence, the ablation experiments illustrate that each component of the proposed GLIDN plays towards improving the model performance, where 41.00% mAP is achieved on Charades.

Which of the contextual views play the roles of the teacher network? In the original KD, the teacher network is larger than the student network. In contrast, in our proposed GLIDN, both student and teacher networks give informative cues about interactions from different contextual views. We hence conduct comprehensive experiments to decide which of the contextual view can better serve the teacher role. Logically, when we take into account the wide range of information provided by the global context, we can consider it as a larger contextual view for HOIs since each human/object learns a relation with all other humans/objects throughout all video frames, while the local context only provides information about how humans/objects attend the others within each individual frame. This idea is evaluated on Charades [1] and CAD-120 datasets [2]. As shown in Table 4.7, best results are usually achieved when we consider the global contextual view as the teacher. Hence, we can conclude that the temporal View (e.g., global contextual view of human-object interactions) is mostly a viable candidate for the teacher.

However, we notice that utilizing different backbones on the same dataset as in Charades [1], leading to different selection of teacher network. This suggests that the features retrieved from different backbones have an impact on determining which of the contextual views play a better role as the teacher. For instance, when training our method with I3D backbone on the Charades dataset [1], we find that using the local contextual view as a teacher achieves better performance. The reason behind this is that the final representation in Slowfast experiments involves concatenating objects relations with fast path features, which are from 64 frames. This means that

Table 4.9: Accuracy results on CAD-120 dataset [2] after applying different values of T (temperature) and λ_2 (weight of the distillation loss).

T	λ_2	**Global-teacher(%)**	**Local-teacher(%)**
2	4	87.56	84.36
1	0.7	**92.85**	86.00
5	0.3	88.36	**87.76**
10	0.3	88.45	83.53
20	0.3	87.62	83.50
5	0.5	84.33	86.84
10	0.5	85.69	84.25
20	0.5	87.47	83.59
5	0.7	86.84	81.89
10	0.7	88.54	82.61
20	0.7	85.27	86.00

Slowfast backbone is richer in temporal information than the I3D backbone, which only uses 32-frame features. Hence, when the temporal range is not large enough to capture better contextual information, especially in clutter background videos as in Charades [1], the spatial local context teacher may outperform the temporal global one. Our findings indicate that distilling the knowledge of interactions between the global and local contextual views outperforms other counterpart approaches in both scenarios, whether a teacher is taking a local or global contextual view.

There are other factors controlling the distillation process, namely the hyper-parameters of T (temperature), λ_1 and λ_2 (weights for balancing the losses in Eq. 4.5). We conduct comprehensive experiments in both CAD-120 [2] and Charades [1] using different values of these hyper-parameters. Two forms of λ settings are used for balancing the weight between the two terms of the objective function as in Eq. 4.5. In the first form of setting, we used the generalized distillation form as in [73] where λ_1 is equal to (1-λ_2). The second form is by setting λ_1 to 1 and λ_2 to 4 or 0.7 as shown at the first two rows in Table 4.9 which shows the results of applying different hyper-parameters on CAD-120 dataset [2] with different settings for teacher and student.

We observe that the best values of T are different for both global contextual view and local contextual view since each network contextual view produces different probability distribution for the logits. We also find in the global teacher, the

Table 4.10: mAP% results on Charades Dataset [1] using I3D backbone after applying different values of T (temperature) and λ_2 for weighting the distillation loss.

T	λ_2	**Global-teacher(%)**	**Local-teacher(%)**
2	4	36.81	35.68
5	0.3	36.60	37.30
1	4	36.99	**37.51**
10	0.7	35.92	36.83
20	0.7	36.03	36.94

Table 4.11: mAP% results on Charades dataset [1] using Slowfast backbone after applying different values of T (temperature) and λ_2 for weighting the distillation loss.

T	λ_2	**Global-teacher(%)**	**Local-teacher(%)**
2	4	**41.00**	39.89
5	0.3	40.82	39.63
10	0.7	40.70	38.86
20	0.7	40.75	38.72

temperature of 1 achieves the best accuracy as in [181] when the weight λ_2 is equal to 0.7. Moreover, when we consider local contextual view as the teacher network, we observe that a large value of T (e.g., 5) with a distillation weight of 0.3 produces the best result of 87.76%. Hence, the optimal values of T and λ can be set empirically based on the predictions of the teacher network. Table 4.10 and Table 4.11 present results after applying different hyper-parameters on Charades dataset using I3D and Slowfast backbones, respectively.

Is teacher-student network design a good choice for distilling object contexts? In order to evaluate our teacher-student network design, we compare it with other collaborative learning approaches, such as Deep Mutual Learning (DML) [182], where the two contexts views are jointly trained. As presented in Table 4.12,

Table 4.12: Comparison between DML and teacher-student networks for distilling knowledge between object contexts on CAD-120 Dataset [2].

Model	**Acc.(%)**
DML (local)	87.73
DML (global)	86.64
our GLIDN (Global-teacher)	**92.85**

we can observed that our teacher-student network achieves a better result of 92.85% with an large increase of 6.21% when we consider the teacher network as the global context of HOIs, while 86.64% is achieved via DML. This is because the teacher-student network approach allows the use of contextual information from the teacher network guiding the student network to capture much structural knowledge about HOIs.

Is context distillation better than conventional fusion? In order to compare our proposed context distillation for recognizing HOIs with standard methods for combining the features and capturing complementary cues from the two contextual views, we conduct two experiments including early fusion and late fusing methods. In early fusion, we concatenate the features from the two contextual views, then fed them to a classifier. In contrast, for late fusion, we average the predictions of the contextual views. As in Table 4.7, we can observe that our model captures better cues of interactions, whereas in the early fusion, some noise in features may affect the network performance. Moreover, as stated in [172] that knowledge distillation can be considered as a late fusion method, we may confirm this statement in Charades dataset [1] where the model performance via a late fusion is similar to knowledge distillation with only the 0.05 % and 0.3% improvement. Although, the late fusion and knowledge distillation results in Charades [1] are close, the results of both approaches outperform the baseline and single view context, proving our claim of exploiting the context of human object interactions from two different views. On the other hand, we find that distilling knowledge between HOI contexts outperforms the late fusion on CAD-120 [2]. The late fusion model achieved an accuracy of 85.97%, but when knowledge distillation is applied with the same training setting, the performance is improved by 6.88%. This supports our claim that knowledge distillation can be used to capture the context of HOIs from many contextual views.

4.3.5 Evaluation Examples

Figure 4.4 and Figure 4.5 show two video examples from CAD-120 Dataset [2]. We found from the examples that inconsistent recognition results may be come up if

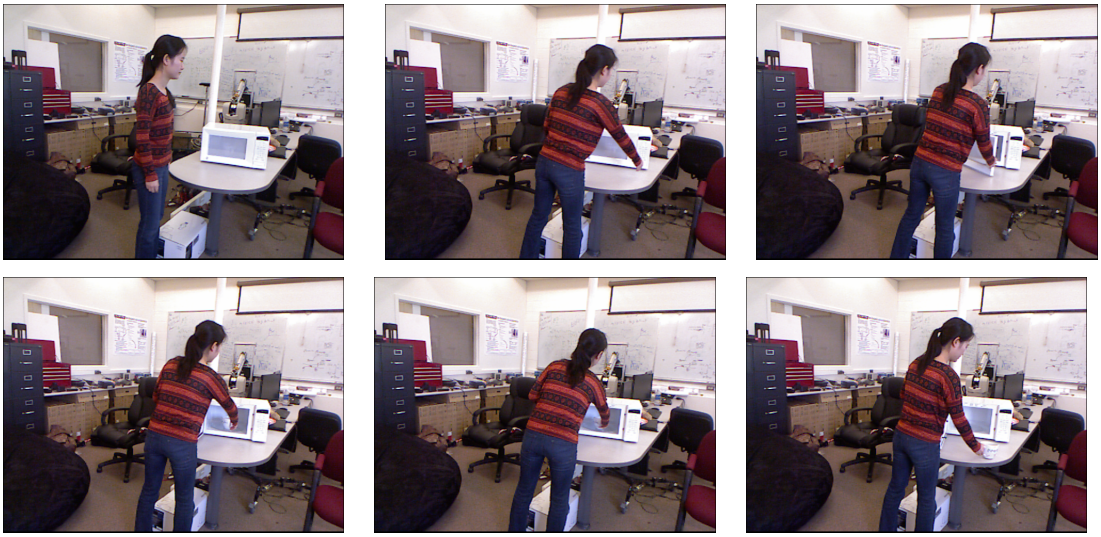


Figure 4.4: Example frames from video ID:0510180218 [2].

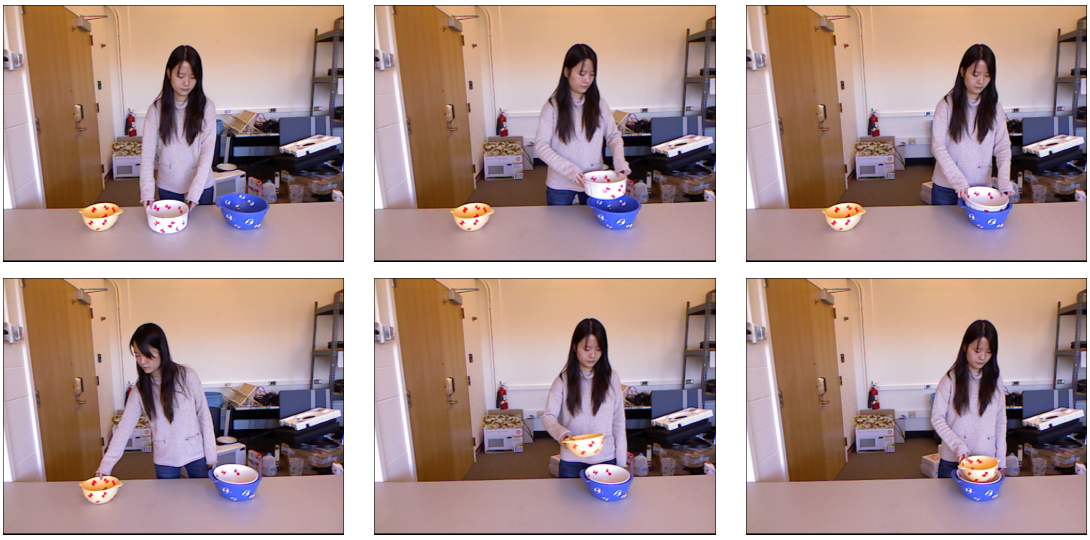


Figure 4.5: Example frames from video ID:1204144736 [2].

only one contextual view is applied. Since the context of HOIs varies, it is difficult to determine which contextual view is more effective.

For example, in Figure 4.4, the video is with the correct label "taking food" and it is mis-classified as "arranging objects" using only the local contextual view. However, the video is recognized correctly by using the global contextual view where temporal interactions between human and objects are learned at the video-level. This indicates the importance of observing the change of object and human status over time, which is captured via the global context.

