Social Scene Understanding from Social Cameras

Hyun Soo Park

April 28, 2014

Mechanical Engineering Department
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Yaser Sheikh, Chair (Carnegie Mellon University)
Jessica K. Hodgins (Carnegie Mellon University)
Levent Burak Kara (Carnegie Mellon University)
Kenji Shimada (Carnegie Mellon University)
Christoph Bregler (New York University)

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Abstract

In social scenes, humans interact with each other by sending visible social signals, such as facial expressions, body gestures, and gaze movements. *Social cognition*, the ability to perceive, model, and predict such social signals, enables people to understand social interactions and to plan their behavior in accordance with the understanding. Computational social cognition is a necessary function allowing artificial agents to enter the social spaces because it enables a socially acceptable behavior. However, two key challenges preclude developing computational social cognition: (1) the core attributes of social cognition such as attention, emotion, and intent are latent quantities that cannot be directly measured by existing sensors; (2) social behaviors are interdependent to each other, i.e., a unified representation is required to understand social behavior as wholes. In this thesis, we address these challenges by **establishing a computational foundation towards social scene understanding from social cameras**.

A social camera is a camera held or worn by a member of a social group that inherits his/her gaze behavior. This social camera is an ideal sensor to capture social signals for three reasons: (1) social cameras naturally secure the best view because the wearers or holders intelligently localize the best view point to attend to what they find interesting; (2) social cameras produce more views of events of greater interest; (3) social cameras efficiently capture socially important events by following social behaviors when the scenes are dynamic. We leverage these advantages of social cameras to understand social scenes.

We present a framework to develop social cognition by perceiving social signals, modeling the relationship between them, and predicting social behaviors.

Social Signal Reconstruction: Reconstructing social signals in a unified 3D coordinate system provides a computational basis to analyze social scenes, e.g., to build a model, reason about relationships, and predict social behaviors. We leverage social cameras to reconstruct three types of social signals: gaze movement, body motion, and general scene motion. (1) Gaze is a strong indicator of attentive behaviors. We model the gaze using the primary gaze direction that is emitted from the center of the eyes and aligned with the head orientation. This gaze model is reconstructed in 3D by leveraging ego- and exo-motion of social cameras. (2) Human body motion such as gestures often conveys intent of social interactions. We model skeletal motion using a set of articulated joint trajectories where the distance between the trajectories of adjacent joints remains constant. This articulation constraint in conjunction with a temporal constraint is applied to reconstruct human body motion without an activity specific prior. (3) We further relax the articulation constraint to model general scene motion occurring in social interactions. We represent a 3D trajectory using a linear combination of predefined trajectory basis vectors. We solve for the parameters of each trajectory by formulating it as a linear least squares system that allows us to reconstruct topology-independent motion and handle missing data.

Social Behavior Understanding: Social behaviors are interactive by definition and therefore, an individual behavioral analysis in isolation cannot fully account for the fundamental relationship between behaviors. For instance, a social signal transmitted by one person can trigger responses in other and the responses can, in turn, affect the behavior of the person. A relational analysis between the signals is needed to characterize the social interactions. We exploit the reconstructed social signals in a unified coordinate system to understand the relationship between them. In particular, our analysis focuses on joint attention, the primary social attribute that is strongly cor-

related with attentive behaviors. We present a method to reconstruct 3D joint attention modeled by social charges—latent quantities that form at locations where primary gaze directions of members in a social group intersect. Inspired by the study of electric fields, we model the relationship between gaze behaviors using a gradient field induced by the social charges. This gradient field allows us to predict gaze behaviors given social charges at any location in the scene.

Our overarching goal is to develop computational social cognition that will enable artificial agents to accomplish their tasks in a socially acceptable way. This thesis takes a first step towards the goal by leveraging social cameras. We present a 3D representation of social signals and based on the reconstructed signals, we build a relational model of social behaviors, which allows us to predict the behaviors. We apply our frameworks in real-world social scenes including sporting events, meetings, and parties.

Acknowledgments

I owe a great intellectual debt to Prof. Yaser Sheikh who has given me tremendous guidance. He has shown the *ideal* role model as a mentor; he has encouraged me with endless patience that allowed me to focus on my research and to dream of pursuing my academic career. I am honored to have an opportunity to work with him and proud of our work. Also I appreciate my committee members including Prof. Jessica Hodgins, Prof. Kenji Shimada, Prof. Levent Burak Kara, and Prof. Christoph Bregler for their valuable comments and advice.

I am fortunate to collaborate with many researchers from Disney Research Pittsburgh, Intel, and Microsoft Research. In particular, I would like to acknowledge Dr. Shiratori who has played a key role to build my insight on structure from motion that constitutes the basis of my thesis. Also I thank all our group members for their advice and help, including Natasha Kholgade, Tomas Simon, Varun Ramakrishna, Yair Movshovitz-Attias, Minh Vo, Zijun Wei, and Hanbyul Joo. I am indebted to the KDisTech members including Sungwook Yang, Junsung Kim, Jongho Lee, and Hyunggi Cho, who have broadened my research horizon.

Finally, I would like to express my great appreciation to my family, in particular, my beloved wife, Soo Jin Kang for her tireless support and sacrifices.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

(a) Social scene (b) Social cameras

Figure 1.1: (a) In a social scene, such as a wedding reception, people interact with others by sending visible social signals, such as gaze direction, facial expressions, or body gestures. These social signals are a strong cue understanding social scenes. In this thesis, we present a computational framework for social cognition—the ability to understand social scenes by perceiving, modeling, and predicting social signals. (b) Cameras are socially immersed in the form of smartphones, camcorders, or wearable cameras. These *social* cameras are ideal sensors to capture social scenes as they encode the gaze behaviors of the camera holders or wearers.

Suppose you are a waiter at the wedding reception shown in Figure 1.1(a). How would you interpret the scene? You see people socializing with others and participating in events such as the wedding toast, dancing, and cake cutting. You interpret their attention, intent, and emotion via their visible social signals, such as their gaze direction, facial expressions, and body gestures, and plan to complete your tasks while complying with their interactions. For instance, you recognize where the main event is happening and that you should not disturb what people want to see; you should not disturb them when they dance; when people form groups to socialize, you should avoid breaking into the group. Now consider a scenario where artificial agents, such as social service robots, were to serve the guests instead of you at the reception. How should we model human understanding of the scene to build such agents? What are the key design factors that would allow them to behave in a socially acceptable way?

A social scene is a scene where human interactions take place and the interactions are the primary focus of their behavior, such as a wedding reception, a conference poster session, or a sporting event. These social scenes are very common in our daily life and, increasingly, artificial agents are entering these social spaces. Vacuum cleaning robots, for example, navigate rooms where humans reside and surgical robots help surgeons assisted by a medical staff. As they become integrated in our lives, we expect them to be equipped with *social cognition*—the ability to understand social scenes by perceiving, modeling, and predicting social signals. Such social cognition will allow them to play roles beyond passive tools that require prompting, but as active team members that organically interact with humans and accomplish tasks, seamlessly and safely. Therefore, computational social cognition is a key design factor for artificial agents interacting with humans.

However, two key challenges preclude developing such computational social cognition: (1) the core attributes of social cognition such as attention, emotion, and intent are latent quantities that cannot be directly measured by existing sensors; (2) social behaviors are interdependent, i.e., a joint representation is required to understand social behavior as a whole. In this thesis, we address these challenges by **establishing a computational representation for social scene understanding from social cameras**—a camera held or worn by social members that encodes his/her gaze behavior (Figure 1.1(b)). We present a 3D representation of social signals evolving in the form of primary gaze direction, human body motion, and general scene motion. This 3D representation allows us to analyze the relationship between the social signals in a unified coordinate system. In particular, our analysis focuses on joint attention, which characterizes attentive behaviors of members in a social group. We exploit the 3D representation of social signals to infer joint attention in social interaction and to build a predictive model that captures the relationship between gaze behavior.

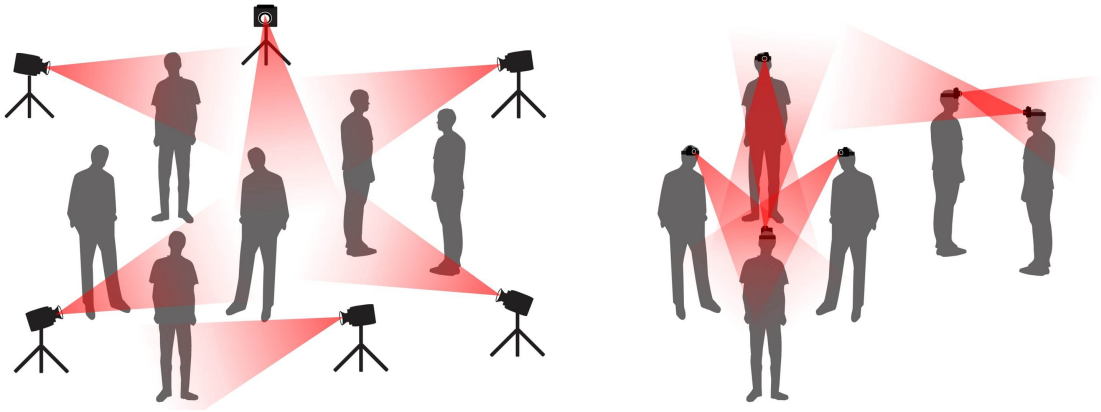
We apply our framework in real-world social scenes where various human interactions frequently occur, including sporting events, meetings, and parties. This work will introduce a new design paradigm of artificial agents interacting with humans.

1.1 Why Social Cameras?

Cameras are socially immersed in our daily life in the form of smartphones, portable camcorders, and wearable cameras, as shown in Figure 1.1(b). Most people carry at least one camera at all times and spontaneously capture scenes that they find interesting. In particular, wearable cameras, such as first person cameras, are poised to broadly enter our social spaces and continually record social interactions. Many collaborative teams (such as search and rescue teams [93], police squads, military patrols, and surgery teams [84]) are already required to wear them. These camera systems are increasingly becoming smaller and more efficient, and will soon be seamlessly integrated into daily life [17].

We define such mobile cameras as *social cameras* as they are embedded in social scenes. **These social cameras are ideal sensors to capture social scenes** because they inherit the gaze behaviors of the camera wearers or holders, i.e., the camera sees what he or she sees. The main benefits of social cameras include:

- Optimal view point: Social cameras naturally secure the optimal view because the camera



(a) Stationary cameras (b) Social cameras

Figure 1.2: (a) Stationary cameras sample a social scene from static points of view. This camera placement often cannot properly observe socially important region. (b) Social cameras sample the scene intelligently. Multiple social cameras densely observe socially important region as they follow social behaviors.

holders or wearers intelligently find the best view point to attend to what they find interesting. This positions the cameras at the optimal view point to observe social interactions while minimizing occlusion.

- Reflection of social interest: Social cameras produce more views of events of greater interest. Socially important events, such as a musical concert, sporting event, or political rally, are of interest to large group of people. Social cameras naturally capture such events densely because many members at the scene likely pay attention to the events.
- Efficient measurement: Stationary cameras measure a space at a fixed placement as shown in Figure 1.2(a). A large number of cameras are needed to cover the entire social space exhaustively, i.e., the number of cameras increases cubically as the space increases. Therefore, the operating space is limited by the number of cameras. In contrast, social cameras follow social activities, which allows them to efficiently capture the social space. They primarily focus on socially important region of the scene even if the scene is dynamic as shown in Figure 1.2(b).

In this thesis, we exploit these advantages of social cameras, as socially immersed sensors, to capture visible social signals.

1.2 Challenges

Developing computational social cognition is challenging because social attributes are latent quantities and social behaviors are interdependent. These two main challenges preclude the application of existing scene understanding frameworks such as structure from motion [133], activity recognition [168], and human affordance identification [51].

1.2.1 Latent Social Attributes

Humans understand social attributes such as attention, intent, and emotion during social interactions and social behaviors are strongly correlated with these social attributes [150]. However, enabling artificial agents to understand social attributes is a challenging task because these attributes are latent quantities that cannot be directly measured by existing sensors, in general. For instance, the social attention of guests at wedding reception cannot be explicitly measured by cameras, microphones, or laser sensors. We can infer what they pay attention to and how much they are engaged in the event based on their gaze directions but there does not exist a directly measurable quantity corresponding to social attention. This latency of social attributes makes existing scene understanding frameworks inapplicable and makes building a computational model for the social cognition difficult.

1.2.2 Interdependent Social Behavior

Social behaviors cannot be understood by analyzing individual behaviors in isolation as social members are, by definition, reliant upon each other. A signal transmitted by a social member triggers the response of other members and the responses can, in turn, affect the behavior of the original member. The relationship between social signals characterizes social interactions and therefore, a relational model of social behaviors is necessary for artificial agents to understand social scenes. However, existing activity recognition frameworks [78, 165, 168] have mainly focused on individual behavioral analysis. A trivial concatenation of these analyses is not sufficient to infer the relationship because each representation is described in different coordinate systems. The relational model must be able to represent the interdependency of social behaviors in a unified coordinate system and predict their behaviors without sensor bias that often occurs when sensor placement with respect to subjects is uneven.

1.3 Our Approach

In this thesis, we address the core challenges in social scene understanding described in Section 1.2. In Part I, we study the 3D representation of social signals, which includes gaze direction, human body motion, and general scene motion by leveraging social cameras. This representation in a unified coordinate system provides a computational basis for understanding social scenes. Given the representation, we present a method to infer joint attention that enables us to predict gaze behavior in Part II.

1.3.1 Part I: Social Signal Reconstruction

3D reconstruction provides a computational representation to analyze the social scenes, e.g., to build a model, reason about relationships, and predict social behaviors. We focus on three types of social signals occurring in social interactions: primary gaze direction, human body motion, and general scene motion.

In Chapter 3, we present two approaches to estimate primary gaze direction of members in a social group in 3D by leveraging ego- and exo-motion of social cameras. We model gaze using a

primary gaze ray emitted from the center of the eyes and aligned with the head orientation. In the ego-motion approach, we estimate gaze direction by exploiting 3D social camera pose and the fixed relationship between the camera and gaze model. The 3D camera poses are reconstructed using structure from motion and the fixed relationship is calibrated via a predefined gaze range-of-motion sequence. In the exo-motion approach, we estimate gaze direction based on faces imaged in the social cameras. We detect faces and associate them to find face correspondences via geometric verification. The landmarks of the associated faces are reconstructed in 3D and gaze direction is inferred from the 3D landmark locations. We further refine them by minimizing reprojection error and maximizing temporal consistency.

In Chapter 4, we present a method to reconstruct human body motion represented by a set of articulated trajectories [103]. An articulated trajectory is a trajectory that remains at a fixed distance with respect to a parent trajectory, i.e., the distance between trajectories remains constant. We apply spatial and temporal constraints simultaneously in the form of a fixed 3D distance to the parent trajectory and smooth 3D point motion. There exist two solutions that satisfy each instantaneous 2D projection and articulation constraint (a ray intersects a sphere at up to two locations) and we show that resolving this ambiguity by enforcing smoothness is equivalent to solving a binary quadratic programming problem.

In Chapter 5, we relax the articulation constraint to reconstruct a general 3D trajectory of a moving point from 2D image correspondences [104]. We model a 3D trajectory using a linear combination of compact trajectory basis vectors, such as the Discrete Cosine Transform basis. This reduces the reconstruction optimization to a linear least squares problem, allowing us to robustly handle missing data that often occur due to motion blur, texture deformation, and self occlusion. We present an algorithm to determine the number of trajectory basis vectors, individually for each trajectory via a cross validation scheme and refine the solution by minimizing geometric error.

1.3.2 Part II: Social Behavior Understanding

In Part II, we study social behaviors from the reconstructed social signals. In particular, our analysis focuses on joint attention, the primary social attribute that is strongly correlated with attentive behaviors of social members. We estimate 3D joint attention from the reconstructed gaze directions and present a predictive model that defines the relationship between gaze behaviors in a unified coordinate system.

In Chapter 6, we present a method to estimate joint attention modeled by 3D social charges—latent quantities that attract the attention of members in a social group [105]. A social charge is formed where gaze directions intersect and it is a socially significant location because the attention of the group is directly linked to that point. We construct a social saliency field by superimposing gaze models in 3D and find modes in that field that correspond to social charges via a mode-seeking algorithm. The number, locations, and magnitudes of the social charges are automatically estimated.

In Chapter 7, we present a method to predict primary gaze behaviors [106]. Inspired by the study of electric fields, we model the interdependency of gaze behaviors using a gradient field induced by social charges, which defines the relationship between the social charges and primary gaze direction. The gradient field is refined by minimizing directional error of the observed gaze

directions. This relational model enables us to predict gaze behaviors at any location or time in the scene.

1.4 Summary of Contributions

This research includes two core contributions: 3D reconstruction and geometric analysis, and social field modeling.

1.4.1 Social Signal Reconstruction

We present a framework to represent social signals in 3D by consolidating measurements from different social cameras. This representation is not biased by sensor placements and therefore, is coordinate independent, which enables a unified relational interpretation of different social signals. Also we study fundamental ambiguity of motion reconstruction through geometric analyses.

1. We formulate trajectory reconstruction into linear least squares that can handle significant amount of missing data and correspondence noise.
2. We characterize the cases when reconstruction becomes ambiguous via a geometric analysis of the relationship between camera motion, point motion, and basis vectors.
3. The fixed relationship between the social camera and gaze model is calibrated by minimizing the alignment error of a point of regard.
4. We introduce a new gaze model that captures primary gaze direction and eye-in-head motion.
5. We show that reconstructing articualted trajectories is equivalent to a binary quadratic program and we use a branch-and-bound method to solve the problem.
6. Automatic selection of the basis vectors and the trajectory refinement by geometric error minimization allows the algorithm to handle a wider variety of realistic motion.
7. We identify real-world scenarios where high reconstruction accuracy is guaranteed and apply our algorithm to reconstruct the time-varying 3D structure of the scenes.

1.4.2 Social Behavior Understanding

Our generative field model captures the relationship between social behaviors. We relate this field with a probabilistic inference framework that allows us to predict gaze direction of members in a social group. Also by describing the activity in the scene in terms of the motion of latent social charges, we move beyond measuring gaze behaviors, and towards understanding the narrative of the events of the scene, as interpreted by the members of the social group itself.

1. We model social behaviors using the fields induced by social charges such as a social saliency field and a gaze field.

2. A membership feature that describes members who pay attention to each social charge allows us to find temporal correspondence of social charges.
3. We introduce a social charge that is a computational model of joint attention.
4. A new mode-seeking algorithm finds social charges from the gaze direction of members in a social group. The number, locations, and magnitudes of social charges are automatically estimated.
5. Predictive validity of our gaze field model is tested on real-world scenes and our method shows superior prediction accuracy compared to state-of-the-art frameworks.
6. We formulate a gaze prediction problem as an Expectation-Maximization problem that allows us to estimate the gaze field given the observed gaze directions.
7. We detect social anomaly using our gaze prediction framework.

Chapter 2

Background

Computational social cognition is an emerging multidisciplinary research domain that aims to understand social scenes, e.g., what people are interested in, how they feel, and what they try to accomplish. As artificial agents are increasingly proliferating in our social spaces, the need for computational social cognition is becoming urgent. Social signal processing [150] is one such domain where various representations of social signals and relational models are studied and different domains of studies such as computer vision, graphics, and robotics start to benefit from their representations. In this chapter, we review prior work related to computational social cognition and establish their relationship with this thesis.

2.1 Social Signal Reconstruction

Humans transmit and respond to many different social signals when they interact with others. These social signals provide a strong cue to infer underlying social attributes [150] and in this section, we review various representations of social signals in the form of primary gaze direction, human body motion, and general scene motion.

2.1.1 Primary Gaze Direction

Gaze direction is one of the most prominent visible signals because it usually indicates what an individual is interested in. In this context, gaze estimation has been widely studied in robotics, human-computer interaction, and computer vision [10, 35, 45, 49, 59, 77, 91, 94, 95, 117, 122, 137, 153]. Gaze direction can be precisely estimated by the eye orientation. Wang and Sung [153] presented a system that estimates the direction of the iris circle from a single image using the geometry of the iris. Guestrin and Eizenman [49] and Hennessey and Lawrence [59] utilized corneal reflections and the vergence of the eye to infer the eye geometry and its motion, respectively. A head-mounted eye tracker is often used to determine the eye orientation [77, 137]. Although all these methods can estimate highly accurate gaze direction, they are usually used in a laboratory setting as the device occludes the viewer's field of view.

While the eyes are the primary source of gaze direction, Emery [35] notes that the head orientation is a strong indication of the direction of attention. For head orientation estimation,

there are two sensing approaches: outside-in and inside-out [158]. An outside-in system takes, as input, a third person image from a particular vantage point and estimates face orientation based on a face model. Murphy-Chutorian and Trivedi [94] have summarized this approach. Geometric modeling of the face has been used to orient the head by Gee and Cipolla [45] and Ballard and Stockman [10]. Rae and Ritter [117] estimated the head orientation via neural networks and Robertson and Reid [122] presented a method to estimate face orientation by learning 2D face features from different views in a low resolution video. With these approaches, a large number of cameras would need to be placed to cover a space large enough to contain all people. Also, the size of faces in these videos is often small, leading to biased head pose estimation depending on the distance from the camera. Instead of the outside-in approach, an inside-out approach estimates head orientation directly from a head-mounted camera looking out at the environment. Munn and Pelz [91] and Takemura et al. [137] estimated the head-mounted camera motion in 3D by feature tracking and visual SLAM, respectively. Pirri et al. [112] presented a gaze calibration procedure based on the eye geometry using four head-mounted cameras.

In our approach, we model gaze using primary gaze direction that is emitted from the center of the eyes and aligned with the head orientation. Given the model, we introduce two approaches to estimate primary gaze direction by leveraging ego-motion (inside-out approach) and exo-motion (outside-in approach) of social cameras. Our gaze model mostly captures the head orientation but the consensus of gaze directions of multiple social members allows us to precisely localize what they simultaneously attend to, i.e., joint attention.

2.1.2 Human Body Motion

Human body motion is a strong social signal that conveys intent in social interactions. For this reason, human body motion has been a core research subject in computer vision, robotics, and graphics. In this section, we focus on 3D reconstruction of human body pose and motion from an image or video.

Human pose estimation from a single image by applying a spatial constraint (skeletal structure) was proposed by Taylor [138] (parameterization of limb lengths by a scalar), by Barron and Kakadiaris [12] (joint motion constraint from anthropometric statistics), by Parameswaran and Chellappa [102] (camera pose estimation from head orientation and rigidity of torso), and by Agarwal and Triggs [1] (silhouette based regression). Shiratori et al. [129] introduced an inside-out approach to reconstruct human body motion using body-mounted cameras.

