DETECTION OF FREE-STANDING CONVERSATIONAL GROUPS WITH GRAPH CONVOLUTIONAL NETWORKS

by

Efehan Atıcı

B.S., Control and Automation Engineering, Istanbul Technical University, 2016

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2022

ACKNOWLEDGEMENTS

I would like to express my earnest gratitude and heartfelt respect to Prof. Lale Akarun and Assist. Prof. Berk Gökberk for their sincere leadership, encouragement and valuable support over the past years. I am grateful for their advice and guidance not only during my Master's study but also all along their lectures. I want to thank them for their patience, consistent support and belief in me.

I sincerely thank my family for their love and support throughout my Master's thesis. Their persistent inspiration and understanding has helped me in difficult days. I want to thank my older brother, Çağdaş, for his unconditional help to me in every situation, I want to thank my father Hakan, for encouraging me to take everything a step further and I want to thank my mother, Zerrin, for her unconditional love and continuous support. This dissertation is passionately dedicated to them.

ABSTRACT

DETECTION OF FREE-STANDING CONVERSATIONAL GROUPS WITH GRAPH CONVOLUTIONAL NETWORKS

Automatically detecting conversational groups from video footage is a very intriguing and practical research area with applications in video activity recognition and human-robot interaction. Therefore, there is a critical need for improved detection of groups to enhance the relationship between humans and robots. In this thesis, we use Graph Convolutional Networks for the group detection problem as the main novel contribution. We base our approach on a method from the community detection domain called Deep Modularity Networks. Our approach improves the group detection quality over state-of-the-art group detection methods. Additionally, we develop a graph construction algorithm using the view frustums, which indicates the individuals' affinities. As a post-processing step, we utilize temporal information in our system and improve our detection results further.

ÖZET

ETKİLEŞİMLİ GRUPLARIN ÇİZGE EVRİŞİMLİ SİNİR AĞLARI İLE TESPİTİ

Video görüntülerinden otomatik olarak etkileşimli grupları tanıma, videolardan aktivite algılama ve insan-robot ilişkilerini geliştirmek için oldukça kullanışlı bir araştırma konusu olmuştur. Bu sebeple, etkileşimli grupların algılanma doğruluğunu iyileştirmek, insan robot ilişkilerini kuvvetlendirmek için doğan ciddi bir ihtiyaçtır. Bu tezde, Çizge Evrişimli Sinir Ağları kullanarak etkileşimli grup tespit problemine özgün bir katkı yapılmıştır. Topluluk tanıma alanında kullanılan Deep Modularity Networks (DMoN) adında bir yöntem baz alınarak yeni bir grup tanıma sistemi geliştirilmiştir. Bu zamana kadar olan teknoloji grup tanıma metotlarında en ileri teknoloji yöntemlerin tanıma kalitesi yükseltilmiştir. Buna ek olarak, insanların yakınlığını görüş konileri kullanarak gösteren bir çizge inşa algoritması geliştirilmiştir. Problemin zaman boyutunu da hesaba katan bir art işleme adımı sisteme entegre edilerek tanıma sonuçları daha ileriye taşınmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

a_{ij}	Adjacency between nodes i and j
$\tilde{w}_{ij}$	Candidate edge weights between nodes i and j
$\mathbf{A}$	Pairwise Euclidean distance matrix
α	Full angle of view-frustum
L	Length of the triangular view-frustum
σ	Standard deviation
$I_{F_iF_j}$	Intersection area between view-frustums of nodes i and j
$\hat{A}$	Adjacency matrix with self-loops added
$\hat{D}$	Diagonal node degree matrix of $\hat{A}$
$H^{(l)}$	Feature vector at the l^{th} hidden layer of the neural network.
$W^{(l)}$	Weights at the l^{th} hidden layer of the neural network.
$\mathcal{Q}$	Modularity
Tr	Trace function
$\mathbf{C}$	Cluster assignment matrix
$\mathbf{B}$	Affinity matrix
$\|\|\|_F$	Frobenius Norm

LIST OF ACRONYMS/ABBREVIATIONS

CMU	Carnegie Mellon University
DANTE	Deep Affinity Networks for Clustering
DMoN	Deep Modularity Networks
EMA	Exponential Moving Average
FCG	Free-Standing Conversational Group
GCFF	Graph Cuts F-Formation
GCN	Graph Convolutional Network
GNN	Graph Neural Network
GTCG	Game Theoretic Clustering Game
HJS-PF	Hybrid Joint-Separable particle filter
MA	Moving Average
MDL	Minimum Description Length
SNAP	Stanford Large Network Dataset Collection
VGA	Video Graphics Array

1. INTRODUCTION

Automatic detection of groups is a crucial problem to solve as it ultimately unlocks numerous possibilities for human-robot interactions. In social robotics, information about an individual's social life their relationships with others employing subtle social clues are significant. Understanding human relationships allow robots to act differently depending on the situation and connections. However, most research in this area up to this day is dominated by traditional handcrafted methods instead of applying deep-learning approaches [2–7]. In recent years, few practices have emerged that use neural networks to compete with conventional systems [8, 9]. In this thesis, we explore the capabilities of deep learning-based networks in a more constrained environment in the absence of enough annotated datasets [10, 11]. We also contribute a new annotated dataset to the domain, in which we provide more extensive annotations than the rest of the datasets in the group detection corpus.

Most people would consider individuals who are connected in terms of relationship or social status; a group [12]. Types of groups can vary considering quantitative information such as size, persistence, structure, and depending on qualitative details, including the severity of relationships, sense of belonging, formality of the event. In our research, we will be focusing on face-to-face conversational groups, free-standing conversational groups (FCGs), or small gatherings of individuals immersed in focused encounters. FCGs are best at displaying the essential features of dynamic engagements, making them an important research topic.

What makes FCGs an important research area is that they depict actual social situations. They typically range from pizza or cocktail parties to more serious scenes such as a meeting or a poster session. Distinguishing the group members and sometimes primary speakers can give us valuable information about the dynamics of the relationships between individuals. Therefore, analyzing and automatically detecting FCGs has become a primary focus of computer vision literature [2, 8, 9, 13–16].

In the context of video surveillance, a 'group' is defined as two or more persons moving at a comparable speed who are physically and temporally adjacent to one another [17]. The main reason for the simplified description of a group in this concept is the difficulty of deducing continuous social arrangements from brief video segments. Most vision-based systems implement tracking in this situation, which entails catching individuals in motion and keeping their identification across video frames while also determining how they are divided into groups [17–21]. The videos are usually taken at public pedestrian areas from elevated viewpoints in the video surveillance domain. Therefore, the encounters in these videos mostly consist of unfocused interactions. Since the videos are taken at a high elevation and from afar, deducing the body pose and orientation of the people is another significant challenge [7].

In another domain where there are focused gatherings, i.e., meeting analysis, people spend time in a fixed place, interacting primarily by speaking or using body language. In a case like this, human actions can be analyzed thoroughly utilizing various audiovisual elements collected by ubiquitous sensors such as portable devices, microphone arrays, or sociometric badges [10, 22, 23].

In sociology, the terms focused and unfocused encounters come from the famous sociologist Erving Goffman [24]. Goffman describes unfocused interactions as the type of gatherings when two or more people come together without any mutually arranged plan. These interactions include scenarios such as waiting at a bus station, forming a queue at the traffic lights, or waiting at a meeting room. The individuals contributing to an unfocused interaction are aware of each other; however, there is no direct engagement in between. As for the focused interactions, people come together for an objective or a situation to communicate face-to-face. These exchanges endure for a long period on a single focus of cognitive and visual attention, such as a meeting, a board game night, or a multi-focused gathering.

Focused interactions can also take the form of free-standing conversational groups (FCGs) [25]. FCGs appear when people instinctively choose to be in each other's

vicinity to interact with one another. Therefore, FCGs happen in numerous and various social events such as a museum visit, a pizza or a cocktail party, a coffee break, or a poster session at the end of a lecture. These reasons identify why FCGs are key social elements, and their automated analysis and detection might lead to a new level of activity and behavioral research.

During a focused event, where people form various FCGs, they express themselves with speech and non-verbal movements, such as body gestures. These expressions are also called social signals [26], in which spatial features are more important than having verbal-only communication. However, having extremely rare data in this subject makes building a nonverbal communication model using social signals challenging. We search for the data that contains individuals' positions and orientations, body gestures, gaze, and facial expressions. To make things easier, we select the most important social interaction identifiers which describe an FCG. These are spatial positions and head/body orientations of people, which are necessary enough to find the F-Formation of a person.

In 1990, Adam Kendon, a researcher in the sociological discipline of nonverbal communication, established the term 'Facing Formation,' mostly known as F-Formation [25, 27, 28]. The individuals forming an F-Formation tend to maintain a circular space (o-space), where everyone in the formation has direct, equal, and easy access. Typical formations are circular, ellipse, horseshoe-shaped, side-by-side, or L-shape. These formations allow individuals to form a direct link and remove the distractions coming from outside the gathering. Automatic extraction of spatial position and orientational information is possible with deep-learning methods such as OpenPose [29], and they open the way for extracting the F-formation and, as a result, finding FCGs.

The impact of robust FCG detection is crucial in many contexts. For example, there is a need for automatic group analysis in video surveillance to recognize abnormal behaviors such as vandalism and violence and prevent them [30]. Also, understanding the social relationships between observed individuals in a setting can be valuable for

advanced suspect profiling. In human-robot interaction, detecting FCGs allows robots to utilize the affinities between people and accordingly in social situations. The approaches in the social robotics field have a limited number of people in their datasets when evaluating their quality of detecting FCGs [31–34]. It could be improved if the new approaches are responsive in more complex scenarios. Another application for the powerful identification of FCGs is the mobile visual search domain. An FCG detector can identify groups more effectively when coupled with 3D pose estimation systems, which opens up the ways to extract social traits of people and social network recommendation systems.

Previous methods that deal with the automatic detection of FCGs span many years. The first example of it is the proposition by Bazzani et al. [20] using positional and orientational features where they find Steady Conversational Groups (SCG). Between the traditional methods that use handcrafted features, the state-of-the-art algorithm is designed by Setti et al. [2] called Graph Cuts for F-Formations (GCFF). Vascon et al. [4] suggested a game-theoretic probabilistic approach (GTCG) that performs better on small datasets with low people density. More recently, deep-learning-based approaches started to arise in which Swofford et al. [8] utilized a Deep Neural Network to find out the affinities between individuals. They named their novel architecture Deep Affinity Networks (DANTE).

Our proposed method is based on Graph Convolutional Networks (GCNs), the backbone of another method submitted by Tsitsulin et al. to the Community Detection domain called Deep Modularity Networks (DMoN) [35]. Their main objective is to detect communities formed inside a single dense graph via clustering, where the graph's edges indicate the possible links between nodes. We modify this architecture and achieve clustering of multiple graphs by altering the training procedure. We introduce a novel approach that employs Graph Convolutional Networks and leverages the temporal information, a first in the social group detection field.

As a second contribution, we present a conversational graph construction algorithm that approximates the affinities between individuals in a group. We employ the techniques researched in the sociological domain, such as the importance of body orientation, using view frustums and combine these techniques with fundamentals of GCNs. We analyze the effect of various elements when building the social graph; the effect of initialization of node features, edge weights, and removal of edges with low weights.

Our other contribution is to create a new dataset valuable in future social graph detection research. Using the footage recorded by Carnegie Mellon University [36] in a dome-like structure with multiple viewpoints, we manually annotate the FCGs in a video taken from a pizza party. Our annotations are denser temporal than any other social event datasets, which we use to leverage the temporal information.

Finally, we re-evaluate previous state-of-the-art methods in the FCG detection problem, using a new training/test split of the datasets. Since deep-learning-based approaches are data-driven, they require a different way of evaluation than traditional methods. We implement K-fold splitting for all the datasets to test for each data point. We apply GRODE [37] metrics that provide more insight into group detection by allowing lenient and strict cases of predictions.

The remainder of this thesis is organized as follows: in Section 2 we explain the terminology presented in social sciences and survey previous methods that were used in social group modeling. In Section 3 we define our proposed method where we first describe our social graph construction and then clarify the mechanism of DMoN. In Section 4 we provide our results with re-evaluations of all previous methods based on our way of splicing the datasets. Finally, we conclude with how we think our model for detecting FCGs could be improved in the future.

2. RELATED WORK

As discussed in the introduction, social groups are made up of two or more people who are in immediate mutual presence at any given time with a prior relationship. From the cognitive science perspective, people tend to develop, dissolve, and redevelop groups in different social environments when they are at a social gathering. These social environments vary from a cocktail party, a night at a club, a day at the office, an evening at a theatre, to a simple walk together around the block or a picnic [38].

Since the context of a social group can vary heavily, the research on this subject is distributed among many fields. The research on group modeling in computer science is shared between cognitive sciences and the sociological field. When placing our approach within these many-sided cases, we need to analyze the difference between related sociological notions. Goffmanian notions are a great starting point for differentiating between concepts like 'Group' and 'Gathering,' 'social occasion' and 'social situation,' and 'unfocused' and 'focused' interactions [24, 39].

Starting with Unfocused Interactions, these interactions happen when two or more people come together without any shared objective. Usually, situations like forming a queue while waiting for traffic lights, waiting rooms for meetings, or waiting at a bus station are included in these kinds of interactions [24]. In these scenarios, an interaction between individuals occurs due to both parties' joint presence regardless of their intentions. The individuals contributing to an unfocused interaction are aware of each other; however, there is no direct engagement in between. Interactions like these are not unusual, and most of the time, they do not involve speech. Goffman classifies them as a form of interaction to reveal the importance of non-verbal forms of communication.

Another way of interaction is called focused interactions, where people come together for an activity or a situation to communicate face-to-face. Most of the time,

the agreement of individuals is rarely verbalized, and these types of interactions last for a time on a single focus of cognitive and visual attention. Adam Kendon further divides Goffman's focused interactions into two main categories: common focused and joint focused [39]. Examples of common focused interactions can be an army platoon on the parade ground or watching a movie at a theatre. There is only weak or no communication between pairs of individuals in common focused interactions due to having a common focused objective. However, there is a sense of a mutual, instead of a common activity, for the jointly focused interactions, where people are more likely to have face-to-face communication. Participation, in this case, is more engaging, and people are mutually involved in conversations with collaborators. For instance, this type of interaction could be a meeting, a board game night, or a multi-focused gathering such as a cocktail party [38].

Focused encounters can also take the form of free-standing conversational groups (FCGs) [25]. FCGs can occur at a party, a social supper, a coffee break, a museum visit, a day at the beach, a stroll through the city plaza, or a visit to the mall; more broadly, when people spontaneously decide to be in each other's close vicinity to connect. FCGs are key social entities for these reasons, and automated study could lead to a new level of activity and behavior analysis.

Adam Kendon came up with the term Facing Formations [25], mostly known as F-Formations, to explain the most important proxemic notions of an FCG. People's spatial position and orientation are mainly required to find the F-Formations. Since individuals in an FCG communicate with other participants, the feature set required to find an F-formation can be extended with other social signals, such as: talking, body language, and other non-verbal social cues. Kendon describes an F-formation as a convex space (o-space) formed around the participants, which people have established and sustained. All participants have equal, straightforward, and immediate entry to the formation in this interaction. Therefore, most of the shapes formed by individuals that count as an F-formation allow these criteria: a side-by-side arrangement, an L-shaped arrangement, a circle, an ellipse, or similar to a horseshoe arrangement. These examples

provide clear and easy access to the formation for participants. In our approach, we are most interested in finding the F-formations in multiple jointly focused interactions that reside within FCGs.