Another example is shown in Figure 4.5, where the correct label of the video

is "stacking objects" but it is mis-classified as "unstacking objects" when using the global contextual view only. However, the video can be correctly classified using the spatial contextual view. This illustrates the importance of having specific time human-object relations, which provides some structure information about an interaction, via its local contextual view.

Notably, our GLIDN model classifies both videos correctly when we consider the global context view (e.g., temporal) as a teacher. This implies that distilling local and global contextual information increases the generalizability of the model. We have achieved the best results of 92.85% on CAD-120 dataset [2].

4.4 Exploring the design of the teacher network

We now investigate alternative designs for distilling human and object (H-O) contexts between different views.

We explore the way of extracting H-O contexts from multi-teacher settings. In this design, the spatial graph at each frame acts as a teacher. These frame-based teachers are trained with shared parameters. Teachers in this situation learn human and object spatial relationships in a frame and generate knowledge (e.g., logits) at the frame-level. In contrast, our GLIDN only has one teacher, which generates predictions based on a video's frame relations. Hence, GLIN teacher performs a video-level prediction. We train the student network, which is the global graph with many teachers from various frames that consider the local relations between human and objects. Hence, the knowledge is distilled from multiple spatial views teachers. The corresponding loss used in training the student can be written as:

$$L_{student} = \lambda_1 L_{CE} + \lambda_2 (\frac{1}{N} \sum_{n=1}^{N} L_{Distill_{(S,T^n)}}) \tag{4.5}$$

where N is the number of teachers that participate in the student network's training. Table 4.13 shows the outcomes of utilizing several instructors by using different samples of frames as teachers (e.g., 30 or 15 frames), while the student network remains the same in both situations (e.g., 30 frames).

Table 4.13: Accuracy results on CAD-120 dataset [2] after applying different designs of teachers. S indicates student network. In the last two rows, student network is trained with 15 and 30 frames, respectively.

Model	**Acc.(%)**
Spatial-Multi-teacher (30 frames)	86.30
Spatial-Multi-teacher (15 frames)	83.49
Spatial-single teacher as in our GLIDN	87.76
Temporal-teacher our GLIDN (S 15 frames)	87.56
Temporal-teacher our GLIDN (S 30 frames)	92.57

As can be observed from the results that considering the spatial relations based on single teacher (e.g., video-level) produces better knowledge, which can be distilled to the student. Also, the temporal teacher outperforms the spatial teachers in both single-teacher and multiple-teacher settings. Furthermore, even with fewer frames, the temporal teacher can still lead the spatial student.

4.5 Conclusion

The context of HOIs gives crucial cues about how human interacts with different objects. Our GLIDN, a novel human objects interaction distillation network, explicitly uses two different views of humans and objects context to capture their interactions at specific time and throughout a video. We also propose context knowledge distillation to transfer knowledge from the teacher contextual view of HOIs to the student network that has information from different context of such interactions. Extensive experiments demonstrate that we outperforms prior works on two datasets including Charades [1] and CAD-120 [2].

While the context of HOI is crucial for understanding interactions by distilling the knowledge across different views, training teacher and student networks can be time-consuming. Moreover, the representation of human-object relations in these contextual views does not consider the hierarchical nature of interactions. Thus, in the following chapter, we explore several architectures to represent human-object interactions hierarchically using transformers, drawing inspiration from the success of the most recent transformers in various computer vision tasks.

CHAPTER 5

Spatio-Temporal Interaction Transformers for Human-Object Interaction Recognition in Videos

5.1 Introduction

Action recognition models perform better when spatio-temporal contexts between two views are distilled for learning human-object interactions, as explored in the previous chapter. Despite this benefit, these contexts do not consider the representation of an interaction that is naturally performed by humans in a hierarchical way. In this hierarchy, the human-object relations at time t should be taken into account before the temporal changes of these relations are learned at a higher level. We extensively explore this concept in this chapter.

Some human-object interactions (HOIs) are difficult to recognize, such as when a human is cleaning an oven or taking food from it. In these cases, the oven can afford different interactions including open, clean, and close. Additionally, the presence of various objects in the scene at the same time can also affect model learning.

Early action recognition models, such as ConvNet [12, 100], recurrent neural networks (RNNs) [11, 166], and 3D convolution models [50, 102], learn a global representation of actions without considering human-object interactions. However, con-

textual information about an interaction, including human-object and object-object relationships, is critical and discriminative at specific times and throughout a video.

Moreover, recent work has explored graph-based techniques for action recognition in videos [120, 124–126, 167], using spatio-temporal graphs to learn object and human relations. Transformers [8, 9] have also been used to learn spatio-temporal relations in videos, e.g., [152] focuses on object layout relationships with a global video representation, and [153] considers spatial and semantic embeddings of objects. However, hierarchical spatio-temporal relations for HOI recognition remain unexplored.

Given that discriminative cues about an interaction can be intensive at specific moments across video frames [125], we propose to learn interactions in a hierarchical manner. Inspired by Transformers in vision tasks, such as image classification [9], we use them to form our spatial and temporal learning network. The relationship between humans and objects is learned through the spatial transformer, revealing local context even in cases where objects are not close to humans within a frame. Subsequently, the long-term temporal dependencies between interactions across different frames are captured by the temporal transformer, which receives compact representations of interactions at each frame across a video.

Unlike other works, such as [14], where different transformer-based architectures were proposed for video classification, the use of hierarchical structures in modeling human and object interactions through transformers is investigated in this study. To our best knowledge, we are the first to study hierarchical modeling in human-object interactions with transformers based solely on visual appearance features. In this chapter, the main contributions are as follows:

- Developing a novel transformer-based framework to learn spatio-temporal interrelations between humans and objects in videos, which captures both long-term and non-local dependencies in HOIs across video frames.

- Investigating how different hierarchical organizations in network design impact HOI learning.

- Evaluating our model on three datasets, namely Charades [1], Something-

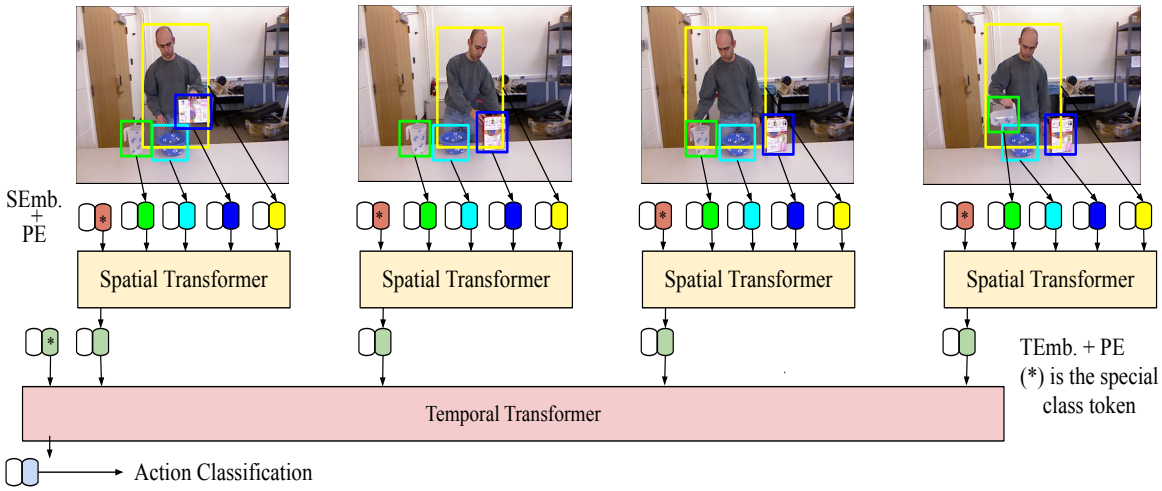


Figure 5.1: Our proposed spatio-temporal transformer (STIT) model. SEmb. and TEmb. stand for spatial and temporal token embeddings, respectively.

Something v1 [3] and CAD-120 [2]. STIT is flexible in adapting any backbone without end-to-end training. All counterpart approaches were outperformed, and state-of-the-art results were achieved on the CAD-120 dataset [2], with an accuracy of 95.93% using RGB data only.

5.2 Methodology

5.2.1 Network Overview of STIT

The overall architecture of STIT is presented in Fig. 5.1.The inputs to the network consist of the extracted human and object region features from a backbone feature map via RoIAlign [173]. These features can serve as tokens for spatial transformer encoders, without the need for dividing each frame into N patches as in [9, 14]. A transformer encoder [8, 9] is primarily made up of numerous layers, each of which contains multi-head self-attentions (MHA), as well as feedforward layers (MLPs). More details regarding transformer architecture, including ViT, are provided earlier in Subsection 2.1.1.

5.2.2 Spatio-Temporal Transformers in STIT

As ViT [9] is flexible in learning token relations, we adapt it forming our spatial and temporal transformer encoders. In spatial encoder, local and non-local dependency relations between human and objects (e.g., tokens) in a frame can be captured. Non-local implies that there are no limitations on the specific distances and positions between objects and humans. It includes all humans and objects that are appeared at time t regardless of where they are located. Spatial-level interactions imply capturing local contextual information where human and object relations at the same time step are learned. This can be done through multi-head attention in the spatial transformer layer where all pairwise interactions between tokens (e.g., humans/objects) in a frame are captured. Hence, each token representation will be refined with respect to all other object tokens appeared at the same moment via self-attention, which effectively captures each object context. Since we adapt ViT, we prepend a learnable class token to objects at each time step, which is proven by ViT that generates a compact representation for an image. Our STIT considers it as a representation for local context at each time step. The input of a spatial transformer at time t is human and objects that are embedded via linear projection to generate tokens of 1D dimension with the size of 2048 each. As in [9, 183], 1D positional encodings are learned throughout the training procedure. These encodings are vectors with the same size as the tokens (e.g., 2048), initialized randomly, and then added to the tokens. Subsequently, the model learns to effectively use them to retain positional information of the tokens. We can write the input to the spatial transformer at time t as:

$$X_t = [\texttt{class}_t, \psi(h_t^1), \psi(o_t^1), \psi(o_t^2), ..., \psi(o_t^N)] + P_t \tag{5.1}$$

$$z_t = \text{Spatial-Transformer}_t(X_t) \tag{5.2}$$

where ψ stands for linear embeddings, h_t^i and o_t^i respectively represent a human and an object visual feature at time t, and N is the number of objects. P_t indicates the learned positional embedding. $\texttt{class}_t$ is an extra token that is prepended to tokens at each time step t. This class token is randomly initialized and via spatial

transformer layers, the token is attended and gathered information from all other tokens in a frame at time t. z_t is the updated version of $\texttt{class}_t$, and is the output of the spatial transformer at time t. Thus, z_t represents the local context of interactions at time t.