Human motion estimation from an image sequence of a monocular camera has been studied as an extension of human pose estimation. Two popular approaches have been explored: data-driven approaches and physics-based approaches. Data-driven approaches learn low dimensional subspace or latent variables that control the underlying human skeletal motion fully using motion capture data or annotated video data. Sidenbladh et al. [130] applied a Bayesian framework for 3D human pose tracking using a generative model of the human body and a prior distribution defined by a temporal dynamics model. Howe et al. [60] showed Baysian learning, Choo and Fleet [27] sampled high dimensional training space from hybrid Monte Carlo method, and Urtasun et al. [146] used Principle Coordinate Analysis (PCA) for learning specific motion (e.g., walking and golfing). Like Taylor's work [138], Wei and Chai [156] introduced a geometric solution of motion reconstruction using the bone symmetric constraint from biomechanical data.

Valmadre and Lucey [147] discussed the validity of Wei and Chai's work [156] and extended their algorithm using a structure from motion scheme. Ramakrishna et al. [119] addressed the activity bias problem of data-driven approaches and achieved activity-independent body pose reconstruction using sparse coding. Recently, physics-based approaches have received attention. Brubaker et al. [23] have shown reconstruction of a bipedal locomotion from a dynamical model and Vondrak et al. [151] have applied multibody dynamics simulation to infer the most plausible human motion in 3D. Wei and Chai [157] have built an interactive system that integrates a dynamical model to capture motion from a video.

Unlike previous methods, our approach represents human body motion using a set of articulated trajectories where the 3D distance between the trajectories of the adjacent joints remains constant across time. We show that resolving pose ambiguity by enforcing temporal smoothness is equivalent to solving a binary quadratic programming problem. As a result, our reconstruction is not biased by specific activities and initialization, which often occurs in data-driven and physics-based approaches, respectively.

2.1.3 General Scene Motion

Some social signals are too subtle to be represented by a human skeletal model such as skin deformation. This requires a generalization of the reconstruction framework to recover topology-independent motion. However, reconstructing a moving point in 3D from a monocular image sequence is a fundamentally ill-posed problem. To overcome its inherent ambiguity, a large body of work has developed algorithms, representations, and scene constraints. Our problem resides at the junction of two lines of research: trajectory triangulation and nonrigid structure from motion.

2.1.3.1 Trajectory Triangulation

When correspondences are provided across 2D images in static scenes, the method proposed by Longuet-Higgins [79] estimates the relative camera poses and triangulates the point in 3D using epipolar geometry. In subsequent research, summarized in Faugeras et al. [37], Ma et al. [82], and Hartley and Zisserman [56], the geometry involved in reconstructing 3D scenes has been systematically developed. While a static 3D point can be reconstructed by triangulation, when the point moves between the capture of both images, the triangulation method becomes inapplicable; the line segments formed by the baseline and the rays from each camera center to the point no longer form a closed triangle.

The principal work in 'triangulating' moving points from a series of images is by Avidan and Shashua [9], who coined the term *trajectory triangulation*. They demonstrated two cases where a moving point can be reconstructed: (1) if the point moves along a line, or (2) if the point moves along a conic section. This work inspired a number of approaches of geometrically constrained trajectory recovery. Shashua and Wolf [127] and Wexler and Shashua [159] introduced homography tensors to represent a point moving on the plane. As an integration of the algebraic curve representation, Wolf and Shashua [160] classified different manifestations of related problems, analyzing them as projections from $\mathbb{P}^N$ to $\mathbb{P}^2$ where N is a factor representing the span of the trajectory space. Kaminski and Teicher [68] extended these ideas to a 3D trajectory represented

by a family of hypersurfaces in the projective space $\mathbb{P}^5$, i.e., a homogeneous polynomial vanishes on the Plücker coordinates of all lines intersecting the trajectory. This method provides a general framework to reconstruct any arbitrary trajectory that can be represented by a polynomial. However, the algorithm is computationally prohibitive and sensitive to noise, which we will discuss in detail in Section 5.3.1.3.

In Chapter 5, we investigate the 3D reconstruction of a point trajectory where the point motion can be described as a linear combination of compact trajectory basis vectors. This representation allows far more natural motions to be linearly reconstructed [2, 4]. We demonstrate its application in reconstructing moving points from a series of image projections where no two image projections necessarily occur at the same time instant.

2.1.3.2 Nonrigid Structure from Motion

Nonrigid structure from motion is another approach to reconstructing dynamic structure in 3D from a monocular sequence. Unlike the trajectory triangulation approach, nonrigid structure from motion approaches recover camera motion as a part of their optimization. The seminal work of Bregler et al. [22] introduced linear shape models as a representation for nonrigid 3D structures, and demonstrated their applicability within the factorization-based paradigm of Tomasi and Kanade [140]. They formulated the problem as a trilinear optimization over camera motion, shape basis vectors, and shape coefficient vectors. However, finding a global solution of the trilinear optimization is difficult [3, 21, 161, 162] because of non-convexity of the objective function. Recent work has considered a number of optimization techniques to overcome the suboptimality issue [31]. Torresani et al. [142, 144] used an alternating linear least squares technique and Brand [20] provided a sophisticated initialization by allowing minimal shape deformation. Paladini et al. [101] proposed a robust metric upgrade method by iteratively projecting the solution onto a metric motion manifold.

Prior knowledge that regularizes deformation on shapes can improve stability of the optimization. Xiao et al. [161, 162] added a shape basis constraint which maximizes the orthogonality of the basis vectors leading to a closed-form solution. An algorithm to learn shape deformation was introduced by Torresani et al. [143]. Torresani and Bregler [141] and Olsen and Bartoli [97] proposed a temporal smoothness prior on the shape basis vectors and camera parameters. Yan and Pollefeys [164] used an articulation constraint that can limit shape subspace. Del Bue [33] proposed a pre-computed prior, which produces reliable reconstruction when there is degeneracy of motion and Fayad et al. [38] introduced piecewise reconstruction by dividing the surface into overlapping patches.

When the shape basis vectors are known, the complexity of the trilinear optimization reduces to a bilinear optimization. This complexity reduction results in robust camera motion and shape estimation. A nonrigid structure registration problem given a template is one such domain. Blanz and Vetter [19] modeled a face using a linear combination of shape basis vectors and registered/manipulated facial deformation given a new face image. Saragih et al. [125] proposed a method to efficiently localize face landmarks using a modified mean-shift algorithm. A surface is another target structure that has been extensively studied. Salzmann et al. [124] utilized a low dimensional shape model made of triangle meshes to represent nonrigid surface. Taylor et al. [139] proposed locally rigid structure from motion by allowing minimal triangular deformation

in 3D and Östlund et al. [99] regularized a deformable surface based on the Laplacian matrix of the structure.

In contrast to these methods, which represent the instantaneous shape of an object as a linear combination of basis shape vectors, Sidenbladh et al. [130] modeled a trajectory using a linear combination of trajectory basis vectors achieved by the eigen analysis and integrated into their Bayesian inference framework. Akhter et al. [2, 4] followed the factorization paradigm by Bregler et al. [22] to reconstruct object independent motion. This allowed them to express any object deformation without prior information while the shape representation is restricted only to the training object shapes. They used a predefined trajectory basis vectors such as the Discrete Cosine Transform (DCT). The method also reduced the complexity of the trilinear optimization to a bilinear optimization and showed accurate reconstruction when shapes cannot be well modeled by compact shape basis vectors such as articulated motion. Gotardo and Martinez [46] also used the DCT trajectory basis vectors to handle missing data. Valmadre and Lucey [148] generalized the trajectory basis concept by formulating the regularization as a temporal filter.

We note that the predefined trajectory basis vectors can be coordinate independent and therefore, social camera poses can be estimated by stationary points in the scene. This enables us to further reduce the complexity of the original trilinear optimization to a linear optimization where we can find a global solution efficiently. We propose a linear solution to reconstruct a moving point from a series of its image projections inspired by the Direct Linear Transform algorithm [56].

2.2 Social Behavior Understanding

Understanding how we socially interact with each other has been a long-standing focus of the social sciences. With the growth of computing and computer science, a significant research thrust has emerged in building computational models for understanding social behaviors driven by efforts in psychology and sociology. In this section, we review representations of social scenes and relational models that can predict social behaviors.

2.2.1 Social Scene Representation

Various models of social interaction have been proposed in psychology, sociology, and computer science. We categorize existing models for the social scene as spatial, temporal, or spatiotemporal models.

A social scene can be represented by a spatial arrangement of interactions. Two representations have been used to model a scene: microscopic and macroscopic representations. The seminal work by Hall [52] introduced the concept of proxemics, a categorization of human individual (microscopic) interactions based on spatial distances. Cristani et al. [30] applied proxemics to infer relationships of people in an image and Yang et al. [167] exploited the touch code in proxemics to estimate body poses of interacting people. A macroscopic representation was introduced by Kendon [69, 70] who modeled group spatial arrangements via F-formations. He showed that similar patterns of the group spatial arrangement (position and orientation of each member) are repetitively observed as the members in the group share their attention. Marshall et

al. [85] further studied how a physical environment can affect F-formations. Cristani et al. [29] used the F-formation model to detect human interactions in a single image. A generalized F-formation concept has been applied to estimate social attention where gaze directions intersect in the scene [14, 36, 83, 105].

Time is another axis to represent a social scene because the social scene often includes dynamic human interactions. Each interaction at each time instant is associated with other interactions at different time instances. The causality test, introduced by Granger [47], is widely used as a measure of causality of two social interactions. Zhou et al. [169] directly applied the causality measure on a pair of trajectories of human activities to detect and recognize interactions. Prabhaker et al. [116] represented a video sequence using a multivariate point-process over activities and estimated the correlation between the activities via the causality measure. Parabhaker and Rehg [115] further extended their work to characterize temporal causality emerging in turn-taking activities. Instead of the causality measure, Gupta et al. [50] showed a method to learn a storyline graph structure and Wang et al. [154] modeled a quasi-periodicity measure to extract a repetitive pattern of activities in a social game.

The full dynamics of a social scene can be modeled by spatiotemporal representations. The social force model proposed by Helbing et al. [57] has successfully emulated crowd dynamics. Each individual experiences repulsive and attractive forces by neighbors and environments. The net force applied to the individual induces motion by Newtonian physics. Johansson et al. [66] and Pellegrini et al. [107] applied the social force model to track pedestrians in a video, and Mehran et al. [86] and Raghavendra et al. [118] detected abnormal events in a crowd scene. A similar social force concept has been used for distributed robot control [44, 71, 121]. Kim et al. [72] represented a dynamic scene with a dense motion field estimated by trajectories of individual players in sports and Wang et al. [155] tracked the ball using the gaze directions of players. Oliver et al. [96] integrated spatiotemporal behaviors into a coupled hidden Markov model to recognize a few types of interactions. Ryoo and Aggarwal [123] proposed a spatiotemporal feature to match between videos of interactions, Choi et al. [26] exploited spatiotemporal relationship of interactions to characterize the scene, and Ramanathan et al. [120] identify the social role via modeling time-varying interactions with a Conditional Markov Random field.

2.2.2 Joint Attention

Gaze in a group setting has been used to identify social interaction or to measure social behavior. Stiefelhagen [135] and Smith et al. [132] estimated the point of interest in a meeting scene and a crowd scene, respectively. Bazzani et al. [14] introduced the 3D representation of the visual field of view, which enabled them to locate the convergence of views. Cristani et al. [29] adopted the F-formation concept that enumerates all possible spatial and orientation configurations of people to define the region of interest. These methods rely on data captured from the third person view point, i.e., outside-in systems and therefore, their capture space is limited and accuracy of head pose estimation degrades with distance from the camera. For an inside-out approach, Fathi et al. [36] present a method that uses a single first person camera to recognize discrete interactions within the wearer's immediate social clique. They analyzed the faces within a single person's field of view.

Unlike previous approaches, our method [105] estimates 3D joint attention modeled by social

charges using the reconstructed primary gaze directions. The social charges form at locations where gaze directions intersect and we estimate the number, locations, and magnitudes of the social charges. Temporal association of the charges is established by membership features. This method analyzes an entire environment where several social cliques may form or dissolve over time.

2.2.3 Gaze Prediction

A large body of research has studied the human ability to predict the gaze direction. Koch and Ullman [74] proposed a computational foundation for visual saliency detection. They modeled selective visual attention using a hierarchical structure of neurons that are sensitive to low level features such as color, orientation, motion, and disparity. This framework was implemented by Itti et al. [63] via their Winner-Take-All networks and showed that their detected visual saliency is matched with eye tracking results. In conjunction with the low level features, Sophie et al. [134] showed that faces are highly salient features in both static and dynamic scenes and adding face features improves the gaze prediction accuracy. However, this feature-based gaze prediction does not generalize to all scenes. Simons and Chabris [131] and Peter and Itti [109] showed that humans are blind to inattentional structure in a scene. When a task is involved, only task related locations are fixated and remembered regardless of visual saliency. For these scenes, data-driven approaches are more accurate predictors [75, 92]. Bernhard et al. [16] learned the gaze patterns from multiple game users to construct an importance map in a 3D game scene to predict visual attention of gamers at run-time. A detailed review of the feature based and data driven approaches can be found in [58].

Social saliency is another stimulus that drives attention. This social saliency states that you are likely to look at what others look at [114]. Friesen and Kingstone [41] showed that gaze is a strong social attention stimulus that can trigger attention shifts. A study by Birmingham and Kingstone [18] further confirmed that gaze is more likely fixated on the eyes than any other stimuli such as low level features. Also the direction of the eyes influences the fixation points more than other directional stimuli such as an arrow.

Similar to social saliency, our predictive model exploits joint attention, or social charges that attract attention of members in a social group. Inspired by the study of electric fields, a gradient field induced by the social charges is used to define the relationship between gaze behaviors in Chapter 7. Within the relational model, we adopt the Winner-Take-All strategy for selective gaze behavior, i.e., humans cannot pay attention to multiple sources of attention simultaneously. Also most previous approaches predict gaze behavior in a 2D plane while our method estimates sources of attention and gaze pose in 3D, which results in view- and feature-independent prediction.

Part I

Social Signal Reconstruction

"To signal is human."

--–A. Pentland (2010) [108]

The attributes of social interactions appear in the form of social signals and thus, these signals provide a strong cue to characterize social scenes. In part I, we present a computational framework for social signal perception and representation using social cameras. We reconstruct social signals in a unified 3D coordinate system, which allows us to further study a relational model of social behaviors. Our analysis focuses on three social signals: primary gaze direction, human body motion, and general scene motion.

Gaze is the most prominent social signal that exhibits attention and humans can detect gaze movement and infer a change in attention without difficulty [114]. In Chapter 3, we develop a method to interpret gaze behaviors for artificial agents by reconstructing gaze behaviors in 3D. We model gaze using primary gaze direction that can be estimated by exploiting the relationship with social cameras.

Human body motion such as gestures constitutes a secondary communication channel that conveys the intent during social interactions. In Chapter 4, we study a 3D representation of such human body motion using a set of articulated trajectories. We apply spatial and temporal constraints simultaneously to resolve the fundamental ambiguity of reconstruction.

Social signals often appear in the form of subtle motion. For such subtle signals, motion undergoes arbitrary deformation where a predefined spatial structure may not suffice to represent it. In Chapter 5, we study a 3D representation of social scene motion where a topological change may occur by modeling each point trajectory independently.

Chapter 3

3D Reconstruction of Primary Gaze Direction

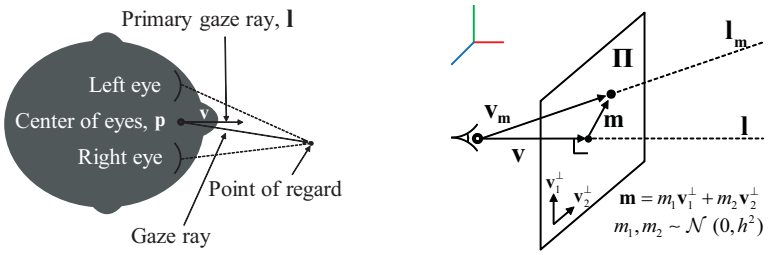
Gaze direction is the most prominent social signal related to one's attention. Humans perceive the gaze direction of others during interactions and detect gaze movement without difficulty. This perceptual ability facilitates inferring the attention of others and in turn, this influences their social behaviors. For instance, when your friend turns her gaze towards the other people behind you, you will perceive the change in her gaze direction and may turn your gaze as well. In this chapter, we present a method to reconstruct gaze direction in 3D from social cameras. We model gaze using primary gaze direction emitted from the center of the eyes and aligned with the head orientation. Given the gaze model, we introduce two approaches to reconstruct the primary gaze direction by leveraging ego- and exo-motion of social cameras. This 3D representation in a unified coordinate system further enables us to analyze attentive behavior in social groups.

3.1 Gaze Model

We represent the viewer's gaze direction as a 3D ray that is emitted from the center of the eyes and is directed towards the point of regard, as shown in Figure 3.1(a). The center of the eyes is fixed with respect to the head position and therefore, the orientation of the gaze ray in the world coordinate system is a composite of the head orientation and the eye orientation (eye-in-head motion). A primary gaze ray can capture only the head position and orientation but not the eye orientation. However, when the motion of the point of regard is stabilized, i.e., when the point of regard is stationary or slowly moving with respect to the head pose, the eye orientation varies by a small degree [7, 73, 88] from the primary gaze ray. We represent the variation of the gaze ray with respect to the primary gaze ray by a Gaussian distribution on a plane normal to the primary gaze ray. The point of regard (and consequently, the gaze ray) is more likely to be near the primary gaze ray.

Let us define the primary gaze ray $\mathbf{l}$ by the center of the eyes $\mathbf{p} \in \mathbb{R}^3$, and the unit direction vector, $\mathbf{v} \in \mathbb{R}^3$ in the world coordinate system, $\mathcal{W}$, as shown in Figure 3.1(a). Any point on the primary gaze ray can be written as $\mathbf{p} + \alpha\mathbf{v}$ where $\alpha > 0$.

Let $\mathbf{\Pi}$ be a plane normal to the primary gaze ray $\mathbf{l}$ at unit distance from $\mathbf{p}$, as shown in



(a) Head top view (b) Gaze ray model

Figure 3.1: (a) The primary gaze ray is a fixed 3D ray with respect to the head coordinate system and the gaze ray can be described by an angle with respect to the primary gaze ray. (b) The variation of the eye orientation is parameterized by a Gaussian distribution of the points on the plane $\mathbf{\Pi}$, which is normal to the primary gaze ray $\mathbf{l}$ at unit distance from $\mathbf{p}$.

Figure 3.1(b). The point $\mathbf{m}$ in $\mathbf{\Pi}$ can be written as $\mathbf{m} = m_1\mathbf{v}_1^{\perp} + m_2\mathbf{v}_2^{\perp}$ where $\mathbf{v}_1^{\perp}$ and $\mathbf{v}_2^{\perp}$ are two orthogonal vectors to $\mathbf{v}$ and m_1 and m_2 are scalars drawn from a Gaussian distribution, i.e., $m_1,\ m_2 \sim \mathcal{N}(0, h^2)$. This point $\mathbf{m}$ corresponds to the ray $\mathbf{l_m}$ in 3D. Thus, the distribution of the points on the plane maps to the distribution of the gaze ray by parameterizing the 3D ray as $\mathbf{l_m}(\mathbf{p}, \mathbf{v_m}) = \mathbf{p} + \alpha\mathbf{v_m}$ where $\mathbf{v_m} = \mathbf{v} + \mathbf{m}$ and $\alpha > 0$.

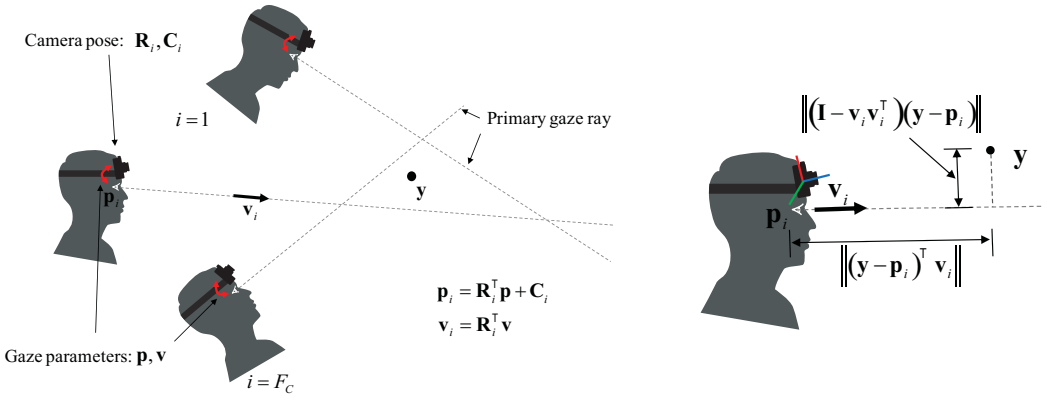
3.2 Estimation of Primary Gaze Direction

Given the gaze model, we estimate a primary gaze direction by leveraging the ego- and exo-motion of social cameras. When a person wears a social camera, the ego-motion of the camera directly encodes the primary gaze direction, i.e., the camera sees what the wearer sees. In Section 3.2.1, we describe a calibration method to find the relative transform between the social camera and primary gaze direction from the predefined gaze range-of-motion sequence. For the exo-motion approach, we introduce a method to reconstruct 3D face poses detected by multiple social cameras in Section 3.2.2. The reconstructed faces allow us to estimate the primary gaze direction.

3.2.1 Ego-motion Approach

A social camera may not be aligned with the direction of the primary gaze ray. In general, its center may not coincide with the center of the eyes either as shown in Figure 3.2(b). The orientation and position offsets between the social camera and primary gaze ray must be calibrated to estimate where the person is looking.

The relative transform between the primary gaze ray and the camera pose is constant across time because the camera is, for the most part, stationary with respect to the head, $\mathcal{C}$, as shown in Figure 3.2(b). Once the relative transform and camera pose have been estimated, the primary gaze ray can be recovered. We learn the primary gaze ray parameters, $\mathbf{p}$ and $\mathbf{v}$, with respect to the camera pose and the standard deviation h of eye-in-head motion.



(a) Gaze calibration (b) Gaze geometry

Figure 3.2: (a) We calibrate the fixed relationship between our gaze model and social camera. We parameterize the gaze model with the center of eye, $\mathbf{p}$, and the primary gaze direction, $\mathbf{v}$ in camera coordinate system. We ask people to move while fixating their gaze at a stationary point, i.e., the point of regard, $\mathbf{y}$. From multiple frames, $i = 1, \cdots, F_c$, the relationship can be estimated. (b) We calibrate the relationship by optimizing Equation (3.1). The objective function minimizes reprojection error of the point of regard.