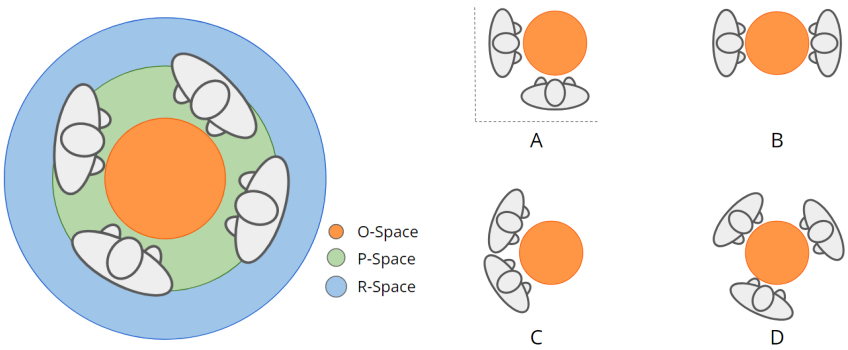

Figure 2.1. Left: O,P and R-space layers, showing the structure of an F-formation. Right: Examples of various F-formation arrangements. A. L-shaped, B. vis-a-vis, C. side-by-side, D. circular.

The research on detecting the conversational groups from focused gatherings in computer science first started with the advanced human-computer interaction and robots. It proliferated with the need for computer-supported cooperative work and socially aware robotics [31, 40–42]. The co-presence, mobility, multimodal communication, and embodied interaction of a service robot operating in the presence of a human user may lead to activities specified by the human's and the robot's behavior. In order to fill this need, a more advanced characteristic study is required, particularly body posture inference other than positional cues extraction. These are challenging tasks for traditional computer vision scenarios, where people are captured at low resolution, under various lighting conditions, and are frequently partially or entirely occluded.

The F-formation analysis incorporates context-aware computing in human computer interaction by taking into account spatial relationships among people, where space considerations become vital in creating applications for devices reacting to a situational change [31, 40]. Ballendat et al. [41], in particular, looked at how proxemic contact is expressive when it comes to indicators like location, identity, movement, and orientation. They discovered that by understanding and utilizing individuals' focused attention to other people, these cues could mediate the simultaneous interaction of sev-

eral people as an F-formation. The hardware design has been a barrier for academics thus far with these applications, and the social dynamics are often not studied.

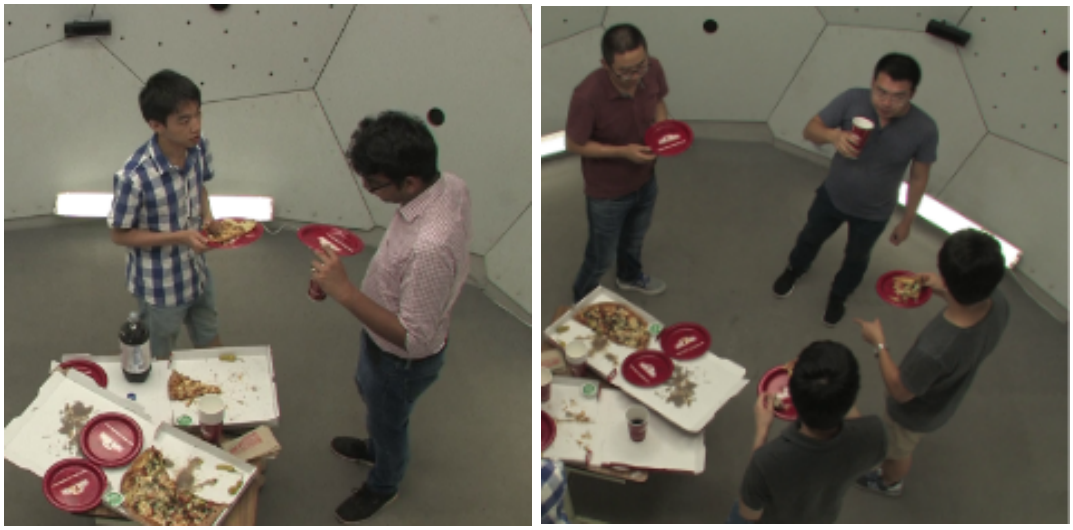

Figure 2.2. Example F-formation arrangements from CMU Pizza Party dataset. Left shows a vis-a-vis arrangement. Right shows a circular arrangement.

When dealing with context-aware computing, one should consider the spatial affinities between people in order to do an F-formation analysis [31]. These spatial factors become crucial when building applications for reacting to a scene change in human-computer interaction. For example, finite cues like a person's position, orientation, identity, and movement help indicate affinity and proxemic interaction. Ballendat et al. [41] studied how these factors take a role in a simultaneous interaction between multiple people to form an F-formation. They interpreted people's directed attention to other people as a measure to find the proxemic relations. However, the hardware design has been a challenge with the application of such models, whereas social dynamics are not explored. Albeit, Jungman et al. [42] researched the social counterpart and studied the relationship between different kinds of interactions with the F-formations formed from them. Some examples of these kinds of F-formations are L-shaped or face-to-face, where there is a cooperative objective in the interaction, whereas there is a competitive objective in the latter.

In another field of research, computer-supported cooperative work is studied, a comparison between face-to-face formation and sitting screen-facing formation is made to explore how the formations affected cooperative interactions on children. Suzuki and Kato [43] analyzed the difference between these two formations and how they changed the children's behavior when working on a collaborative goal. In another study by Morrison et al. [44], the difference between various F-formations formed during hospital ward rounds and how it affected the adoption of electronic patient records is examined. Another paper written by Marshall et al. [45] inspected the F-formations in a tourist information center and explained how different F-Formations encouraged or discouraged certain interactions. They showed how the physical facilities in the space accounted for the behavior of the individuals. Although none of these studies applied automatic F-formation detection in their works, they are a great example for indicating that spatial features are crucially necessary when analyzing F-formations.

Moreover, social robots also play a big role in human-robot interactions. Robotinho [46] is a human-like robot used as a tour guide and undertakes the role of a human tour guide. It guides people to form a semi-circular F-formation when viewing the exhibits in a museum, attends visitors, and explains the exhibits to them. Robothino does all of this by detecting people's bodies, legs, and faces using a built-in laser. Although it is unclear how F-formations are recognized in this work, Yousuf et al. [33] improve Robotinho by detecting an F-formation before explaining the exhibit. An algorithm detects the F-formations; however, it is very limited, only checking for a single formation side-by-side arrangement.

One of the widely popular previous works from Bazzani et al. [13] analyzed the F-formations by using the view frustum intersections of the people forming it. These view frustums are automatically inferred from the head orientation of each person using a head direction estimator. Since the head pose may become a challenging estimation, in cases where a person's head is very small in the image, they apply a rough estimator that estimates only four possible directions then extend it with a multi-class classifier for pedestrian detection. After that fine-grained estimation, they apply a polyhedron

view frustum to every person in 3D space. In their terms, this estimation is called Subjective View Frustum. From a sociological perspective, although it is possible to use an approximation of a person's head orientation as visual and cognitive attention, the more conventional way of detecting an F-formation is to add the body orientation and foot positions to the algorithm.

Furthermore, another practice that takes advantage of the view frustums came from Vascon et al. [4] where they also introduced a game-theoretic approach to the group detection problem. Authors propose a game-theoretic framework that models the uncertainty with the position and orientation of the people while considering the geometrical configurations of how F-formations formed naturally. They achieve this framework by the following steps. First, for every person in the frame, a view-frustum is fitted on their heads based on their position and head orientation. They provide an improved view frustum calculated by the combination of two probability distributions, a Gaussian and a Beta distribution. This modification results in higher performances and a speedup of the view frustum calculation algorithm. Given these improved view-frustums, they calculate a pairwise affinity matrix for each pair of people using the frustum overlap amount. From this affinity matrix and also integrating the temporal information to the calculations by smoothing the derived pair affinities across multiple frames, they compute a weight vector based on the theory of multiple payoff functions. This weight vector acts as a trade-off between the significance of the frames. Finally, by using evolutionary stable strategy clusters, they achieve the final F-formations for the given frame.

In another research proposed by Hung and Kröse [3], the F-formations are considered as a dominant-set cluster of the graph constructed from positions of the people in a scene. If we consider the newly created graph with edge weights as the affinity between participants, then a dominant set solves the maximal clique problem that provides a clustering of groups. The affinity between all nodes within the dominant set must be higher than those external to it. The maximal clique problem is also investigated by Pavan and Pelillo [47], where they applied a game-theoretic approach to cluster the

edge-weighted graph.

Among the graph-based solutions, Setti et al. [2] provide a method that uses efficient graph-cut to find the F-formations by pruning unsupported groups in each iteration with a prior. Their proposed algorithm starts by placing an arbitrary F-formation on the graph constructed from people's ground positions, orientations, and head/body poses. A hill-climbing optimization is used at every iteration to assign participants to possible F-formations depending on the efficient graph-cut-based optimization. Then, update the F-formations centers according to the result gathered from efficient graph-cuts while removing the impossible groups that do not obey a prior rule. This prior rule is called the 'Minimum Description Length' (MDL), and it is defined as follows. In simple terms, MDL is used to prevent the cases where a person blocks another by standing between the o-space center and them. Since this kind of positioning prevents the second person from gaining access to the o-space center of the F-formation, this grouping is removed due to 'Minimum Description Length.' The iterations continue until the convergence of the algorithm, which is guaranteed.

Lately, the research within the group detection field has re-emerged using Deep Learning algorithms. One of the most popular approaches by Swofford et al. [9] uses a novel Deep Affinity Network (DANTE) method to find the optimal clustering in a free-standing conversational setting. They turn the group detection problem into an affinity prediction problem where the network learns from data the appropriate affinity values between the nodes of the graph, which constitutes edge weights. The authors assume that the spatial configuration of the scene could be enough to calculate the relationships, and the relevant social context should be utilized to maximize the performance. They employ this idea by using the position and orientation of individuals to derive a context. They are inspired by Setti et al.'s work [5] where there can not be a possible group if a person is in-between the o-space of an F-formation and another person. However, the main contribution of their work is to find an affinity matrix and construct an edge-weighted graph from that to finally cluster nodes using the Dominant Sets algorithm. When finding the affinity matrix, they use two transform

functions: Dyad transform and context transform. Both of these transform functions use the power of Neural Networks. Firstly, Dyad transform, which consists of multiple independent multi-layer perceptrons, applies a non-linear function to input feature vectors of the intersection graph constructed from the social scene. This results in a feature encoding that helps cluster people based on their spatial features. Secondly, context transform computes a single global feature that characterizes the context of the social scene. Context transform also uses multi-layer perceptrons with dense layers followed by ReLU activation to compute a context feature vector. DANTE is one of the first methods that use Neural Network-based solutions for the group detection problem.

3. PROPOSED METHOD

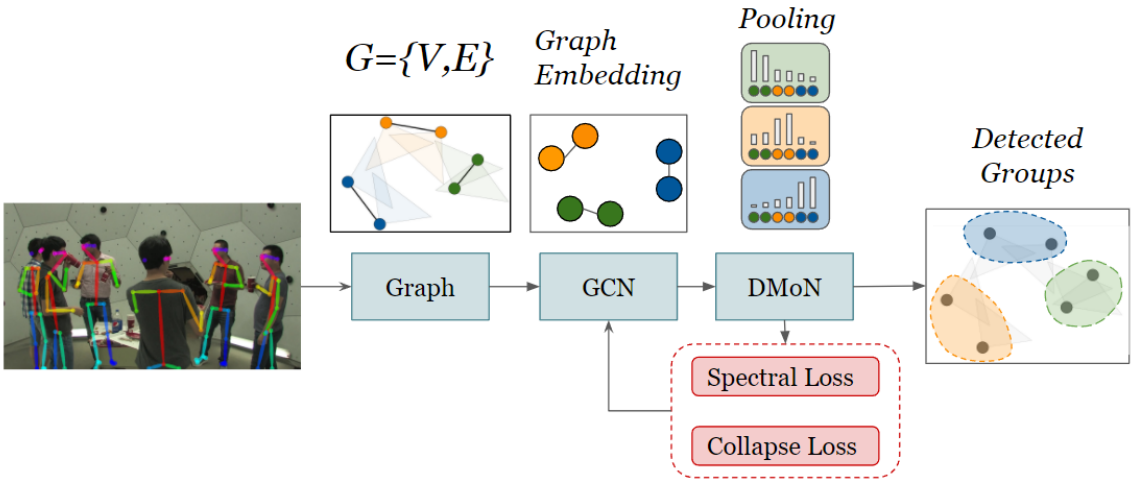

Figure 3.1. Main diagram of our method.

Our approach is based on 'Deep Modularity Networks' (DMoN) [35], which accomplishes graph clustering with Graph Convolutional Networks (GCN). Graph Convolutional Networks are simple yet effective message-passing networks that can help emerge various information on graphs. The striking power of using graphs is defined across many fields, including social sciences [48], natural sciences [49], and knowledge graphs [50]. Mainly, there are two different approaches to graph-related problems; semi-supervised learning and unsupervised learning.

Semi-supervised learning handles the general problem of learning from labeled and unlabeled data. This type of learning is mainly used on node classification on graphs. Utilizing the graph structure of data enables learning with very few labels. Most graph-based research that uses semi-supervised learning assumes a cluster structure that implies that adjacent nodes on a graph share common features. This assumption is natural for most of the datasets available in graph formats. For example, in a citation network, the citation links between documents describe their citation relations, connecting papers within similar research domains. Many semi-supervised learning approaches aim to represent this network structure's clusterability as well as possible.

On the other hand, unsupervised methods solve the problem of learning different representations from given data features and adjacencies between graph nodes. Unsupervised learning with graphs is used in various fields featuring community detection, graph clustering, graph embedding, and anomaly detection. Many researchers try to solve these problems by using dimensionality reduction. Most of the data used in these fields contain high-dimensional data. However, the vast volumes of data present several obstacles and render traditional methodologies ineffective in real-world applications. Instead of using traditional methods, dimensionality reduction achieved through Graph Neural Networks comes as a more helpful resource.

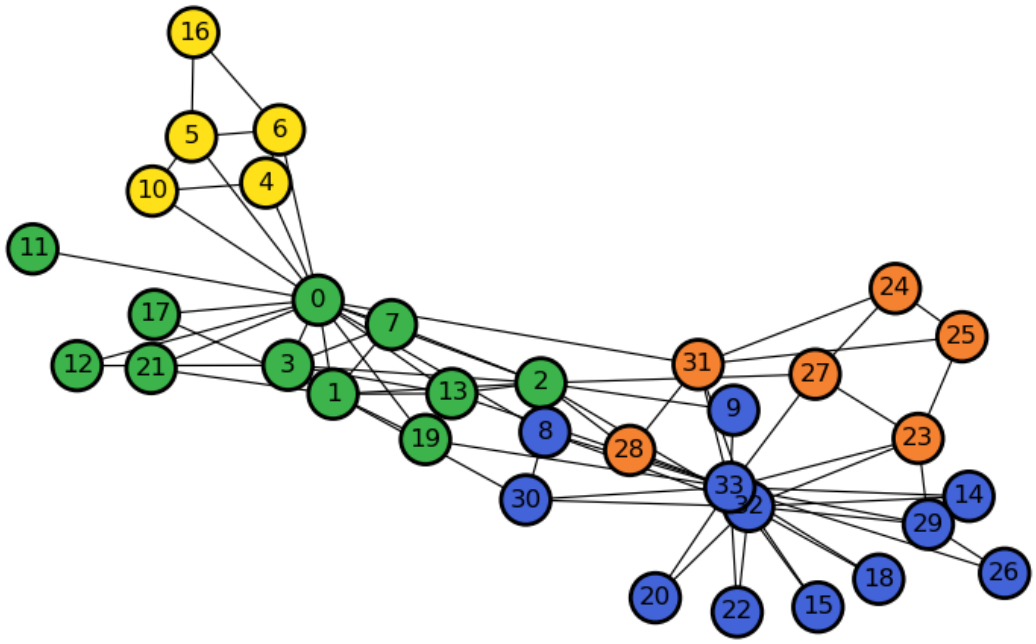

Figure 3.2. Zachary's karate club network [1] showing the community structure. The nodes are labeled into four different groups based on their decisions on the financial conflict between the primary instructors.