To capture long-term HOI dependency, we add a second-level transformer for modeling temporal HOI evolution. The input tokens of the temporal transformer encoder are the updated class tokens outputted from the spatial transformers, that retain an abstract representation of interactions at each frame. The input to temporal transformer is then:

$$H = [\texttt{class}_{video}, \phi(z_1), \phi(z_2), \phi(z_3), ..., \phi(z_T)] + P_I \tag{5.3}$$

$$Y_{interaction} = \text{Temporal-Transformer}(H) \tag{5.4}$$

where ϕ is a linear transformation. Similar to spatial transformers, we prepended new class token to the token sequence which is $\texttt{class}_{video}$ in this level. z_i is the latent token generated by spatial transformers at temporal index i and T is the number of frames in a sequence. P_I is the positional encoding that learns and preserves the position of each token in a sequence. In the temporal transformer, the class token is attended to other tokens in the sequence, which are the compact representations of interactions at different time steps. $Y_{interaction}$ is the updated version of $\texttt{class}_{video}$ and output of the temporal transformer, where a high-level hierarchy of interactions is learned, providing discriminative cues of an action.

5.3 Experiments

We validate STIT on CAD-120 [2], Charades [1] and Something-Something v1 (SSv1) [3] datasets. The following three sections provide details of the experiments conducted on these datasets.

Table 5.1: A summary of training settings for our STIT model on CAD-120 [2] and Charades [1].

	CAD-120 [2]	**Charades [1]**
Optimizer	Adam	SGD
Learning rate (LR)	2.e-6	0.018
Epochs	100	60,30
Decay	each 50 steps	each 40 steps
Training procedure	LOSO-CV	Stage-Wise

5.3.1 Experiments on the CAD-120 Dataset

Training details

To train STIT, we take 30 evenly sampled frames from each video. To extract human and objects features, we follow [164] where the region of interest (RoIs) that indicate the bounding boxes of human and objects are cropped and reshaped to 224×224×3 for meeting the input size of 2D ResNet backbone [48]. Hence, 2048 features for each human and objects are extracted from ResNet-50 [48]. These human and object features are used as initial tokens for the proposed STIT model. As in Chapter 4 experiments on the same dataset, we only use the token features (e.g., without concatenating with the frame-level ResNet-50 features) for recognizing actions. Training hyper-parameters are presented in Table 5.1. We employ Leave-One-Out Cross-Validation and cross-entropy loss for training our STIT model. Moreover, we train the spatial transformer with six layers and two heads while the temporal transformer is trained with one layer and two of heads. Table 5.2 presents input and output details for spatial and temporal transformers on CAD-120 [2] dataset.

Table 5.2: Configuration of our spatial and temporal transformers for CAD-120 Dataset [2]. B and CLS indicate batch size and class token, respectively. 30 is the number of frames that we use to extract human objects features and 6 is the number of human and objects at each frame.

Model	**Input size**	**Output size**
Spatial Transformer	B*30×6+ `CLS` × 2048	B*30× 2048
Temporal Transformer	B×30 + `CLS` × 2048	B× 2048

Model Variants and prior works

We analyse four variants of our model, namely `LSTM-Spatial-T`, `LSTM-Pool`, `LSTM-GAT` and `GAT-Temporal-T`, to investigate the importance of our STIT model components, with `T` standing for transformer. `LSTM-Spatial-T` investigates the impact of a temporal transformer on learning temporal dependencies across frames. We replace the temporal transformer with two layers of Long-short Term memory (LSTMs) [58] which can be used in sequence learning of videos [11, 166]. The `LSTM-Pool` and `LSTM-GAT` models investigate the role of the spatial transformer in understanding the spatial context of HOIs. We hence use pooling and Graph Attention Networks (GAT) [66] to replace the spatial transformer. Finally, `GAT-Temporal-T` investigates how the spatial transformer affects the temporal transformer when it is replaced by GAT [66]. An additional model (`NO-Relation`) is trained to ignore the spatial relationship between humans and objects. Instead, their features are aggregated and pooled across time. Table 5.3 shows our STIT model outperforms all other variants.

Table 5.3: Performance of model Variants on CAD-120 [2].

Model	**Acc.(%)**
LSTM-Spatial-T	93.31
LSTM- Pool	90.26
LSTM-GAT,	92.47
GAT-Temporal-T	88.39
NO-Relation	86.69
STIT **(ours)**	**95.93**

We observe a 2.62% drop in accuracy when replacing the the temporal transformer with LSTM, indicating temporal modeling via transformers is superior to LSTMs. We further observe our spatial transformers outperform GAT when they work with either LSTMs or temporal transformer. Finally, the performance of the model degrades by 9.24% when the spatial relations and temporal modeling of HOIs are disregarded.

Figure 5.2 compares HOIs recognition by model variants. In the first example, the spatial transformer gives more discriminative context than other models, successfully identifying the human is having a meal. In the second example, the human

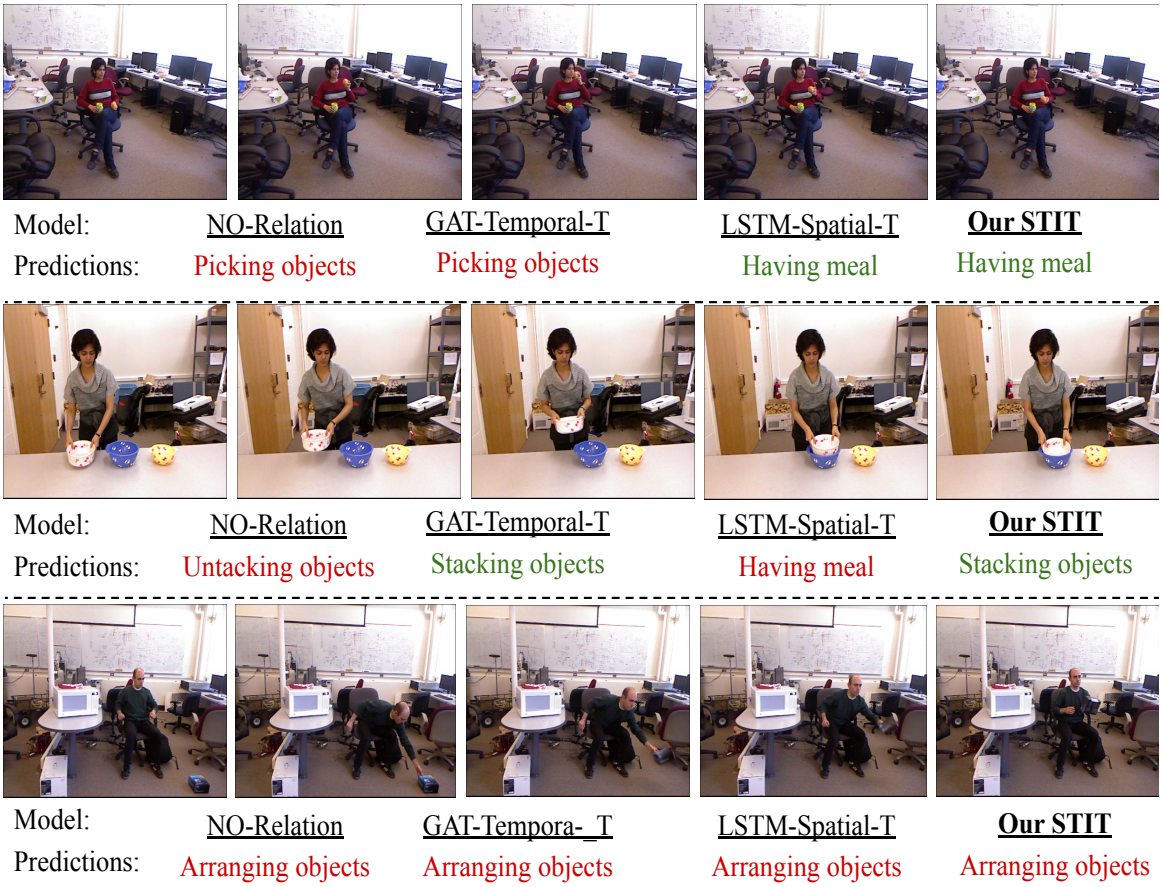


Figure 5.2: Prediction results of some actions by applying four different models on CAD-120 [2].

is stacking objects, which is analogous to the reversed action of "unstacking objects". The spatial transformer cannot understand such an action on its own. Yet our STIT correctly recognizes the action via the temporal transformer. Similarly, GAT with a temporal transformer can also correctly predict the action. This confirms that the temporal modelling via transformer outperforms LSTMs to recognize these type of interactions. Some examples of failure are shown in the last example, where the person is picking an object and all models incorrectly identify it as arranging objects. This may be because that in some videos the arranging and picking action of the same object could be similar but the difference is based on the human pose. We leave this for future work by considering human skeleton information.

Notably, as in Table 5.4, comparing to previous works, which mostly use depth and skeleton data, our STIT model still achieves the best results even with RGB data only. Even [89] has proposed a 3D model to leverages RGB data for action

recognition, our STIT model achieves 2.33% of higher accuracy. This demonstrates the importance of performing HOI reasoning both at each frame and over a course of HOIs. Also, transformer properties, such as multi-head attention and learnable token placements, along with a two-level hierarchy of human-object relation modelling, help our STIT model achieve state-of-the-art accuracy on CAD-120 [2].

Table 5.4: Results with CAD-120 [2]. Note that [15], [16], [2] and [17] have employed additional skeleton or depth information.

Model	**Acc.(%)**
Wang et al. [15]	81.2
*Liu et al. [16]	93.3
*koppula et al. [2]	80.6
*Tayyub et al. [17]	95.2
Sanou et al. [89]	93.6
STIT **(ours)**	**95.93**

Ablation studies

To validate the effectiveness of each component of our STIT model, we conduct two main experiments with STIT-Spatial and STIT-Temporal. In STIT-Spatial, the temporal transformer is replaced by average pooling (e.g., over time dimension) whereas in STIT-Temporal the spatial transformers are replaced by pooling (e.g., pooling over nodes at time t). As shown in Table 5.5, ignoring either spatial or temporal hierarchy leads to decreased model performance. Moreover, a 2.60% performance loss over our STIT model is observed when omitting the temporal transformer, because long term dependencies between HOIs over time is not explicitly modeled. We also notice that model performance decreases significantly by 13.5% when when replacing the spatial transformer with pooling. Because human and objects features are merely extracted from ResNet-50 [48] that is pre-trained on ImageNet [184]. In contrast, embeddings in spatial transformers enhance the token features besides learning the relations between human and objects at each frame, which lead to model accuracy improvement to 95.93%.

As shown in Table 5.5, we conduct additional experiments to explore the affect of using class token as a representation of the spatial context at time t and for the video

Table 5.5: Ablation results on CAD-120 [2].