We ask people to move while fixating their gaze at a stationary point in 3D (predefined gaze range-of-motion). Let $\mathbf{y}$ be the stationary point, i.e., the point of regard. The gaze ray parameters, $\mathbf{p}$ and $\mathbf{v}$, are relative vectors represented by the camera coordinate system, $\mathcal{C}$. These vectors can be represented in the world coordinate system, i.e., $\mathbf{p}_i = \mathbf{R}_i^\mathsf{T}\mathbf{p} + \mathbf{C}_i$ and $\mathbf{v}_i = \mathbf{R}_i^\mathsf{T}\mathbf{v}$ where $\mathbf{R}_i$ and $\mathbf{C}_i$ be the camera orientation and the camera center at the i^{th} time instant, respectively. The primary gaze ray represented by $\mathbf{p}_i$ and $\mathbf{v}_i$ must pass through the point of regard, $\mathbf{y}$, i.e., $\mathbf{y} = \mathbf{p}_i + \alpha_i\mathbf{v}_i$ for some α_i, if there is no eye-in-head motion. All rays, $\{\mathbf{p}_i + \alpha_i\mathbf{v}_i\}_{i=1,\cdots,F_c}$, must intersect at the point, $\mathbf{y}$, where F_c is the number of frames. Due to the eye-in-head motion, the rays do not meet at a point in general. We estimate the stationary point, $\mathbf{y}$, and the gaze ray parameters, $\mathbf{p}$ and $\mathbf{v}$, simultaneously by minimizing the following objective function,

$$\begin{aligned} &\underset{\mathbf{p},\mathbf{v},\mathbf{y}}{\text{minimize}} \sum_{i=1}^{F_c} \left\| \frac{\left(\mathbf{I} - \mathbf{v}_i\mathbf{v}_i^\mathsf{T}\right)\left(\mathbf{y} - \mathbf{p}_i\right)}{\left(\mathbf{y} - \mathbf{p}_i\right)^\mathsf{T}\mathbf{v}_i} \right\|^2 + \lambda_c \left\|\mathbf{p}\right\|^2 \\ &\text{subject to} \quad \mathbf{p}_i = \mathbf{R}_i^\mathsf{T}\mathbf{p} + \mathbf{C}_i, \quad \mathbf{v}_i = \mathbf{R}_i^\mathsf{T}\mathbf{v}, \end{aligned} \tag{3.1}$$

where λ_c is a weight on the regularization. The first term in Equation (3.1) accounts for the sum of reprojection error across the time instances: $\| \left(\mathbf{I} - \mathbf{v}_i\mathbf{v}_i^\mathsf{T}\right)\left(\mathbf{y} - \mathbf{p}_i\right) \|$ is distance from the ray to $\mathbf{y}$ and $\| \left(\mathbf{y} - \mathbf{p}_i\right)^\mathsf{T}\mathbf{v}_i \|$ is distance from the center of eyes to the projection of $\mathbf{y}$ onto the ray. The second term in Equation (3.1) prevents a trivial solution: if the center of eyes, $\mathbf{p}$, is along the ray but is infinitely far away from $\mathbf{y}$, the reprojection error becomes zero because $\| \left(\mathbf{y} - \mathbf{p}_i\right)^\mathsf{T}\mathbf{v}_i \| \rightarrow \infty$. We look for $\mathbf{p}$ close to the center of eyes. Equation (3.1) is a nonlinear optimization which requires good initial estimates. Note that $\mathbf{y}$ is unknown, which makes Equation (3.1) a bilinear optimization. We initialize $\mathbf{p} = \mathbf{0}$ such that the center of eyes coincides with the camera center and $\mathbf{v} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^\mathsf{T}$ such that the primary gaze ray aligns with the camera facing direction

Figure 3.3: We reconstruct primary gaze direction in 3D based on faces detected in social cameras. We find faces from an off-the-shelf face detector, align the landmarks of the faces, and reconstruct the face poses in 3D. Reprojection of the 3D face landmarks and primary gaze directions are illustrated.

(the camera facing direction is the z axis in the camera coordinate system. $\mathbf{y}$ is initialized as the least squares solution from the triangulation of all initialized rays. This is a valid initialization when the primary gaze ray does not significantly deviate from the camera facing direction. Once $\mathbf{p}$ and $\mathbf{v}$ are estimated from Equation (3.1), the standard deviation of the reprojection, h, can be obtained, i.e., $h = \texttt{std}\left\{\frac{\left(\mathbf{I}-\mathbf{v}_i\mathbf{v}_i^\mathsf{T}\right)(\mathbf{y}-\mathbf{p}_i)}{(\mathbf{y}-\mathbf{p}_i)^\mathsf{T}\mathbf{v}_i}\right\}_{i=1,\cdots,F_c}$.

3.2.2 Exo-motion Approach

A face has a discriminative pattern [126] that can be reliably detected and aligned in an image. The detected faces from multiple social cameras can be used to reconstructed their poses in 3D. In this section, we present a framework to estimate primary gaze direction via face reconstruction.

We find faces in an image using an off-the-shelf face detector (OpenCV or PittPatt) that roughly estimates the location and orientation of the faces, which provides a good initialization for 2D face landmark alignment. The face alignment algorithm using the supervised descent method (IntraFace) by Xiong and De la Torre [163] precisely localizes the landmarks of the faces efficiently given the initialization. We apply the face detection and alignment algorithms on all images taken at the same time instant. This generates a set of the detected faces, $\mathcal{G} = \{\mathbf{g}_i\}_{i=1,\cdots,I}$, where $\mathbf{g}_i = \left(\mathbf{x}_i^1, \cdots, \mathbf{x}_i^P\right)$ is the i^{th} face that consists of P 2D landmarks, $\mathbf{x}_i \in \mathbb{P}^2$ and I is the number of the detected faces.

In the set, some faces correspond to the same face. We decompose the set of the detected faces into mutually exclusive sets, i.e.,

$$\mathcal{G} = \bigcup_{j=1}^{J} \mathcal{G}_j \qquad \text{and} \qquad \mathcal{G}_k \bigcap \mathcal{G}_l = \emptyset \quad \text{for } k \neq l, \tag{3.2}$$

where J is the number of faces and $\mathcal{G}_j$ is the subset of the detected faces corresponding to the j^{th}

face in the scene, i.e., each subset corresponds to one face. Once we decompose the set, $\mathcal{G}$, we can reconstruct the landmarks of each face in 3D given the camera poses.

To decompose the set, we sequentially apply the RANSAC framework that verifies geometric consistency of the detected faces. We randomly select two faces in $\mathcal{G}$ which do not belong to the same camera[1] and triangulate the landmarks of the faces in 3D given the camera poses. The triangulated landmarks are projected to each camera of the detected face to evaluate reprojection error. If the reprojection error is lower than a threshold, we include the face in the inlier set, i.e, the reprojection error of face correspondences must be low. Once we find the largest inlier set, $\mathcal{G}_{\text{in}}$, we re-initialize the set by excluding the inlier set, i.e., $\mathcal{G} \leftarrow \{\mathbf{g}_k | \mathbf{g}_k \notin \mathcal{G}_{\text{in}}\}$, and iterate the RANSAC process until $\mathcal{G}$ becomes empty. This process allows us to decompose the $\mathcal{G}$ into mutually exclusive subsets, $\{\mathcal{G}_j\}_{j=1,\cdots,J}$.

From each subset corresponding to one face, we reconstruct the face via triangulation, followed by nonlinear refinement by minimizing the following reprojection error of the landmarks:

$$\underset{\mathbf{G}_j}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{p=1}^{P} d\left(\mathbf{P}_k \mathbf{X}_j^p, \mathbf{x}_k^p\right)^2, \quad K = |\mathcal{G}_j|, \tag{3.3}$$

where $\mathbf{G}_j = \left(\mathbf{X}_j^1, \cdots, \mathbf{X}_j^P\right)$ is the j^{th} face that consists of P 3D landmarks, $\mathbf{X}_i \in \mathbb{P}^3$ and $\mathbf{P}_k$ is a camera projection matrix corresponding to the k^{th} detected face. $d(\mathbf{y}_1, \mathbf{y}_2)$ measures the Euclidean distance between the inhomogeneous coordinates of $\mathbf{y}_1$ and $\mathbf{y}_2$.

Given the reconstructed faces, we find a temporal association by a nearest neighbor search in 3D landmark locations between frames. We refine the face poses over time initialized by the temporal association to minimize reprojection error and temporal inconsistency:

$$\underset{\{\mathbf{G}_j^t\}_{t=1,\cdots,T}}{\operatorname{argmin}} \sum_{t=1}^{T} \sum_{k=1}^{K} \sum_{p=1}^{P} d\left(\mathbf{P}_k(t) \mathbf{X}_j^p(t), \mathbf{x}_k^p(t)\right)^2 + \lambda \sum_{t=1}^{T-1} \sum_{p=1}^{P} \left\| \mathbf{X}_j^p(t) - \mathbf{X}_j^p(t+1) \right\|^2, \tag{3.4}$$

where $\mathbf{P}_k(t)$, $\mathbf{X}_j^p(t)$, and $\mathbf{x}_k(t)$ are $\mathbf{P}_k$, $\mathbf{X}_j^p$, and $\mathbf{x}_k$ at time t, respectively. T is the sequence length and λ controls the weight on the temporal consistency term.

Based on the 3D landmarks of the reconstructed faces, we estimate the corresponding primary gaze ray. The origin of the ray is set to the landmark location of the center of the eyes. We estimate the direction that is orthogonal to a plane made of the extremities of the landmark locations: these extremity points are approximately coplanar and the plane made of these points can represent the head orientation. This allows us to parameterize the primary gaze ray using face landmark locations in 3D as shown in Figure 3.3.

3.3 Summary

In this section, we present a method to reconstruct the social signal transmitted via a person's gaze behavior in 3D from social cameras. We represent gaze using primary gaze direction emitted from the center of the eyes and directed towards the point of regard. Eye-in-head motion

[1]Faces detected in the same camera cannot be the same face.

is modeled using the variance of the gaze direction with respect to primary gaze direction. We introduce two approaches to estimate the gaze direction in a unified 3D coordinate system: an ego-motion approach and an exo-motion approach. As a social camera directly encodes the gaze direction, we demonstrate an algorithm to calibrate the fixed relationship between the social camera and primary gaze direction from the predefined gaze range-of-motion sequence for the ego-motion approach. In the exo-motion approach, the faces captured by multiple social cameras are detected, aligned, and reconstructed. The reconstructed faces are refined by minimizing reprojection error and maximizing temporal consistency, and used to estimate primary gaze direction.

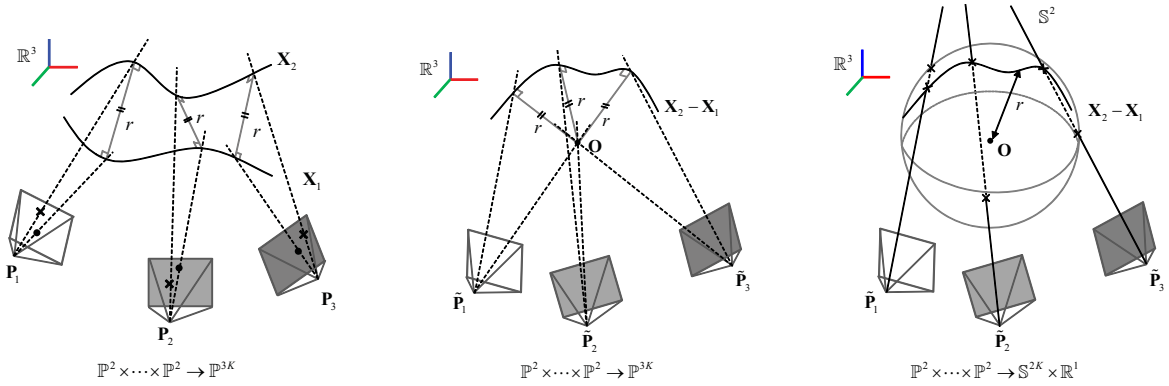
Chapter 4

3D Reconstruction of Human Body Motion

Reconstructing a moving point in 3D from a sequence of two dimensional projections is an ill-posed problem; any point on the line of projection connecting the camera's optical center and an image measurement can be a solution. Yet, humans can effortlessly perceive depth if the 2D points correspond to articulations of a known skeleton [65]. We study the conjecture that if 3D points move smoothly with a known articulation structure, then it is possible to reconstruct their 3D locations from their 2D projections — without any activity specific prior. The reconstruction of an *articulated* trajectory has a fundamental ambiguity because there are two intersecting points that satisfy an articulation constraint and an image measurement at each time instant [76]: for a 2D trajectory of F frames, there are 2^F 3D trajectories that remain at fixed distance to a parent trajectory[1]. The reconstruction of a *smooth* trajectory without spatial constraints is also known to be fundamentally ambiguous when the camera trajectory is smooth [100, 104]. We present an algorithm to reconstruct a smooth articulated trajectory in 3D by *simultaneously* applying articulation and smoothness constraints. The algorithm takes as input 2D projections of the trajectory, its parent trajectory in 3D, and the camera pose at each time instant. We present a measure of reconstructability of an articulated trajectory which characterizes the stability of estimation under articulation and smoothness constraints.

Each trajectory is parameterized by coefficients of trajectory basis vectors in the spherical coordinate system to enforce smoothness and articulation constraints. We show that if a trajectory is embedded in the trajectory basis vectors and articulation constraints are applied, the reconstruction problem is equivalent to a binary quadratic program which is known to be NP-hard [43]. A number of algorithms exist that produce an approximate solution [87, 98, 113] and we use a branch-and-bound method to produce an initialization. We refine the articulated trajectories by minimizing reprojection error. The results are smooth, length preserved 3D trajectories. We have applied our algorithm to recursively reconstruct the 3D motion of a human given the 3D motion of its root. Two general approaches have been explored in prior literature to reconstruct human articulated body motion. Data-driven approaches use repositories of exemplars to overcome the ambiguity [27, 60, 146, 147, 156] and physics-based approaches use dynamical models of the human body to fit to the image stream [23, 151, 157]. Unlike these approaches, our approach reconstructs human motion from *purely geometric constraints*. Thus, the target motion is not confined to predefined activities or view points.

[1]The parent trajectory in a skeleton hierarchy is the proximal trajectory to the root trajectory and the child trajectory is the distal trajectory.



(a) Fixed length trajectory (b) Relative trajectory (c) Trajectory on a sphere

Figure 4.1: (a) An articulated trajectory is defined as a trajectory $\mathbf{X}_2$ which preserves distance from its parent trajectory $\mathbf{X}_1$ across all time instances. (b) The articulated trajectory is transformed to the relative trajectory, $\mathbf{X}_2 - \mathbf{X}_1$, by collapsing $\mathbf{X}_1$ to the origin. (c) The articulated trajectory lies on a sphere of radius r. There are two intersecting points at each time instant between the sphere and the ray connecting the camera's optical center and an image measurement, which allow 2^F possible 3D trajectories.

4.1 Geometry of an Articulated Trajectory

A point trajectory in 3D without any constraint can be represented by a series of points:

$$\mathbf{X} = \left(\begin{bmatrix} x_1 \\ \vdots \\ x_F \end{bmatrix}, \begin{bmatrix} y_1 \\ \vdots \\ y_F \end{bmatrix}, \begin{bmatrix} z_1 \\ \vdots \\ z_F \end{bmatrix} \right), \tag{4.1}$$

where (x_i, y_i, z_i) is the Cartesian coordinate of a point at i^{th} time instant and F is the number of frames. If a trajectory is smooth, it is known that the trajectory can be expressed by a linear combination of compact trajectory basis vectors [4], i.e.,

$$\mathbf{X} = (\mathbf{\Theta}\mathbf{a}_x, \mathbf{\Theta}\mathbf{a}_y, \mathbf{\Theta}\mathbf{a}_z) \tag{4.2}$$

where $\mathbf{\Theta}$ is a $F \times K$ matrix composed of a collection of linear trajectory basis vectors, $\mathbf{a}$ is the coefficients or the parameters of a trajectory, and K is the number of basis vectors.

If two trajectories, $\mathbf{X}_1$ and $\mathbf{X}_2$, are articulated, the distance between trajectories remains constant across all time instances as shown in Figure 4.1(a), i.e.,

$$\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2 = r^2, \quad i = 1, \cdots, F, \tag{4.3}$$

where $\Delta\mathbf{X} = \mathbf{X}_2 - \mathbf{X}_1$ is the relative trajectory.

When a perspective camera captures these two trajectories, points on the trajectories at the time instant are projected onto the camera plane. The camera representation in this chapter is a 3×4 projection matrix, $\mathbf{P}_i = \mathbf{K}\mathbf{R}_i \left[\begin{array}{c|c} \mathbf{I}_3 & -C_i \end{array} \right]$ where $\mathbf{I}_3$, $\mathbf{K}$, $\mathbf{R}_i$, and C_i are a 3×3 identity

matrix, the upper triangular intrinsic matrix, the camera rotation matrix, and the camera's optical center vector at the i^{th} time instant, respectively.

If we transform one of the trajectories, $\mathbf{X}_1$, to the origin, $\mathbf{O}$, the other trajectory, $\mathbf{X}_{\widetilde{2}}$, maps to the relative trajectory, $\Delta\mathbf{X}$, and a camera, $\mathbf{P}_i$, maps to the relative camera pose, $\widetilde{\mathbf{P}}_i$ with respect to $\mathbf{X}_1$ as shown in Figure 4.1(b). The transformed relative trajectory lies on a sphere with radius r. There are two points intersecting the sphere and the ray connecting the camera's optical center and an image measurement at each time instant as shown in Figure 4.1(c). All intersecting points are candidate 3D points which the relative trajectory passes and thus, there are 2^F possible relative trajectories.

The representation of a relative trajectory between the articulated trajectories from Equation (4.2), which is a Cartesian coordinate representation, has to meet the additional quadratic equality constraints of Equation (4.3). Instead of the Cartesian coordinate representation, we introduce the spherical coordinate representation for a relative trajectory to control the distance between trajectories, explicitly, i.e.,

$$\Delta\mathbf{X} = (\mathbf{\Theta}\mathbf{a}_\theta, \mathbf{\Theta}\mathbf{a}_\phi, r)\,, \tag{4.4}$$

where θ is inclination from the z axis, ϕ is azimuth from the x axis in the xy plane, and r is the radius . This representation enables us to describe an articulated trajectory precisely because it satisfies the temporal constraint and the length constraint simultaneously regardless of parameters by setting the radius constant explicitly. It also enforces that all imputed points between frames satisfy the articulation constraint while the Cartesian representation does not. From a topological point of view, the reconstruction from the spherical coordinate system is the mapping of $\mathbb{P}^{2F} \rightarrow \mathbb{S}^{2K} \times \mathbb{R}^1$ while the reconstruction from the Cartesian coordinate system is the mapping of $\mathbb{P}^{2F} \rightarrow \mathbb{P}^{3K}$ as shown in Figure 4.1(c).

4.2 Articulated Trajectory Reconstruction

In this section, we present an algorithm for recovering a trajectory which satisfies spatial and temporal constraints using the spherical coordinate representation of a relative trajectory described in the previous section.

4.2.1 Objective Function of 3D Reconstruction

From the spherical coordinate representation, we reconstruct smooth articulated trajectories which minimize the reprojection errors:

$$\underset{\Delta\mathbf{X}_1,\cdots,\Delta\mathbf{X}_P}{\operatorname{argmin}} \sum_{i,j}^{F,P} d\left(\mathbf{x}_{ij}, \hat{\mathbf{x}}_{ij}\right), \tag{4.5}$$

where $\Delta\mathbf{X}_j$ is the j^{th} articulated (or relative) trajectory parameterized by $(\mathbf{\Theta}\mathbf{a}_{\theta,j}, \mathbf{\Theta}\mathbf{a}_{\phi,j}, r_j)$, $d(\cdot,\cdot)$ is the L_2 distance between two arguments, P is the number of articulated points, and $\mathbf{x}_{ij}$ and $\hat{\mathbf{x}}_{ij}$ are a 2D image measurement and a reprojection of the j^{th} point trajectory at the i^{th} time instant, respectively.

If articulated trajectories are sequentially linked, the trajectories are

$$\mathbf{X}_j = f(\mathbf{X}_R; \Delta\mathbf{X}_1, \cdots, \Delta\mathbf{X}_{j-1}), \tag{4.6}$$

where $f(\cdot)$ is the forward kinematic function that takes the root trajectory, $\mathbf{X}_R$, and all parent relative trajectories, $\Delta\mathbf{X}_1, \cdots, \Delta\mathbf{X}_{j-1}$, and outputs the j^{th} trajectory, $\mathbf{X}_j$, in the Cartesian coordinate system. The reprojection, $\hat{\mathbf{x}}_{ij}$ is

$$\hat{\mathbf{x}}_{ij} = \left(\frac{\mathbf{P}_i^1 \widetilde{\mathbf{X}}_j(i)}{\mathbf{P}_i^3 \widetilde{\mathbf{X}}_j(i)}, \frac{\mathbf{P}_i^2 \widetilde{\mathbf{X}}_j(i)}{\mathbf{P}_i^3 \widetilde{\mathbf{X}}_j(i)} \right), \tag{4.7}$$

where $\mathbf{P}_i^l$ is the l^{th} row of the camera projection matrix at the i^{th} time instant and $\widetilde{\mathbf{X}}_j(i)$ is the homogeneous representation of the i^{th} point in the j^{th} trajectory, $\mathbf{X}_j(i)$.

4.2.2 Initialization of Objective Function

The objective function of Equation (4.5) is highly nonlinear and direct optimization falls into a local minimum. Therefore, a good initialization of trajectory parameters is necessary. When the parent joint position and the length between trajectories are known, there are two intersecting points between a sphere whose origin is the parent joint position, X_p, and a line connecting an image measurement and camera optical center, C, at each time instant as shown in Figure 4.2(a). A point lying on the line is $C + s\mathbf{v}$ where s is an unknown scalar and $\mathbf{v}$ is the direction of the projection, i.e.,$\mathbf{v} = \mathbf{R}^\mathsf{T}\mathbf{K}^{-1} \begin{bmatrix} \mathbf{x}^\mathsf{T} 1 \end{bmatrix}^\mathsf{T}$. Then, the intersecting points are

$$^1X = C + s_1\mathbf{v}, \;\; ^2X = C + s_2\mathbf{v}, \tag{4.8}$$

where

$$s_{1,2} = \frac{-\mathbf{v}^\mathsf{T}\Delta C \pm \sqrt{(\mathbf{v}^\mathsf{T}\Delta C)^2 - \|\mathbf{v}\|^2 \left(\|\Delta C\|^2 - r^2\right)}}{\|\mathbf{v}\|^2} \tag{4.9}$$

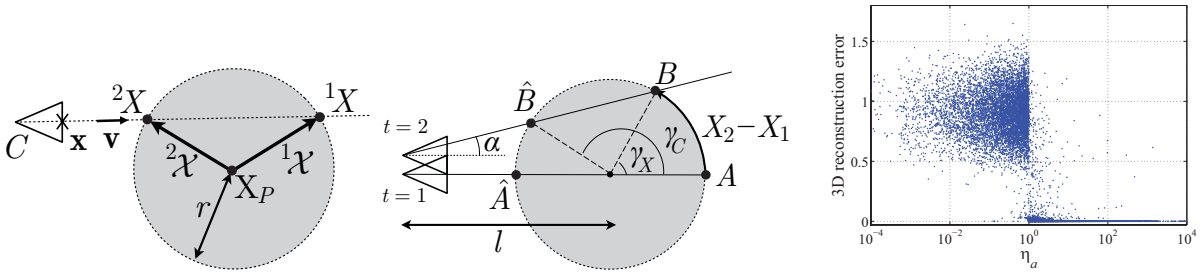
and $\Delta C = C - X_p$. For each time instant, we have two candidate 3D points through which the reconstructed trajectory must pass. Across all time instances, there are 2^F possible trajectories which satisfy the image measurements. Among those trajectories, we look for the trajectory best described by the trajectory basis vectors.