DMoN exploits the power of the Graph Convolutional Networks to solve the unsupervised community detection problem. In its essence, it extracts the graph embedding vectors from the graph's node features which enables more straightforward clustering of the data. The datasets mainly used in the unsupervised clustering field are relational

datasets with numerous nodes. For example, Cora [51] is a dataset of 2708 scientific publications roughly classified into seven classes. Citeseer [52] is another publication network consisting of 3312 scientific publications with 4732 links. Graph Convolutional Networks are highly efficient when approaching unsupervised clustering problems with large networks. Another high-efficiency use case for the GCNs is for community detection. Starting the seeds as early as 1977, Zachary introduced the historically famous community dataset Karate Club [1]. With the increasing popularity of social network websites such as Facebook, Google+, and Twitter, there has been a demand increase in community research. SNAP (Stanford Large Network Dataset Collection) [53] contains various datasets created from the connections and information gathered from these websites. Many different GCN methods have been used on these datasets with outstanding results. Since our problem can be simplified as a community detection problem, we use GCN-based DMoN to detect the free-standing conversational groups.

In order to use DMoN, which is GCN-based, we need to convert our input datasets into a graph format. We use the idea of using view frustums as an indicator of affinity between individuals. We construct a graph for each scenario in the datasets from this affinity information. The rest of this chapter is structured as follows. Section 3.1 explains how we construct the graph from input data, and Section 3.2 provides information about the method we use DMoN.

3.1. Conversational Graph Construction

A scene taken from a video is usually described with each person's position in 2D or 3D coordinates or with their human joint locations in each frame. In our work, we recreate each frame as a graph to visually analyze group distributions. Specifically, we construct an undirected graph $G = (V, E)$ on a single sequence with V nodes and E edges featuring people's positions and their relationships with each other. In this graph, the node-set contains all the human beings present in the current frame. The feature vector on a node $F(v_i)$ holds the person's position in a 2D vector which corresponds to a top-down view coordinate. Some example graphs created from CMU Pizza Party and

SALSA dataset ground-truth annotations can be seen in 3.3, 3.4, and 3.5 respectively. Since these graphs are generated from ground-truth annotations, we have to come up with a new way that can produce graphs when there are no ground-truth f-formations available.

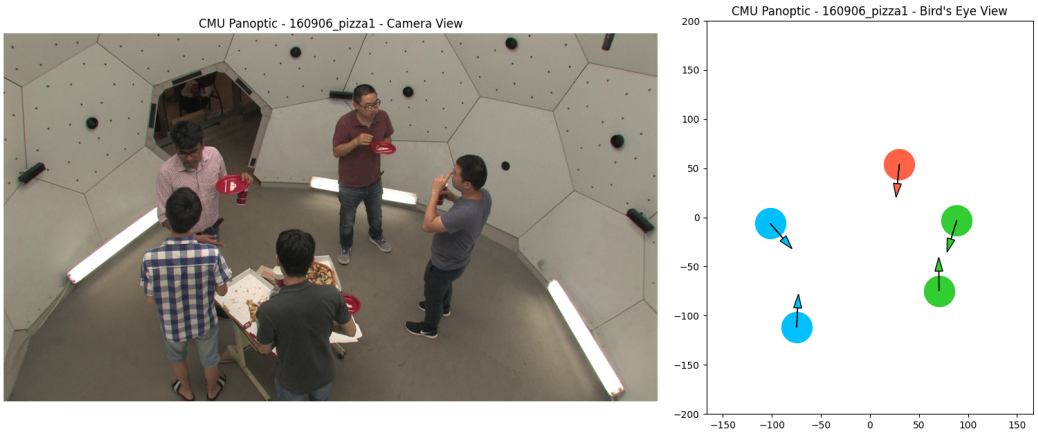

Figure 3.3. Example ground truth formation and a camera view of the same frame side-by-side taken from CMU Pizza Party dataset.

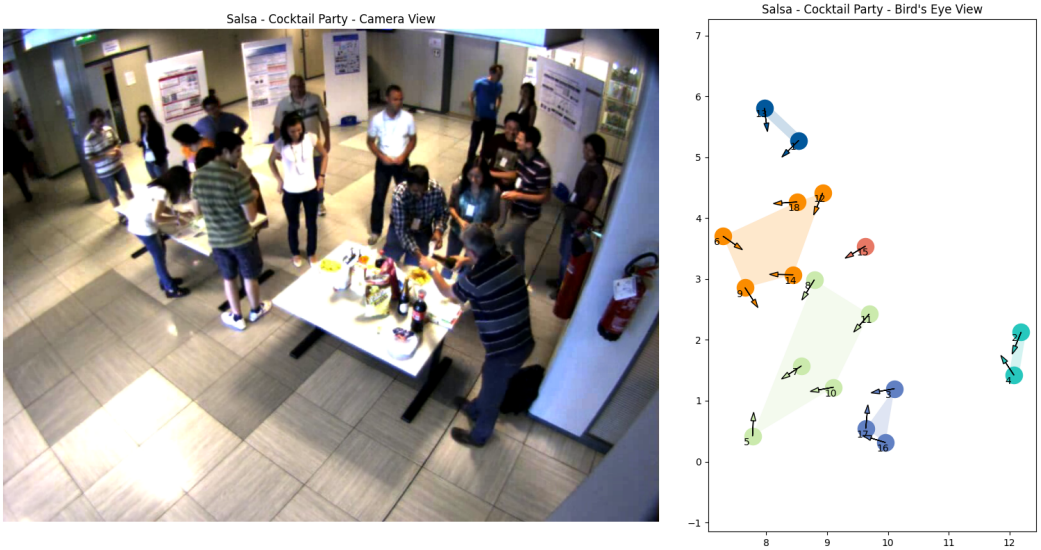

Figure 3.4. Example ground truth formation and a camera view of the same frame side-by-side taken from SALSA Cocktail Party dataset.

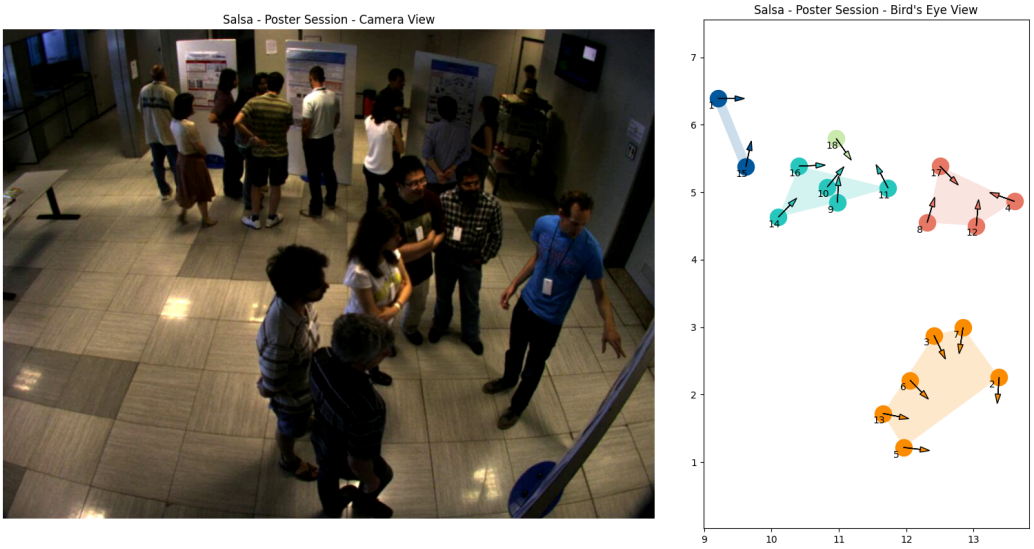

Figure 3.5. Example ground truth formation and a camera view of the same frame side-by-side taken from SALSA Poster Session dataset.

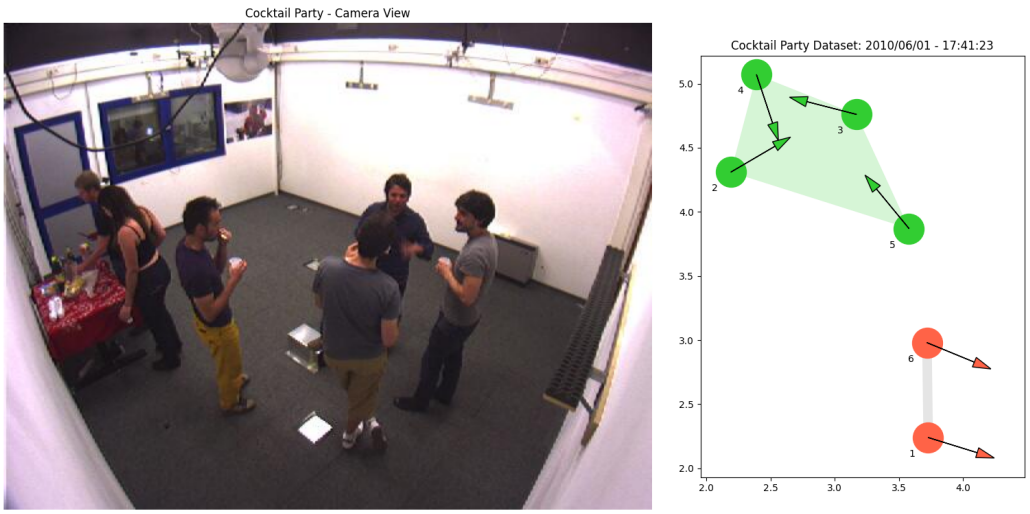

Figure 3.6. Example ground truth formation and a camera view of the same frame side-by-side taken from Cocktail Party dataset.

We construct our graph from people's locations in two steps. First, we calculate pairwise Euclidean distances between all individuals. This distance inversely

contributes to the weights of edges in the graph. Considering $\tilde{w}_{ij}$ the candidate edge weight between nodes i and j, it is calculated from:

$$\tilde{w}_{ij} = \frac{\max(\mathbf{A})}{d(n_i, n_j)} \tag{3.1}$$

where $\mathbf{A}$ is the pairwise Euclidean distance matrix and $d(n_i, n_j)$ is the distance between i and j nodes.

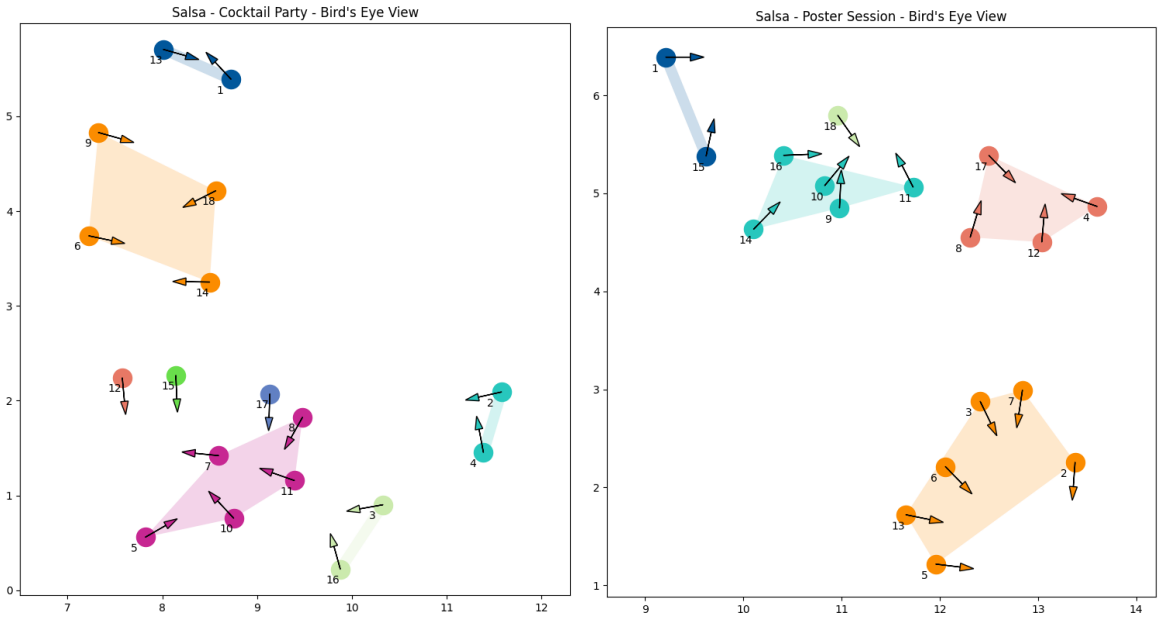

Figure 3.7. Example graphs created from ground-truth annotations of SALSA dataset. Left: Cocktail Party, right: Poster Session.

To calculate the finalized edge weights, we first consider a triangular view frustum for each person. This frustum starts out from the base location of the individual and stretches out to l length with α degrees to each side 3.8. These parameters are tuned differently for each dataset available to achieve the best performance possible. We calculate the area of intersections of view frustums between people using Sutherland-Hodgman convex polygon clipping algorithm [54] 3.9. And then, using the Shoelace formula, we can calculate the area of the resulting polygon after clipping. After calculating the area of intersection between view frustums, we finalize edge weights by

multiplying this value with the intermediate edge weight:

$$w_{ij} = \frac{\max(\mathbf{A})}{d(n_i, n_j)} \times \frac{1}{2} \left| \sum_{i=1}^{N-1} (x_i y_{i+1}) + x_n y_1 - \sum_{i=1}^{N-1} (x_{i+1} y_i) - x_1 y_n \right|, \tag{3.2}$$

where x_i and y_i are the vertex positions of the final clipped polygon.

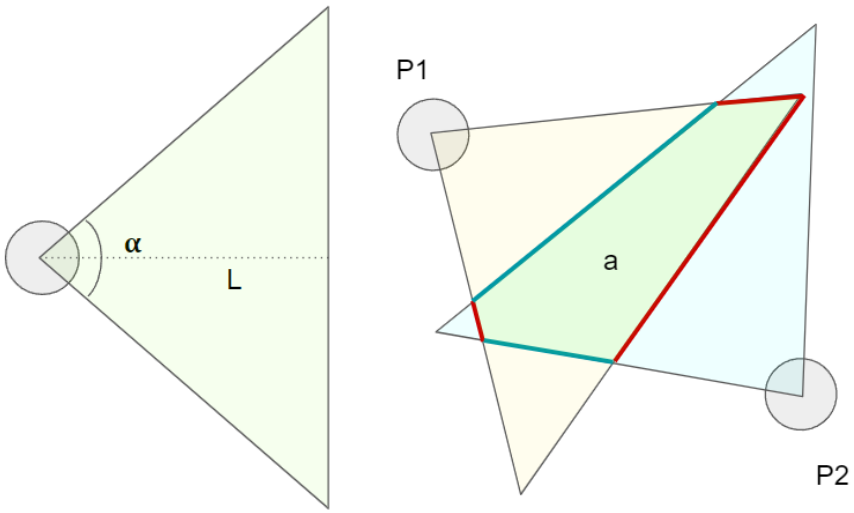

Figure 3.8. Showcasing the triangular view frustum. Left: shows an example view frustum of a single person where L is the length of frustum and α is the frustum angle. Right: shows the intersection of two view frustums. Red and cyan lines represent the edges obtained from Sutherland-Hodgman algorithm with respect to P1 and P2.