Model	**Acc.(%)**
Baseline	86.69
STIT-Spatial	93.34
STIT-Temporal	82.43
STIT-spatial-mean	95.13
STIT-Temporal-mean	93.34
STIT-mean	95.04
STIT **(ours)**	**95.93**

which is used as the output of the temporal transformer. STIT-spatial-mean, STIT-Temporal-mean and STIT-mean indicate replacing the output of spatial, temporal and both spatial and temporal transformers in STIT with mean token instead of latent class token, respectively. Notably, using latent token as the output of spatial and temporal transformers leading to better results. The confusion matrix of the prediction results on CAD-120 using STIT model is presented in Fig. 5.3. It can be observed from the confusion matrix that the vast majority of human interactions in the CAD-120 dataset [2] are accurately recognized by the proposed STIT model. Some actions, such as stacking and picking objects, are mistakenly identified as unstacking and arranging objects, respectively. This may be because these activities are performed by a person in a very similar way, and even when arranging objects, the human picks an object to arrange, which shares the same interactions as when the human is mainly picking objects.

5.3.2 Experiments on the Charades Dataset

Implementation details

To train our STIT, we employ two models as our backbones including Inflated 3D ConvNet (I3D) [50] with Resnet-50 and Slowfast-R50 [100]. We initialize I3D with pre-trained parameters on Kinetics-400 dataset [69] from [177]. For Slowfast-R50, we access the model via the Slowfast Github repository [177] where it has previously been trained on Charades. As input, we sample 32 (as in [120]) and 64 (as in [100]) frames from each video clip with 224×224 pixels for I3D and Slowfast-R50,

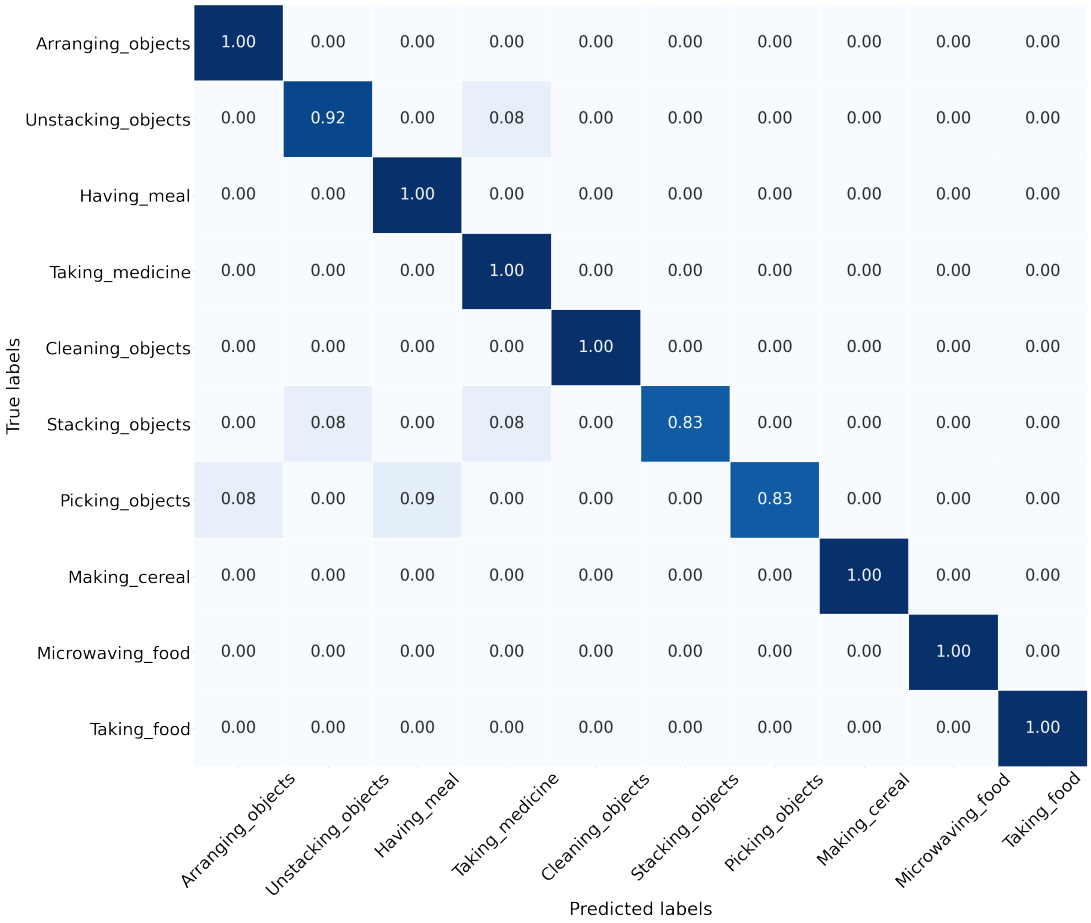


Figure 5.3: Confusion matrix for the CAD-120 [2] when using our STIT model.

respectively. We use a 2-stage training, which is different from [120, 125, 167], where we do not train the backbone and our model together for the third stage as end-to-end. This indicates the flexibility of our model to be integrated to any backbone with fewer number of training stages and with different settings of backbones including the one that is already trained on the same dataset as in Charades or using pretraind model as we used for training our model in CAD-120 [2].

Moreover, similar to [120], in I3D backbone, we concatenate the backbone features with the interaction features (e.g., latent class token from the temporal transformer). The concatenated features are then fed to a fully connected layer (FC). Similarly, in Slowfast we concatenate both Fast and Slow paths with the the interaction features. Thus, the final feature to represent a video is a concatenation of 256 (Fast), 2048 (Slow), 2048 (interaction representation) features. Six heads in 16 transformer layers were used for spatial and temporal transformers. Details of the input and output for the spatial and temporal transformers on the Charades dataset [1] are shown in Table 5.6.

We employ binary cross-entropy loss to train our STIT model with multi-label videos in charades. From each video, we apply multi-view inference where 10 clips are sampled from a video as in [100,120]. The evaluation metric is the mean average precision (mAP) where scores from different views are fused to report the results. The rest of the implementation details are the same as the experiments in Chapter 4.

Table 5.6: Configuration of our spatial and temporal transformers for Charades Dataset [1]. B and `CLS` indicate batch size and class token, respectively. 16 is the number of frames that we use to extract human objects features and 15 is the number of human object proposals at each frame.

Model	**Input size**	**Output size**
Spatial Transformer	B*16×15+ `CLS` × 2048	B*16× 2048
Temporal transformer	B×16 + `CLS` × 2048	B× 2048

Comparison with state-of-the-art approaches

Table 5.7 shows the results of all prior methods that applied on the same dataset. The most close methods are those using the same backbone network as ours. Please note that our proposed method can be applied to any CNN backbone network where RoIAlign [173] is employed to extract human and object features. However, more advanced backbones, such as those based on transformers [185], generate features that are not compatible with RoIAlign [173] for the extraction of human and object features. We defer this investigation for future research. It is observed that considering pose (P) information is not enough for correctly capturing HOIs. This indicates the importance of learning human-object relations in both space and time. Although we utilize fewer number of proposals (e.g., 15), our results are better than [120] where 50 proposals were used. Also, we achieve superior results comparing to STAG [125] that considers relations between a compact interactions, which include visual phrase (e.g., union box of both human and object). This indicates the power of learning the local context of human and objects through spatial transformers even without the visual phrases. Furthermore, learning the relation between visual tokens of human and objects gives more cues rather than considering the layout of human and objects as in STLT+I3D model [152]. Also, the proposed STIT model can be incorporated with any backbone model rather than I3D without end-to-end training. As a result, our STIT model with Slowfast 16 x 8 surpasses its baseline. Thus, the results show that our STIT outperforms all other counterpart approaches which reflects the power of structure learning of HOIs through our two-level hierarchy of transformers.

Ablation studies

We also evaluated the effectiveness of STIT by conducting ablation studies on the Charades dataset [1] using I3D backbone. The purpose was to showcase how each component of STIT contributes to learning HOIs. In order to study the impact of each hierarchy of the model on developing discriminative HOI representation, we carried out similar ablation experiments to the ones in Section 5.3.1, including STIT-Spatial, STIT-Temporal, STIT-spatial-mean, STIT-Temporal-mean and STIT-mean. The results can be seen in Table 5.8. We find that removing the

Table 5.7: Comparison with prior approaches on Charades dataset [1]. Note that slowfast network achieved 45.2%mAP on charades using R101 network but for fair comparison we report Slowfast results with R50 network.

Model	**Backbone**	**Modality**	**mAP**(%)
2-Stream [179]	VGG-16	RGB+Flow	18.6
2-Stream+LSTM [179]	VGG-16	RGB+Flow	17.8
Async-TF [179]	VGG-16	RGB+Flow	22.4
Multiscale TRN [96]	Inception	RGB	25.2
I3D [50]	Inception	RGB	32.9
I3D [120]	R50-I3D	RGB	31.8
STRG [120]	R50-I3D	RGB	36.2
STAG [125]	R50-I3D	RGB	37.2
Pose and Joint-Aware [180]	R50-I3D	Pose+RGB	32.81
LFB Max [151]	R50-I3D-NL	RGB	38.6
STLT+I3D [152]	R50-I3D	RGB	38.5
I3D+STIT **(ours)**	R50-I3D	RGB	**39.62**
Slowfast 16 x 8 [100]	R50-3D	RGB	38.9
Slowfast 16 x 8+STIT **(ours)**	R50-3D	RGB	**42.49**

temporal transformer leads to a 3% decline in model performance whereas the performance loses only 0.84% when replacing the spatial transformers with pooling. This indicates the importance of temporal dependencies between interactions that can be captured via the temporal transformer.

Moreover, we apply different settings in using latent class token over the mean of transformer tokens. We observe that latent token provides better compact representation for spatial and temporal contexts, which are learned via spatial and temporal transformers, respectively. Also, learning human-object relations via our STIT outperforms the I3D baseline, achieving 5.39% mAP improvement. Fig. 5.4 shows examples of HOIs that our STIT performs better than I3D. Our STIT model can distinguish between different interactions with the same objects, such as taking, holding, and placing a laptop, whereas I3D cannot. Furthermore, our model can discriminate between how the same HOI can be performed with various objects, such as holding a towel versus holding a box. More importantly, interactions that occur simultaneously can be recognized. For example, it can be seen in Fig. 3, the human in the third example is washing a window and this interaction involved

Table 5.8: Ablation results on Charades [1] using I3D-R50 backbone.

Model	**mAP(%)**
I3D	34.23
STIT-Spatial	36.60
STIT-Temporal	38.78
STIT-spatial-mean	38.94
STIT-Temporal-mean	38.64
STIT-mean	37.06
STIT **(ours)**	**39.62**

another interaction, which is holding towel at the same time.