Let χ be the relative direction vector with respect to the parent point as shown in Figure 4.2(a). For each time instant, χ_i takes either $^1\chi_i$ or $^2\chi_i$, i.e.,

$$\begin{aligned} \chi_i &= {}^1\chi_i b_i + {}^2\chi_i(1 - b_i), \\ &= ({}^1\chi_i - {}^2\chi_i)b_i + {}^2\chi_i, \;\; \text{where } b_i \in \{0, 1\}. \end{aligned} \tag{4.10}$$

Then, all possible trajectories can be represented as:

$$\begin{aligned} \begin{bmatrix} \chi_1 \\ \vdots \\ \chi_F \end{bmatrix} &= \begin{bmatrix} \Delta\chi_1 & & \\ & \ddots & \\ & & \Delta\chi_F \end{bmatrix} \mathbf{b} + \begin{bmatrix} {}^2\chi_1 \\ \vdots \\ {}^2\chi_F \end{bmatrix} \\ \text{or} \;\; \boldsymbol{\chi} &= \mathbf{E}\mathbf{b} + \mathbf{F}, \end{aligned} \tag{4.11}$$



(a) Intersection (b) Relative trajectory (c) Reconstructability

Figure 4.2: (a) There are two solutions, ${}^1X$ and ${}^2X$ which satisfy the articulation constraint and an image measurement. (b) The articulated trajectory and the camera pose are transformed with respect to the parent trajectory. (c) The accuracy of the reconstruction is high when η_a is greater than 1 where the trajectory basis vectors span the ground truth trajectory better than the impostor trajectory.

where $\mathbf{b}$ is a binary variable vector, ${}^1\chi_i$ and ${}^2\chi_i$ are two relative direction vectors, and $\Delta\chi_i = {}^1\chi_i - {}^2\chi_i$. Finding the best trajectory is equivalent to finding the binary vector, $\mathbf{b}$, which minimizes the following cost,

$$\begin{aligned} \mathbf{b}^* = \underset{\mathbf{b}}{\operatorname{argmin}} & \left\| \left(\boldsymbol{\Theta}\boldsymbol{\Theta}^\mathsf{T} - \mathbf{I} \right) \left(\mathbf{E}\mathbf{b} + \mathbf{F} \right) \right\|^2, \\ \text{subject to} \quad & \mathbf{b} \in \{0,1\}^F. \end{aligned} \tag{4.12}$$

Note that $\boldsymbol{\Theta}\boldsymbol{\Theta}^\mathsf{T} - \mathbf{I}$ is the projection operation onto the null space of the trajectory basis vectors, $\boldsymbol{\Theta}$. Equation (4.12) is a quadratic problem over binary variables.

A binary quadratic programming problem is NP-hard in general. The structure of our problem does not fall into one of the solvable cases; our quadratic matrix has positive off-diagonal elements [111], is a non-singular matrix [5, 39], and cannot be represented by a tri-/five-diagonal matrix [48]. Also, the underlying graph structure is not series parallel [11]. Thus, in theory, this is an intractable problem. However, a number of approaches have been proposed to approximate a solution of the problem efficiently using spectral or semidefinite relaxation. A branch-and-bound routine with binary relaxation is one technique for global optimization. Since our quadratic matrix is positive definite, the objective function behaves convexly in a branched rectangle, which enables us to define a tight lower bound of the rectangle in polynomial time.

Once $\mathbf{b}^*$ is recovered, we project $\boldsymbol{\chi} = \mathbf{E}\mathbf{b}^* + \mathbf{F}$ onto the trajectory basis space of the spherical coordinate system to produce low dimensional parameters, i.e., $\Delta\mathbf{X} = (\boldsymbol{\Theta}\mathbf{a}_\theta, \boldsymbol{\Theta}\mathbf{a}_\phi, r)$. This yields an accurate initialization which can be refined by nonlinear optimization of Equation (4.5).

When the relative trajectory, $\Delta\mathbf{X}$, passes a singular point in the spherical coordinate system in the process of projecting $\boldsymbol{\chi}$ onto the spherical coordinate system, a discontinuity of angular trajectory occurs. For example, when ϕ passes from $\epsilon > 0$ to $2\pi - \epsilon$, this results in a discontinuity of the angular trajectory because ϕ is defined in the interval $[0, 2\pi)$. To deal with discontinuous trajectories, we find the best angular representation among all spherical representations of χ which preserves local continuity by allowing the domains of θ and ϕ to be $(-\infty, \infty)$.

4.3 Geometric Analysis

We now explore the reconstruction ambiguity of an articulated trajectory and analyze configurations in which the reconstruction is accurate. Let $\mathbf{X}_1$ be a known parent trajectory and $\mathbf{X}_2$ be an articulated child trajectory which are observed at two time instances as shown in Figure 4.2(b). The ground truth relative trajectory between $\mathbf{X}_1$ and $\mathbf{X}_2$ moves from A to B. $\hat{A}$ and $\hat{B}$ are impostor points that satisfy the image measurements as well as the articulation constraint. In this section, we show that the relationship between the true trajectory and the impostor trajectory inherently determines the reconstruction accuracy.

We define a measure of *reconstructability of an articulated trajectory*, η_a, as a criterion to characterize reconstruction accuracy where

$$\eta_a = \frac{\left\| \boldsymbol{\Theta}^{\perp} \mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{C}} \right\|}{\left\| \boldsymbol{\Theta}^{\perp} \mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{X}} \right\|}, \tag{4.13}$$

$\boldsymbol{\gamma}_\mathbf{X} = \boldsymbol{\Theta}\mathbf{a}_{\boldsymbol{\gamma}_\mathbf{X}} + \boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{X}}$, $\boldsymbol{\gamma}_\mathbf{C} = \boldsymbol{\Theta}\mathbf{a}_{\boldsymbol{\gamma}_\mathbf{C}} + \boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{C}}$, and $\boldsymbol{\Theta}^{\perp}$ is the null space of the trajectory basis vectors. If the reconstructability of an articulated trajectory goes to infinity, there exists a unique solution and it corresponds to the ground truth trajectory. This can be proven by the following. For each time instant, there are two intersecting points and an estimation should be one of them:

$$\hat{\gamma} = (1-b)\gamma_X + b\gamma_C, \quad b = 1 \text{ or } 0 \tag{4.14}$$

where $\hat{\gamma}$ is an estimated angle. For an estimated angular trajectory,

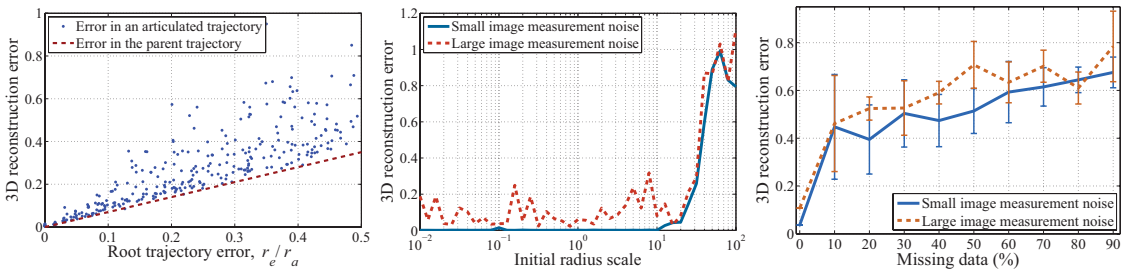
$$\hat{\boldsymbol{\gamma}} = (\mathbf{I} - \mathbf{B})\,\boldsymbol{\gamma}_\mathbf{X} + \mathbf{B}\boldsymbol{\gamma}_\mathbf{C}, \tag{4.15}$$

where $\mathbf{B}$ is a diagonal matrix whose entries are either 1 or 0. The best trajectory represented by the trajectory basis vectors minimizes the following:

$$\begin{aligned}
& \underset{\hat{\mathbf{a}},\mathbf{B}}{\operatorname{argmin}} \left\| \boldsymbol{\Theta}\hat{\mathbf{a}} - \hat{\boldsymbol{\gamma}} \right\|^2 && (4.16)\\
= \; & \underset{\hat{\mathbf{a}},\mathbf{B}}{\operatorname{argmin}} \left\| \boldsymbol{\Theta}\hat{\mathbf{a}} - (\mathbf{I} - \mathbf{B})\,\boldsymbol{\gamma}_\mathbf{X} - \mathbf{B}\boldsymbol{\gamma}_\mathbf{C} \right\|^2 && (4.17)\\
= \; & \underset{\hat{\mathbf{a}},\mathbf{B}}{\operatorname{argmin}} \left\| \boldsymbol{\Theta}\hat{\mathbf{a}} - (\mathbf{I} - \mathbf{B})\,\boldsymbol{\Theta}\mathbf{a}_{\boldsymbol{\gamma}_\mathbf{X}} - \mathbf{B}\boldsymbol{\Theta}\mathbf{a}_{\boldsymbol{\gamma}_\mathbf{C}} - (\mathbf{I} - \mathbf{B})\,\boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{X}} - \mathbf{B}\boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{C}} \right\|^2 . && (4.18)
\end{aligned}$$

Reconstructability of an articulated trajectory goes to infinity when $||\boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{C}}|| \to \infty$ or $||\boldsymbol{\Theta}^{\perp}\mathbf{a}^{\perp}_{\boldsymbol{\gamma}_\mathbf{X}}|| \to 0$. For either case, $\mathbf{B}$ has to approach $\mathbf{0}$ to eliminate the residual of the null components in Equation (4.18), which leads to $\hat{\mathbf{a}} \to \mathbf{a}_{\boldsymbol{\gamma}_\mathbf{X}}$.

From the method of Park et al. [104], if the camera motion is slow or stationary, there is no way to reconstruct an accurate trajectory using the trajectory basis vectors because it spans the camera trajectory well. The reconstructability of an articulation states that if the parent trajectory is independent of the camera trajectory, the trajectory reconstruction is still possible because mixed motion between the camera and the parent motions induces $\boldsymbol{\alpha}$ motion where $\boldsymbol{\alpha}$ is



(a) Error in the root trajectory (b) Initialization error of radius (c) Effect of missing data

Figure 4.3: (a) Performance of our algorithm against error in the root trajectory, (b) the initialization error of the radius, (c) amount of missing data are illustrated.

the trajectory of viewing angles from a camera, α, as shown in Figure 4.2(b). Even when camera and parent motions are stationary, the reconstruction is possible if $\boldsymbol{\gamma}_{\mathbf{X}} \in \boldsymbol{\Theta}$ because each α is a nonlinear function of γ_X, i.e., $\alpha = \tan^{-1}\left(\sin\gamma_X/(l+\cos\gamma_X)\right)$ where l is the distance between the parent trajectory and camera trajectory, and thus $\boldsymbol{\alpha} \notin \boldsymbol{\Theta}$ and $\boldsymbol{\gamma}_{\mathbf{C}} \notin \boldsymbol{\Theta}$ unless $l = 0$ or $l = \infty$ (i.e., orthographic projection) as shown in Figure 4.2(b).

Figure 4.2(c) shows the distribution of 3D reconstruction error with respect to reconstructibility of an articulated trajectory, η_a, from the CMU motion capture data[2]. A trajectory initialized by binary quadratic programming is the best fitted trajectory by the trajectory basis vectors. When η_a is high ($\gg 1$), 3D reconstruction error of an articulated trajectory is low because the ground truth trajectory is well described by the trajectory basis vectors and the ground truth trajectory and the impostor trajectory are well separable. In contrast, when η_a is low ($\ll 1$), our solution converges to the impostor trajectory because the trajectory basis vectors span the impostor trajectory better.

4.4 Results

To validate our method, we tested it with the HumanEva-II dataset, synthesized trajectories, and the CMU motion capture data quantitatively and with real human motion examples taken by video cameras qualitatively. We use the first K Discrete Cosine Transform (DCT) basis vectors[3] in order of increasing frequency and the number of basis vectors is chosen manually to span the trajectory well.

4.4.1 Quantitative Evaluation

We compare our method with the state-of-art in human pose estimation [6, 15, 61, 110] using the HumanEva-II dataset[4]. Subject S2 with camera C1 is used to reconstruct the articulated trajecto-

[2] `http://mocap.cs.cmu.edu/`

[3] Hamidi and Pearl [53] have shown that the DCT provides the optimal performance to encode a signal sampled from the first order Markov processes. Ahkter et al. [4] have empirically justified its optimality on motion capture data.

[4] `http://vision.cs.brown.edu/humaneva/`

ries. Our method results in 128.8mm of 3D mean error with 17.75mm standard deviation. This error is comparable to the error of the state-of-art pose estimation algorithms (82mm$\sim$211.4mm). It should be noted that while all methods rely on activity specific training data to reconstruct motions, our approach uses only activity independent geometric constraints.

We generate synthetic 2D perspective projections from synthetic data and the CMU motion capture data and evaluate for three aspects: error in the root trajectory, error in radius of an articulated trajectory, and missing data. For evaluation of errors in the root trajectory and radius, we set the camera stationary and vary error of the root trajectory and radius error while the root position is moving. For the evaluation of missing data, we artificially remove 2D projections randomly.

We measure 3D reconstruction error of an articulated trajectory by varying the ratio between the average distance error of the root trajectory, r_e, and the radius of the articulated trajectory, r_a, as shown in Figure 4.3(a). The error in the parent trajectory is a lower bound on the reconstruction error of the articulated trajectory. While the variance of the distribution for small root trajectory error (< 0.2) is low, i.e., the reconstruction can be done reliably, the reconstruction from high root trajectory error (> 0.3) causes high error in the child trajectory as well.

For the evaluation of the error in radius, we measure 3D reconstruction error for erroneous radii multiplied by scale[5]. Figure 4.3(b) illustrates robustness to erroneous initialization. Even though the initial scale is small (i.e., $10^{-2} \sim 10^{0}$), the 3D reconstruction can be done reliably because before solving the binary quadratic programming, we adjust the radius of the sphere to intersect with the line of projection at one point at least. When the initial scale is high ($> 10^{1}$), however, the reconstruction becomes unreliable because the ray intersects with the sphere at all time instances and the optimization falls into a local minimum around a mis-estimated trajectory.

We also test with the CMU motion capture data for the evaluation of missing data caused by occlusion or measurement failure. When there are missing data, our spatial and temporal constraints enable us to impute missing points. For this experiment, we artificially introduce length errors, image measurement noise, and root trajectory error while the camera is stationary. Our algorithm produces an average relative error[6] of 13% for 5% missing data as shown in Figure 4.3(c).

4.4.2 Qualitative Evaluation

We apply our algorithm to reconstruct human body motion in 3D from 2D perspective projections. Reconstruction from a stationary camera and a moving camera are tested and the statistical anthropometric length ratio of the human body is used for the initialization of the length ratio with some modifications for accurate skeleton estimation purpose. The scale of the skeleton is roughly initialized and we manually label image measurements for articulated points.

Figure 4.4(a) and Figure 4.4(b) show the reconstruction of the *juggling motion* and the *motion in front of a webcam*, respectively, in 3D from a stationary video camera. We project the 2D root trajectory to the unit depth plane and use it as the 3D root trajectory because the depth of the root trajectory is underdetermined from a stationary camera. For both experiments, we use the torso

[5]Initial radius scale error 1 means the ground truth.

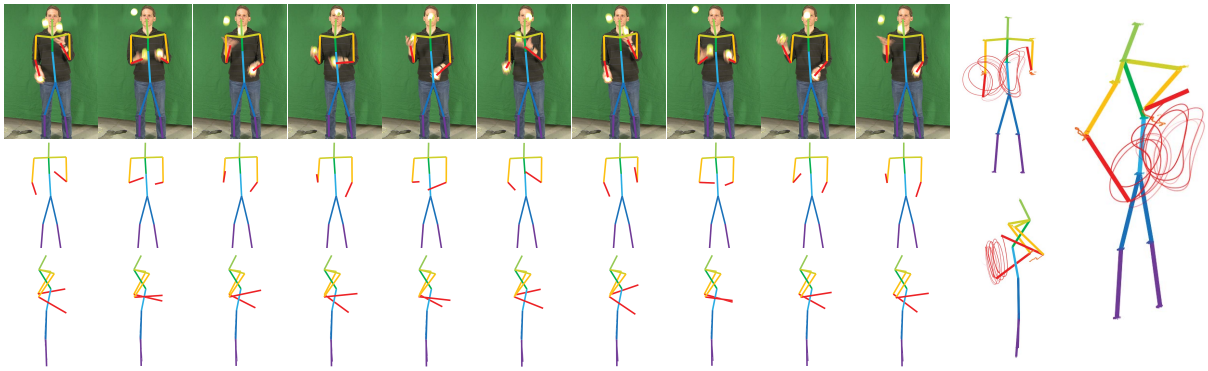
[6]error = $||\mathbf{X} - \hat{\mathbf{X}}||/||\mathbf{X}||$, where $\mathbf{X}$ is the ground truth trajectory and $\hat{\mathbf{X}}$ is the estimated trajectory.

as the root. From the root trajectory, all articulated trajectories are reconstructed recursively.

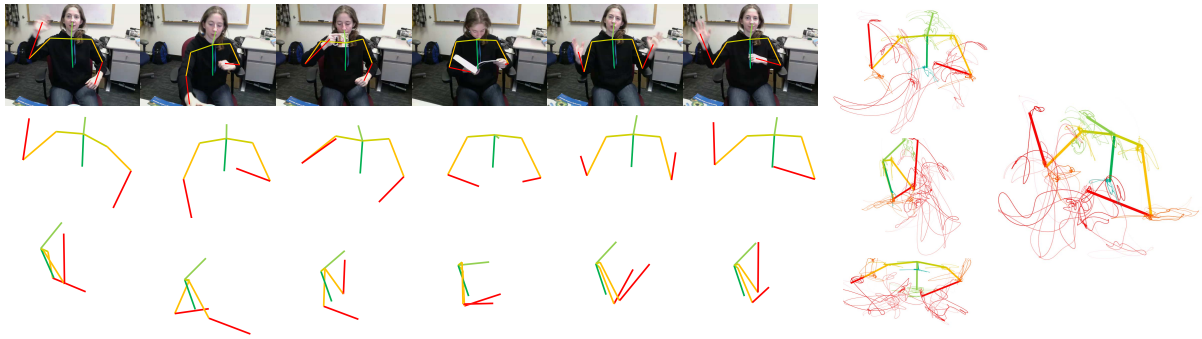
We also apply our method to data captured from a moving camera to recover the *playing card motion* and the *yoga motion* as shown in Figure 4.4(c) and Figure 4.4(d), respectively. Both camera trajectories are smooth and well spanned by the trajectory basis vectors. For the reconstruction of the root trajectory, we choose a relatively rigid part of the human body through a sequence and reconstruct them using the structure from motion algorithm. Once relative camera poses are estimated from the rigid part of the human body, we estimate the similarity transform between the relative camera poses and the original camera poses estimated by using 3D static structure. Head and torso are used as the root for playing card motion and yoga motion, respectively.

4.5 Summary

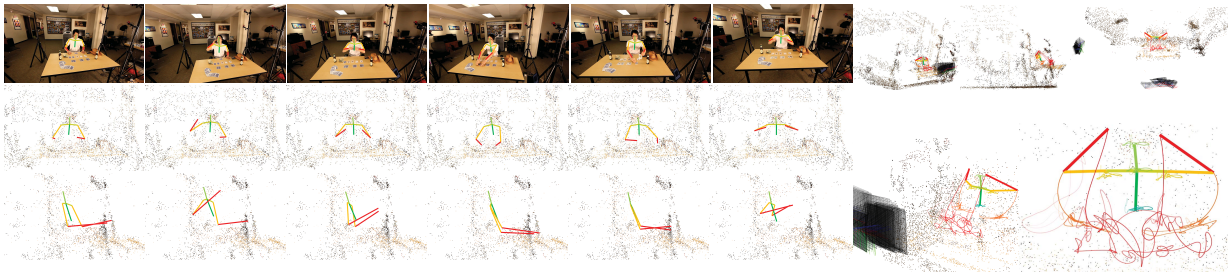
In this chapter, we study an articulated trajectory which remains at a constant distance with respect to the parent trajectory. The relative trajectory is a trajectory on a sphere and there are 2^F trajectories that meet the spatial constraint and image measurements. Among those trajectories, we look for the best trajectory spanned by the trajectory basis vectors and we identify that this is equivalent to solving a binary quadratic programming problem. The relative trajectory obtained by the binary quadratic program is parameterized by compact trajectory basis vectors in the spherical coordinate system, which satisfies spatial and temporal constraints, simultaneously. We optimize the trajectory by minimizing reprojection error. Reconstruction of the articulated trajectory is fundamentally limited by the motion induced by the camera and the parent trajectory. We propose a measure of reconstructability of an articulated trajectory, which characterizes the reconstruction accuracy. Our results show that we are able to reconstruct highly articulated human motions from a stationary camera and a moving camera.