Require $\mathbf{v_i^1},\ i = 1, 2, \ldots, N$ the vertices of the clipping polygon
Require $\mathbf{v_j^2},\ j = 1, 2, \ldots, M$ the vertices of the subject polygon
Create new polygon P' from clipping polygon
for clippingEdge $[v_i^1, v_{i+1}^1], i \to (N-1) \in P'$ **do**
  Initialize empty P'' array
  for subjectEdge $[v_j^2, v_{j+1}^2], i \to (M-1)$ **do**
    Compute intersectionPoint between clippingEdge and subjectEdge
    if clippingEdge inside subjectEdge **then**
      $P'' \overset{+}{\leftarrow} v_{j+1}^1$
    end if
    if v_j^2 inside clippingEdge **and** v_{j+1}^2 outside clippingEdge **then**
      $P'' \overset{+}{\leftarrow}$ intersectionPoint
    end if
    if v_{j+1}^2 inside clippingEdge **and** v_j^2 outside clippingEdge **then**
      $P'' \overset{+}{\leftarrow}$ intersectionPoint & v_{j+1}^1
    end if
    $P' \leftarrow P''$
  end for
end for
return P'

Figure 3.9. Sutherland-Hodgman Polygon Clipping Algorithm.

3.1.1. Node Features

When constructing the graph, we need to define a feature vector for each node in the graph. These node vectors are similar to features obtained from convolutional filters in convolutional neural networks. The features obtained from convolutional filters represent local patterns in images. Whereas in the graph case, these node vectors can be assigned by the user. Therefore, there are multiple ways of initializing node

features.

Firstly, in traditional unsupervised GNN and GCN training, the node feature matrix is initialized as an identity matrix where each node is distinct. This way of initializing feature vectors assures that each node has features that are different from one another. As the GCN training continues, each node will learn its features in addition to the features of its neighboring nodes.

Furthermore, in DMoN, the node features are gathered from the information available on the datasets. For example, datasets that contain citation networks provide abstracts for the published material. In this case, they used published papers as nodes where the bag-of-words abstracts are used as node features. For different datasets such as Amazon PC and Amazon Photos, where the graph represents the co-purchase of Amazon PC products, the node features are represented by bag-of-words product reviews. In another dataset that contains co-authorship networks, each node is an author, and the node features are the keywords collected from papers of the authors.

Consequently, we use this idea of using information gathered from the dataset when initializing the node feature matrix. In our datasets, we have the 2D configuration of each frame where the location of each person is available. We use the 2D position of each person as their node feature. For each person, we create a 2D feature vector that creates an $N \times 2$ node feature matrix.

Moreover, we analyzed the advantages of crafting the feature matrix using the position information compared to naively initializing it as an identity vector. We used our augmented CMU dataset as the baseline for comparing both initialization routines. We have found that using the latter method yielded up to 3% increase in $T = 1$ F1 scores over all of the folds on the SALSA dataset with Poster Session and Cocktail Party on Table 4.7 and 4.8 respectively.

3.1.2. Edge Weighting

As explained in Section 3.1, we construct graph edges using triangular view frustums. We get inspiration from one of the previous methods that also utilize the notion of view frustums. We want to initialize weights of edges to be proportional to how much people are interacting. This way, we can make sure that our Graph Convolutional Network can learn the ideal graph structure. Other alternative methods use different edge weight initializations.

Previously Hung and Kröse [3], in their method, acknowledged an F-formation as a dominant-set cluster of an edge-weighted graph. The edges between two nodes (person) measure the affinity between pairs. They calculate the proximity between two persons using a symmetric distance function. Considering a pair of nodes i and j:

$$A_{ij}^{prox} = e^{-\frac{d_{ij}}{2\sigma^2}} \tag{3.3}$$

where d_{ij} is the euclidean distance and σ is the variance of pairwise distances depending on the dataset (approximately two meters for all datasets). A_{ij}^{prox} gives a relative proximity value between i and j node pairs.

However, just using distance as a proximity metric may produce some errors as the location gets more crowded. Hung and Kröse [3] improve their affinity matrix using people's orientation. They multiply the angle difference between head orientations of two people with the proximity value. This is especially important in times when the pair of people are close to each other while facing the opposite directions. This kind of gathering would be undesirable when constructing an F-Formation with only the position in the proximity equation.

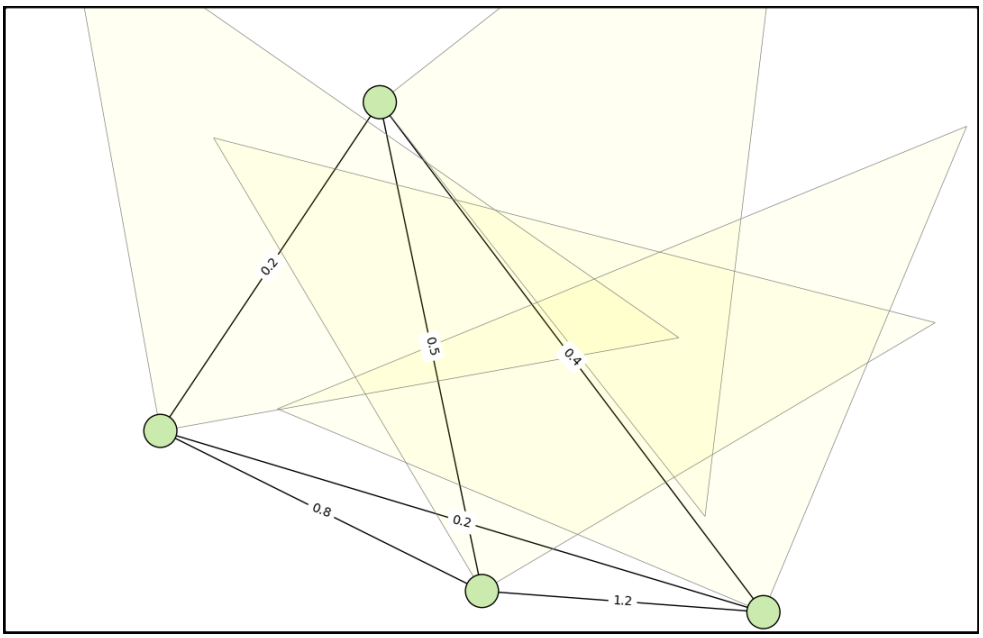

Figure 3.10. Example calculated edge weights from a group in SALSA Poster Session dataset.

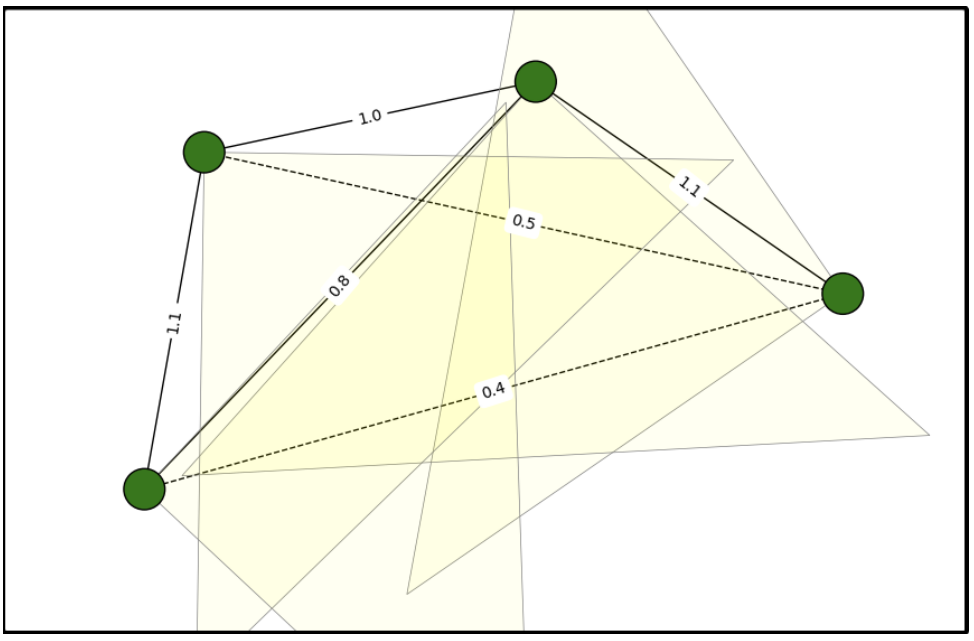

Figure 3.11. Edge weight calculation for an example group formation. The edge pruning tolerance is set at an arbitrary value of 0.6. The dashed edges indicate edges to be removed from graph.

One of the frustum-based methods, Bazzani et al. [13] focuses on using people's head orientations to induce a 3D view frustum. These view frustums are used as an approximation of an individual's Focus of Attention (FoA). They combine the idea of proximity information with the intersections of view frustums to find the interactions between people.

They use a 3D view frustum polyhedron that spans 120° in both directions. We use this idea of view frustums in our approach. However, since not all of our datasets consist of 3D head-pose vectors, we use a 2D view frustum. To find the optimal viewing frustum length, we explore the optimal length by changing it gradually between 0.5 and 2.0 units for each dataset individually. Furthermore, we also experiment with frustum angle between 60° and 150° in both directions.

Finally, we calculate the weight of the edges with the following equation:

$$w_{ij} = \frac{A_{ij}^{prox} \cdot I_{F_i F_j}}{(L^2 \cdot \tan \frac{\theta}{2})} \tag{3.4}$$

where $I_{F_i F_j}$ denotes the intersection area between frustum of i^{th} and j^{th} nodes. L is the frustum length, and θ is the frustum angle on a single side. $L^2 \cdot \tan \frac{\theta}{2}$ gives the area of a single person's frustum.

3.1.3. Edge Pruning

As a final touch to our weighted graph, we propose an idea called 'edge pruning.' We observe that if we feed a completely connected weighted graph to GCN, the embedded representations of the nodes do not get separated for optimal clustering. Therefore, we cull the edges that contain very low affinity between people. We introduce a new variable called "edge weight threshold" and remove the edges under this threshold in every graph. By doing this, we improve the clusterability of the graph embeddings, as seen in 3.12.

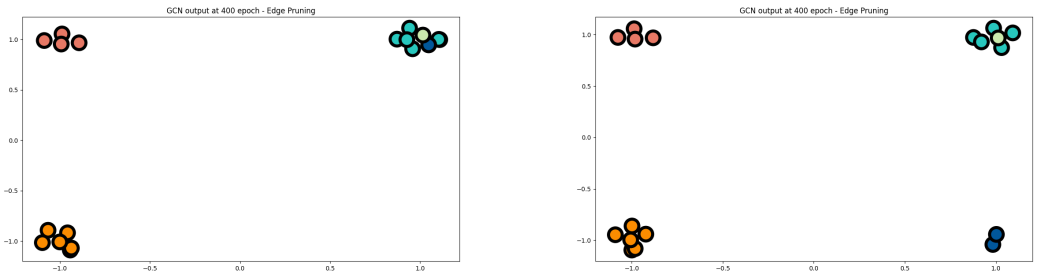

Figure 3.12. Graph embeddings of nodes at 400^{th} epoch of GCN training on SALSA dataset. Left: shows graph embeddings without edge pruning, right: shows with edge pruning. Darker blue group is separated from cyan group which shows the effect of edge pruning.

3.2. Deep Modularity Networks - DMoN

Deep Modularity Networks (DMoN) is an unsupervised pooling method that supersedes the methods that use modularity to measure clustering quality. It undertakes the challenging problem of clustering massive real-world graphs. DMoN is able to simultaneously use the signal from the node attributes and graph structure as an advantage. With that, DMoN displays state-of-the-art result clusters in real-world data, which compare significantly with ground truth labels.

3.2.1. Graph Convolutional Networks

Graph neural networks (GNNs) are a type of neural network that works with graph-structured data in a natural way. Compared to models that analyze individual entities in isolation, GNNs may generate more educated predictions about entities in these interactions by extracting and exploiting characteristics from the underlying graph.

Many systems and interactions may be represented as graphs, including social networks, molecules, organizations, citations, physical models, and transactions. However, the real challenge is doing computations by using a graph. Since graphs are very adjustable mathematical models, they lack consistency in structure between instances.

It is not trivial to represent graphs so that processors can compute over and relies heavily on the problem to achieve a meaningful result. Also, there is no intrinsic ordering among the nodes in many graphs. In contrast with an image, where each pixel is individually identified by its absolute position, graphs do not contain a grid-like structure, making it hard to define the ordering between the nodes.

The most straightforward GNN architecture usually consists of a differentiable model used as the update function. This type of structure is, called the 'message passing neural network' proposed by Gilmer et al., is the foundation of the GNNs. This framework applies the differentiable update function for each node vector and outputs a learned embedding. Edges and global-context vectors can have features just like nodes. The exact process is repeated for edges and global-context vectors to achieve edge embeddings and a single embedding for the entire graph.

The most common architecture for Graph Neural Networks contains a convolutional pooling function since the parameters used are generally shared for all the locations in the graph. Nevertheless, the convolution on graphs is not as simple as it is on images. Therefore, Kipf et al. came up with a 'fast approximate convolution' on graphs that produce the traditional convolution's intended result.

The method they provide requires node features (x_i) and adjacency matrix (A) as input. First, they modify the adjacency matrix by adding an identity matrix so that while summing up the feature vectors of the neighboring node in order to not lose the information from the self node. This is called adding a self-loop to the graph $\hat{A} = A + I$. Another modification they apply is that the adjacency matrix of a graph requires normalization before applying it directly because if it is multiplied directly, the feature vectors of the initial graph will be scaled out of proportion. The adjacency matrix is normalized with the inverse of the degree matrix to combat this issue $\hat{D}^{-1}\hat{A}$. In practice, symmetric normalization is applied rather than having naive averaging of

neighboring nodes $\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}$. The final aggregation function in this case becomes:

$$f(H^{(l)}, A) = \sigma(\hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{3.5}$$

where $H^{(l)}$ and $W^{(l)}$ are the feature vector and the network weights at the l^{th} layer of the network and σ is a non-linear activation function, usually a $ReLU$.

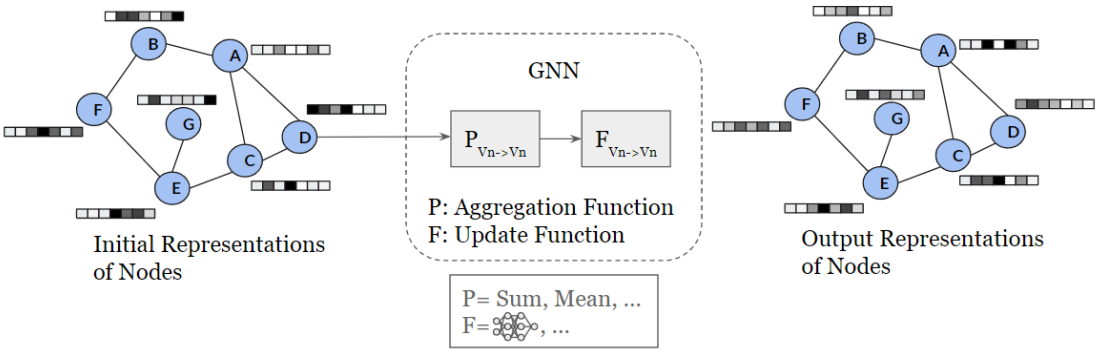

Figure 3.13. A GNN architecture diagram, which updates node representations of a graph by aggregating neighboring nodes.

At its core, DMoN uses GCN with minor modifications to obtain soft cluster assignments. They change the traditional GCN architecture by introducing a trainable skip connection instead of adding self-loops to the graph structure. Another adjustment they make is to use SeLU activation instead of ReLU for improved convergence [55].

3.2.2. DMoN Network Architecture

We combine the abilities of a graph clustering method called Deep Modularity Networks (DMoN) [35] with our constructed graphs to improve group detection. DMoN is an approach used in the social network clustering problem which proposes an unsupervised clustering objective function that optimizes soft cluster assignments.