5.3.3 Experiments on the Something-Something v1 Dataset

We also evaluate our model on different types of interaction videos where only hands are interacting with objects without the appearance of the human body. In these types of interactions, learning temporal dependencies is very important [96]. For training the proposed STIT, as in [120], we sample 32 frames and use 10 object proposals that are generated as in the Charades experiments from each frame. We train our model on the top of a fixed I3D backbone where we extract the tokens features from. We train our model for 50 epochs with batch size of 8 videos. We start with a 0.02 learning rate and it is reduced by a factor of 10 at 35 and 45 epochs. We train I3D backbone, and initialize it as in [120] with Kinetics pre-trained model and use the same training schedule as our STIT model. We add 1×1 convolution layer on top of I3D output to reduce the channel number from 2048 to 512. Subsequently, each human and object token is with the size of 512. As in [120], we fuse our STIT model features with I3D features for final representation of the action. In our STIT model, we use eight and four heads for spatial and temporal transformers, respectively. Table 5.9 provides information on the input and output of spatial and temporal transformers on SSv1 [3] dataset.

For inference, we use multi-view from each video as in [18, 100, 120] with 2 clips. We also applied multi-crop (left, center, right) as in [120,125]. The evaluation metric for SSv1 is the accuracy where scores from different views are fused via max to report

Figure 5.4: Comparison between I3D and our STIT on Charades [1].

Table 5.9: Configuration of our spatial and temporal transformers for SSv1 Dataset [3]. B and `CLS` indicate batch size and class token, respectively. 16 is the number of frames that we use to extract human objects features and 10 is the number of human object proposals at each frame.

Model	**Input size**	**Output size**
Spatial Transformer	B*16×10+ `CLS` × 512	B*16× 512
Temporal transformer	B×16 + `CLS` × 512	B× 512

the results.

As shown in Table 5.10, our STIT model outperforms other approaches, which confirms the importance of our proposed hierarchical learning of human-object interactions, even when learning different nature of interactions, such as hand-object interaction in Something-Something v1. Our hierarchical representation of actions outperforms other relation approaches without hierarchical representation, such as similarity graph [120], and other models relying on global representation of actions. However, it's important to note that the STM model [18] achieves better performance than our STIT model when modeling additional motion features.

Table 5.10: Performance of STIT model on Something-Something v1 dataset [3] compared with prior works. Top-1 accuracy is reported on the validation set. CSTM [18] represents a variant of STM [18] that exclusively takes into account spatiotemporal features.

Model	**Backbone**	**Top-1 Acc.(%)**
MultiScale TRN [96]	Inception	34.4
I3D [120]	R50-I3D	41.6
I3D+similarity graph [120]	R50-I3D	42.7
STRG [120]	R50-I3D	43.4
ECO [186]	BNInception+3D ResNet-18	46.4
TSM [97]	R50	44.8
TSN [41]	R50	19.9
CSTM [18]	R50	47.7
STM [18]	R50	49.2
STIT **(ours)**	R50-I3D	47.92

5.3.4 Structure Learning of HOIs via Hierarchical Designs

We now justify our network design in employing two-level hierarchies including spatial and temporal. We also consider different time windows (e.g., number of frames) for aggregating the local contexts with different number of temporal transformers, which are referred as close to a small time window (Close) and wide to a large window (Wide). For simplicity, in Fig. 5.5, we show example of close window with two frames. Note that for the Charades [1] experiments in Table 5.11, we extract token features from a total of 16 frames, and a window of 4 and 8 frames are used for Close and Wide windows, respectively. For CAD-120 Dataset [2], we choose 5 and 15 frames for Close and Wide windows, respectively. We run experiments with different designs for modeling HOIs including our model as shown in Fig. 5.5. Explanations of these designs are as follows:

- The first design implies that hierarchical learning is not taken into consideration. Instead, the spatio-temporal transformer is used to learn the pairwise relations between all tokens from different time steps, as shown in Fig. 5.5 (A).

- (B) The second design is our STIT, which utilizes a two-level hierarchy to

obtain the latent representation of HOIs, as depicted in Fig. 5.5 (B).

- In the third design, we use a small window of 2 where three mid-level temporal transformers are used to learn the relation between compact representations of HOIs with a two-frame range. Then, a higher-level temporal transformer is used to model the relations between the mid-level representations of HOIs to produce the final representation of HOIs. Thus, three-level hierarchies of transformers, including spatial, mid-level, and high-level transformers, are formed. This design is illustrated in Fig. 5.5 (C).

- The last design amends the previous one by using a larger window of frames (e.g., Wide).

As in Table 5.11, we find that using more than two levels of transformers leads to model overfit where deeper levels of transformers can affect model generalization. Also, without hierarchical learning and using only one level of spatio-temporal transformer as in Fig. 5.5 (A), the model produces better results than three levels of hierarchy with specific temporal range because it captures the whole relations from different time steps. Hence, long-term temporal relations are captured well. Among all these architectures, we verify that our STIT with two-level hierarchy is the best for modelling HOIs and for capturing discriminative cues of action context.

Due to the different natures of actions and how they are being performed by human, some actions can be recognized with no-hierarchy, while others may require deeper-hierarchies. For example, recognizing a picking object action requires a deeper hierarchy in STIT while no-hierarchy fails to identify the action. In contrast, without a hierarchy in STIT, stacking objects actions in some videos are easier to be recognized. Because in picking objects actions, the spatial reasoning for objects at specific time is critical. However, we believe that in stacking objects, recognizing such action requires information about how the status of each object changes across time.

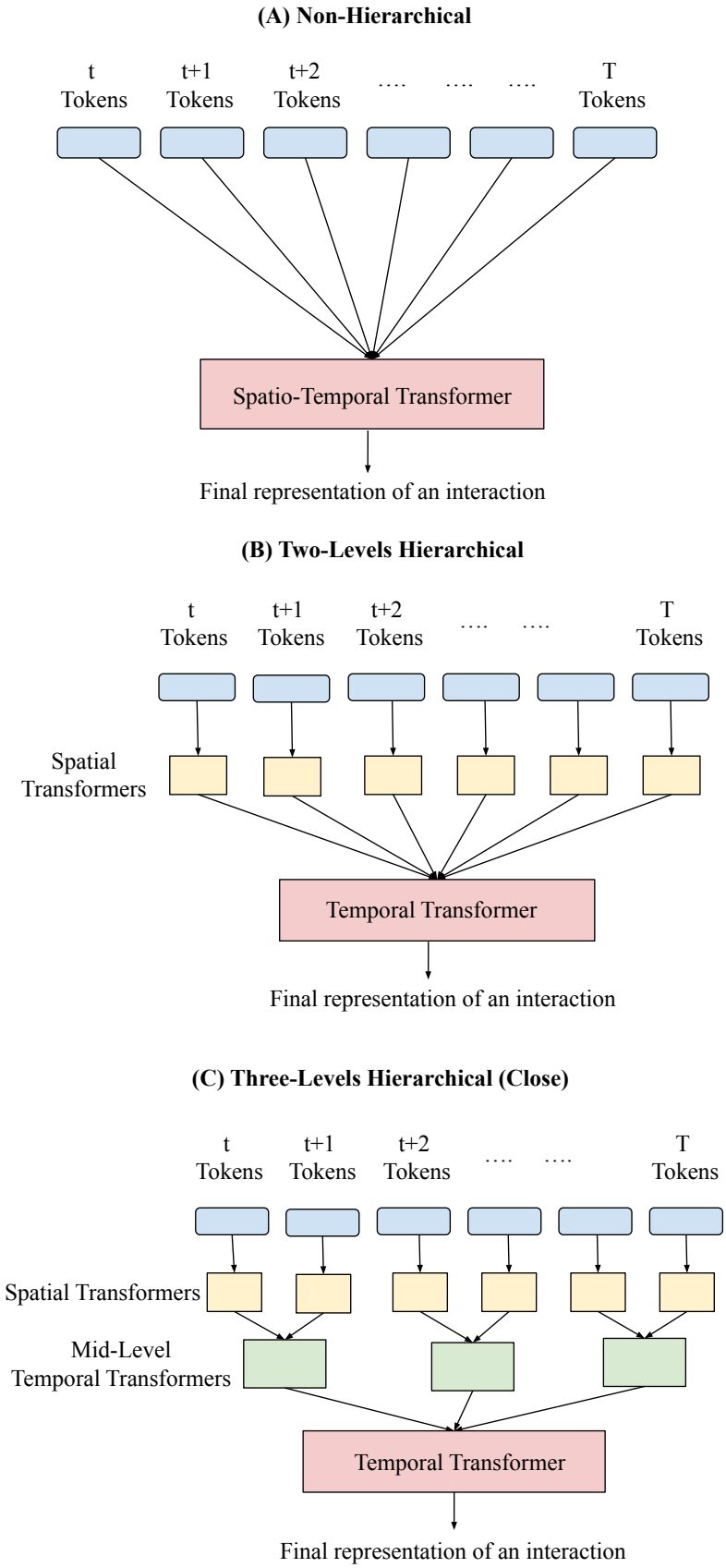


Figure 5.5: Different network designs for modeling HOIs. STs and TTs stand for spatial transformers and temporal transformers, respectively. For simplicity, we use six frames as an example.

Table 5.11: Results of applying different hierarchical designs in modeling HOIs. H stands for Hierarchical. The results are presented in terms of mean average precision (mAP) and accuracy (Acc) for the Charades [1] and CAD-120 [2] datasets, respectively.

Architecture	**Charades [1] (I3D)**	**Charades [1] Slowfast**	**CAD-120 [2]**
Three-Levels H (Close)	35.15	40.66	89.02
Three-Levels H (Wide)	35.23	40.86	84.44
Non-Hierarchical	38.91	41.24	94.21
Two-Levels H **(our STIT)**	**39.62**	**42.49**	**95.93**

5.4 Conclusion

The structural learning of HOIs captures crucial cues about how humans interact with different objects. Our STIT model explicitly uses hierarchical learning of the context of humans and objects to capture their interactions both at specific times and across a video. We show that STIT outperforms existing approaches, especially on the Charades and CAD-120 datasets. By studying different levels of hierarchy for modeling HOIs, we observed that two levels of hierarchy are enough to capture the local and global context of interactions via spatial and temporal transformers, respectively. Thus, this chapter emphasizes the significance of hierarchical learning in the recognition of human-object interactions.

CHAPTER 6

Conclusions

This thesis presents several important contributions to the area of recognizing human actions in videos, particularly actions that involve interactions with objects. Detailed descriptions of these contributions are provided in earlier chapters. This chapter concludes the thesis by providing a summary of the research contributions in Section 6.1, as well as a discussion of the study's limitations and possible directions for future research in Section 6.2.

6.1 Thesis Summary and Contributions

This thesis introduces and evaluates three novel deep learning-based models for recognizing human-object interactions in videos within a supervised learning context. The contributions of the thesis are summarized in the following three paragraphs.