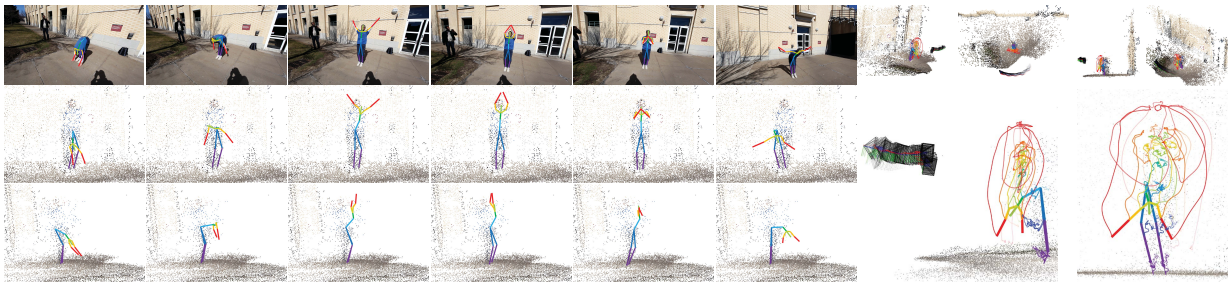
(a) Juggling motion from a stationary camera



(b) Motion in front of a webcam from a stationary camera



(c) Playing motion from a moving camera



(d) Yoga motion from a moving camera

Figure 4.4: (a) Juggling motion, (b) motion in front of the webcam from a stationary camera, (c) playing card motion, and (d) yoga motion from a moving camera. Image measurements are superimposed on images in the top row and 3D reconstruction of the motion corresponding to the images are shown from different views in the second and third rows.

Chapter 5

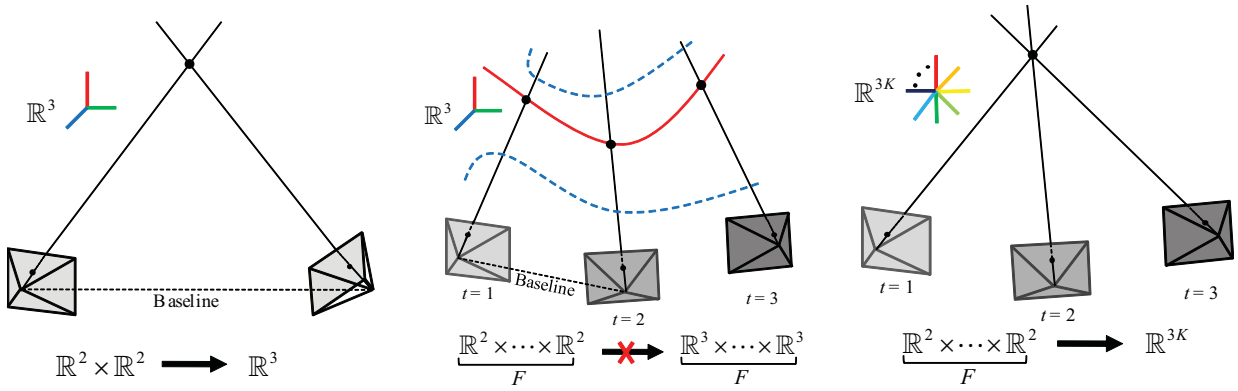
3D Reconstruction of General Scene Motion

It is impossible to reconstruct a 3D scene from a single image without making prior assumptions about scene structure. Binocular stereoscopy is a solution used by both biological and artificial systems to localize the position of a point in 3D via correspondences in two views. Classic triangulation used in stereo reconstruction is geometrically well-posed, as shown in Figure 5.1(a). The rays connecting each image location to its corresponding camera center intersect at the true 3D location of the point—this process is called triangulation, as the two rays form a triangle with the baseline that connects the two camera centers. The triangulation constraint does not apply when the point moves between image captures, as shown in Figure 5.1(b). This case abounds as most artificial vision systems are monocular and most real scenes contain moving elements.

3D reconstruction of a trajectory is directly analogous to monocular image reconstruction. Just as it is impossible to reconstruct a 3D point from a single image without making assumptions about scene structure, it is impossible to reconstruct a moving point without making assumptions about the way it moves. In this chapter, we present an algorithm to reconstruct a moving point from a series of 2D perspective projections given the camera poses. We represent the 3D trajectory using a linear combination of compact trajectory basis vectors [2, 4, 130] and demonstrate that, under this model, we can recover 3D point motion, linearly. We generalize the problem of 3D point triangulation, which is a mapping from $\mathbb{R}^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}^3$, to 3D trajectory reconstruction, as a mapping $\mathbb{R}^2 \times \cdots \times \mathbb{R}^2 \rightarrow \mathbb{R}^{3K}$. $3K$ is the number of trajectory basis vectors required to represent the 3D point trajectory[1] as shown in Figure 5.1(c).

Dynamic 3D reconstruction using shape or trajectory basis vectors requires three types of variables to be estimated [22]: camera motion, model description (often represented as shape or trajectory basis vectors), and model parameters (often represented as basis coefficients). Simultaneously estimating these three types of parameters results in a trivariate optimization, and constitutes the problem definition of nonrigid structure from motion (NRSfM). The optimization suffers from suboptimality, in general, due to the non-convex objective function, and is sensitive to noise and missing data. Akhter et al. [2, 4] reduced the complexity of the trilinear relationship by exploiting the fact that a compact set of trajectory basis vectors can be object independent

[1] Related observations have been made in Shashua and Avidan [127] and Hartley and Vidal [55].



(a) Static point (point in $\mathbb{R}^3$) (b) Moving point (trajectory in $\mathbb{R}^3$) (c) Moving point (point in $\mathbb{R}^{3K}$)

Figure 5.1: (a) A 3D point can be triangulated from two or more views; (b) 3D trajectory reconstruction is impossible without any constraint on the trajectory because any trajectory (dotted trajectories) passing through the optical rays can be a solution; (c) We represent a 3D trajectory with a linear combination of compact trajectory basis vectors, which is a point in $\mathbb{R}^{3K}$. This enables us to linearly reconstruct the point trajectory.

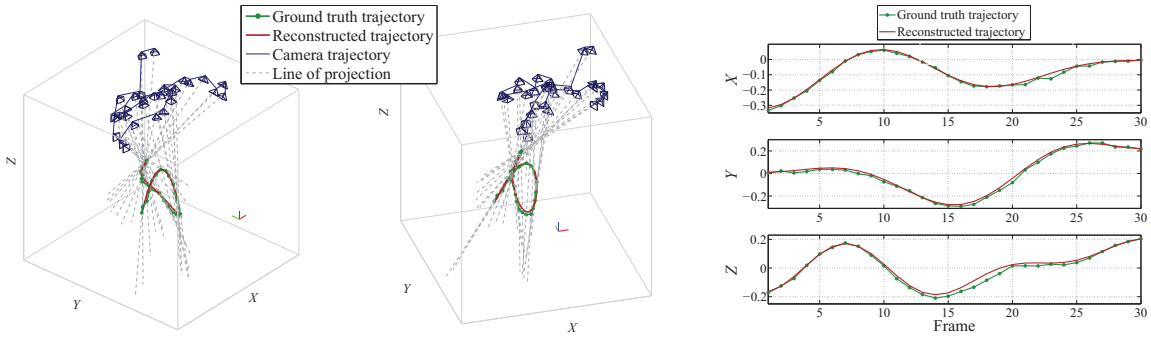
and therefore, can be predefined. This yielded a bilinear optimization over camera motion and coefficient vectors. We note that the predefined trajectory basis vectors are also *coordinate* independent, and as a result, we can use stationary points in the scene to separately estimate camera poses using classic structure from motion[2]. Thus, unlike NRSfM, we take cameras estimated by the stationary areas of the scene as input into our algorithm. The resulting optimization can be solved using linear least squares providing stable, accurate, and efficient estimates in the presence of missing data. We demonstrate 3D reconstruction results of dynamic scenes that include whole body motion, multiple interacting people, and activity with significant locomotive displacement.

The stability of classic triangulation is known to depend on the baseline between camera centers [56]. We study the instability encountered when interference occurs between the point trajectory and camera trajectory, and characterize the cases when trajectory reconstruction is ambiguous. In particular, we define a criterion called *reconstructability*, a measure of reconstruction accuracy defined by the point trajectory, camera trajectory, and basis vectors. We show that when reconstructability approaches infinity, the obtained solution from least squares approaches the ground truth solution.

5.1 Trajectory Reconstruction

In this section, we present an algorithm to reconstruct the 3D trajectory of a moving point from 2D perspective projections given the 3D camera poses and time of capture of the cameras. We represent each trajectory using a linear combination of compact trajectory basis vectors and solve for the trajectory coefficient vector via linear least squares. The number of trajectory basis vectors is automatically chosen by a cross validation scheme and the estimated trajectory is refined by minimizing the geometric error.

[2]A similar approach has been used in Del Bue et al. [34] and Bartoli et al. [13].



(a) Trajectory reconstruction from Equation (5.7)

(b) Comparison with the ground truth

Figure 5.2: We reconstruct a trajectory using linear least squares. (a) The reconstructed trajectory is illustrated in two views. The trajectory which is represented by a linear combination of trajectory basis vectors passes through all lines of projections. The blue pyramid structures are camera poses. (b) We project the ground truth trajectory and the reconstructed trajectory into the X, Y, and Z axis to show accuracy of trajectory reconstruction. Trajectory reconstruction via Equation (5.7) produces an accurate solution.

5.1.1 Linear Trajectory Reconstruction

For a given i^{th} camera projection matrix, $\mathbf{P}_i \in \mathbb{R}^{3\times 4}$, let a point in 3D, $\mathbf{X}_i = \begin{bmatrix} X_i & Y_i & Z_i \end{bmatrix}^\mathsf{T}$, be imaged as $\mathbf{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^\mathsf{T}$. The index i represents the i^{th} time sample. This projection is defined up to scale,

$$\begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix} \simeq \mathbf{P}_i \begin{bmatrix} \mathbf{X}_i \\ 1 \end{bmatrix}, \text{ or } \begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}_\times \mathbf{P}_i \begin{bmatrix} \mathbf{X}_i \\ 1 \end{bmatrix} = \mathbf{0}, \tag{5.1}$$

where $[\cdot]_\times$ is the skew symmetric representation of the cross product [56]. This can be rewritten as an inhomogeneous equation,

$$\begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}_\times \mathbf{P}_{i,1:3}\mathbf{X}_i \;\; = \;\; -\begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}_\times \mathbf{P}_{i,4} \;\; ,$$

where $\mathbf{P}_{i,1:3}$ and $\mathbf{P}_{i,4}$ are the matrices made of the first three columns and the last column of $\mathbf{P}_i$, respectively, or simply as $\mathbf{Q}_i\mathbf{X}_i = \mathbf{q}_i$, where,

$$\mathbf{Q}_i = \left(\begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}_\times \mathbf{P}_{i,1:3} \right)_{1:2}, \; \mathbf{q}_i = -\left(\begin{bmatrix} \mathbf{x}_i \\ 1 \end{bmatrix}_\times \mathbf{P}_{i,4} \right)_{1:2},$$

and $(\cdot)_{1:2}$ is the matrix made of first two rows from $(\cdot)$. By taking into account all time instances, the 3D point trajectory, $\mathbf{X}$, can be written as,

$$\begin{bmatrix} \mathbf{Q}_1 & & \\ & \ddots & \\ & & \mathbf{Q}_F \end{bmatrix} \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_F \end{bmatrix} = \begin{bmatrix} \mathbf{q}_1 \\ \vdots \\ \mathbf{q}_F \end{bmatrix}, \text{ or } \mathbf{QX} = \mathbf{q}, \tag{5.2}$$

where F is the number of time samples in the trajectory. Since Equation (5.2) is an underconstrained system (i.e., $\mathbf{Q} \in \mathbb{R}^{2F \times 3F}$), there are an infinite number of solutions for a given set of measurements (2D projections). We constrain the solution space in which $\mathbf{X}$ lies by approximating the point trajectory using a linear combination of compact trajectory basis vectors,

$$\begin{aligned} \mathbf{X} &= \begin{bmatrix} \mathbf{X}_1^\mathsf{T} & \cdots & \mathbf{X}_F^\mathsf{T} \end{bmatrix}^\mathsf{T} \\ &\approx \boldsymbol{\Theta}_1 \beta_1 + \ldots + \boldsymbol{\Theta}_{3K} \beta_{3K} \\ &= \boldsymbol{\Theta}\boldsymbol{\beta}, \end{aligned} \tag{5.3}$$

where $\boldsymbol{\Theta}_j \in \mathbb{R}^{3F}$ is a trajectory basis vector, $\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{\Theta}_1 & \ldots & \boldsymbol{\Theta}_{3K} \end{bmatrix} \in \mathbb{R}^{3F \times 3K}$ is the trajectory basis matrix, $\boldsymbol{\beta} = \begin{bmatrix} \beta_1 & \ldots & \beta_{3K} \end{bmatrix}^\mathsf{T} \in \mathbb{R}^{3K}$ is a trajectory coefficient vector, and K is the number of the trajectory basis vectors per coordinate.

From Equation (5.2) and (5.3), we can derive the following a system of equations,

$$\mathbf{Q}\boldsymbol{\Theta}\boldsymbol{\beta} = \mathbf{q}. \tag{5.4}$$

To reconstruct moving points in 3D, we have to solve the following trilinear system [22],

$$\underset{\{\mathbf{P}_i\}_{i=1,\cdots,F}, \boldsymbol{\Theta}, \boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{Q}\boldsymbol{\Theta}\boldsymbol{\beta} - \mathbf{q}\|^2, \tag{5.5}$$

given 2D projections, $\{\mathbf{x}_i\}_{i=1,\cdots,F}$. Akhter et al. [2] identified that the trajectory basis vectors could be *object* independent. This allowed them to use predefined trajectory basis vectors such as the DCT (Discrete Cosine Transform) and to remove $\boldsymbol{\Theta}$ from the trilinear optimization. This reduced the optimization to a bilinear system,

$$\underset{\{\mathbf{P}_i\}_{i=1,\cdots,F}, \boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{Q}\boldsymbol{\Theta}_{\mathrm{DCT}}\boldsymbol{\beta} - \mathbf{q}\|^2, \tag{5.6}$$

where $\boldsymbol{\Theta}_{DCT}$ is the predefined DCT trajectory basis vectors.

We note that these trajectory basis vectors are also *coordinate* independent, i.e., the trajectory basis vectors can compactly represent a trajectory equally well in any arbitrary orthogonal world coordinate system by the following Theorem 1.

Theorem 1. *The spectral distribution of a 3D trajectory basis is invariant to 3D similarity transforms.*

See Section B.1 for a proof.

From Theorem 1, we can estimate the camera motion, $\mathbf{P}_{\mathrm{SfM}}$, independently, using structure from motion on the stationary points in a scene [56] as discussed in Section 5.3.2. This further reduces the bilinear system to a linear system as follows,

$$\underset{\boldsymbol{\beta}}{\operatorname{argmin}} \|\mathbf{Q}_{\mathrm{SfM}}\boldsymbol{\Theta}_{\mathrm{DCT}}\boldsymbol{\beta} - \mathbf{q}_{\mathrm{SfM}}\|^2. \tag{5.7}$$

Solving Equation (5.7) for the trajectory coefficient vector, $\boldsymbol{\beta}$, is a linear least squares system if $2F \geq 3K$, which provides an efficient, numerically stable, and globally optimal solution. Figure 5.2 shows 3D trajectory reconstruction via Equation (5.7) in the presence of measurement

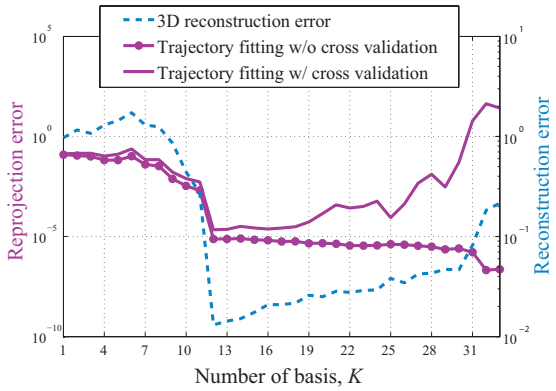


Figure 5.3: We select the number of the DCT basis vectors using a cross validation scheme. As the number of the basis vectors, K, increases, reprojection error decreases in general because the higher K can express the detail of the point motion. The purple line with markers shows reprojection error as K increases (reprojection error decreases). The purple line without markers shows reprojection error measured by our cross validation scheme. When $K = 12$, reprojection error is minimized and the most consistent trajectory through all image measurements is achieved. This also minimizes 3D reconstruction error. Note that the graph has two-sided Y axes, where the left and right Y axes represent reprojection error and 3D reconstruction error in log scale, respectively.

noise. Figure 5.2(a) illustrates the camera trajectory and point trajectory with the lines of projections from two perspectives. The reconstructed trajectory is a trajectory that passes through all lines of projections and that is represented by a linear combination of the trajectory basis vectors. Figure 5.2(b) shows how the reconstructed trajectory and ground truth trajectory are similar.

If there are missing data by self-occlusion or measurement noise, corresponding rows in $\mathbf{Q}$ and $\mathbf{q}$ may be dropped in Equation (5.7). As long as the resulting $\mathbf{Q\Theta}$ matrix satisfies the least squares criterion, i.e., $2\widehat{F} > 3K$ where $\widehat{F}$ is the remaining number of measurements, the estimation of $\boldsymbol{\beta}$ is robust. This allows us to handle the problem of missing data.

5.1.2 Selection of the Number of Basis Vectors

Our approach requires the selection of the number of basis vectors, K. In Akhter et al. [2, 4] and Park et al. [104], the number of the DCT basis vectors was manually tuned and all trajectories were reconstructed with the same number of the basis vectors. This is undesirable because different points may undergo different degrees of motion. The number of the basis vectors controls the complexity of the trajectory motion. For example, a point that undergoes complex motion such as the hands in the dance scene shown in Figure 11, requires higher K, i.e., high frequency DCT trajectory basis vectors are needed to represent and reconstruct the complex motion; a point that undergoes simple motion such as the left leg can be represented by more aggressive truncation, retaining only low frequency DCT trajectory basis vectors. If K is too high, the algorithm overfits measurement noise, and conversely, if it is too low, the reconstructed trajectory cannot express the detail of the point motion. In this section, we present

an approach to automatically select K_i for the i^{th} trajectory rather than manually setting a global value of K. Note that Bartoli et al. [13] also presented a method to select K for shape basis vectors via coarse-to-fine reconstruction while our method can determine it per point.

To select the number of basis vectors automatically and individually, we use an N-fold cross validation scheme to check the consistency of the reconstructed trajectory. The 2D trajectory is divided into N sets such that each set contains F/N samples which are uniformly distributed in time across the 2D trajectory. When the j^{th} set, $\mathcal{S}_j$, is considered, the reprojection error, e_j, is evaluated from a 3D trajectory reconstructed from the rest of the $N-1$ sets for a given K_i. This is iterated until all N sets are tested. When K_i is too high, the trajectory overfits measurement noise, which results in high reprojection error. When K_i is too low, the reprojection error is also high because of limited expressiveness of the basis vectors. We choose the number of the basis vectors for the i^{th} trajectory, which minimizes cross-validated reprojection error, i.e.,

$$
\begin{aligned}
K_i^* &= \underset{K_i}{\operatorname{argmin}} \sum_{j=1}^{N} e_j(K_i), \qquad (5.8)\\
\text{where } K_i &= 1, 2, \cdots, \lfloor 2F/3 \rfloor,\\
e_j(K_i) &= \sum_{s \in \mathcal{S}_j} \left(\frac{\mathbf{P}_s^1 \mathcal{X}_s^{K_i}}{\mathbf{P}_s^3 \mathcal{X}_s^{K_i}} - x_s \right)^2 + \left(\frac{\mathbf{P}_s^2 \mathcal{X}_s^{K_i}}{\mathbf{P}_s^3 \mathcal{X}_s^{K_i}} - y_s \right)^2,\\
\mathcal{X}_s^{K_i} &= \begin{bmatrix} \boldsymbol{\Theta}(s)^{K_i} \boldsymbol{\beta}^{K_i} \\ 1 \end{bmatrix},
\end{aligned}
$$

where $\lfloor \cdot \rfloor$ is the floor operator (the largest integer not greater than $\cdot$). $\boldsymbol{\Theta}(s)^{K_i} \in \mathbb{R}^{3 \times 3K_i}$ is the trajectory basis vectors evaluated at the s^{th} time instant with the $3K_i$ trajectory basis vectors and $\mathbf{P}^l$ is the l^{th} row of the matrix $\mathbf{P}$. x_s and y_s are a 2D measurement at the s^{th} time instant. In Figure 5.3, the purple line with markers is the reprojection error as K_i increases. The higher K_i, the lower the reprojection error because the details of the trajectory can be expressed. However, high K_i may overfit to the measurement noise of the trajectory. From our cross validation scheme, we are able to automatically select the K_i that is the most expressive but the least overfitted. The purple line without markers shows reprojection error and $K_i = 12$ produces the most consistent trajectory for all image measurements (minimum reprojection error) in the presence of measurement noise. This K_i also minimizes 3D reconstruction error.

5.1.3 3D Trajectory Refinement

Trajectory reconstruction from Equation (5.7) minimizes the algebraic error [56]. The solution, $\boldsymbol{\beta}$, is not necessarily the maximum likelihood solution under Gaussian measurement noise. We refine the linearly reconstructed trajectory by minimizing the reprojection error, i.e.,

$$
\begin{aligned}
&\underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_{i=1}^{F} \left(\frac{\mathbf{P}_i^1 \mathcal{X}_i}{\mathbf{P}_i^3 \mathcal{X}_i} - x_i \right)^2 + \left(\frac{\mathbf{P}_i^2 \mathcal{X}_i}{\mathbf{P}_i^3 \mathcal{X}_i} - y_i \right)^2, \qquad (5.9)\\
&\text{where } \mathcal{X}_i = \begin{bmatrix} \boldsymbol{\Theta}(i) \boldsymbol{\beta} \\ 1 \end{bmatrix},
\end{aligned}
$$

$\boldsymbol{\Theta}(i) \in \mathbb{R}^{3 \times 3K}$ is the trajectory basis vectors evaluated at thei^{th} time instant.

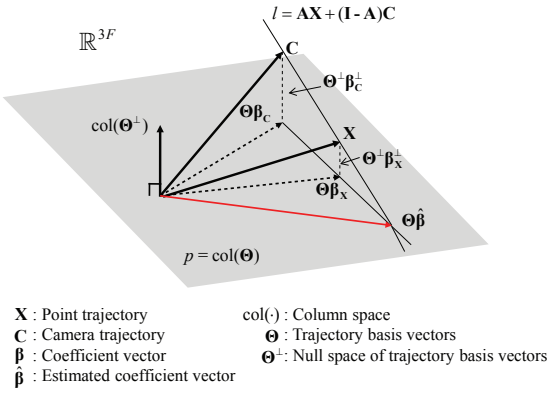


Figure 5.4: Geometric illustration of the least squares solution of Equation (5.7). ours trajectory $\mathbf{\Theta}\widehat{\boldsymbol{\beta}}$ is placed on the intersection between l hyperplane containing the camera trajectory space and the point trajectory, and the p space spanned by the trajectory basis vectors, $\mathrm{col}(\mathbf{\Theta})$.