The primary objective function of DMoN is to maximize the modularity [56] of the graph. Modularity is a metric that assesses how well a network can be divided

into clusters. High-modularity networks feature powerful connections between nodes inside clusters and weak connections between nodes in different clusters. In optimization approaches for finding community structure in networks, modularity is commonly utilized. It compares the given graph with a fully random graph to evaluate the significance of clustering. In a random graph, given nodes i and j with degrees d_i and d_j, the expected connection probability would be $\frac{d_i d_j}{2m}$, where m is the total number of edges in the graph. The divergence between the intra-cluster edges and the anticipated one is measured by:

$$\mathcal{Q} = \frac{1}{2m} \sum_{ij} \left[\mathbf{A}_{ij} - \frac{d_i d_j}{2m} \right] \delta(c_i, c_j), \tag{3.6}$$

where $\delta(c_i, c_j) = 1$ if i and j are in the same cluster and zero otherwise.

However, maximizing modularity is an NP-hard problem [57] and cannot be solved feasibly for more comprehensive networks. Instead, the authors of DMoN choose to approach the problem with a spectral relaxation to solve it efficiently [58]. The modularity $\mathcal{Q}$ can be reformulated as:

$$\mathcal{Q} = \frac{1}{2m} \mathrm{Tr}(\mathbf{C}^\top \mathbf{B} \mathbf{C}), \tag{3.7}$$

where $\mathbf{B}$ is the affinity matrix defined as $\mathbf{B} = \mathbf{A} - \frac{\mathbf{d}\mathbf{d}^\top}{2m}$ and $\mathbf{C} \in 0, 1^{nxk}$ is the cluster assignment matrix. The optimal cluster assignment matrix $\mathbf{C}$ is found from the top-k eigenvectors of the affinity matrix $\mathbf{B}$, maximizing the modularity $\mathcal{Q}$. The solution can be optimized quickly with iterative methods such as power iteration or Lanczos algorithm [35].

Finding a spectral maximum has a massive flaw when using gradient-based optimization methods. We can produce a locally optimal solution that disables further learning by assigning every node to a single cluster. The main contribution of DMoN is to regularize cluster assignments so that this situation can be circumvented. They introduce 'collapse regularization' in their objective function, which prevents the opti-

mizer from being trapped in a local maximum while not limiting the optimization of the main objective as follows:

$$L_{DMoN} = -\frac{1}{2m}\mathrm{Tr}(\mathbf{C}^\top \mathbf{B}\mathbf{C}) + \frac{\sqrt{k}}{n}\left\|\sum_i \mathbf{C}_i^\top\right\|_F - 1, \tag{3.8}$$

where the Frobenius norm of the cluster membership counts $\left\|\sum_i \mathbf{C}_i^\top\right\|_F$, normalized to the range [0, 1], is used as the regularizer. This value will be maximized when all nodes are assigned to a single cluster and will be zero when each cluster has the same number of nodes. They also added a dropout layer after obtaining the latent representations of GNN to further prevent the optimizer from getting trapped in the local optima of the objective function [59].

4. EXPERIMENTAL RESULTS

4.1. Datasets

We present our annotations of a publicly available dataset on top of five other datasets we use to compare with other methods. We annotated the 'Pizza Party' setting under the 'Special Events' category of CMU Panoptic Studio Dataset [36]. For other datasets, we rely on the ground truth annotations given by their authors. These datasets are: Synthetic Dataset and Coffee Break from [60], Cocktail Party from [5], Salsa Cocktail Party and Poster Session from [10].

4.1.1. CMU Panoptic Studio

CMU Panoptic Studio dataset provides a way of capturing the 3D motion of multiple individuals engaged in social interaction. A dome-shaped geodesic structure is built to capture footage from inside the dome, with a total of 511 cameras covering it. The structure is held by 40 hexagonal, six pentagonal, and ten trimmed panels, where each hexagonal panel contains 24 VGA cameras. Also, there are 10 Kinect sensors that provide point cloud data to create more robust keypoint detection [36].

The annotations consist of 3D joint locations in COCO19 format [61] of each individual in the frame on top of camera calibration matrices, face, hand [62], and feet keypoints, Kinect RGB-D cloud point data.

As of right now, there are eight main categories of recorded video. Some video types such as: 'Range of Motion', 'Musical Instruments', and 'Dance' mainly consist of a single person interacting with the environment. As for other categories, 'Ultimatum' and 'Mafia' are social games where multiple people engage in role-playing. Although these videos introduce more than one person and have free-standing conversational groups, they consist of only a single group which is not feasible for our research. Simi-

larly, the 'Haggling' and 'Toddler' categories consist of three people playing a bargaining game and mom/toddler interaction, respectively. We choose to work on the 'Special Events' section as we think it qualifies as an example of a free-standing conversational group context.

Among other videos among 'Special Events', we select '160906_pizza1' which includes the footage of six people having a pizza party while having conversations with each other. We discard videos taken in an office setting, car repair, and sports practice because of not having enough people to form an FCG. On top of that, we ignore meetings and various social games because of having only a single group throughout the whole sequence. From now on, we will refer to the video called '160906_pizza1' as 'Pizza Party' for clarity.

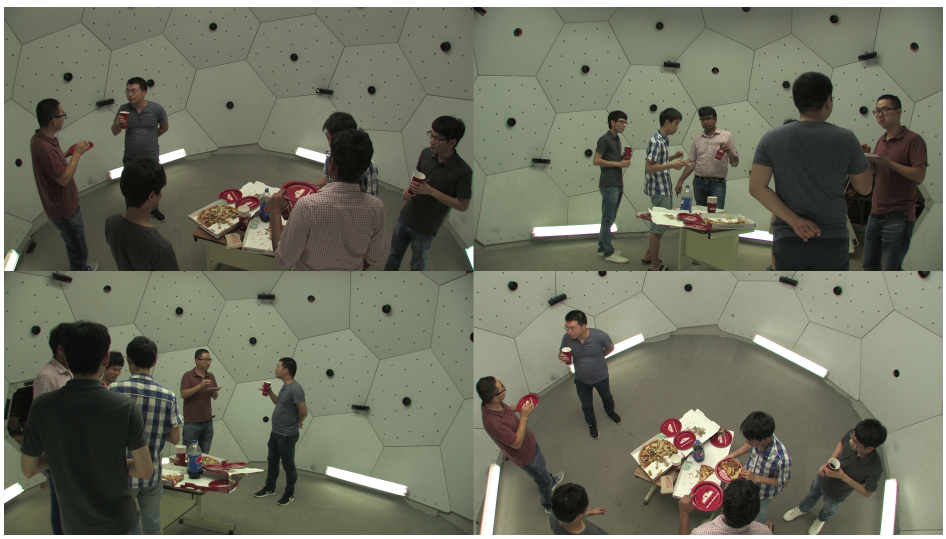

Figure 4.1. Example HD video angles taken from CMU Pizza Party video.

Pizza Party video lasts for five minutes and has HD quality videos available in 31 different angles. We annotate FCGs throughout the video at every 30^{th} frame (one second) while inspecting people's positions with four main camera angles (cam no. 0, 8, 15, 23). With this method, we gather 220 annotated frames out of 6582 available. User interface of our annotation tool can be seen in 4.2.

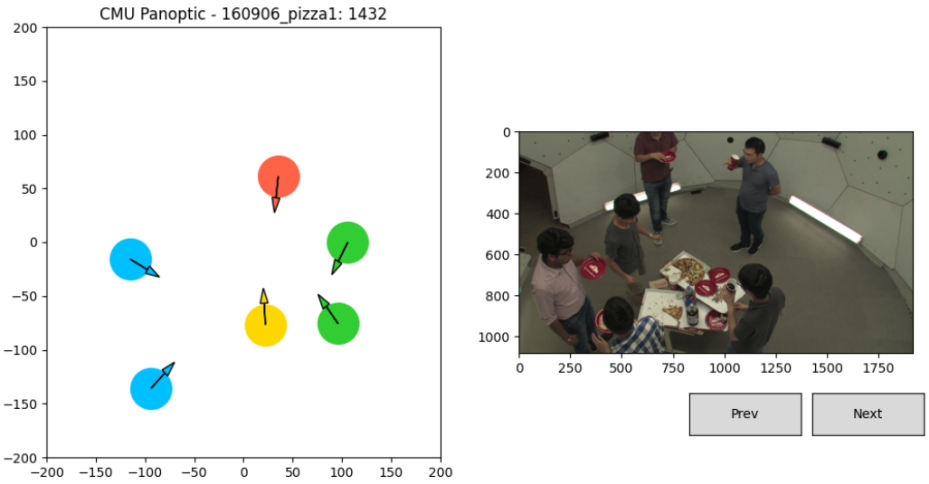

Figure 4.2. Interface of our annotation tool. 'Previous' and 'Next' buttons change camera positions. Video can be skipped by one second using right and left keyboard keys. To annotate groups, we can select people from the left view with mouse pointer.

4.1.2. Synergetic sociAL Scene Analysis (SALSA)

The SALSA dataset is an exceptional resource for researching human social interactions in multiple variations. The authors provide visual data from four different camera angles as well as audio and accelerometer sensor data. Besides sensor outputs, the dataset also contains f-formation annotations (annotated in three seconds intervals) personality scores for each big-five personality trait [63] of every individual.

There are two recordings available with 30,000 (Poster Session) and 25,000 frames (Cocktail Party) which last for a total of 60 minutes. At each scene, the body positions and head orientations of 18 individuals are tracked using a Hybrid Joint-Separable particle filter (HJS-PF) [64]. Using these tracking results, they annotated the position, head, and body orientation of each target in a semi-automatic fashion every 45 frames (three seconds). These annotated data were then used to deduce the F-Formations. The rule for annotating F-Formations is that if an individual's body/head orientation is converging to an *o-space* without any obstructions, then he is considered as a member of that group. Example scenes taken from Poster Session and Cocktail Party videos

taken at 10^{th} minute can be seen in 4.3 and 4.4. There are 645 annotated frames for Poster Session and 500 for Cocktail Party.

Figure 4.3. Example frame taken from Poster Session video of SALSA dataset showcasing four different camera angles.

Figure 4.4. Example frame taken from Cocktail Party video of SALSA dataset showcasing four different camera angles.

4.2. Performance Evaluation Metrics

This section will introduce the performance metrics used to measure the quality of the detected groups. Remarkably, each study in the literature defines the term 'group' differently, and there is no agreed-upon formal definition. As a result, many detection systems have improvised figures of their particular group definition, making it notably challenging to test comparatively with various algorithms. Therefore, we use specific group detection metrics that are agreed upon in most of the previous methods.

Firstly, a generic group clustering quality measure was introduced by Cristiani et al. [6], stating that a correct estimation of a group is if at least $\lceil (T \cdot |G|) \rceil$ of their members are detected correctly, where $|G|$ is the cardinality of the group G and T is the *tolerance threshold*. Cristiani et al. [6] chose the *tolerance threshold* as $2/3$ and calculated precision and recall metrics accordingly. Setti et al. [2] improve this idea by choosing the threshold as a free variable and calculating the area under the curve (AUC) for F_1 vs. T graph with T varying from $1/2$ to 1. Though, they are particularly interested in two values of T: $2/3$ and 1.

In 2019, Setti et al. [37] introduced the GRODE metrics to consider the ideal group detector behavior. They state that the ideal group detector must contain some crucial properties. These properties include; invariance to 'Group cardinality,' 'Group number,' 'Group appearance,' and 'Singletons.' Essentially, the ideal group detector should be unbiased to group cardinality the appearance of individuals and should detect all groups present in the scene even if it consists of single members. Setti et al. [37] also define essential properties for group detection metrics. These include Compactness, Generality, and Essentiality. The ideal group detection metric should consist of only a few free parameters to have any rule applied on top of it will be simple to be reproduced. It should also be general enough to apply to any domain of group detection methods in the literature.

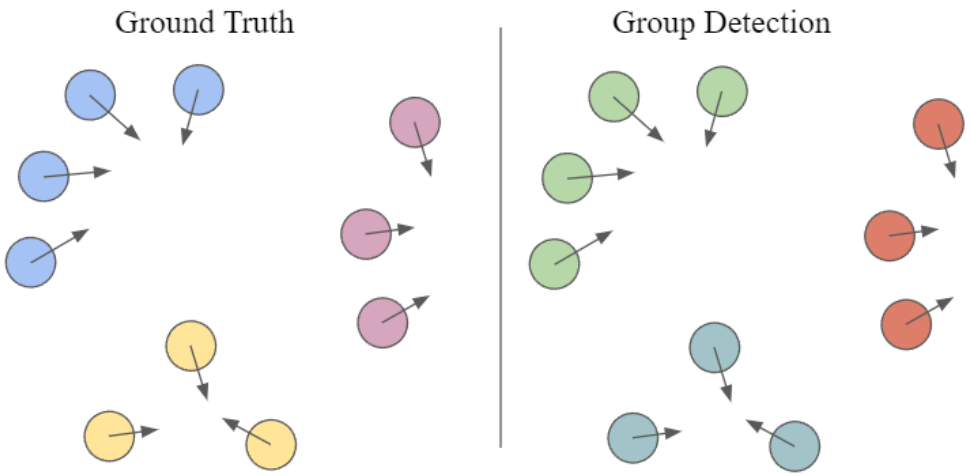

Figure 4.5. Case when tolerance threshold is 1. All detected groups must match exactly to the ground-truth groups.

4.3. Effect of Node Features

As explained in Section 3.1.1, we experiment with the initialization of the input feature matrix in two different ways. First, we experiment with initializing the node feature matrix as an NxN identity matrix where N is the number of nodes in the current graph. This way allows initializing each node feature as a different one-hot encoded vector, essentially initializing each one as a unique feature. This initialization method designates each person as a separate individual, which forces the GCN to learn from the adjacency matrix alone. It is common to use this technique in cases where nodes represent individuals, and no identifying feature can be extracted from the dataset. Kipf et al. have used this initialization of the feature vectors when training the Karate Club dataset with GCNs.

Another way of setting the input feature matrix is to use 2D person position information supplied by the datasets. When initialized this way, we have a $2xN$ node feature matrix that contributes to people's affinity. Instead of having each individual as unique, we consider people closer to each other as more similar in the input matrix. We then experiment with both ways of constructing the input features and analyze

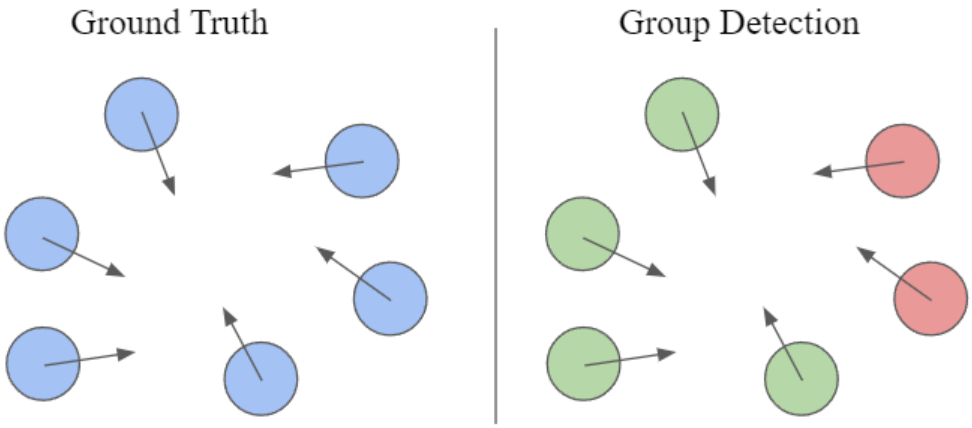

Figure 4.6. Case when tolerance threshold is 2/3. Lowered tolerance value allows misdetections of people in groups. In this example, there are six people in a single group in the ground truth. Therefore with 2/3 tolerance threshold even though two people are not detected in the same group in the detection, this case is considered as correct.

the effect on the F1 Score with tolerance threshold at both $T = 1$ and $T = 2/3$. We use the SALSA dataset as our experimentation dataset. The results are in 4.7 and 4.8 for the SALSA Poster Session and Cocktail Party datasets respectively. Here, we can conclude that having node features as people's positions improved the F1 Score on the Poster Session subset. However, there was no significant improvement with the Cocktail Party dataset.