Recognizing human interactions in videos requires temporal modeling of these interactions, which we focus on in Chapter 3. We present an LSTM-based network where the temporal changes of humans and objects can be captured independently. However, not all of the temporal changes of humans and objects provide a discriminative cue for an interaction. Therefore, we apply attention to the output of the

LSTM so that the significant parts of the temporal changes for each human and object could be captured. Then, we learn the pairwise relations between humans and objects via a bilinear layer, which is then fed to a deep LSTM network to capture high-level information on HOIs. Experimentally, we observe the importance of each part of the network design, where attention and the bilinear layer play crucial roles in improving the network's performance. Moreover, we validate the model's design by replacing the attentive-LSTMs with transformers, which have recently been employed in numerous computer vision tasks [9,14]. This substitution leads to in superior results when compared to LSTMs.

In Chapter 4, we address the drawback of the aforementioned approach where spatial relations between objects are not considered. We introduce a teacher-student-based network that captures both spatial and temporal relations between humans and objects from two different contextual views based on Graph Attention Network (GAT). The local contextual view focuses on learning the relations between humans and objects at each time step t (e.g., spatial), while the global contextual view learns the relations between humans and objects at different time steps (e.g., temporal). We observe that distilling knowledge from the global contextual view to the local one mostly boosts the performance of action recognition models and helps identify human-object interactions.

The final contribution of this work is to explore spatio-temporal modeling using transformer-based networks to recognize human-object interactions in a hierarchical manner. This approach involves learning the relations between humans and objects at each time step, leading to a local context representation at a specific time. Subsequently, higher-level temporal modeling is applied to learn the relationships between these time-specific representations. We study different hierarchical designs, including one, two, and three levels of hierarchical architectures to represent interactions. Our experimental results in Chapter 5 on two datasets, CAD-120 [2] and Charades [1], show that the two-level hierarchy is the best design for recognizing human-object interactions.

In summary, this thesis presents three novel approaches for recognizing human-object interactions using only visual data. The proposed methods leverage spatial

and temporal information of human and objects to learn discriminative representations of interactions. Among the methods we introduced, the most robust one that yielded superior results for human-object interactions is the third method outlined in Chapter 5. This method employs a crucial design for recognizing human-object interactions, implying the capture of local context of HOIs at specific times. This concept can be seen as analogous to the local contextual view of HOIs proposed in Chapter 4. However, in Chapter 4, temporal ordering is not learned, a facet that the method in Chapter 5 effectively capture. Additionally, even though the method presented in Chapter 3 involves sequence modeling and employs a hierarchical structure for learning HOIs, it doesn't capture the local context of such interactions, a feature that is preserved in the method described in Chapter 5. Thus, the approach proposed in Chapter 5 suggests that by jointly learning temporal, contextual, and hierarchical representations of human-object interactions, the most informative aspects of HOIs can be captured. This ultimately leads to achieving the most favorable results.

6.2 Limitation and Future Work

Although the thesis effectively demonstrates the superiority of the proposed methods over current approaches, it is important to acknowledge their limitations.

Firstly, not all datasets used for action recognition provide the necessary human and object annotations, such as bounding boxes. Therefore, in this thesis, we employ the Region Proposal Network (RPN) [174] to generate bounding boxes of potential humans and objects. To capture the target and relevant objects, a large number of proposals is necessary. We select the top ten RPN proposals as a minimum, even though their precision is lower than ground truth annotations, if available. This step is a prerequisite for training our proposed models. We use 10 or 15 proposals for this study, depending on the videos in the datasets. For instance, we use 15 proposals for the Charades dataset [1], which has a cluttered background. However, we select only 10 proposals for datasets containing videos with clear backgrounds, such as the something-something dataset [3]. Furthermore, when bounding boxes are

generated by RPN, RoIAlign [173] is employed to extract features of these bounding boxes from CNN backbones. However, with the recent advancements in models such as transformers that have achieved state-of-the-art results in various computer vision tasks, the applicability of RoIAlign [173] becomes limited and requires further investigation as a potential avenue for future research.

Furthermore, while the teacher-student network design with Knowledge Distillation (KD) presented in Chapter 4 improves the performance of the student network, it relies on a pre-trained teacher network. As a result, training teacher-student networks is a challenging task that requires proper training and fixing of the teacher network before it can effectively guide the student network. In addition, training the student network in the proposed network in Chapter 4 requires access to ground truth labels of videos. Therefore, it may not work if there are insufficient labels or if they are unavailable. Moreover, the experiments conducted in Chapter 4 illustrate the importance of carefully adjusting the temperature factor in knowledge distillation. This factor is used to soften the logits of both the teacher and student models. Consequently, the use of inappropriate temperature values could potentially have a negative impact on the quality of transferred knowledge and the overall performance of the student model.

Finally, in accordance with previous research, this study demonstrates that for certain backbones such as I3D, stage-wise training is the optimal strategy for feature learning in the proposed models presented in Chapters 4 and 5 resulting in superior results. Therefore, exploring the possibility of replacing multi-stage training with single-stage training in videos while preserving optimal performance could be a promising avenue for future research.

There are also other potential avenues for future research. Building upon the insights gained from the transformer-based proposal presented in Chapter 5, additional data types such as geometry, skeleton data, and word embeddings can be incorporated. The challenge will be to design a network that can effectively learn structural representations and semantic knowledge of human-object interactions from different data modes.

Additionally, the GLIDN network introduced in Chapter 4 provides a platform

Table 6.1: The accuracy results of the proposed multi-view transformer network on the CAD-120 dataset [2]

Model	**Acc.(%)**
Baseline	86.69
View 1(Spatio-temporal)	94.21
View 2 (Structural STIT)	95.93
Combining two views (multi-view transformers)	97.55

for potential expansion with the objective of transfer a rich knowledge from the teacher model. This goal could be achieves through the incorporation of multi-level distillation for human-object interactions. In this approach, in addition to utilizing logit outputs, features from multiple levels or layers of the teacher model can be transferred to the student model, thereby capturing diverse aspects of an interaction.

Moreover, exploring more complex action scenarios where occlusion or partial occlusion occurs in videos, such as multiple actions performed by multiple individuals with the presence of multiple objects, would be an intriguing area for future research. This could also be extended to include the recognition of unusual actions, where humans interact with objects in unconventional ways or with previously unseen objects. Besides, conducting cross-dataset validation, such as training on the Charades dataset and evaluating on different datasets, can also be explored to validate the generalizability of the proposed models for recognizing human-object interactions. Given that our proposed methods rely exclusively on recognizing HOIs, there would be benefits in extending them to also learn the exact frames at which an interaction actually begins and ends within the video.

In addition, in light of our second contribution, it would be interesting to explore multi-view transformer-based networks that consider different spatio-temporal relations. We conducted initial experiments on the CAD-120 dataset [2] by incorporating two views: the spatio-temporal view, which considers connections between humans and objects in the same frame and across frames from different time steps in videos, and the structural view, which is the STIT network proposed in Chapter 5. Our best results of 97.5% accuracy were achieved using this approach. Table 6.1 presents the ablation studies of this idea. This approach can be further extended

by applying contrastive learning between these views. Other ideas can be explored, such as using pure transformers without convolutions for feature extraction and randomly selecting patches for each view to learn spatio-temporal relations between them contrastively. Expanding the usage of these views could also involve incorporating multiple teacher networks (e.g., an ensemble of teachers) and applying various distillation methods. This concept presents an avenue for future research.

Furthermore, expanding the interpretability of our models could be a potential future path to better understanding the model's decision-making process. By using explainable AI models, we could determine the most informative frame(s) and human-object interactions, as well as the most important objects for recognizing actions in situations with multiple objects around humans. This would provide insights into the internal mechanisms of the models, and potentially reveal areas for improvement.

Alternative methodologies, such as reinforcement learning or active learning techniques, may be employed to obtain generalizations for human-object interaction recognition. Also, there is a promising avenue for future research in which generative AI models can be utilized to explore human-object interactions. This includes generating diverse types of interactions under varying motions and background conditions (e.g. low-light). Besides, it involves investigating methods to generate a variety of human-object interaction scenarios conditioned on different modalities, such as text, in order to achieve a deeper comprehension of these interactions. Also, it is worthwhile to study human-object interactions in 3D, utilizing 3D representations for human body and object shapes, and to explore how to learn the spatial-temporal relations between them under complex scenarios in order to recognize such interactions in videos. While our research focuses on daily life interactions, it is worth exploring the use of the proposed models in different interaction situations, such as driving, to detect collision or near-collision events.

Bibliography

[1] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *European Conference on Computer Vision*, pp. 510–526, Springer, 2016.

[2] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[3] R. Goyal, S. Ebrahimi Kahou, V. Michalski, J. Materzynska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. 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*, pp. 5842–5850, 2017.

[4] U. ROBOTS, "6 examples of industrial robots in the automotive industry," 2020.

[5] I. Rodríguez-Moreno, J. M. Martínez-Otzeta, I. Goienetxea, I. Rodriguez-Rodriguez, and B. Sierra, "Shedding light on people action recognition in social robotics by means of common spatial patterns," *Sensors*, vol. 20, no. 8, p. 2436, 2020.

[6] A. olutions, "Is amazon go store technology the end of supermarket checkouts?," 2023.

[7] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

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

[11] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2625–2634, 2015.

[12] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, vol. 27, 2014.

[13] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7794–7803, 2018.

[14] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," *arXiv preprint arXiv:2103.15691*, 2021.

[15] K. Wang, X. Wang, L. Lin, M. Wang, and W. Zuo, "3d human activity recognition with reconfigurable convolutional neural networks," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 97–106, 2014.

[16] Z. Liu, Y. Yao, Y. Liu, Y. Zhu, Z. Tao, L. Wang, and Y. Feng, "Learning dynamic spatio-temporal relations for human activity recognition," *IEEE Access*, vol. 8, pp. 130340–130352, 2020.

[17] J. Tayyub, A. Tavanai, Y. Gatsoulis, A. G. Cohn, and D. C. Hogg, "Qualitative and quantitative spatio-temporal relations in daily living activity recognition," in *Asian Conference on Computer Vision*, pp. 115–130, Springer, 2014.

[18] B. Jiang, M. Wang, W. Gan, W. Wu, and J. Yan, "Stm: Spatiotemporal and motion encoding for action recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2000–2009, 2019.

[19] S. M. Kang and R. P. Wildes, "Review of action recognition and detection methods," *arXiv preprint arXiv:1610.06906*, 2016.

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

[21] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng, "Learning to detect human-object interactions," in *2018 ieee winter conference on applications of computer vision (wacv)*, pp. 381–389, IEEE, 2018.

[22] A. Prest, C. Schmid, and V. Ferrari, "Weakly supervised learning of interactions between humans and objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 601–614, 2011.

[23] L. Shen, S. Yeung, J. Hoffman, G. Mori, and L. Fei-Fei, "Scaling human-object interaction recognition through zero-shot learning," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1568–1576, IEEE, 2018.

[24] G. Gkioxari, R. Girshick, P. Dollár, and K. He, "Detecting and recognizing human-object interactions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8359–8367, 2018.

[25] B. Xu, J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Interact as you intend: Intention-driven human-object interaction detection," *IEEE Transactions on Multimedia*, vol. 22, no. 6, pp. 1423–1432, 2019.