5.2 Geometric Analysis

Empirically, we observe the point trajectory reconstruction approaches the ground truth if the camera motion is fast or random when the DCT trajectory basis vectors are used. Conversely, if the camera moves slowly or smoothly, the solution tends to deviate from the ground truth. In this section, we analyze the stability of trajectory reconstruction using Equation (5.7) by considering the geometric relationship between the point trajectory, camera trajectory, and trajectory basis vectors. We categorize trajectory reconstruction as a solvable or unsolvable system. Trajectory reconstruction is possible only when our least squares system is solvable. More importantly, a solvable system does not guarantee an accurate estimate of the trajectory parameters. We define a measure of reconstruction accuracy, *reconstructability*, for solvable systems. Reconstructability enables us to precisely characterize when accurate reconstruction of a 3D trajectory is possible.[3]

5.2.1 Geometry of Camera Trajectory and Point Trajectory

Let $\mathbf{X}$ and $\widehat{\mathbf{X}}$ be a ground truth trajectory and an estimated point trajectory, respectively. The camera matrix can, without loss of generality, be normalized by intrinsic and rotation matrices, $\mathbf{K}$ and $\mathbf{R}$, respectively, (as all camera matrices are known), i.e., $\mathbf{R}_i^{\mathsf{T}}\mathbf{K}_i^{-1}\mathbf{P}_i = \left[\ \mathbf{I}_3 \ \middle|\ -\mathbf{C}_i\ \right]$, where $\mathbf{P}_i = \mathbf{K}_i\mathbf{R}_i\left[\ \mathbf{I}_3 \ \middle|\ -\mathbf{C}_i\ \right]$, $\mathbf{C}_i$ is the camera center, and $\mathbf{I}_3$ is a 3×3 identity matrix. This follows from the fact that triangulation and 3D trajectory reconstruction are both geometrically unaffected by the rotation of the camera about its center. All $\mathbf{P}_i$ subsequently used in this analysis are normalized camera matrices, i.e., $\mathbf{P}_i = \left[\ \mathbf{I}_3 \ \middle|\ -\mathbf{C}_i\ \right]$. Then, a measurement is a projection of $\mathbf{X}_i$ onto the image plane from Equation (5.1). Since Equation (5.1) is defined up to scale, the

[3]Ambiguity analyses have been investigated by [162], [149], [55], and [3]. However, these analyses consider the ambiguity with the use of a shape basis representation, which utilizes the correlation across *multiple* points. In this section, we consider the case of the reconstruction of a *single* 3D point trajectory.

measurement, $\mathbf{x}_i$, can be replaced as follows,

$$\left[\mathbf{P}_i \begin{bmatrix} \mathbf{X}_i \\ 1 \end{bmatrix}\right]_\times \mathbf{P}_i \begin{bmatrix} \widehat{\mathbf{X}}_i \\ 1 \end{bmatrix} = 0. \tag{5.10}$$

Plugging in $\mathbf{P}_i = \begin{bmatrix} \mathbf{I}_3 \mid -\mathbf{C}_i \end{bmatrix}$ results in,

$$[\mathbf{X}_i - \mathbf{C}_i]_\times \left(\widehat{\mathbf{X}}_i - \mathbf{C}_i\right) = 0, \tag{5.11}$$

or equivalently,

$$[\mathbf{X}_i - \mathbf{C}_i]_\times \widehat{\mathbf{X}}_i = [\mathbf{X}_i]_\times \mathbf{C}_i. \tag{5.12}$$

To satisfy Equation (5.12), $\widehat{\mathbf{X}}_i$ has to lie in the space spanned by $\mathbf{X}_i$ and $\mathbf{C}_i$, or $\widehat{\mathbf{X}}_i = a_1\mathbf{X}_i + a_2\mathbf{C}_i$. It can be easily verified that $a_2 = 1 - a_1$ by substituting in Equation (5.12). Thus, the solution of Equation (5.12) is,

$$\widehat{\mathbf{X}}_i = a_i\mathbf{X}_i + (1 - a_i)\mathbf{C}_i, \tag{5.13}$$

where a_i is an arbitrary scalar. Geometrically, Equation (5.13) is a constraint for the perspective camera model that enforces the solution to lie on the ray joining the camera center and the point in 3D. By generalizing the i^{th} point to a point trajectory, Equation (5.13) becomes,

$$\widehat{\mathbf{X}} = \mathbf{A}\mathbf{X} + (\mathbf{I} - \mathbf{A})\mathbf{C}, \tag{5.14}$$

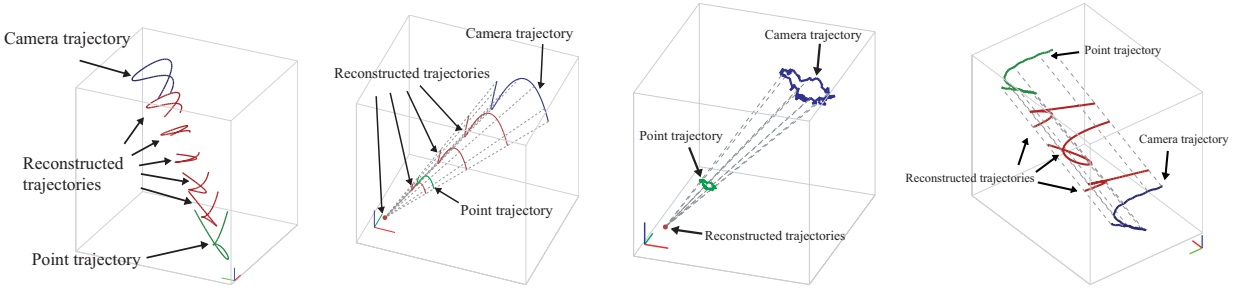
where $\mathbf{A} = \mathbf{D} \otimes \mathbf{I}_3$[4]. From Equation (5.3), Equation (5.14) can be rewritten as $\boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} \approx \mathbf{A}\mathbf{X} + (\mathbf{I} - \mathbf{A})\mathbf{C}$.

Figure 5.4 illustrates the geometry of the solution of Equation (5.7). Let the subspace, p, be the space spanned by the trajectory basis vectors, $\mathrm{col}(\boldsymbol{\Theta})$. The solution $\boldsymbol{\Theta}\widehat{\boldsymbol{\beta}}$, has to simultaneously lie in l and $\mathrm{col}(\boldsymbol{\Theta})$ where l is a hyperplane that contains the camera trajectory and the point trajectory. Thus, $\boldsymbol{\Theta}\widehat{\boldsymbol{\beta}}$ is the intersection of the hyperplane l and the subspace p. Note that the line and the plane are a conceptual 3D vector space representation for the $3F$-dimensional space. The camera center trajectory, $\mathbf{C} = \begin{bmatrix} \mathbf{C}_1^\mathsf{T} & \dots & \mathbf{C}_F^\mathsf{T} \end{bmatrix}^\mathsf{T}$, and the point trajectory, $\mathbf{X}$, are projected onto $\mathrm{col}(\boldsymbol{\Theta})$ as $\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{C}$ and $\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{X}$, respectively.

5.2.2 Characterization of Trajectory Reconstruction

Recovering $\boldsymbol{\beta}$ depends on the camera trajectory as shown in Figure 5.4. We study the degeneracy of the solution of Equation (5.7) to characterize the cases when trajectory reconstruction is impossible. The least squares system of Equation (5.7) is solvable if $\mathrm{rank}(\mathbf{Q}\boldsymbol{\Theta}) = 3K$ (i.e., it has full column rank).

[4] $\otimes$ is the Kronecker product and $\mathbf{D}$ is a diagonal matrix which consists of $\{a_1, \cdots, a_F\}$, the scalar for each point along the trajectory.



(a) $\mathbf{C}, \mathbf{X} \in \text{col}(\mathbf{\Theta})$ (b) $\mathbf{C}, \mathbf{X} \in \text{col}(\mathbf{\Theta})$ and $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ (c) $\mathbf{C}, \mathbf{X} \notin \text{col}(\mathbf{\Theta})$ and $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ (d) $\mathbf{C}, \mathbf{X} \notin \text{col}(\mathbf{\Theta})$ and $\mathbf{X} = \mathbf{C} + \mathbf{1} \otimes \mathbf{d}$

Figure 5.5: We illustrate unsolvable systems that produce an infinite number of solutions or a trivial solution. (a) Trajectory reconstruction is ambiguous when $\mathbf{C}, \mathbf{X} \in \text{col}(\mathbf{\Theta})$ because there exists $\text{null}(\mathbf{Q\Theta})$, which is an unsolvable system. Plausible reconstructed trajectories that satisfy Equation (5.7) are illustrated. (b) Plausible reconstructed trajectories that satisfy Equation (5.7) when $\mathbf{C}, \mathbf{X} \in \text{col}(\mathbf{\Theta})$ and $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ are shown. (c) When $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ where $c \neq 1$, the solution of the system is always $\mathbf{1} \otimes \mathbf{d}/(1-c)$, which is trivial. (d) When $\mathbf{X} = \mathbf{C} + \mathbf{1} \otimes \mathbf{d}$, the system is unsolvable because $\text{rank}(\mathbf{Q\Theta}) = 2K$.

5.2.2.1 Unsolvable systems

When the system is unsolvable, there is a space of solutions where trajectory estimation is ambiguous. We characterize such an unsolvable system as follows,

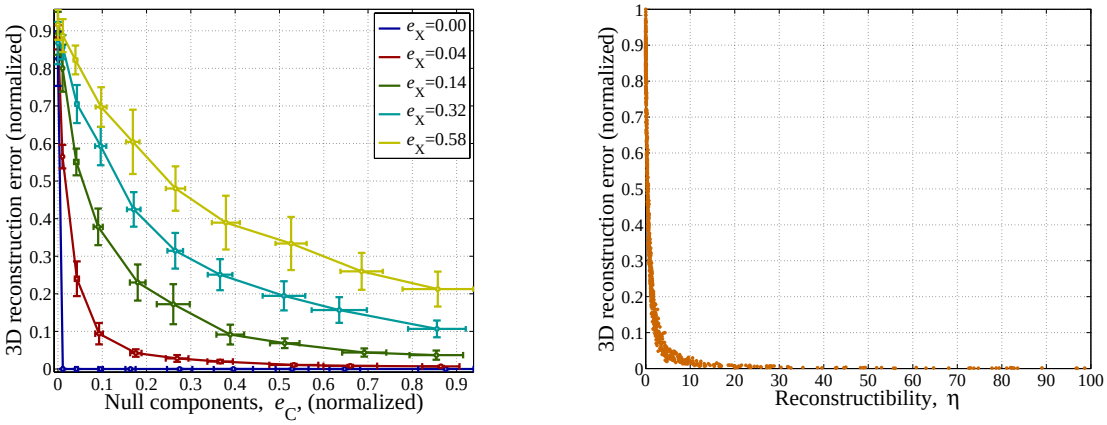
Theorem 2. Trajectory reconstruction via Equation (5.7) is unsolvable if
(*i*) $\mathbf{X}, \mathbf{C} \in \text{col}(\mathbf{\Theta})$, or
(*ii*) $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ where c is a nonzero scalar, $\mathbf{1}$ is an F dimensional vector whose entries are all ones, and $\mathbf{d} \in \mathbb{R}^3$ is an arbitrary vector.

See Section B.2 for a proof.

Figure 5.5 illustrates solutions of unsolvable systems. For Theorem 2.*i*, Figure 5.5(a) shows an ambiguous solution of Equation (5.7) when $\mathbf{X}, \mathbf{C} \in \text{col}(\mathbf{\Theta})$. All reconstructed trajectories lie in one dimensional subspace $\boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}}$. When $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ (i.e., Theorem 2.*ii*), the system is also unsolvable. When $c \neq 1$, the solution is $\alpha\mathbf{C}_i + (1-\alpha)/(1-c)\mathbf{d}$. α can be nonzero only when $\mathbf{C} \in \text{col}(\mathbf{\Theta})$. Figure 5.5(b) illustrates the space of solutions by varying α. When $\mathbf{C} \notin \text{col}(\mathbf{\Theta})$, $\alpha = 0$ and the solution is always $\mathbf{1} \otimes \mathbf{d}/(1-c)$ (i.e., stationary point) which is a trivial solution as shown in Figure 5.5(c). Figure 5.5(d) shows trajectory reconstruction when $c = 1$, which results in $\text{rank}(\mathbf{Q\Theta}) = 2K$. Any trajectory in K dimensional subspace (i.e., $\text{null}(\mathbf{Q\Theta})$) is a solution lying on a surface made by the point trajectory and the camera trajectory, which is shown by gray dotted lines.

5.2.2.2 Solvable systems

Theorem 2 considers an unsolvable system or a system resulting in a trivial solution. For a solvable system, Equation (5.7) can be solved without ambiguity in a least squares sense and there exists a unique solution, $\widehat{\boldsymbol{\beta}}$. However, the solvable system does not guarantee an accurate solu-



(a) Orthogonality (b) Reconstructability

Figure 5.6: (a) As the null component of the camera trajectory, $e_{\mathbf{C}}$, decreases, the solution of Equation (5.7) deviates from the ground truth. (b) Reconstructability, η, provides the degree of interference between the camera trajectory and point trajectory. Reconstructability is inversely proportional to 3D reconstruction error.

tion: how much $\widehat{\boldsymbol{\beta}}$ deviates from $\boldsymbol{\beta}_{\mathbf{X}}$. We observe that the accuracy of trajectory reconstruction depends on the relationship between the camera trajectory, point trajectory, and trajectory basis vectors. Given this observation, we characterize the case when reconstruction is accurate.

Solving the least squares system, $\widehat{\mathbf{X}} = \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}}$, minimizes the residual error by Equation (5.14),

$$\underset{\widehat{\boldsymbol{\beta}},\mathbf{A}}{\operatorname{argmin}} \left\| \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} - \mathbf{A}\mathbf{X} - (\mathbf{I} - \mathbf{A})\,\mathbf{C} \right\|^2. \tag{5.15}$$

Let us decompose the point trajectory and the camera trajectory into the column space of $\boldsymbol{\Theta}$ and that of the null space, $\boldsymbol{\Theta}^{\perp}$ as follows, $\mathbf{X} = \boldsymbol{\Theta}\boldsymbol{\beta}_{\mathbf{X}} + \boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{X}}^{\perp}$, $\mathbf{C} = \boldsymbol{\Theta}\boldsymbol{\beta}_{\mathbf{C}} + \boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{C}}^{\perp}$, where $\boldsymbol{\beta}^{\perp}$ is the coefficient vector for the null space. Let us also define a measure of *reconstructability*, η, of the 3D point trajectory reconstruction,

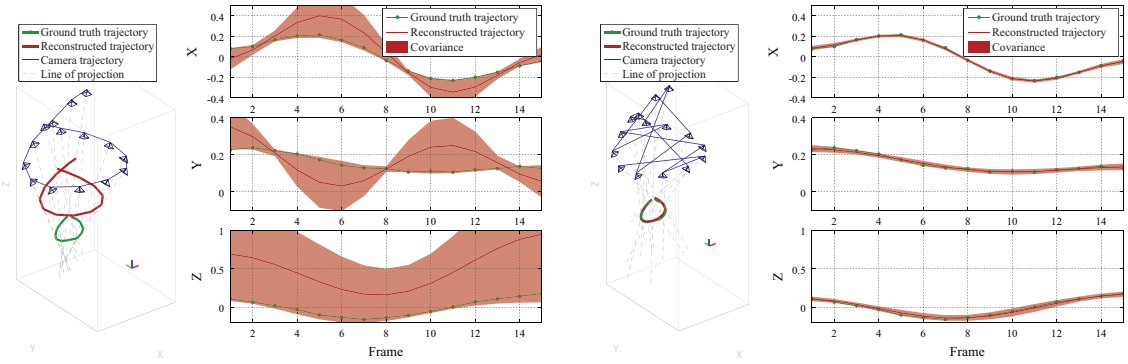
$$\eta\left(\boldsymbol{\Theta}\right) \;=\; \frac{\left\|\boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{C}}^{\perp}\right\|}{\left\|\boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{X}}^{\perp}\right\|} = \frac{\text{How poorly } \boldsymbol{\Theta} \text{ describes } \mathbf{C}}{\text{How poorly } \boldsymbol{\Theta} \text{ describes } \mathbf{X}}. \tag{5.16}$$

Reconstructability enables us to define the accuracy of the trajectory reconstruction by the following Theorem.

Theorem 3. $\lim_{\eta\to\infty} \widehat{\boldsymbol{\beta}} = \boldsymbol{\beta}_{\mathbf{X}}$.

See Section B.3 for a proof.

Figure 5.6(a) shows how reconstructability is related to the accuracy of the 3D reconstruction error. In each reconstruction, the residual error (null components) of the point trajectory, $e_{\mathbf{X}} = \left\|\boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{X}}^{\perp}\right\|$, and the camera trajectory, $e_{\mathbf{C}} = \left\|\boldsymbol{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{C}}^{\perp}\right\|$, are measured. Increasing $e_{\mathbf{C}}$ for a given point trajectory enhances the accuracy of the 3D reconstruction, while increasing $e_{\mathbf{X}}$ lowers accuracy. Even though we cannot directly measure the reconstructability (we never know



(a) Low reconstructability (b) High reconstructability

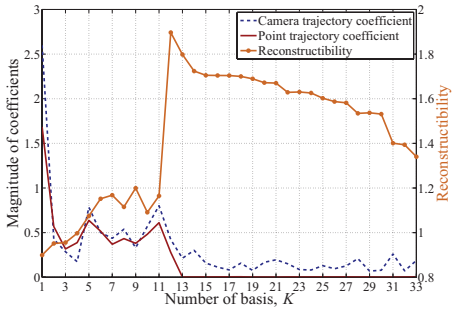
Figure 5.7: Stability or uncertainty of trajectory reconstruction depends on reconstructability. We reconstruct the same point trajectory with the same camera location but different ordering. (a) The order of captures forms a smooth camera trajectory (left column), which results in low reconstructability ($\eta = 0.77$). The reconstructed point trajectory is inaccurate and the covariance of the trajectory is large (right column). (b) We shuffle the order of captures that produces a random camera trajectory (see the camera trajectory on left column). This results in high reconstructability ($\eta = 54.78$). The reconstructed point trajectory is accurate and the covariance of the trajectory is small (right column).

the true point trajectory in a real example), it is useful to demonstrate the direct relation with 3D reconstruction accuracy. Figure 5.6(b) illustrates that the reconstructability is inversely proportional to the 3D reconstruction error.

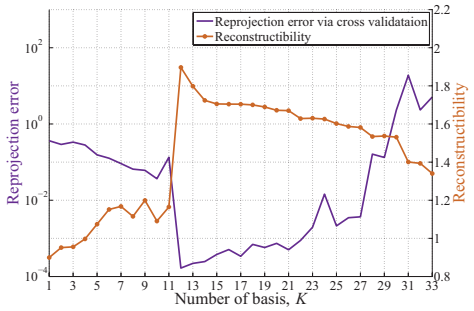
5.2.3 Discussion on Reconstructability

Reconstructability provides key insights into the fundamental relationship between the camera trajectory, point trajectory, and trajectory basis vectors for trajectory reconstruction in 3D and it explains why certain types of the camera motion produce high 3D reconstruction error. It is analogous to the baseline which connects two camera centers in classic triangulation as shown Figure 5.1(a). Stability or uncertainty of point reconstruction is dependent on the baseline between camera centers. If the baseline is wide, the uncertainty of the 3D reconstructed point is small and the stability of estimation is high. If the baseline is narrow, reconstructing the point is highly unstable (i.e., high uncertainty along the rays of projections) in the presence of Gaussian noise. Thus, the baseline provides a key insight of the stability of the reconstruction. Reconstructability is the corresponding concept to the baseline for nonrigid structure from motion in trajectory space.

Figure 5.7 illustrates how reconstructability relates to the reconstruction accuracy and the covariance of the reconstructed trajectory when the DCT trajectory basis vectors are used. We generate a smooth camera trajectory and point trajectory as shown in the left column of Figure 5.7(a). The smooth camera trajectory forms low reconstructability ($\eta = 0.77$). Trajectory reconstruction is inaccurate and the covariance of the reconstructed trajectory is large (the right column of Figure 5.7(a)). In Figure 5.7(b), we shuffle the order of capture, which produces a



(a) Reconstructability and trajectory spectrum



(b) Reconstructability and reprojection error

Figure 5.8: Reconstructability and the cross validation scheme are highly related; when reconstructability is maximized, the reprojection error used for the cross validation is minimized. (a) The magnitude of coefficient vectors of the point and camera trajectories is plotted and reconstructability when K basis vectors are used is overlaid. Reconstructability is maximized when the magnitude of coefficients of the point trajectory is diminished ($K = 12$). (b) Reprojection error for the cross validation is minimized where reconstructability is maximized ($K^* = 12$) because that number of basis vectors is the most expressible and the least overfitted.

random camera trajectory while the camera poses remain the in Figure 5.7(a). Note that the locations of the camera centers are the same but the camera trajectory is random in the left column of Figure 5.7(b). The random camera trajectory results in high reconstructability ($\eta = 54.78$). This camera trajectory reconstructs the accurate point trajectory with low covariance as shown in the right column of Figure 5.7(b).

In practice, the infinite reconstructability criterion is difficult to satisfy because the actual $\mathbf{X}$ is unknown. To enhance reconstructability we can maximize $e_{\mathbf{C}}$ with constant $e_{\mathbf{X}}$. Thus, the best camera trajectory for a given trajectory basis matrix is the one that lives in the null space, $\mathrm{col}(\mathbf{\Theta}^{\perp})$. This explains our observation about slow and fast camera motion described at the beginning of this section. When the camera motion is slow, the camera trajectory is likely to be represented well by the DCT basis vectors, which results in low reconstructability and vice versa. However, for a given camera trajectory, there is no deterministic way to define trajectory basis vectors because it is coupled with both the camera trajectory and the point trajectory. If one simply finds the orthogonal space to the camera trajectory, in general, it is likely to nullify space that also spans the point trajectory space. Geometrically, simply changing the orientation of p in Figure 5.4 may result in a greater deviation between $\mathbf{\Theta}\boldsymbol{\beta}_{\mathbf{X}}$ and $\mathbf{\Theta}\widehat{\boldsymbol{\beta}}$.

Reconstructability is highly related to the selection of the number of basis vectors via our cross validation scheme described in Section 5.1.2. Given camera motion, reconstructability varies as the number of basis vectors changes as shown in Figure 5.8. Figure 5.8(a) shows the relationship between the magnitude of the coefficient vectors used to reconstruct the point and camera trajectories, and the reconstructability principle. The selected $K^* = 12$ is the minimum number of trajectory basis vectors that also minimizes the 3D reconstruction error. K^* is the automatically selected number of basis vectors via the cross validation scheme. When $K < K^*$, $||\mathbf{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{X}}^{\perp}||$ is not minimized because there are some coefficients at higher than the K frequency.