4.4. Effect of Edge Pruning

In Section 3.1.3, we introduce a novel idea of pruning edges with low weights when building the input graph for GCN from given datasets. We observed that if a completely connected weighted graph is fed to the GCN as input, the network will have difficulty separating the nodes into clusters in embedded space. Therefore, as a solution, we introduce an edge weight threshold that acts as a regularizer for the GCN input. Edges under this arbitrary threshold represent very weak affinities between

people and are sometimes misleading. We explore different settings of the edge weight threshold between 0 to 0.2, where at 0, there will be no edge pruning. We remove the edges below this threshold and analyze its effect on the F1 Score on the SALSA dataset. The results are given in 4.5 and 4.6 for the SALSA Poster Session and Cocktail Party datasets respectively. We observe that in Cocktail Party, there is a slight improvement in F1 Score at $T = 1$ from initially 48.36% to 51.6% with a 0.2 edge weight threshold. However, there is no clear evidence that edge pruning improves the F1 Score for Poster Session.

4.5. Temporal Fusion Post-Processing

As we explained in Chapter 2, the previous methods that concentrate on the social group detection problem do not use the temporal information that resides innately within the videos. Therefore, one of our novel contributions in this paper is to use this temporal information as a post-processing step. The nature of the social groups indicates that when a group or a bond is formed between individuals, the group participants do not change rapidly. These groups tend to last for a certain time before breaking up and forming another group elsewhere. We observe this condition in videos where the groups are more static such as a poster session. In a more dynamic environment such as the cocktail party, the groupings lean towards a shorter lifespan.

Another observation that we have is that the output of our neural network can have noisy inputs. If the network is unsure about the assignment of a person in a group, it can lead to wrong assignments in some cases. Although the prediction probability is not high, since we are assigning groups using a single frame at a time, the probability of him being in the other group may win by a small margin. We apply a smoothing function through the temporal dimension that fixes these noisy group assignments to prevent these cases.

We use exponential moving averages to smooth out the noisy predictions in the time domain. Even though it is a simple approach and more popularly used with

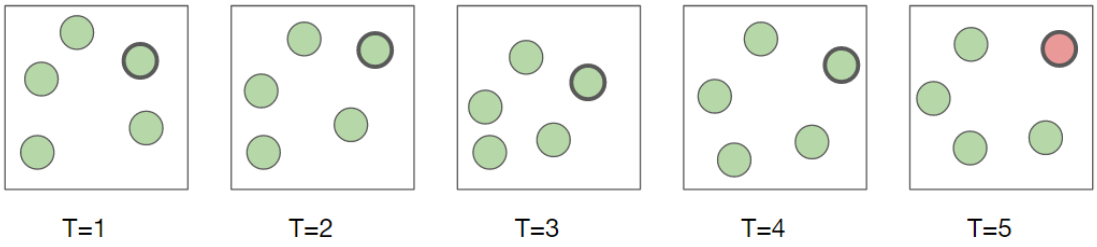

Figure 4.7. Example scene showing noisy detection at time T=5. Red indicates that the neural network assigned that individual to a different group.

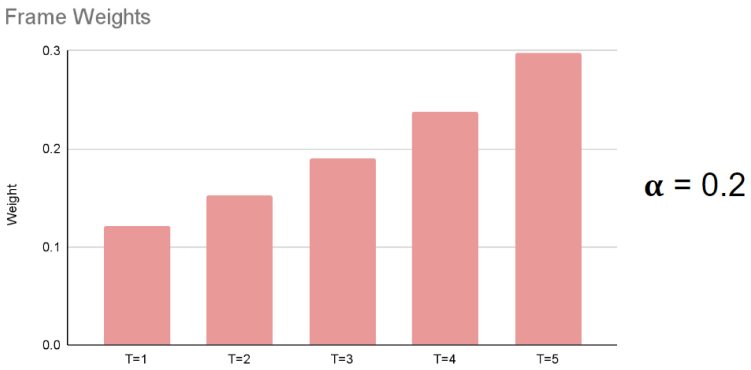

Figure 4.8. Weights for each prediction for every frame in 4.7 when using exponential moving average.

observing trends in the stock markets, it works quite well for our use case. After gathering the cluster assignments from the activation layer of the DMoN, we apply an exponential moving average to the last seen frame and frames that come before it. Therefore we avoid making predictions for the frames we have not seen yet. When using exponential average, we experiment with the exponential coefficient α by changing it from 0.1 to 0.5. The α coefficient determines how big of a frame window we are considering behind the current frame to make a decision. When the constant is bigger, the weights of the frames long before the current frame will be more effective. As this constant gets smaller, the frame window will be shorter, and the effect of the older frames will be diminished heavily. We also apply a simple moving average with a fixed window. However, using the simple average is more ineffective than the exponential

average. The optimal frustum length (L) is chosen as two meters, and the optimal frustum angle θ is found to be 45 degrees. The α parameter of the exponential moving average filter is selected as 0.3 for the Cocktail Party dataset and 0.1 for the Poster Session dataset. This result is expected because the groups in the Poster Session dataset are more static and change less frequently than the groups in the Cocktail Party dataset.

4.6. Training and Evaluation Details

In order to test our method on each frame from the dataset, we use 5-fold cross-validation. Our 5-fold cross-validation is consistent with the 5-fold used in the prior Deep Learning-based methods such as DANTE [9]. Due to the nature of time-series data, each fold is treated as a continuous data segment to preserve spatial information in the videos. We choose data for validation from the training set in such a way that it isolates the data that is used for training from the data that is used for testing as much as feasible. After hyper-parameters are determined based on the validation set, the test data of a specific fold is solely utilized to compute final results.

To train DMoN, we try to find the optimal hyper-parameters for each dataset using an exhaustive grid search algorithm. To quickly find them, we select random n frames from the dataset to train and evaluate other frames. We select n to be as small as possible while giving logical results. We find that GCN architecture (depth and kernel size) and learning rate does not differ among datasets. However, parameters such as frustum length, frustum angle, edge cutoff threshold, collapse regularization, number of maximum clusters, and dropout rate may have different values between datasets.

4.7. Qualitative and Quantitative Results

We illustrate the predictions of our method for both datasets in 4.9. We show two examples for Cocktail Party and Poster Session datasets, where the first two rows

belong to Cocktail Party, and the last two rows are from the Poster Session dataset. We illustrate our accurate and inaccurate group predictions for both datasets where the first row of each dataset shows the accurate detections and the second row shows the inaccurate results. In the first row, we see that our predictions only misplaced the person number 8. We also show the results of GCFF, our best competitor in this dataset, where they failed to identify most of the groups correctly. In the second row, our model failed to detect all groups except the blue group with three members. In this frame, it is observed that our approach inaccurately prefers *vis-a-vis* groups (See 2.1 for the *vis-a-vis* group formation) over the larger group formations whereas the GCFF method performs better. In the third row -a scene taken from the Poster Session dataset- our approach perfectly matches every group. For this dataset, our best competitor is DANTE. Looking at their results, DANTE could not detect the orange group in the ground truth and falsely assigns 3 and 10 to the same group. For the last row which exhibits complex small group formations, our system erroneously combines dark blue, light blue, and red groups into a single group (orange). In this case, DANTE misplaces only person 12 to the wrong group.

We show more results in 4.10 and 4.11 for our accurate predictions. In 4.10 our model accurately predicts three groups (shown in red, light green, and dark green in ground truth), and it misidentifies two small groups (orange and blue) as one. When observed from the camera view, it is not clear how the group should be annotated, but since the individuals are so close to each other, they remain each others' r-space (see 2.1), which indicates that we should consider it as a single group. For the 4.11 our method perfectly predicts the groups as given in the ground truth.

We present the results for the CMU Panoptic Pizza Party dataset in 4.12, 4.13 and, 4.14. In 4.12 our predictions match perfectly with the ground truth. For 4.13 and 4.14 we observe that our model is incorrectly assigning person number 6 to a group who is taking a pizza from the table. These inaccuracies indicate that if we can integrate key inanimate objects into our pipeline, we can prevent cases similar to this. We discuss this issue in the Section 5.

We provide the quantitative group detection results (F_1 scores) at two different group tolerance thresholds ($T = 2/3$ and $T = 1$) in 4.1 and 4.2 for the Poster Session and Cocktail Party datasets, respectively. If we look at the F_1 scores at $T = 1$ for the Poster Session dataset in 4.1, we observe that our approach performs better than the other approaches. When $T = 2/3$, DANTE seems slightly better with $F_1 = 84.97\%$. With the incorporation of temporal information, the proposed system outperforms all other approaches, as can be seen at last row in 4.1

In the 4.2, we present the results for Cocktail Party dataset which is more difficult than the Poster Session dataset due to highly dynamic group formations. The best competitor to our method in this dataset is the GCFF method. At $T = 1$, we improve the F_1 score from 57.23% to 61.90%. When temporal fusion is employed, F_1 score further improves to 64.51%. Similar improvements can also be observed for $T = 2/3$ in 4.2.

We demonstrate the effect of using edge pruning and node initialization in 4.6, 4.5, 4.8, and 4.7. In 4.6 and 4.5 we can observe the effect of edge pruning in Cocktail Party and Poster Session dataset respectively. Although it is not clear in the Poster Session dataset, the effect of edge pruning shows a slight improvement at the $T = 1$ threshold F1 Score in the Cocktail Party dataset. For node initialization, the result is the opposite of the edge pruning experiments where we see a significant improvement in the Poster Session dataset and there is no correlation in the Cocktail Party dataset. If the spatial positions of individuals are used for node features, we observe a 10.75% increase in the $T = 1$ F1 Score.

We also provide our experimental results for the both SALSA datasets combined in 4.9. The authors of DANTE share their results for $T = 1$ threshold as 64.4%. However, in our experiments we cannot achieve the same results and find the F1 score as 56.04%. In this dataset, our method improves the submitted results of DANTE from 64.4% to 65.99%.

For the CMU Panoptic Pizza Party dataset, we include the results in 4.10. In this dataset we cannot improve the full match accuracy of the GCFF and fall below by 3.18%. However, in the $T = 2/3$ threshold, we achieve almost perfect F1 score of 99.82%, well above of GCFF by 8.65%. Since our method is based on a deep-learning framework, it requires more data to generalize the situation. In the CMU dataset, traditional methods may perform better than data-driven models as the number of data points are limited.

Lastly, we analyze the group detection times for all of the approaches. The proposed method detects groups in a single frame in 19 milliseconds on average. The group detection times for the other approaches are as follows: 114 ms for the DANTE, 45 ms for the GTCG, and 8 ms for the GCFF approach.

Table 4.1. Averaged F1-Score results of conversational group detection methods on SALSA Poster Session dataset. The results are given in percentages.

	F1 Score ($T = \frac{2}{3}$)	**F1 Score** ($T = 1$)
GCFF	81.35	60.64
GTCG	74.87	58.42
DANTE	84.97	64.90
DMoN	84.20	67.48
DMoN - MA	85.05	67.56
DMoN - EMA	85.17	68.03

Table 4.2. Averaged F1-Score results of conversational group detection methods on SALSA Cocktail Party dataset. The results are given in percentages.

	F1 Score ($T = \frac{2}{3}$)	**F1 Score** ($T = 1$)
GCFF	77.83	57.23
GTCG	58.42	38.82
DANTE	74.84	56.04
DMoN	79.36	61.90
DMoN - MA	81.57	63.72
DMoN - EMA	81.71	64.51

Table 4.3. F1-scores of all methods on SALSA Poster Session dataset at each fold. MA and EMA represent post-processing methods, Simple Moving Average and Exponential Moving Average respectively. All results are given in percentages.

	Method	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	DMoN	84.20	93.23	92.48	70.02	87.14	78.12
	DMoN - MA	85.05	93.71	93.11	70.16	90.22	78.03
	DMoN - EMA	85.17	93.57	93.63	70.00	89.34	79.31
	GCFF	81.35	79.38	87.06	81.59	78.59	80.11
	GTCG	74.87	78.15	80.03	69.24	73.21	73.74
	DANTE	84.97	66.39	97.85	81.02	90.45	89.16
$T = 1$	DMoN	67.48	90.80	80.40	58.66	55.07	52.46
	DMoN - MA	67.56	91.14	81.67	56.18	54.63	54.18
	DMoN - EMA	68.03	91.22	80.61	58.73	56.32	53.28
	GCFF	60.64	67.55	68.99	50.89	51.80	63.96
	GTCG	58.42	43.16	51.87	50.10	70.53	76.42
	DANTE	64.90	40.66	80.82	57.67	75.87	69.46

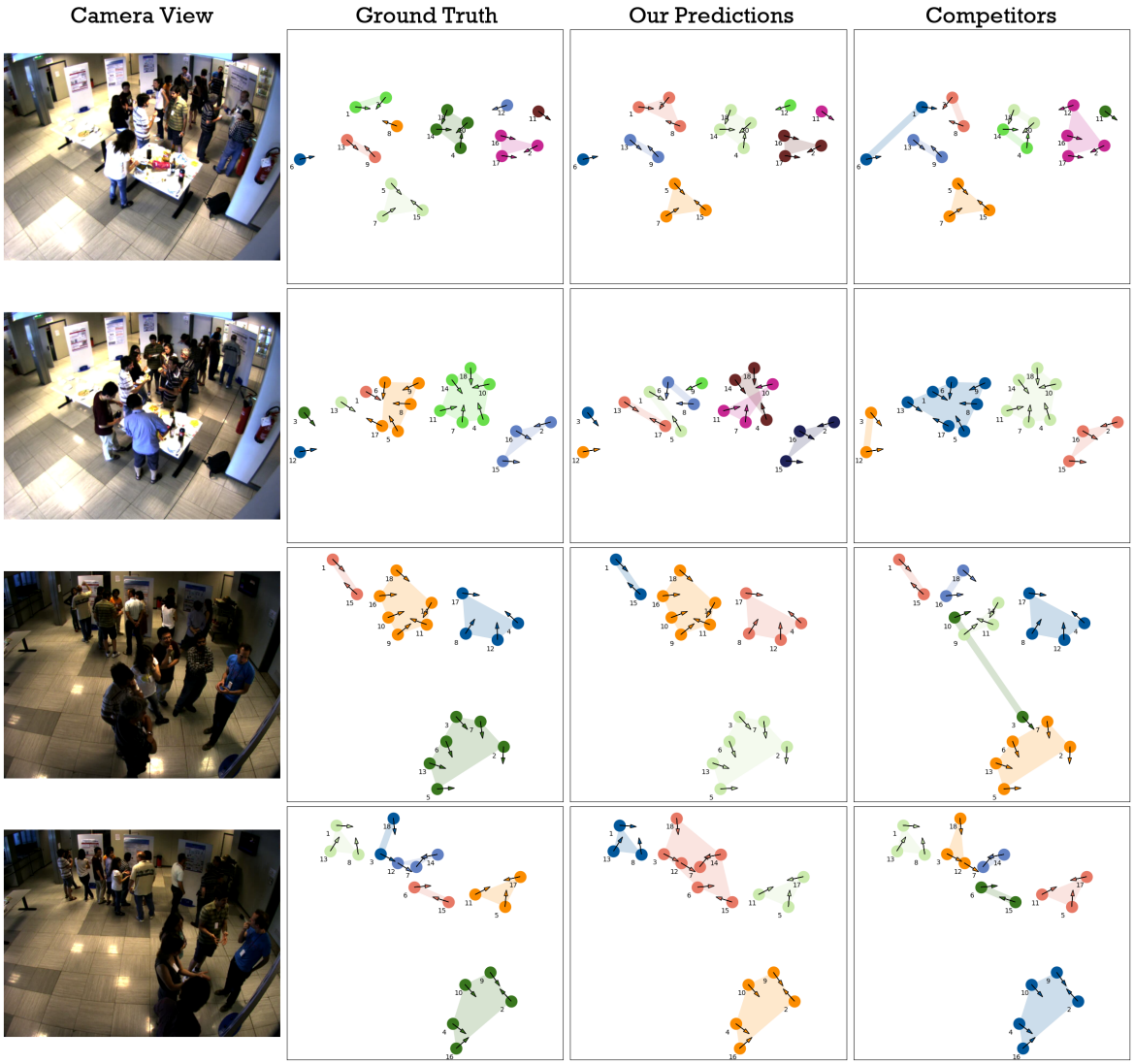

Figure 4.9. Visual group detection results. The first two rows of results are from the SALSA Cocktail Party dataset, and the next two rows are from the SALSA Poster Session. The first column shows the video frame, the second column shows the ground truth, the third column shows the results from our approach, and the last column shows the results from the best competitors, e.g., the GCFF approach for the Cocktail Party and DANTE for the Poster Session dataset.