[26] B. Yao and L. Fei-Fei, "Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses," *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 9, pp. 1691–1703, 2012.

[27] W.-M. Deng, H.-B. Zhang, Q. Lei, J.-X. Du, and M. Huang, "Pose attention and object semantic representation-based human-object interaction detection network," *Multimedia Tools and Applications*, pp. 1–18, 2022.

[28] L. Cao, Y. Tian, Z. Liu, B. Yao, Z. Zhang, and T. S. Huang, "Action detection using multiple spatial-temporal interest point features," in *2010 IEEE International Conference on Multimedia and Expo*, pp. 340–345, IEEE, 2010.

[29] A.-N. Sharkawy, "Human-robot interaction: Applications," *Proceedings of the 1st IFSA Winter Conference on Automation, Robotics Communications for Industry 4.0*, pp. 98–103, 2021.

[30] M. Babiker, O. O. Khalifa, K. K. Htike, A. Hassan, and M. Zaharadeen, "Automated daily human activity recognition for video surveillance using neural network," in *2017 IEEE 4th international conference on smart instrumentation, measurement and application (ICSIMA)*, pp. 1–5, IEEE, 2017.

[31] Y. Xu, X. Yang, L. Gong, H.-C. Lin, T.-Y. Wu, Y. Li, and N. Vasconcelos, "Explainable object-induced action decision for autonomous vehicles," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9523–9532, 2020.

[32] B. T. Naik, M. F. Hashmi, and N. D. Bokde, "A comprehensive review of computer vision in sports: Open issues, future trends and research directions," *Applied Sciences*, vol. 12, no. 9, p. 4429, 2022.

[33] A. Polacco and K. Backes, "The amazon go concept: Implications, applications, and sustainability," *Journal of Business and Management*, vol. 24, no. 1, pp. 79–92, 2018.

[34] A. olutions, "Number of amazon go retail stores in the united states in 2023," 2023.

[35] A. Bansal, S. S. Rambhatla, A. Shrivastava, and R. Chellappa, "Detecting human-object interactions via functional generalization," *arXiv preprint arXiv:1904.03181*, 2019.

[36] D. Tu, W. Sun, X. Min, G. Zhai, and W. Shen, "Video-based human-object interaction detection from tubelet tokens," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23345–23357, 2022.

[37] A. Gupta, A. Kembhavi, and L. S. Davis, "Observing human-object interactions: Using spatial and functional compatibility for recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 10, pp. 1775–1789, 2009.

[38] Y. Kong and Y. Fu, "Human action recognition and prediction: A survey," *International Journal of Computer Vision*, vol. 130, no. 5, pp. 1366–1401, 2022.

[39] H.-S. Fang, J. Cao, Y.-W. Tai, and C. Lu, "Pairwise body-part attention for recognizing human-object interactions," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 51–67, 2018.

[40] M. G. Morshed, T. Sultana, A. Alam, and Y.-K. Lee, "Human action recognition: A taxonomy-based survey, updates, and opportunities," *Sensors*, vol. 23, no. 4, p. 2182, 2023.

[41] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European conference on computer vision*, pp. 20–36, Springer, 2016.

[42] P. Weinzaepfel and G. Rogez, "Mimetics: Towards understanding human actions out of context," *International Journal of Computer Vision*, vol. 129, no. 5, pp. 1675–1690, 2021.

[43] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *arXiv preprint arXiv:1603.07285*, 2016.

[44] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

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

[46] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.

[47] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, 2015.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

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

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

[51] H. Salehinejad, S. Sankar, J. Barfett, E. Colak, and S. Valaee, "Recent advances in recurrent neural networks," *arXiv preprint arXiv:1801.01078*, 2017.

[52] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3156–3164, 2015.

[53] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1110–1118, 2015.

[54] B. Singh, T. K. Marks, M. Jones, O. Tuzel, and M. Shao, "A multi-stream bi-directional recurrent neural network for fine-grained action detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1961–1970, 2016.

[55] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[56] R. Pascanu, T. Mikolov, and Y. Bengio, "On the difficulty of training recurrent neural networks," in *International conference on machine learning*, pp. 1310–1318, PMLR, 2013.

[57] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[58] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[59] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[60] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[61] Y. Li, W. Ouyang, B. Zhou, J. Shi, C. Zhang, and X. Wang, "Factorizable net: an efficient subgraph-based framework for scene graph generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 335–351, 2018.

[62] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018.

[63] M. Welling and T. N. Kipf, "Semi-supervised classification with graph convolutional networks," in *J. International Conference on Learning Representations (ICLR 2017)*, 2016.

[64] R. v. d. Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," *arXiv preprint arXiv:1706.02263*, 2017.

[65] E. Choi, M. T. Bahadori, L. Song, W. F. Stewart, and J. Sun, "Gram: graph-based attention model for healthcare representation learning," in *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 787–795, 2017.

[66] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[67] D. Sarkar, R. Bali, and T. Ghosh, *Hands-On Transfer Learning with Python: Implement advanced deep learning and neural network models using TensorFlow and Keras*. Packt Publishing Ltd, 2018.

[68] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.

[69] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[70] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[71] F. M. Thoker and J. Gall, "Cross-modal knowledge distillation for action recognition," in *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 6–10, IEEE, 2019.

[72] V. Vapnik and A. Vashist, "A new learning paradigm: Learning using privileged information," *Neural networks*, vol. 22, no. 5-6, pp. 544–557, 2009.

[73] D. Lopez-Paz, L. Bottou, B. Schölkopf, and V. Vapnik, "Unifying distillation and privileged information," *arXiv preprint arXiv:1511.03643*, 2015.

[74] Z. Luo, J.-T. Hsieh, L. Jiang, J. C. Niebles, and L. Fei-Fei, "Graph distillation for action detection with privileged modalities," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 166–183, 2018.

[75] N. C. Garcia, S. A. Bargal, V. Ablavsky, P. Morerio, V. Murino, and S. Sclaroff, "Distillation multiple choice learning for multimodal action recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2755–2764, 2021.

[76] N. Crasto, P. Weinzaepfel, K. Alahari, and C. Schmid, "Mars: Motion-augmented rgb stream for action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7882–7891, 2019.

[77] R. Dai, S. Das, and F. Bremond, "Learning an augmented rgb representation with cross-modal knowledge distillation for action detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13053–13064, 2021.

[78] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE symposium on security and privacy (SP)*, pp. 582–597, IEEE, 2016.

[79] J. Vongkulbhisal, P. Vinayavekhin, and M. Visentini-Scarzanella, "Unifying heterogeneous classifiers with distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3175–3184, 2019.

[80] Z. Zhang, C. Zhou, and Z. Tu, "Distilling inter-class distance for semantic segmentation," *arXiv preprint arXiv:2205.03650*, 2022.

[81] Y. Wang, W. Zhou, T. Jiang, X. Bai, and Y. Xu, "Intra-class feature variation distillation for semantic segmentation," in *European Conference on Computer Vision*, pp. 346–362, Springer, 2020.

[82] L. Zhang and K. Ma, "Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors," in *International Conference on Learning Representations*, 2020.

[83] J. Guo, K. Han, Y. Wang, H. Wu, X. Chen, C. Xu, and C. Xu, "Distilling object detectors via decoupled features," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2154–2164, 2021.

[84] Z. Yang, Z. Li, X. Jiang, Y. Gong, Z. Yuan, D. Zhao, and C. Yuan, "Focal and global knowledge distillation for detectors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4643–4652, 2022.

[85] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: a large video database for human motion recognition," in *2011 International conference on computer vision*, pp. 2556–2563, IEEE, 2011.

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

[87] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, 2014.

[88] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[89] I. Sanou, D. Conte, and H. Cardot, "An extensible deep architecture for action recognition problem," in *14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP 2019)*, 2019.

[90] L. Gorelick, M. Blank, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 12, pp. 2247–2253, 2007.

[91] I. Laptev, "On space-time interest points," *International journal of computer vision*, vol. 64, pp. 107–123, 2005.

[92] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Action recognition by dense trajectories," in *CVPR 2011*, pp. 3169–3176, 2011.

[93] H. Wang and C. Schmid, "Action recognition with improved trajectories," in *Proceedings of the IEEE international conference on computer vision*, pp. 3551–3558, 2013.

[94] A. B. Sargano, P. Angelov, and Z. Habib, "A comprehensive review on handcrafted and learning-based action representation approaches for human activity recognition," *applied sciences*, vol. 7, no. 1, p. 110, 2017.

[95] J. Yue-Hei Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici, "Beyond short snippets: Deep networks for video classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4694–4702, 2015.

[96] B. Zhou, A. Andonian, A. Oliva, and A. Torralba, "Temporal relational reasoning in videos," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 803–818, 2018.

[97] J. Lin, C. Gan, and S. Han, "Tsm: Temporal shift module for efficient video understanding," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7083–7093, 2019.

[98] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[99] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1510–1517, 2017.

[100] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6202–6211, 2019.

[101] C. Feichtenhofer, "X3d: Expanding architectures for efficient video recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 203–213, 2020.

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

[103] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," *Advances in neural information processing systems*, vol. 28, 2015.

[104] D. Dwibedi, P. Sermanet, and J. Tompson, "Temporal reasoning in videos using convolutional gated recurrent units," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1111–1116, 2018.

[105] X. Wang, Z. Miao, R. Zhang, and S. Hao, "I3d-lstm: A new model for human action recognition," in *IOP Conference Series: Materials Science and Engineering*, vol. 569, p. 032035, IOP Publishing, 2019.

[106] F. Baradel, N. Neverova, C. Wolf, J. Mille, and G. Mori, "Object level visual reasoning in videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 105–121, 2018.

[107] D. Damen, H. Doughty, G. M. Farinella, S. Fidler, A. Furnari, E. Kazakos, D. Moltisanti, J. Munro, T. Perrett, W. Price, *et al.*, "Scaling egocentric vision: The epic-kitchens dataset," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 720–736, 2018.

[108] C.-Y. Ma, A. Kadav, I. Melvin, Z. Kira, G. AlRegib, and H. Peter Graf, "Attend and interact: Higher-order object interactions for video understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6790–6800, 2018.

[109] C. Sun, A. Shrivastava, C. Vondrick, K. Murphy, R. Sukthankar, and C. Schmid, "Actor-centric relation network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 318–334, 2018.

[110] C. Gu, C. Sun, D. A. Ross, C. Vondrick, C. Pantofaru, Y. Li, S. Vijayanarasimhan, G. Toderici, S. Ricco, R. Sukthankar, *et al.*, "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6047–6056, 2018.

[111] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5308–5317, 2016.

[112] A. M. Truong and A. Yoshitaka, "Structured lstm for human-object interaction detection and anticipation," in *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–6, IEEE, 2017.

[113] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1933–1941, 2016.