When $K > K^*$, $||\mathbf{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{X}}^{\perp}||$ is already minimized but $||\mathbf{\Theta}^{\perp}\boldsymbol{\beta}_{\mathbf{C}}^{\perp}||$ is not maximized. When $K = K^*$, reconstructability is maximized and reprojection error that is used for the cross validation is simultaneously minimized as shown in Figure 5.8(b).

5.3 Results

In this section, we evaluate our algorithm quantitatively on motion capture data and qualitatively on real data. In all cases, the trajectory basis vectors are the first K_i DCT basis vectors in order of increasing frequency where K_i is determined by our cross validation scheme. The DCT basis vectors have been shown to provide optimal performance in encoding a signal under the first order Markov process [53] and demonstrated to accurately and compactly model point trajectories [2, 4]. If a 3D trajectory is continuous and smooth, the DCT basis vectors can represent it accurately with relatively few low frequency components. We make the realistic assumption that each point trajectory is continuous and smooth and use the DCT basis as the trajectory basis, $\mathbf{\Theta}$. Also for numerical stability, we normalize 2D measurements of the each trajectory such that the mean of 2D measurements is $\mathbf{0}$ and the average distance from the origin is $\sqrt{2}$ before solving Equation (5.7) [54]. We obtained correspondences of moving points across images, manually. 3D trajectories of moving points are estimated linearly as described in Section 5.1.1. The number of basis vectors is chosen per point using the cross validation method and each linearly estimated trajectory is refined by the nonlinear optimization as described in Section 5.1.2 and in Section 5.1.3, respectively. The results, data, and the code of real data are available on the project webpage[5].

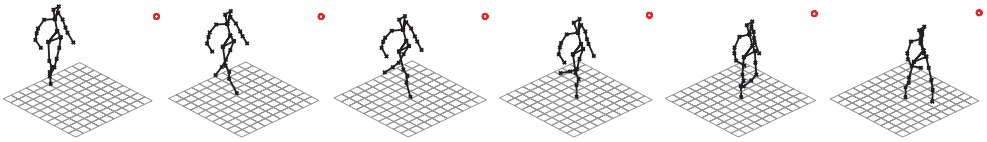
5.3.1 Quantitative Evaluation

To quantitatively evaluate our method, we generate synthetic 2D images from 3D motion capture data and test it from three perspectives: reconstructability, handling missing data, and accuracy. For reconstructability, we compare reconstruction by increasing the null component, $e_{\mathbf{C}}$, of the camera trajectory. For robustness, we test with missing data and lower sampling rates. Finally, for accuracy, we compare our algorithm with a state-of-the-art algorithm (trajectory triangulation) by Kaminski and Teicher [68]. The results show our method outperforms their method.

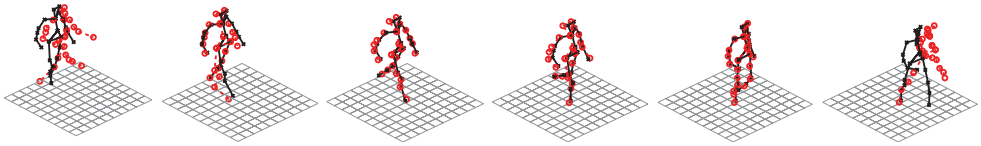
5.3.1.1 Reconstructability

Earlier, we defined the reconstructability of a 3D trajectory as the trade off between the ability of the chosen trajectory basis vectors to accurately reconstruct the point trajectory vs. its ability to reconstruct the camera trajectory. To evaluate this effect empirically we relative generate camera trajectories by varying $e_{\mathbf{C}}$ and measure the 3D reconstruction error. Each trajectory is normalized to have zero mean and unit variance so that errors can be compared across different sequences. Figure 5.9 shows examples (walking sequences) of trajectory reconstructions under various reconstructability. When the reconstructability is zero shown in Figure 5.9(a), the

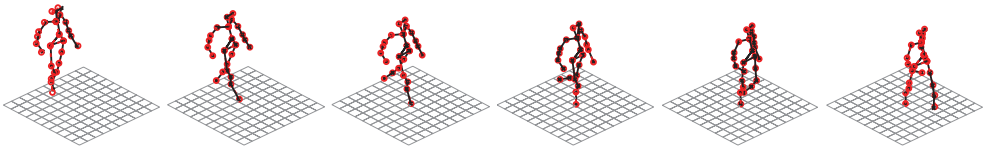
[5] http://www.cs.cmu.edu/~hyunsoop/trajectory_reconstruction.html



(a) Trajectory reconstruction at zero reconstructability



(b) Trajectory reconstruction with low reconstructability



(c) Trajectory reconstruction with high reconstructability

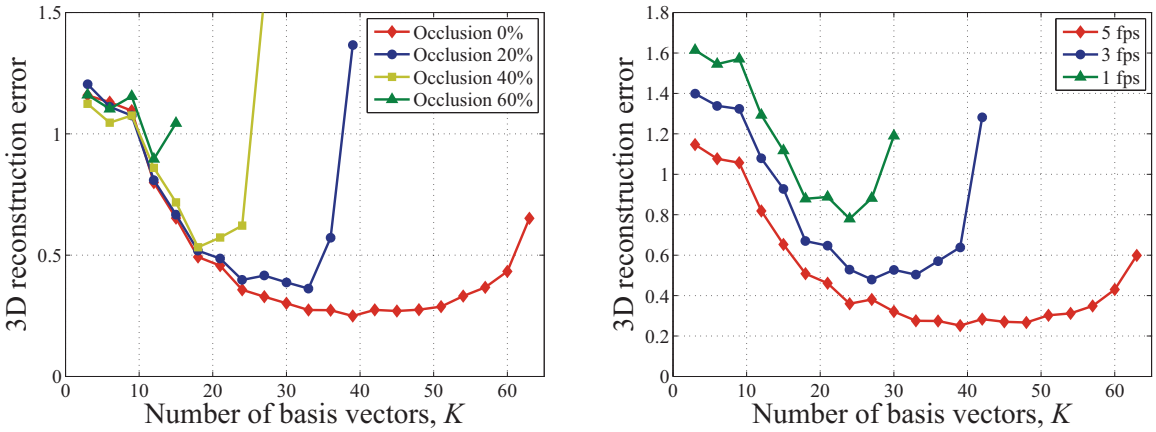
Figure 5.9: Qualitative comparison of trajectory reconstruction from various reconstructability. Black: ground truth, red: reconstructed trajectory. (a) Zero reconstructability, $\eta = 0$. The relative camera trajectory is stationary and the reconstructed trajectory is exactly the same as the camera trajectory. (b) Low reconstructability, $\eta = 0.32$ results in inaccurate reconstruction at the beginning and the end of the sequence. (c) All trajectories are reconstructed accurately under high reconstructability, $\eta = 5.31$.

reconstructed trajectories are exactly the same as the camera motions because the camera trajectory is the intersection of the hyperplane, l, and the space of trajectory basis vectors, $\mathrm{col}(\mathbf{\Theta})$, as shown in Figure 5.4. When reconstructability is low, $\eta = 0.32$, shown in Figure 5.9(b), the reconstruction deviates from the ground truth because there is interference from the camera trajectory. High estimation error can be observed at the beginning and the end of the sequence. If the reconstructability is high, $\eta = 5.31$, reconstruction is very close to the ground truth.

5.3.1.2 Handling Missing Data

We test for the effects of missing data and low frame rate (sparse measurements) with high reconstructability. Missing samples occur in practice due to occlusion, self-occlusion, or measurement failure.

In general, as the number of the basis vectors, K, increases, the 3D reconstruction error decreases because the high frequency components of a point trajectory can be described by the basis vectors. However, when there is occlusion, reconstruction instability occurs due to measurement noise. Figure 5.10(a) shows the reconstruction error as the amount of occlusion varies (0%, 20%, 40%, and 60% of the sequence) for different numbers of the DCT basis vectors, K. A walking motion capture sequence was used and each experiment was repeated 10 times with



(a) Occlusion

(b) Frame rate

Figure 5.10: (a) While a large number of basis vectors results in low 3D reconstruction error in general, reconstruction instability is observed when there is missing data. Reconstruction instability results from overfitting of trajectories. Nevertheless, our algorithm can handle 40% missing data with 19 basis vectors, which results in relatively low 3D reconstruction error. (b) As frame rate increases, visibility of motion also increases, which results in low 3D reconstruction error.

random occlusion. As long as the visibility of a point in a sequence is sufficient to overconstrain Equation (5.7), the solution is robust to moderate occlusion. Figure 5.10(a) shows that our algorithm can handle relatively high number of missing data (40%) with $K = 19$.

Figure 5.10(b) evaluates the robustness to the frequency of input samples, i.e., varying the effective frame rate of the input sequence given camera motion and point motion. Note that since the camera motion and point motion are fixed, relative motion, or reconstructability, is constant. Visibility of moving points is important to avoid poor conditioning of the solution, and intuitively more frequent visibility results in better reconstruction. The results confirm this observation. As was observed in the occlusion experiment, the higher K, the less the reconstruction error but reconstruction instability can be observed when frame rate is low (1 fps).

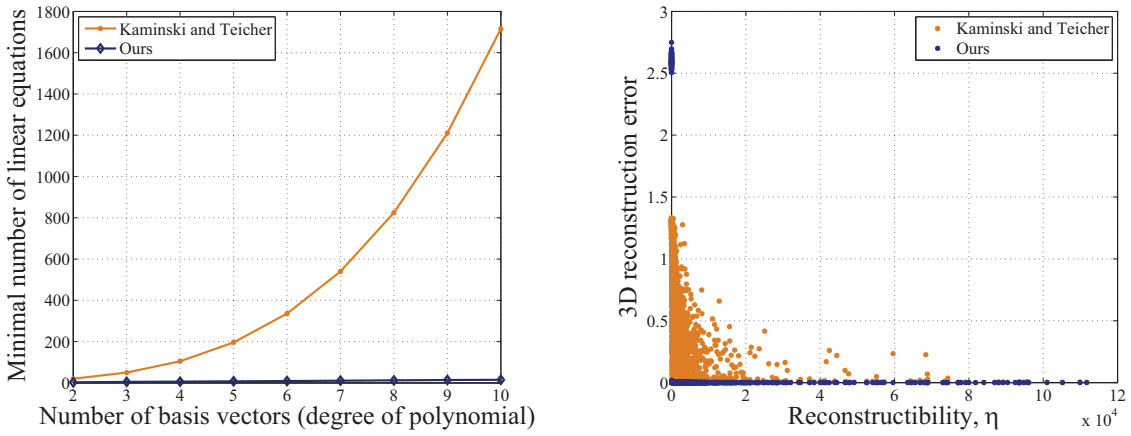
5.3.1.3 Accuracy

We evaluate our algorithm by comparing with a trajectory reconstruction algorithm proposed by Kaminski and Teicher [68][6]. The result of this comparison indicates that their method is computationally prohibitive and less fault tolerant.

Kaminski and Teicher [68] introduced a method to reconstruct a trajectory from 2D projections given camera poses similar to our method. They represented a trajectory (algebraic curve) as a hypersurface in $\mathbb{P}^5$ where all lines of projections intersect, i.e., a homogeneous polynomial vanishes on Plücker coordinates of 3D lines intersecting the trajectory.

Their algorithm is composed of two optimizations: to estimate the Chow polynomial from lines of projections and to find points on a trajectory that satisfy the Chow polynomial. To solve

[6]The method by Avidan and Shashua [9] can only reconstruct a linear or conic trajectory.



(a) Number of linear equations

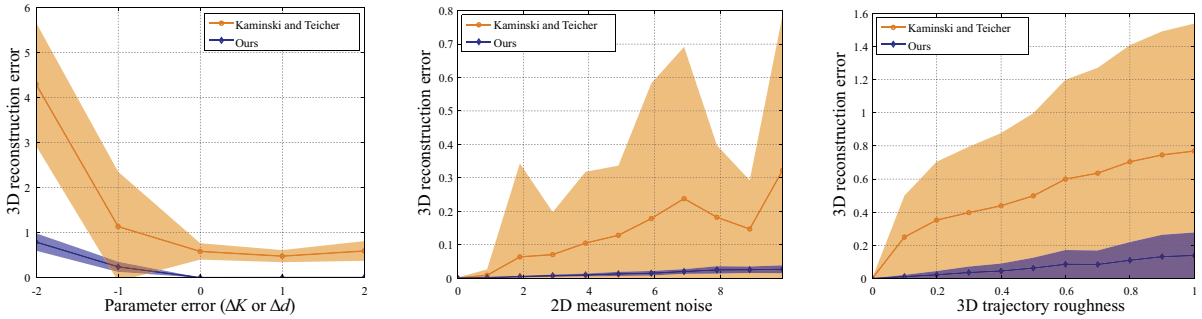
(b) Reconstructability

Figure 5.11: (a) The minimal number of linear equations increases exponentially as the degree of polynomial (degree of motion) increases for the method by Kaminski and Teicher [68] while it increases linearly for our method. This computationally precludes them from reconstructing a trajectory with high complexity. (b) We compare reconstruction accuracy by varying reconstructability. Both methods show an inverse relationship between 3D reconstruction error and reconstructability. Our method achieves smaller errors than their method.

the Chow polynomials from lines of projections, $N_d = \begin{pmatrix} d+5 \\ d \end{pmatrix} - \begin{pmatrix} d+3 \\ d-2 \end{pmatrix} - 1$ measurements have to be made, and each measurement produces one linear equation. Therefore, N_d linear equations have to be solved[7] while our method needs to solve $N_K = \lceil 3K/2 \rceil$ linear equations. d is the degree of the homogeneous polynomial that determines degree of motion (complexity of the trajectory), which is equivalent to the number of trajectory basis vectors, K, for our method. N_d increases exponentially while N_K increases linearly as shown in Figure 5.11(a). This indicates that their method is computationally prohibitive as the degree of motion, d, increases. Inaccurate trajectory reconstruction caused by low reconstructability is also observed from their method as shown in Figure 5.11(b). 3D reconstruction error is inversely related to reconstructability while their method is more sensitive to reconstructability than ours, which is shown as a heavy-tailed distribution.

We evaluate both algorithms based on a fault tolerance criterion; how far the system can tolerate erroneous input parameters in Figure 5.12. Three sources of error are tested: degree of motion, camera poses, and a point trajectory model. Mis-estimated degree of motion, d or K, results in inaccurate reconstruction, i.e., the reconstructed trajectory can be overfitted or oversmoothed. We randomly generate a trajectory with K basis vectors or d degree of polynomial and reconstruct it with K_r and d_r. Figure 5.12(a) shows that their algorithm breaks when the trajectory is reconstructed with smaller d_r, i.e., $\Delta d = d_r - d < 0$, while our method does not break significantly for $\Delta K = K_r - K < 0$. When $\Delta d > 0$ and $\Delta K > 0$, the reconstruction

[7]To solve the second part of the optimization, they have to additionally solve $\begin{pmatrix} d+2 \\ d \end{pmatrix}$ linear equations.



(a) Parameter error (b) 2D measurement noise (c) Trajectory smoothness

Figure 5.12: We compare our algorithm with the method proposed by Kaminski and Teicher [68]. We measure reconstruction error as changing error of input parameters. (a) We show our algorithm can reconstruct a trajectory with high accuracy although the number of basis vectors is mis-estimated while their method cannot. ΔK and Δd are difference between ground truth parameters and estimated parameters. (b) We illustrate the cases where camera poses or 2D projections are inaccurate. (c) We show how much our method can tolerate a trajectory that cannot be modeled by its representation, i.e., non-smooth trajectories. For all cases, our method outperforms their method, i.e., less error and more stable reconstruction. Also our method exhibits graceful degradation when the error of input parameters increases. Note that the shaded area represents standard deviation of each 3D reconstruction error.

is comparable with $d_r = d$ and $K_r = K$. Inaccurate camera pose estimation can produce 2D image measurement noise. We measure 3D reconstruction error as varying Gaussian noise of the projections. Their method easily breaks in the presence of the image noise while our method can still reconstruct with high accuracy at high noise levels as shown in Figure 5.12(b). Finally, we test how much an algorithm can handle a trajectory that cannot be modeled by its representation. Both algorithms model point motion as a smooth trajectory. We generate a 3D smooth trajectory and mix with Gaussian noise to create a non-smooth trajectory. Our algorithm is more tolerant on non-smooth trajectory with high accuracy than their algorithm as shown in Figure 5.12(c). For all cases, our method degrades gracefully as the error of input parameters increases while their method easily breaks.

5.3.2 Qualitative Evaluation

The theory of reconstructability states that it is possible to reconstruct 3D point trajectories using the DCT basis vectors if a camera trajectory is random (non-smooth). An interesting real world example of this case occurs when many independent photographers take temporally non-coincidental images of the same event from different locations. A collection of non-coincidental photos can be interpreted as the random motion of a camera center. Using multiple photographers, we collected data in several 'media event' scenarios: a person *rock climbing*, a photo-op *hand shake*, a public *speech*, *greeting*, and *dance*.

The parameters for each scenario are summarized in Table 5.1. We were able to use the DCT basis vectors for all scenes. The required number of the basis vectors implies the complexity

Table 5.1: Parameters of real data sequences.

	F (sec)	# of photos	# of photographers
Rock climbing	39	107	5
Handshake	10	32	3
Speech	24	67	4
Greeting	24	66	4
Dance	16	49	4

(a) Reprojection error (b) Selection per point

Figure 5.13: Reprojections of trajectories from manually selected K and automatically selected K_i are shown for the dance scene. (a) Red cross: measurement, cyan circle: manually selected K, and green triangle: automatically and individually selected K_i. Trajectory from K_i has smaller reprojection error. Average reprojections for K and K_i are 11.55 and 6.47, respectively. (b) The number of basis vectors per point is color-coded. The points on the hands require many basis vectors while the points on the left leg which barely move requires few basis vectors.

of the trajectory. A long sequence such as the rock climbing scene generally requires a larger number of basis vectors than a short sequence such as the hand shake scene as shown in Figure 5.14. Figures 5.15, 5.16, 5.17, 5.18, and 5.19 show some of input images and reconstructed point trajectories (the number of basis vectors is color-coded into a trajectory).

Selection of the Number of Basis Vectors To validate the proposed method of selecting the number of basis vectors described in Section 5.1.2, we tested on static points of real scenes where we know $K_i = 1$. As a result, static points of most scenes are classified as $K_i = 1$ ($> 96\%$) except for the speech scene ($> 70\%$). For the speech scene, since the baselines between photographers are very small uncertainty of the depth of points is relatively high. This causes some static points in the speech scene to be classified as points with motion along the depth direction. Figure 5.13 shows results of automatic selection of the number of basis vectors for the dance scene. It is compared with $K = 14$ which is set manually for all trajectories. Automatic selection produces smaller reprojection error and it describes point motions better than manual selection.

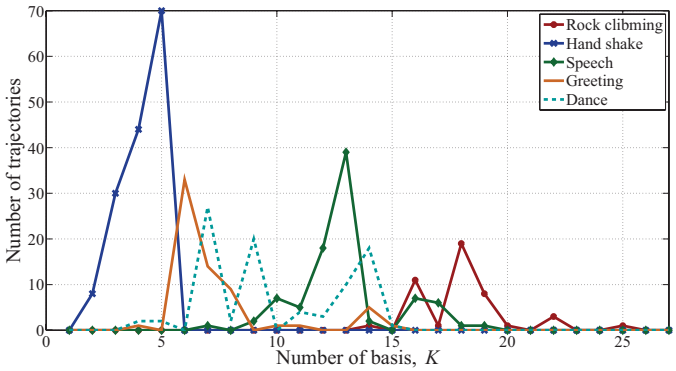


Figure 5.14: The distribution of the number of basis vectors. Scenes which are long or contain complex trajectories such as the rock climbing scene or the speech scene (complex hand motions), require the high number of basis vectors while short or simple motion scenes such as the hand shake scene or the greeting scene require the low number of basis vectors. In the greeting scene, there are several trajectories that exhibit a relatively the high number of basis vectors ($14 \sim 15$), which correspond to the hand motion (there is hand waving motion.).

5.4 Summary

We present an algorithm to robustly estimate the general motion of a 3D point from monocular perspective projections. The algorithm is stable in the presence of missing data and measurement error. We characterize the cases when 3D reconstruction is possible and how accurate it can be, based on the relationship between camera motion and point motion. We also categorize systems as solvable or unsolvable and further define a criterion called reconstructability to characterize the stability of solvable systems. The algorithm automatically selects the number of trajectory basis vectors for each trajectory individually using a cross validation scheme, so as to maximize reconstructability. In addition, we refine the trajectories initialized by the least squares system by minimizing image reprojection error directly.

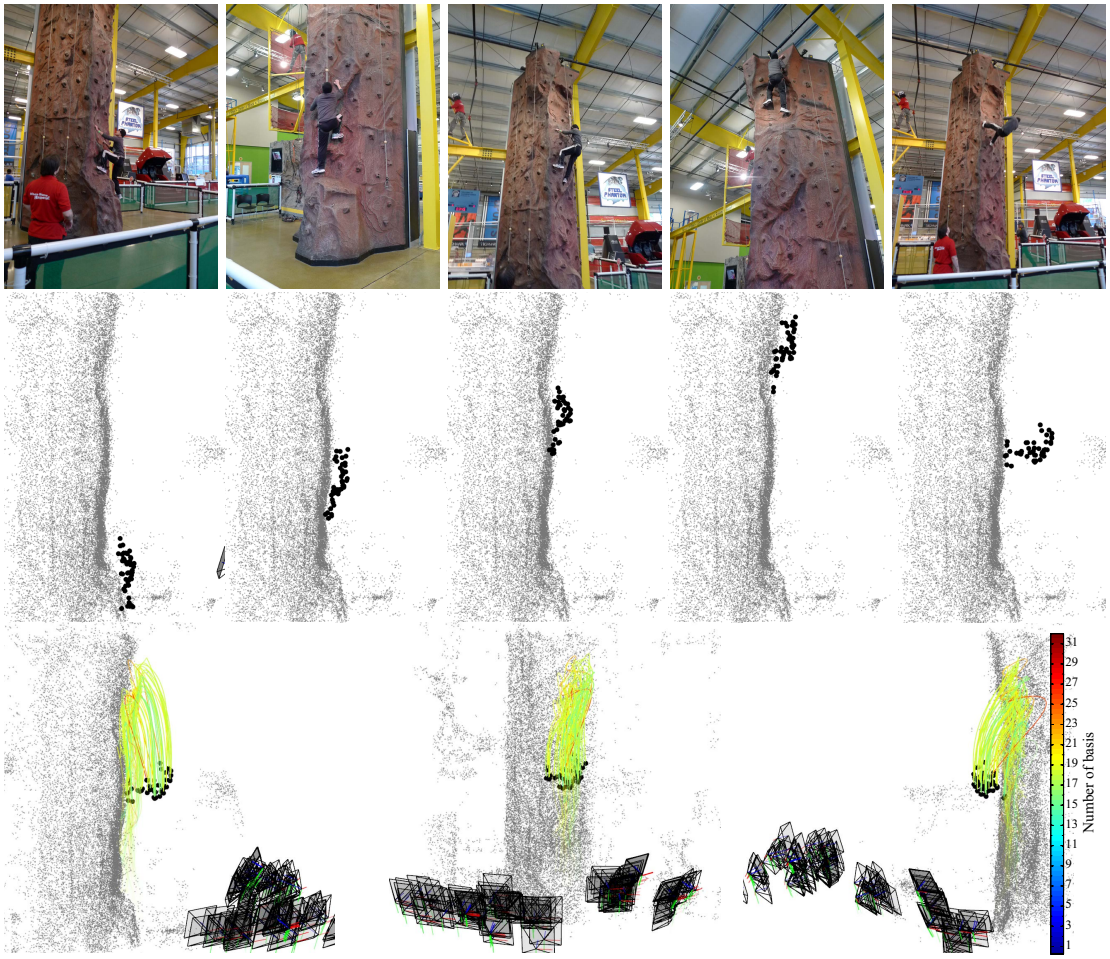


Figure 5.15: Results of the rock climbing scene. Top row: sampled image input, second row: five snap shots of 3D reconstruction of motion of the rock climber, and bottom row: reconstructed trajectories in different views. The number of basis vectors is color-coded.