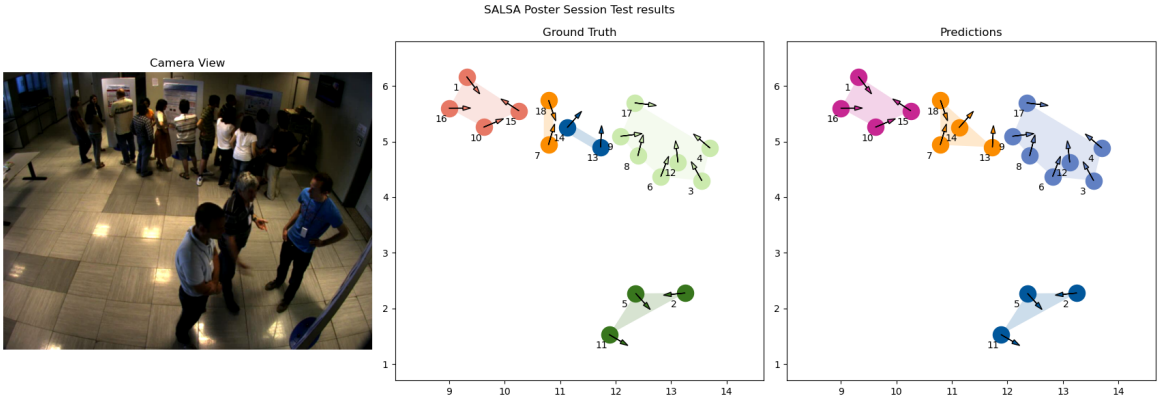

Figure 4.10. Sampled test result from SALSA Poster Session dataset. Left: original image from the SALSA Poster Session dataset. Middle: ground truth conversational group. Right: Our results with DMoN with Exponential Moving Average post-processing method. Here the only misidentified group is the group with persons number 13 and 14.

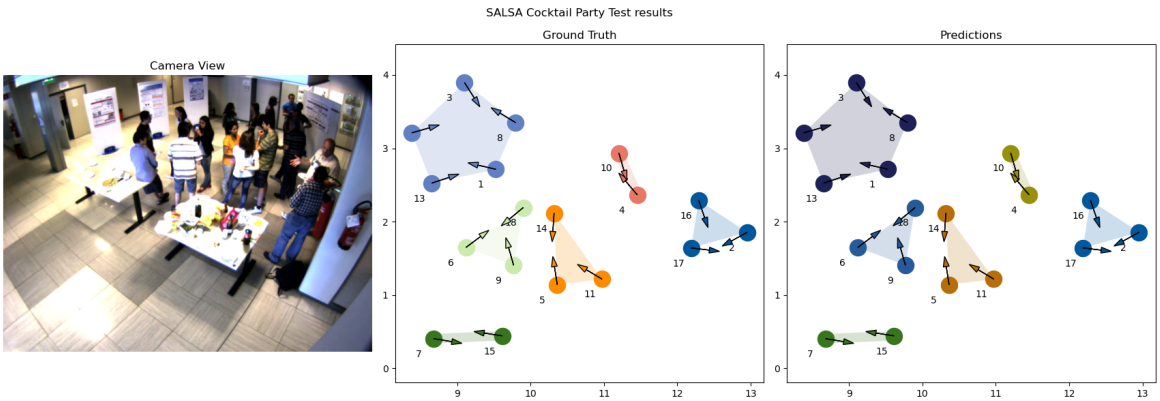

Figure 4.11. Sampled test result from SALSA Poster Session dataset. Left: original image from the SALSA Poster Session dataset. Middle: ground truth conversational group. Right: Our results with DMoN with Exponential Moving Average post-processing method. A perfect example aligning with ground truth exactly.

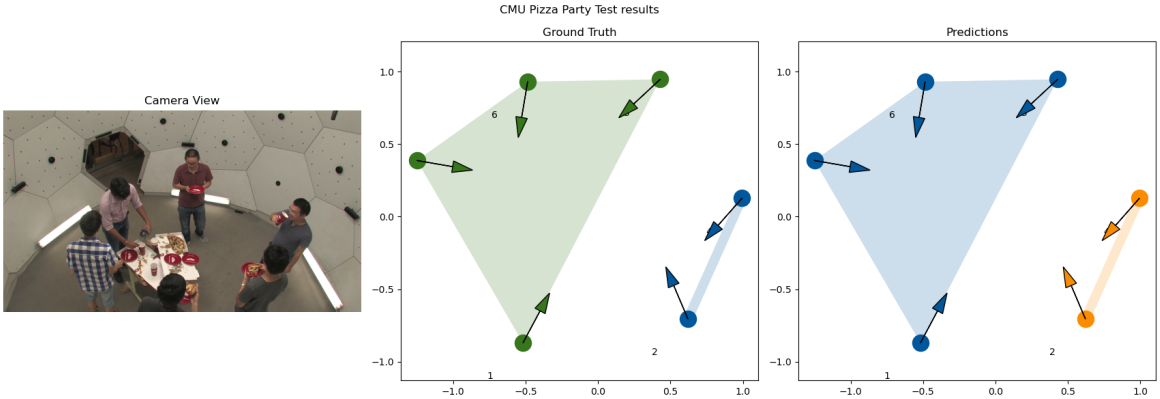

Figure 4.12. Sampled test result from CMU Pizza Party dataset. Left: original image from the SALSA Poster Session dataset. Middle: ground truth conversational group. Right: Results obtained from DMoN with Exponential Moving Average post-processing method.

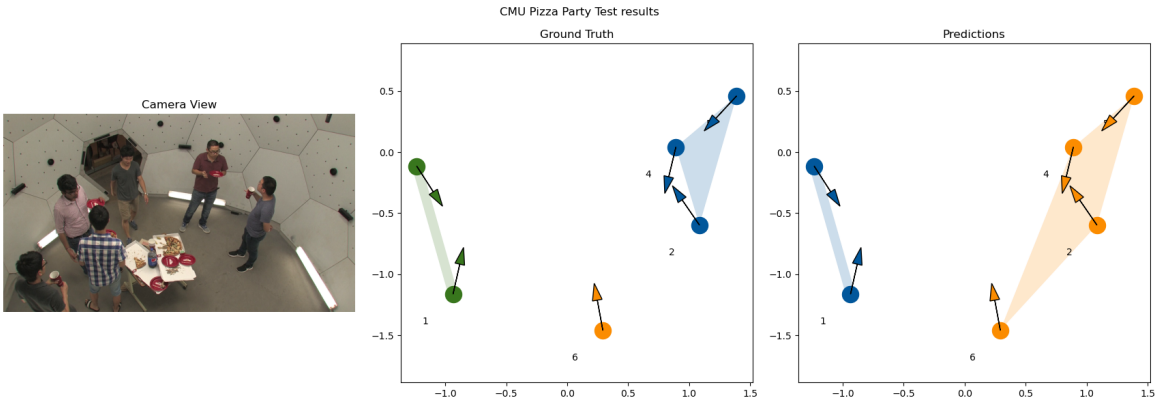

Figure 4.13. Fail case in the CMU Pizza Party Dataset. Although person number six can be seen entering the room and heading for the pizza table from the camera view, DMoN wrongly assigns him to the other groups.

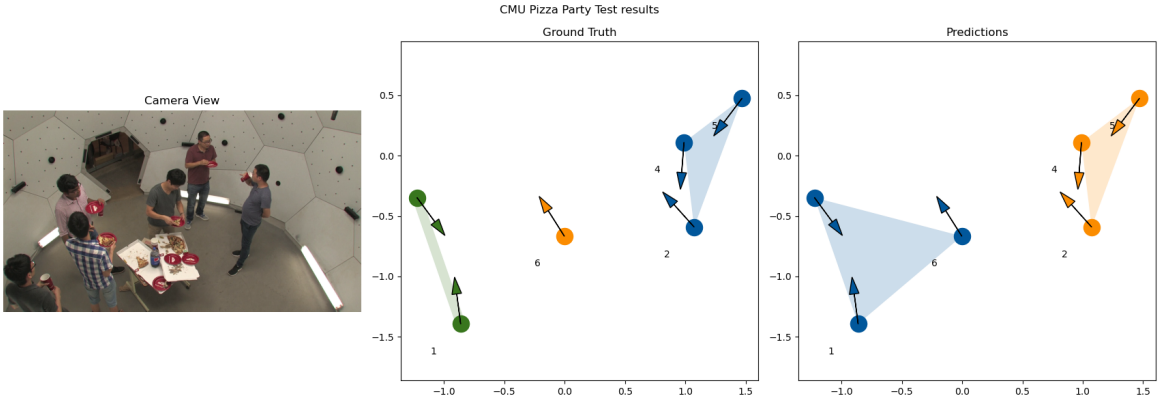

Figure 4.14. Fail case in the CMU Pizza Party Dataset. Although person number six can be seen entering the room and heading for the pizza table from the camera view, DMoN wrongly assigns him to the other groups.

Table 4.4. F1-scores of all methods on SALSA Cocktail Party dataset at each fold. MA and EMA represent post-processing methods, Simple Moving Average and Exponential Moving Average respectively. All results are given in percentages.

	Method	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	GCFF	77.83	66.64	74.64	70.98	86.82	90.09
	GTCG	58.42	43.16	51.87	50.10	70.53	76.42
	DANTE	74.84	64.47	78.56	65.69	90.03	75.48
	DMoN	79.36	75.88	72.02	92.22	93.50	63.19
	DMoN - MA	81.57	75.64	77.50	92.30	95.54	68.05
	DMoN - EMA	81.71	75.80	77.50	93.04	95.07	67.14
$T = 1$	GCFF	57.23	39.78	50.73	50.89	67.84	76.89
	GTCG	38.82	25.25	34.12	29.69	51.66	53.37
	DANTE	56.04	42.79	55.87	48.93	74.55	58.07
	DMoN	61.90	53.20	44.60	78.42	86.93	46.37
	DMoN - MA	63.72	57.76	42.13	78.33	91.60	48.77
	DMoN - EMA	64.51	56.92	44.46	79.45	91.75	49.97

Table 4.5. Effect of edge pruning on SALSA Poster Session dataset. There's no conclusive evidence that pruning edges with low weights helps in this case. All results are given in percentages.

	Model	**Avg F1**	**Threshold**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	DMoN	83.96	no prune	97.96	92.33	66.62	84.89	78.00
		83.49	0.1	97.76	93.24	66.55	81.88	78.00
		83.11	0.2	97.76	92.57	65.73	81.32	78.17
$T = 1$	DMoN	67.08	no prune	90.80	77.45	58.54	56.13	52.46
		67.48	0.1	90.80	80.40	58.66	55.07	52.46
		66.72	0.2	90.77	79.00	58.02	53.17	52.63

Table 4.6. Effect of edge pruning on SALSA Cocktail Party dataset. A slight improvement can be seen when $T = 1$ at the edge pruning threshold 0.2. All results are given in percentages.

	Model	**Avg F1**	**Details**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	DMoN	75.77	no prune	59.04	75.67	71.17	85.94	87.03
		75.16	0.1	58.59	74.50	71.40	85.20	86.13
		73.96	0.2	61.07	72.33	59.90	84.42	92.10
$T = 1$	DMoN	48.36	no prune	29.16	45.41	48.02	58.46	60.75
		47.92	0.1	29.40	37.94	51.52	60.84	59.92
		51.60	0.2	31.55	42.98	44.24	67.98	71.22

Table 4.7. Effect of using people's 2D spatial position as node features on SALSA Poster Session dataset. Up to 10% increase in F1-Score can be seen at $T = 1$ when position information is used. All results are given in percentages.

	Model	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	DMoN - No Pos.	76.78	66.26	93.86	65.21	91.43	67.13
	DMoN - Pos.	83.96	97.96	92.33	66.62	84.89	78.00
$T = 1$	DMoN - No Pos.	56.33	51.35	74.71	50.65	63.48	41.47
	DMoN - Pos.	67.08	90.80	77.45	58.54	56.13	52.46

Table 4.8. Effect of using people's 2D spatial position as node features on SALSA Cocktail Party dataset. Since Cocktail Party is more dynamic in nature, using position values as features does not yield any improvements. All results are given in percentages.

	Model	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	DMoN - No Pos.	75.77	59.04	75.67	71.17	85.94	87.03
	DMoN - Pos.	72.57	53.80	69.22	67.88	80.38	91.59
$T = 1$	DMoN - No Pos.	48.36	29.16	45.41	48.02	58.46	60.75
	DMoN - Pos.	47.23	30.01	39.66	41.58	56.55	68.34

Table 4.9. F1-scores of all methods on both SALSA Cocktail Party and SALSA Poster Session combined at each fold. MA and EMA represent post-processing methods, Simple Moving Average and Exponential Moving Average respectively. All results are given in percentages. Our re-evaluation of DANTE on the combined dataset for both $T = 1$ and $T = 2/3$, and their results from the paper for $T = 1$ are given.

	Method	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	GCFF	79.59	73.01	80.85	76.29	82.71	85.10
	GTCG	66.64	60.65	65.95	59.67	71.87	75.08
	DANTE	74.84	64.47	78.56	65.69	90.03	75.48
	DMoN	80.58	71.89	98.48	81.38	86.75	64.39
	DMoN - MA	81.10	72.80	98.69	81.01	88.82	64.15
	DMoN - EMA	81.14	73.45	98.42	80.69	89.02	64.13
$T = 1$	GCFF	58.93	53.66	59.86	50.89	59.82	70.43
	GTCG	48.62	34.20	43.00	39.90	61.09	64.89
	DANTE	56.04	42.79	55.87	48.93	74.55	58.07
	DANTE (PAPER)	64.40	76.00	57.00	70.00	50.00	69.00
	DMoN	64.32	56.02	88.40	57.77	71.61	47.79
	DMoN - MA	65.54	56.79	90.22	59.20	73.34	48.14
	DMoN - EMA	65.99	56.71	90.16	59.84	74.11	49.11

Table 4.10. F1-scores of all methods on both CMU Pizza Party dataset at each fold. MA and EMA represent post-processing methods, Simple Moving Average and Exponential Moving Average respectively. All results are given in percentages. Although DMoN achieves the best group detection accuracy with tolerance threshold at $T = 2/3$, it falls below GCFF at $T = 1$. This might be due to nature of Deep Learning algorithms requiring much more data to produce meaningful results.