[114] R. Christoph and F. A. Pinz, "Spatiotemporal residual networks for video action recognition," *Advances in neural information processing systems*, vol. 3, 2016.

[115] X. Wang, L. Gao, P. Wang, X. Sun, and X. Liu, "Two-stream 3-d convnet fusion for action recognition in videos with arbitrary size and length," *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 634–644, 2017.

[116] C. Gao, Y. Zou, and J.-B. Huang, "ican: Instance-centric attention network for human-object interaction detection," *arXiv preprint arXiv:1808.10437*, 2018.

[117] S. Gupta and J. Malik, "Visual semantic role labeling," *arXiv preprint arXiv:1505.04474*, 2015.

[118] M. Antoun and D. Asmar, "Human object interaction detection: Design and survey," *Image and Vision Computing*, p. 104617, 2022.

[119] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[120] X. Wang and A. Gupta, "Videos as space-time region graphs," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 399–417, 2018.

[121] Z. Liang, J. Liu, Y. Guan, and J. Rojas, "Visual-semantic graph attention networks for human-object interaction detection," *arXiv e-prints*, pp. arXiv–2001, 2020.

[122] J. Ji, R. Krishna, L. Fei-Fei, and J. C. Niebles, "Action genome: Actions as compositions of spatio-temporal scene graphs," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10236–10247, 2020.

[123] N. Wang, G. Zhu, H. Li, M. Feng, X. Zhao, L. Ni, P. Shen, L. Mei, and L. Zhang, "Exploring spatio-temporal graph convolution for video-based human-object interaction recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[124] M. Tomei, L. Baraldi, S. Calderara, S. Bronzin, and R. Cucchiara, "Video action detection by learning graph-based spatio-temporal interactions," *Computer Vision and Image Understanding*, vol. 206, p. 103187, 2021.

[125] R. Herzig, E. Levi, H. Xu, H. Gao, E. Brosh, X. Wang, A. Globerson, and T. Darrell, "Spatio-temporal action graph networks," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 0–0, 2019.

[126] J. Materzynska, T. Xiao, R. Herzig, H. Xu, X. Wang, and T. Darrell, "Something-else: Compositional action recognition with spatial-temporal interaction networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1049–1059, 2020.

[127] J. Liu, G. Wang, P. Hu, L.-Y. Duan, and A. C. Kot, "Global context-aware attention lstm networks for 3d action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1647–1656, 2017.

[128] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," *arXiv preprint arXiv:1511.04119*, 2015.

[129] J. Li, X. Liu, W. Zhang, M. Zhang, J. Song, and N. Sebe, "Spatio-temporal attention networks for action recognition and detection," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2990–3001, 2020.

[130] R. Girdhar and D. Ramanan, "Attentional pooling for action recognition," *Advances in neural information processing systems*, vol. 30, 2017.

[131] W. Du, Y. Wang, and Y. Qiao, "Recurrent spatial-temporal attention network for action recognition in videos," *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1347–1360, 2017.

[132] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. Snoek, "Videolstm convolves, attends and flows for action recognition," *Computer Vision and Image Understanding*, vol. 166, pp. 41–50, 2018.

[133] D. Li, T. Yao, L.-Y. Duan, T. Mei, and Y. Rui, "Unified spatio-temporal attention networks for action recognition in videos," *IEEE Transactions on Multimedia*, vol. 21, no. 2, pp. 416–428, 2018.

[134] C. Dai, X. Liu, and J. Lai, "Human action recognition using two-stream attention based lstm networks," *Applied soft computing*, vol. 86, p. 105820, 2020.

[135] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos "in the wild"," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1996–2003, IEEE, 2009.

[136] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2929–2936, IEEE, 2009.

[137] X. Guo, X. Guo, and Y. Lu, "Ssan: Separable self-attention network for video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12618–12627, 2021.

[138] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International Conference on Machine Learning*, pp. 10347–10357, PMLR, 2021.

[139] P. Chu, J. Wang, Q. You, H. Ling, and Z. Liu, "Transmot: Spatial-temporal graph transformer for multiple object tracking," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 4870–4880, 2023.

[140] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," *arXiv preprint arXiv:2104.11227*, 2021.

[141] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," *arXiv preprint arXiv:2106.13230*, 2021.

[142] D. Neimark, O. Bar, M. Zohar, and D. Asselmann, "Video transformer network," *arXiv preprint arXiv:2102.00719*, 2021.

[143] D. Ahn, S. Kim, H. Hong, and B. C. Ko, "Star-transformer: A spatio-temporal cross attention transformer for human action recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3330–3339, 2023.

[144] J. Yang, X. Dong, L. Liu, C. Zhang, J. Shen, and D. Yu, "Recurring the transformer for video action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14063–14073, 2022.

[145] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3202–3211, 2022.

[146] S. Yan, X. Xiong, A. Arnab, Z. Lu, M. Zhang, C. Sun, and C. Schmid, "Multiview transformers for video recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3333–3343, 2022.

[147] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?," *arXiv preprint arXiv:2102.05095*, 2021.

[148] D. Damen, H. Doughty, G. M. Farinella, A. Furnari, E. Kazakos, J. Ma, D. Moltisanti, J. Munro, T. Perrett, W. Price, *et al.*, "Rescaling egocentric vision," *arXiv preprint arXiv:2006.13256*, 2020.

[149] M. Monfort, A. Andonian, B. Zhou, K. Ramakrishnan, S. A. Bargal, T. Yan, L. Brown, Q. Fan, D. Gutfreund, C. Vondrick, *et al.*, "Moments in time dataset: one million videos for event understanding," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 2, pp. 502–508, 2019.

[150] R. Girdhar, J. Carreira, C. Doersch, and A. Zisserman, "Video action transformer network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 244–253, 2019.

[151] C.-Y. Wu, C. Feichtenhofer, H. Fan, K. He, P. Krahenbuhl, and R. Girshick, "Long-term feature banks for detailed video understanding," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 284–293, 2019.

[152] G. Radevski, M.-F. Moens, and T. Tuytelaars, "Revisiting spatio-temporal layouts for compositional action recognition," *The British Machine Vision Conference (BMVC)*, 2021.

[153] Y. Cong, W. Liao, H. Ackermann, B. Rosenhahn, and M. Y. Yang, "Spatial-temporal transformer for dynamic scene graph generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16372–16382, 2021.

[154] Q. Li, X. Xie, J. Zhang, and G. Shi, "Few-shot human–object interaction video recognition with transformers," *Neural Networks*, vol. 163, pp. 1–9, 2023.

[155] A. Prest, C. Schmid, and V. Ferrari, "Weakly supervised learning of interactions between humans and objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 3, pp. 601–614, 2012.

[156] B. Xu, J. Li, Y. Wong, M. S. Kankanhalli, and Q. Zhao, "Interact as you intend: Intention-driven human-object interaction detection," *arXiv preprint arXiv:1808.09796*, 2018.

[157] M. S. Ibrahim, S. Muralidharan, Z. Deng, A. Vahdat, and G. Mori, "A hierarchical deep temporal model for group activity recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1971–1980, 2016.

[158] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[159] Z. Yu, J. Yu, C. Xiang, J. Fan, and D. Tao, "Beyond bilinear: Generalized multimodal factorized high-order pooling for visual question answering," *IEEE transactions on neural networks and learning systems*, no. 99, pp. 1–13, 2018.

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

[161] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[162] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3464–3468, IEEE, 2016.

[163] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[164] S. P. R. Sunkesula, R. Dabral, and G. Ramakrishnan, "Lighten: Learning interactions with graph and hierarchical temporal networks for hoi in videos," in *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 691–699, 2020.

[165] B. Xu, Y. Wong, J. Li, Q. Zhao, and M. S. Kankanhalli, "Learning to detect human-object interactions with knowledge," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[166] F. Li, C. Gan, X. Liu, Y. Bian, X. Long, Y. Li, Z. Li, J. Zhou, and S. Wen, "Temporal modeling approaches for large-scale youtube-8m video understanding," *arXiv preprint arXiv:1707.04555*, 2017.

[167] H. Tan, L. Wang, Q. Zhang, Z. Gao, N. Zheng, and G. Hua, "Object affordances graph network for action recognition," BMVC, 2019.

[168] J. Yang, W.-S. Zheng, Q. Yang, Y.-C. Chen, and Q. Tian, "Spatial-temporal graph convolutional network for video-based person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3289–3299, 2020.

[169] L. Lu, Y. Lu, R. Yu, H. Di, L. Zhang, and S. Wang, "Gaim: Graph attention interaction model for collective activity recognition," *IEEE Transactions on Multimedia*, vol. 22, no. 2, pp. 524–539, 2019.

[170] D. Huang, P. Chen, R. Zeng, Q. Du, M. Tan, and C. Gan, "Location-aware graph convolutional networks for video question answering," in *Proceedings*

of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 11021–11028, 2020.

[171] P. Ghosh, Y. Yao, L. Davis, and A. Divakaran, "Stacked spatio-temporal graph convolutional networks for action segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 576–585, 2020.

[172] B. Pan, H. Cai, D.-A. Huang, K.-H. Lee, A. Gaidon, E. Adeli, and J. C. Niebles, "Spatio-temporal graph for video captioning with knowledge distillation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10870–10879, 2020.

[173] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

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

[175] C. Bian, W. Feng, L. Wan, and S. Wang, "Structural knowledge distillation for efficient skeleton-based action recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 2963–2976, 2021.

[176] Y. Liu, L. Sheng, J. Shao, J. Yan, S. Xiang, and C. Pan, "Multi-label image classification via knowledge distillation from weakly-supervised detection," in *Proceedings of the 26th ACM international conference on Multimedia*, pp. 700–708, 2018.

[177] H. Fan, Y. Li, B. Xiong, W.-Y. Lo, and C. Feichtenhofer, "Pyslowfast." `https://github.com/facebookresearch/slowfast`, 2020.

[178] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[179] G. A. Sigurdsson, S. Divvala, A. Farhadi, and A. Gupta, "Asynchronous temporal fields for action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 585–594, 2017.

[180] A. Shah, S. Mishra, A. Bansal, J.-C. Chen, R. Chellappa, and A. Shrivastava, "Pose and joint-aware action recognition," *arXiv preprint arXiv:2010.08164*, 2020.

[181] R. Girdhar, D. Tran, L. Torresani, and D. Ramanan, "Distinit: Learning video representations without a single labeled video," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 852–861, 2019.

[182] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4320–4328, 2018.

[183] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[184] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[185] M. Ryoo, A. Piergiovanni, A. Arnab, M. Dehghani, and A. Angelova, "Tokenlearner: Adaptive space-time tokenization for videos," *Advances in Neural Information Processing Systems*, vol. 34, pp. 12786–12797, 2021.

[186] M. Zolfaghari, K. Singh, and T. Brox, "Eco: Efficient convolutional network for online video understanding," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 695–712, 2018.