	Method	**Avg F1**	**Fold-1**	**Fold-2**	**Fold-3**	**Fold-4**	**Fold-5**
$T = \frac{2}{3}$	GCFF	91.17	88.64	92.61	79.15	100.00	95.45
	GTCG	52.63	10.70	58.05	66.29	62.96	65.15
	DMoN	97.36	97.73	100.00	89.09	100.00	100.00
	DMoN - MA	99.82	100.00	100.00	99.09	100.00	100.00
	DMoN - EMA	99.82	100.00	100.00	99.09	100.00	100.00
$T = 1$	GCFF	75.91	60.23	68.56	79.15	87.50	84.09
	GTCG	23.95	0.91	27.17	47.29	26.18	18.18
	DMoN	67.00	56.82	57.95	50.91	79.55	89.77
	DMoN - MA	70.18	59.09	57.95	60.00	84.09	89.77
	DMoN - EMA	72.73	70.45	57.95	61.36	84.09	89.77

5. CONCLUSION & FUTURE WORK

In this thesis, we present a novel approach for detecting the free-standing conversational groups by clustering the constructed social graphs from still images. In social group detection, our method is the first to leverage the power of Graph Convolutional Networks. We transfer the domain of a technique used in community detection to FCG detection, where we utilize multiple sparse graphs instead of a vast single graph. We introduce the way of constructing a graph from given spatial information of people in a scene and apply it to work with GCNs. We analyze the importance of spatial information to the affinities of people in engagements by evaluating the effect of different proxemic features. We present a solution that individuates groups with high accuracy, improving the state-of-the-art methods on the SALSA dataset. As an extension of this study, consideration and modeling of key inanimate objects such as a food table in a cocktail party or a poster in a poster session could improve the accuracy of FCG detections as they provide insight into group dynamics.

As for future work, there are several directions we can take to improve our method. Firstly, a supervised approach can be implemented. All of the datasets we use contain ground truth F-Formations, but our method uses modularity as the objective function to find the best clustering. If a supervised learning system could be constructed, the results would improve even further. Secondly, if the key elements in the scene (e.g., pizza table, poster stand) can be integrated into the pipeline, we can discriminate between forming a group and a simple attraction to the resources, therefore decreasing the frequency of falsely assigned groups. A similar topic is to detect between people in motion to prevent them from joining other groups while in motion. Another future work path could be using a more modern deep-learning approach for handling temporal information. Using exponential moving average is beneficial; however, it is a very shallow solution in its current state. Lastly, since the ground truth labels in the datasets are manual annotations, they are prone to human mistakes. Therefore, smoothing the labels in the time domain could be applied to reduce noisy ground truth labels. If a

hierarchical grouping approach could be accepted, the evaluation metrics would give more granular information.

REFERENCES

1. Zachary, W. W., "An Information Flow Model for Conflict and Fission in Small Groups", *Journal of Anthropological Research*, Vol. 33, No. 4, pp. 452–473, 1977.

2. Setti, F., C. Russell, C. Bassetti and M. Cristani, "F-Formation Detection: Individuating Free-Standing Conversational Groups in Images", *PloS one*, Vol. 10, No. 5, p. e0123783, 2015.

3. Hung, H. and B. Kröse, "Detecting F-Formations as Dominant Sets", *Proceedings of the 13th International Conference on Multimodal Interfaces*, pp. 231–238, 2011.

4. Vascon, S., E. Z. Mequanint, M. Cristani, H. Hung, M. Pelillo and V. Murino, "A Game-Theoretic Probabilistic Approach for Detecting Conversational Groups", *Asian Conference on Computer Vision*, pp. 658–675, Springer, 2014.

5. Setti, F., O. Lanz, R. Ferrario, V. Murino and M. Cristani, "Multi-Scale F-Formation Discovery for Group Detection", *2013 IEEE International Conference on Image Processing*, pp. 3547–3551, IEEE, 2013.

6. Cristani, M., L. Bazzani, G. Paggetti, A. Fossati, D. Tosato, A. Del Bue, G. Menegaz and V. Murino, "Social Interaction Discovery by Statistical Analysis of F-Formations.", *BMVC*, Vol. 2, p. 4, Citeseer, 2011.

7. Cristani, M., R. Raghavendra, A. Del Bue and V. Murino, "Human Behavior Analysis in Video Surveillance: A Social Signal Processing Perspective", *Neurocomputing*, Vol. 100, pp. 86–97, 2013.

8. Swofford, M., J. Peruzzi and M. Vázquez, "Conversational Group Detection With Deep Convolutional Networks", *arXiv preprint arXiv:1810.04039*, 2018.

9. Swofford, M., J. Peruzzi, N. Tsoi, S. Thompson, R. Martín-Martín, S. Savarese

and M. Vázquez, "Improving Social Awareness Through DANTE: Deep Affinity Network for Clustering Conversational Interactants", *Proceedings of the ACM on Human-Computer Interaction*, Vol. 4, No. CSCW1, pp. 1–23, 2020.

10. Alameda-Pineda, X., J. Staiano, R. Subramanian, L. Batrinca, E. Ricci, B. Lepri, O. Lanz and N. Sebe, "Salsa: A Novel Dataset for Multimodal Group Behavior Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 38, No. 8, pp. 1707–1720, 2015.

11. Ricci, E., J. Varadarajan, R. Subramanian, S. Rota Bulo, N. Ahuja and O. Lanz, "Uncovering Interactions and Interactors: Joint Estimation of Head, Body Orientation and F-Formations From Surveillance Videos", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4660–4668, 2015.

12. Forsyth, D. R., *Group Dynamics*, Cengage Learning, 2014.

13. Bazzani, L., M. Cristani, D. Tosato, M. Farenzena, G. Paggetti, G. Menegaz and V. Murino, "Social Interactions by Visual Focus of Attention in a Three-Dimensional Environment", *Expert Systems*, Vol. 30, No. 2, pp. 115–127, 2013.

14. Vascon, S., E. Z. Mequanint, M. Cristani, H. Hung, M. Pelillo and V. Murino, "A Game-Theoretic Probabilistic Approach for Detecting Conversational Groups", *Asian Conference on Computer Vision*, pp. 658–675, Springer, 2014.

15. Alameda-Pineda, X., Y. Yan, E. Ricci, O. Lanz and N. Sebe, "Analyzing Free-Standing Conversational Groups: A Multimodal Approach", *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 5–14, 2015.

16. Zhang, L. and H. Hung, "Beyond F-Formations: Determining Social Involvement in Free Standing Conversing Groups From Static Images", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1086–1095, 2016.

17. Gárate, C., S. Zaidenberg, J. Badie and F. Brémond, "Group Tracking and Behav-

ior Recognition in Long Video Surveillance Sequences", *2014 International Conference on Computer Vision Theory and Applications (VISAPP)*, Vol. 2, pp. 396–402, IEEE, 2014.

18. Lau, B., K. O. Arras and W. Burgard, "Multi-Model Hypothesis Group Tracking and Group Size Estimation", *International Journal of Social Robotics*, Vol. 2, No. 1, pp. 19–30, 2010.

19. Qin, Z. and C. R. Shelton, "Improving Multi-Target Tracking via Social Grouping", *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1972–1978, IEEE, 2012.

20. Bazzani, L., M. Cristani and V. Murino, "Decentralized Particle Filter for Joint Individual-Group Tracking", *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1886–1893, IEEE, 2012.

21. Mazzon, R., F. Poiesi and A. Cavallaro, "Detection and Tracking of Groups in Crowd", *2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance*, pp. 202–207, IEEE, 2013.

22. Zen, G., B. Lepri, E. Ricci and O. Lanz, "Space Speaks: Towards Socially and Personality Aware Visual Surveillance", *Proceedings of the 1st ACM International Workshop on Multimodal Pervasive Video Analysis*, pp. 37–42, 2010.

23. Zhang, D., D. Gatica-Perez, S. Bengio, I. McCowan and G. Lathoud, "Modeling Individual and Group Actions in Neetings: a Two-Layer HMM Framework", *2004 Conference on Computer Vision and Pattern Recognition Workshop*, pp. 117–117, IEEE, 2004.

24. Goffman, E., *Encounters: Two Studies in the Sociology of Interaction*, Ravenio Books, 1961.

25. Kendon, A., *Conducting Interaction: Patterns of Behavior in Focused Encounters*,

Vol. 7, CUP Archive, 1990.

26. Vinciarelli, A., M. Pantic and H. Bourlard, "Social Signal Processing: Survey of an Emerging Domain", *Image and Vision Computing*, Vol. 27, No. 12, pp. 1743–1759, 2009.

27. Kendon, A., "The Negotiation of Context in Face-to-Face Interaction", , 1992.

28. Ciolek, T. M. and A. Kendon, "Environment and the Spatial Arrangement of Conversational Encounters", *Sociological Inquiry*, Vol. 50, No. 3-4, pp. 237–271, 1980.

29. Cao, Z., G. Hidalgo Martinez, T. Simon, S. Wei and Y. A. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

30. Zaidenberg, S., B. Boulay, C. Garate, D. P. Chau, E. Corvée and F. Bremond, "Group interaction and group tracking for video-surveillance in underground railway stations", *International Workshop on Behaviour Analysis and Video Understanding (ICVS 2011)*, p. 10, 2011.

31. Hornecker, E., "A Design Theme forTtangible Interaction: Embodied Facilitation", *ECSCW 2005*, pp. 23–43, Springer, 2005.

32. Hüttenrauch, H., K. S. Eklundh, A. Green and E. A. Topp, "Investigating Spatial Relationships in Human-Robot Interaction", *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5052–5059, IEEE, 2006.

33. Yousuf, M. A., Y. Kobayashi, Y. Kuno, A. Yamazaki and K. Yamazaki, "Development of a Mobile Museum Guide Robot That Can Configure Spatial Formation With Visitors", *International Conference on Intelligent Computing*, pp. 423–432, Springer, 2012.

34. Nieuwenhuisen, M. and S. Behnke, "Human-Like Interaction Skills for the Mobile Communication Robot Robotinho", *International Journal of Social Robotics*, Vol. 5, No. 4, pp. 549–561, 2013.

35. Tsitsulin, A., J. Palowitch, B. Perozzi and E. Müller, "Graph Clustering with Graph Neural Networks", *arXiv preprint arXiv:2006.16904*, 2020.

36. Joo, H., H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara and Y. Sheikh, "Panoptic Studio: A Massively Multiview System for Social Motion Capture", *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3334–3342, 2015.

37. Setti, F. and M. Cristani, "The GRODE Metrics: Exploring the Performance of Group Detection Approaches", *2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 36–42, 2015.

38. Goffman, E., "Behavior in Public Places: Notes on the Social Organization of Gatherings", *New York*, Vol. 3, 1963.

39. Kendon, A., "Goffman's Approach To Face-to-Face Interaction", *Erving Goffman: Exploring the Interaction Order*, 1988.

40. Schnädelbach, H., "Hybrid Spatial Topologies", *The Journal of Space Syntax*, Vol. 3, No. 2, pp. 204–222, 2012.

41. Ballendat, T., N. Marquardt and S. Greenberg, "Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment", *ACM International Conference on Interactive Tabletops and Surfaces*, pp. 121–130, 2010.

42. Jungmann, M., R. Cox and G. Fitzpatrick, "Spatial Play Effects in a Tangible Game With an F-Formation of Multiple Players", *Proceedings of the Fifteenth Australasian User Interface Conference-Volume 150*, pp. 57–66, 2014.

43. Suzuki, H. and H. Kato, "Interaction-Level Support for Collaborative Learning: AlgoBlock—An Open Programming Language", *The First International Conference on Computer Support for Collaborative Learning*, CSCL '95, p. 349–355, L. Erlbaum Associates Inc., USA, 1995.

44. Morrison, C., M. Jones, A. Blackwell and A. Vuylsteke, "Electronic Patient Record Use During Ward Rounds: A Qualitative Study of Interaction Between Medical Staff", *Critical Care*, Vol. 12, No. 6, pp. 1–8, 2008.

45. Marshall, P., Y. Rogers and N. Pantidi, "Using F-Formations To Analyse Spatial Patterns of Interaction in Physical Environments", *Proceedings of the ACM 2011 Conference on Computer Supported Cooperative Work*, pp. 445–454, 2011.

46. Faber, F., M. Bennewitz, C. Eppner, A. Gorog, C. Gonsior, D. Joho, M. Schreiber and S. Behnke, "The Humanoid Museum Tour Guide Robotinho", *RO-MAN 2009-The 18th IEEE International Symposium on Robot and Human Interactive Communication*, pp. 891–896, IEEE, 2009.

47. Pavan, M. and M. Pelillo, "Dominant Sets and Pairwise Clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 1, pp. 167–172, 2006.

48. Perozzi, B., R. Al-Rfou and S. Skiena, "Deepwalk: Online Learning of Social Representations", *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 701–710, 2014.

49. Fout, A. M., *Protein Interface Prediction Using Graph Convolutional Networks*, Ph.D. Thesis, Colorado State University, 2017.

50. Schlichtkrull, M., T. N. Kipf, P. Bloem, R. van den Berg, I. Titov and M. Welling, "Modeling Relational Data with Graph Convolutional Networks", A. Gangemi, R. Navigli, M.-E. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai and M. Alam

(Editors), *The Semantic Web*, pp. 593–607, Cham, 2018.

51. Sen, P., G. Namata, M. Bilgic, L. Getoor, B. Galligher and T. Eliassi-Rad, "Collective Classification in Network Data", *AI Magazine*, Vol. 29, No. 3, p. 93, Sep. 2008.

52. Giles, C. L., K. D. Bollacker and S. Lawrence, "CiteSeer: An Automatic Citation Indexing System", *Proceedings of the Third ACM Conference on Digital Libraries*, pp. 89–98, 1998.

53. Leskovec, J. and A. Krevl, *SNAP Datasets: Stanford Large Network Dataset Collection*, 2014, `https://snap.stanford.edu/data/`, accessed in September 2021.

54. Sutherland, I. E. and G. W. Hodgman, "Reentrant Polygon Clipping", *Communications of the ACM*, Vol. 17, No. 1, pp. 32–42, 1974.

55. Klambauer, G., T. Unterthiner, A. Mayr and S. Hochreiter, "Self-normalizing Neural Networks", *arXiv preprint arXiv:1706.02515*, 2017.

56. Newman, M. E., "Modularity and Community Structure in Networks", *Proceedings of the National Academy of Sciences*, Vol. 103, No. 23, pp. 8577–8582, 2006.

57. Brandes, U., D. Delling, M. Gaertler, R. Görke, M. Hoefer, Z. Nikoloski and D. Wagner, "Maximizing Modularity is Hard", *arXiv preprint physics/0608255*, 2006.

58. Newman, M. E., "Finding Community Structure in Networks Using the Eigenvectors of Matrices", *Physical Review E*, Vol. 74, No. 3, p. 036104, 2006.

59. Srivastava, N., G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, "Dropout: A Simple Way To Prevent Neural Networks From Overfitting", *The Journal of Machine Learning Research*, Vol. 15, No. 1, pp. 1929–1958, 2014.

60. Lan, T., Y. Wang, W. Yang, S. N. Robinovitch and G. Mori, "Discriminative Latent Models for Recognizing Contextual Group Activities", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 34, No. 8, pp. 1549–1562, 2011.

61. Lin, T.-Y., M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, "Microsoft COCO: Common Objects in Context", *European Conference on Computer Vision*, pp. 740–755, Springer, 2014.

62. Simon, T., H. Joo, I. Matthews and Y. Sheikh, "Hand Keypoint Detection in Single Images Using Multiview Bootstrapping", *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1145–1153, 2017.

63. Subramanian, R., Y. Yan, J. Staiano, O. Lanz and N. Sebe, "On the Relationship Between Head Pose, Social Attention and Personality Prediction for Unstructured and Dynamic Group Interactions", *Proceedings of the 15th ACM on International Conference on Multimodal Interaction*, pp. 3–10, 2013.

64. Lanz, O., "Approximate Bayesian Multibody Tracking", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28, No. 9, pp. 1436–1449, 2006.