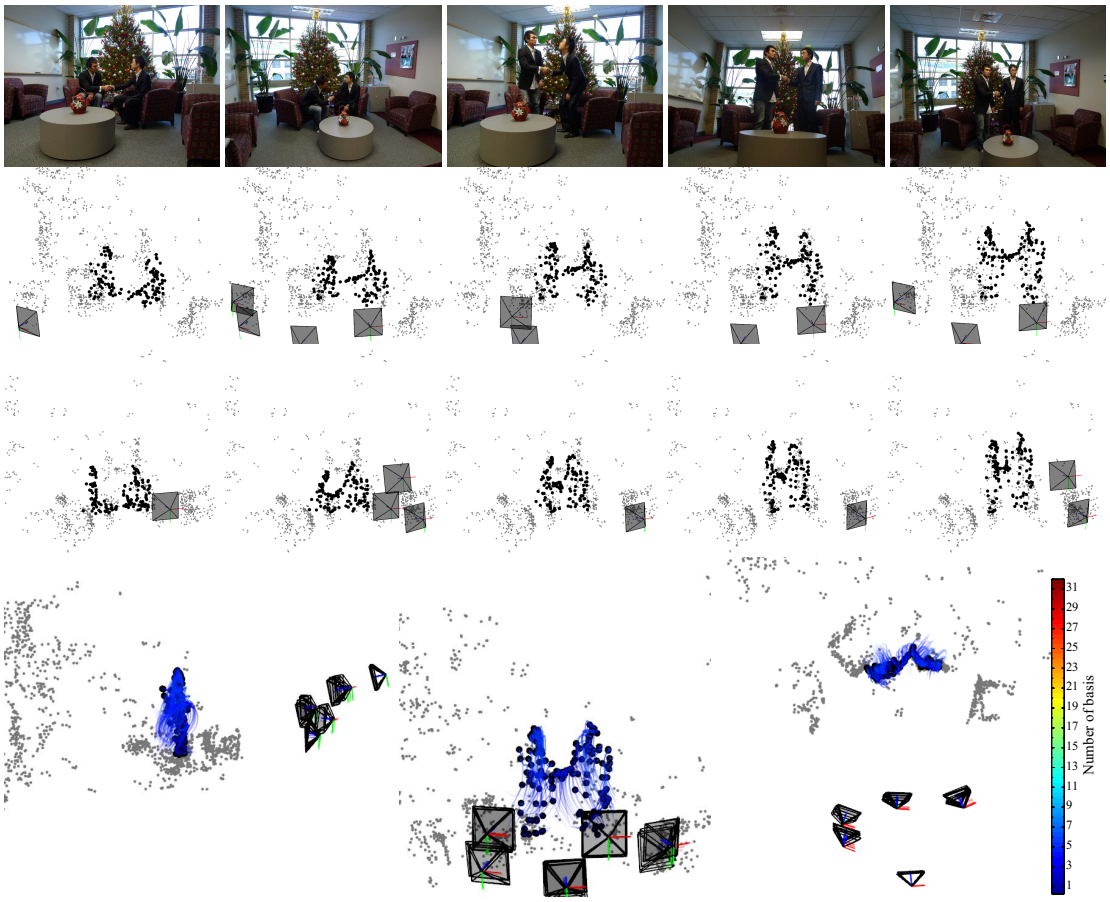


Figure 5.16: Results of the handshake scene. Top row: sampled image input, second and third row: five snap shots of 3D reconstruction in different views, and bottom row: reconstructed trajectories. The number of basis vectors is color-coded.

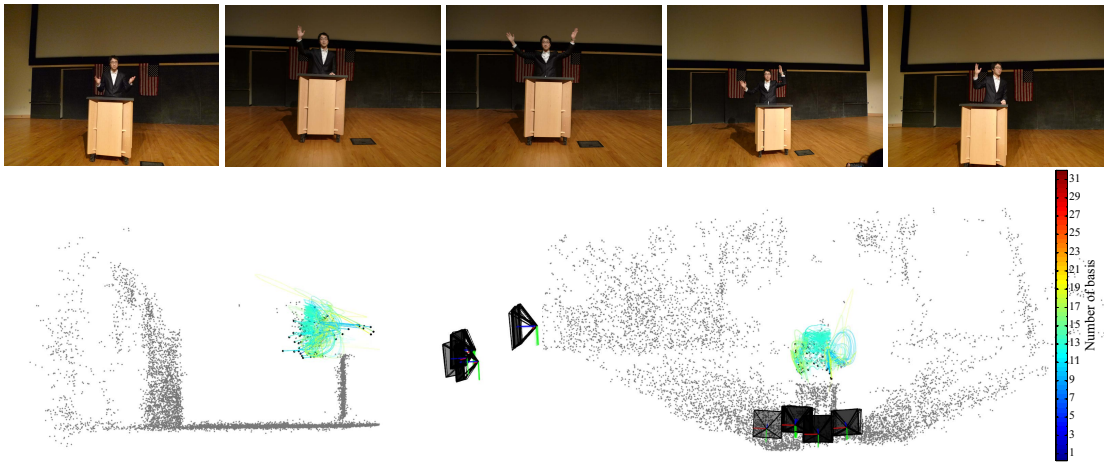


Figure 5.17: Results of the speech scene. Top row: sampled image input, and bottom row: reconstructed trajectories in different views. The number of basis vectors is color-coded.

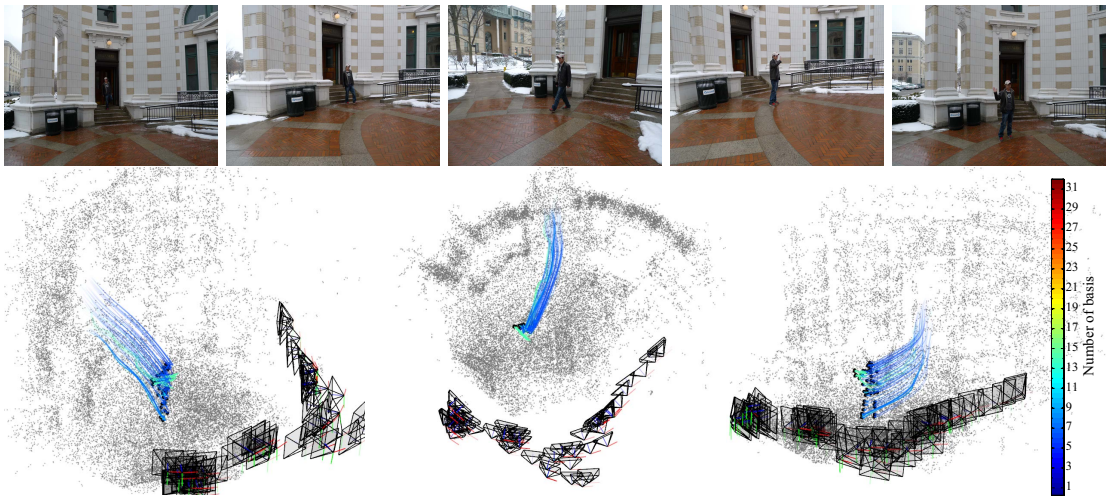


Figure 5.18: Results of the greeting scene. Top row: sampled image input and bottom row: reconstructed trajectories in different views. The number of basis vectors is color-coded.

Figure 5.19: Results of the dance scene. Top row: sampled image input, and bottom row: reconstructed trajectories in different views. The number of basis vectors is color-coded.

Part II

Social Behavior Understanding

"A pertinent form of statistical treatment would be one which deals with social configurations as wholes, and not with single series of facts, more or less artificially separated from the total picture."

–-J. L. Moreno and H. H. Jenning (1938) [89]

Social signals convey a transmitter's attention, emotion, or intent. These signals are interdependent, and therefore a direct concatenation of individual behavioral analyses is not sufficient to understand a social interaction. Therefore, a unified representation of social signals is required. In this part, we focus on interpreting attentive behaviors using joint attention and modeling the relationship between them from the reconstructed gaze rays in a unified 3D coordinate system.

Joint attention plays a central role in social interactions. We begin an interaction by engaging joint attention, communicate through joint attention, and change the subject of joint attention in turn-taking interactions. In Chapter 6, we study joint attention represented by social charges—latent quantities that drive attentive behaviors. We reconstruct the social charges that form at the intersections of gaze directions of members in a social group.

In Chapter 7, we build a relational model of social gaze behaviors inspired by the study of electric fields. The social charges induce a gradient field that aligns with the gaze direction of social members. This gradient field defines the relationship between gaze behaviors. We present a method to reconstruct the gradient field and predict gaze behaviors at any location and time.

Chapter 6

3D Joint Attention Reconstruction

Humans transmit visible social signals about what they find important and these signals are powerful cues for social scene understanding [150]. For instance, humans spontaneously orient their gaze to the target of their attention. When multiple people simultaneously pay attention to the same point in three dimensional space, e.g., at an obnoxious customer at a restaurant, their gaze rays[1] converge to a hypothetical point that we refer to as a *social charge*. Social charges are foci of the 3D social saliency field of a scene. It is an effective approximation because although an individual's gaze indicates what he or she is subjectively interested in, a social charge encodes the consensus of multiple individuals. In a scene occupied by a larger number of people, multiple such concurrences may emerge as social cliques form and dissolve. In this chapter, we present a method to reconstruct the social charges from the primary gaze direction estimated in Section 3.

6.1 3D Social Charge Reconstruction

Social saliency is a measure of social significance of a 3D point. The more people attending to it, the higher social saliency of the point. Let us define a social saliency field, $f : \mathbb{R}^3 \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ where $f(\mathbf{x}, t)$ denotes the density of social saliency at a 3D location, $\mathbf{x} \in \mathbb{R}^3$, and time instant, t. We model the social saliency field based on people's gaze directions, as their attention/interest is directly associated with their gaze directions [35]. High density in the social saliency field is formed at a social charge where multiple gaze directions intersect in 3D, i.e., when people's attention simultaneously coincides at the social charge as shown in Figure 6.1.

6.1.1 Social Saliency Field Construction

Our observations from social cameras are primary gaze rays. The gaze ray model discussed in Section 3.1 generates a distribution of points of regard for each primary gaze ray as shown in Figure 6.2(b). The superposition of these distributions of all people's gaze ray models in a scene yields a 3D social saliency field.

[1] A gaze ray is a three dimensional ray emitted from the center of eyes and oriented to the point of regard as shown in Figure 3.1(a).

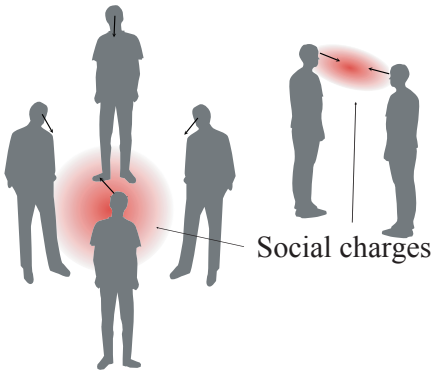


Figure 6.1: We present a method to reconstruct 3D joint attention represented by social charges—latent quantities that form at where the gaze directions of the members in a social group intersect.

For any point in 3D, $\mathbf{x} \in \mathbb{R}^3$, a density function (social saliency field), f, is generated by our gaze ray model. Let $f_i(\mathbf{x}, t)$ be a distribution of social saliency generated by the i^{th} single gaze ray model, $\mathbf{l}_i$ that is made of $\mathbf{p}_i^t$ and $\mathbf{v}_i^t$, at a given time instant, t. f_i can be written as follows,

$$f_i(\mathbf{x}, t) = K\left(\frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i}\right) = \frac{1}{h_i\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{\|\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})\|^2}{h_i^2}\right) \tag{6.1}$$

where h_i is a bandwidth of gaze ray model set to be the standard deviation of eye-in-head motion obtained from the gaze ray calibration (Section 3.2.1) for the i^{th} gaze ray. Note that $\mathbf{l}_i^t$ is time-varying because gaze directions shift over time. $K(\cdot)$ is a Gaussian kernel density function and $\mathbf{d} \in \mathbb{R}^3$ is a perspective distance vector defined as

$$\mathbf{d}(\mathbf{l}_i^t(\mathbf{p}_i^t, \mathbf{v}_i^t), \mathbf{x}) = \begin{cases} \frac{\mathbf{x} - \widehat{\mathbf{x}}_i}{(\mathbf{x} - \mathbf{p}_i^t)^\mathsf{T} \mathbf{v}_i^t} & \text{for } (\mathbf{x} - \mathbf{p}_i^t)^\mathsf{T} \mathbf{v}_i^t \geq 0 \\ \infty & \text{otherwise,} \end{cases} \tag{6.2}$$

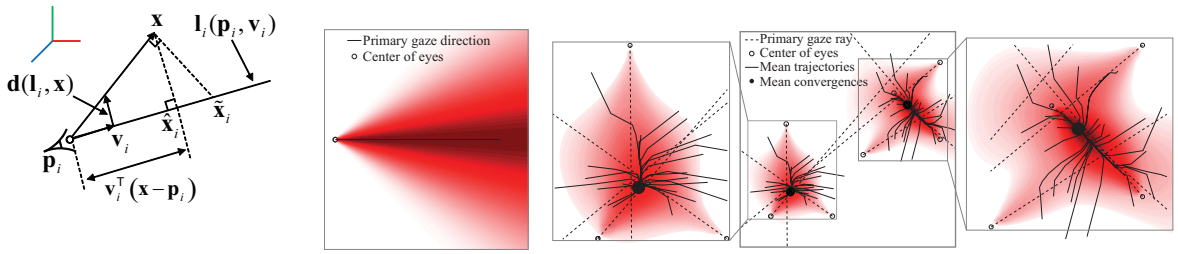
where $\widehat{\mathbf{x}}_i = \mathbf{p}_i^t + \left((\mathbf{x} - \mathbf{p}_i^t)^\mathsf{T} \mathbf{v}_i^t\right) \mathbf{v}_i^t$, which is the projection of $\mathbf{x}$ onto the primary gaze ray as shown in Figure 6.2(a). $\mathbf{p}_i^t$ is the center of eyes and $\mathbf{v}_i^t$ is the direction vector for the i^{th} primary gaze ray. Note that when $(\mathbf{x} - \mathbf{p}_i^t)^\mathsf{T} \mathbf{v}_i^t < 0$, the point is behind the eyes, and therefore is not visible. This distance vector directly captures the distance between the primary gaze ray, $\mathbf{l}$, and the point of regard, $\mathbf{l_m}$, in Section 3.1 and therefore, this kernel density function yields a cone-shaped density field (Figure 6.2(b)).

The social saliency field is a superposition (average) of the distribution generated by all gaze ray models as follows:

$$f(\mathbf{x}, t) = \frac{1}{N} \sum_{i=1}^{N} f_i(\mathbf{x}, t) \tag{6.3}$$

$$= \frac{1}{N} \sum_{i=1}^{N} K\left(\frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i}\right) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h_i\sqrt{2\pi}} \exp\left(-\frac{1}{2}\frac{\|\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})\|^2}{h_i^2}\right), \tag{6.4}$$

where N is the number of gaze rays and the social saliency field is normalized by N. This social saliency modeling emphasizes on a region where gaze directions overlaps, which is social



(a) Geometry (b) Gaze model (c) Social saliency field and mean trajectories

Figure 6.2: (a) $\widehat{\mathbf{x}}_i$ is the projection of $\mathbf{x}$ onto the primary gaze ray, $\mathbf{l}_i$, and $\mathbf{d}$ is a perspective distance vector defined in Equation (6.2). (b) Our gaze ray representation results in the cone-shaped distribution in 3D. (c) Two social charges are formed by seven gaze rays. High density is observed around the intersections of rays. Note that the maximum intensity projection [152] is used to visualize the 3D density field. Our mean-shift algorithm allows any random points to converge to the highest density point accurately.

charges. Note that all social saliency derived by gaze directions are at the same time instant, t. Figure 6.2(c) shows a social saliency field (density field) generated by seven gaze rays. The regions of high density are the social charges, where the gaze ray models intersect. Note that the maximum intensity projection [152] of the density field is used to visualize a 3D density field.

6.1.2 Social Charge Estimation via Mode-seeking

3D social charges are formed at the intersections of multiple gaze rays, not at the intersection of multiple primary gazes (see Figure 3.1(a)). If we knew the 3D gaze rays, and which of the rays shared a social charge, the point of intersection could be directly estimated via least squares estimation, for example. In our setup, neither one of these are known, nor do we know the number of social charges. With a social camera, only the primary gaze ray is computable; the eye-in-head motion is an unknown quantity (see Chapter 3). This precludes estimating the 3D social charge by finding a point of intersection, directly. In this section, we present a method to estimate the number and the 3D locations of social charges given primary gaze rays.

The modes in a social saliency field correspond to the social charges. We seek the modes via a mean-shift algorithm [25, 28, 42]. The mean-shift algorithm finds the modes by evaluating the weights between the current mean and observed points. We derive the closed form of the mean-shift vector from the social saliency field constructed by our gaze ray models. While our observations are gaze ray models represented by $\mathbf{p}_i^t$, $\mathbf{v}_i^t$, and h_i, the estimated modes are points in 3D. This formulation differs from the classic mean-shift algorithm where the observations and the modes lie in the same space.

The social saliency field in Equation (6.4) can be rewritten as

$$f(\mathbf{x}, t) = \frac{1}{N}\sum_{i=1}^{N} K\left(\frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i}\right) = \frac{c}{N}\sum_{i=1}^{N}\frac{1}{h_i}k\left(\frac{\|\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})\|^2}{h_i^2}\right) \tag{6.5}$$

k is the profile of the kernel density function, i.e., $K(\cdot) = ck(\|\cdot\|^2)/h$ and c is a scaling constant.

The updated mean is the location where the maximum density increase can be achieved from the current mean. Thus, it moves along the gradient direction of the density function evaluated at the current mean. The gradient of the density function, $f(\mathbf{x}, t)$, is

$$\begin{aligned}
\nabla_{\mathbf{x}} f(\mathbf{x}, t) &= \frac{2c}{N} \sum_{i=1}^{N} \frac{1}{h_i^3} k' \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right) \mathbf{d}(\mathbf{l}_i^t, \mathbf{x})^{\mathsf{T}} \left(\nabla_{\mathbf{x}} \mathbf{d}(\mathbf{l}_i^t, \mathbf{x}) \right) \\
&= \frac{2c}{N} \sum_{i=1}^{N} \frac{1}{h_i^3} k' \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right) \left(\frac{(\mathbf{x} - \widehat{\mathbf{x}}_i)}{(\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t} - \frac{\|\mathbf{x} - \widehat{\mathbf{x}}_i\|^2}{\left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^3} \mathbf{v}_i^t \right)^{\mathsf{T}} \\
&= \frac{2c}{N} \sum_{i=1}^{N} \frac{k' \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2} (\mathbf{x} - \widetilde{\mathbf{x}}_i)^{\mathsf{T}} \\
&= \frac{2c}{N} \sum_{i=1}^{N} \frac{g \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2} (\widetilde{\mathbf{x}}_i - \mathbf{x})^{\mathsf{T}} \\
&= \frac{2c}{N} \left[\sum_{i=1}^{N} \frac{g \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2} \right] \left[\frac{\sum_{i=1}^{N} \frac{g \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2} \widetilde{\mathbf{x}}_i}{\sum_{i=1}^{N} \frac{g \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2}} - \mathbf{x} \right]^{\mathsf{T}} \\
&= \frac{2c}{N} \left[\sum_{i=1}^{N} w_i \right] \left[\frac{\sum_{i=1}^{N} w_i \widetilde{\mathbf{x}}_i}{\sum_{i=1}^{N} w_i} - \mathbf{x} \right]^{\mathsf{T}}, \qquad (6.6)
\end{aligned}$$

where

$$w_i = \frac{g \left(\left\| \frac{\mathbf{d}(\mathbf{l}_i^t, \mathbf{x})}{h_i} \right\|^2 \right)}{h_i^3 \left((\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t \right)^2}, \quad \widetilde{\mathbf{x}}_i = \widehat{\mathbf{x}}_i + \frac{\|\mathbf{x} - \widehat{\mathbf{x}}_i\|^2}{(\mathbf{x} - \mathbf{p}_i^t)^{\mathsf{T}} \mathbf{v}_i^t} \mathbf{v}_i^t,$$

and $g(x) = -k'(x)$. $\widetilde{\mathbf{x}}_i$ is the location that the gradient at $\mathbf{x}$ points to with respect to $\mathbf{l}_i^t$ as shown in Figure 6.2(a). Note that the gradient direction at $\mathbf{x}$ is perpendicular to the ray connecting $\mathbf{x}$ and $\mathbf{p}_i^t$. The last term of Equation (6.6) is the difference between the current mean estimate and the weighted mean. The new mean location, $\mathbf{x}^{j+1}$, can be achieved by adding the difference to the current mean estimate, $\mathbf{x}^j$:

$$\mathbf{x}^{j+1} = \frac{\sum_{i=1}^{N} w_i^j \widetilde{\mathbf{x}}_i^j}{\sum_{i=1}^{N} w_i^j}. \qquad (6.7)$$

Equation (6.7) shows the update rule for a mode-seeking algorithm that does not require any prior knowledge of the number and the locations of modes. The modes where the weighted

means converge correspond to social charges in the social saliency field. Figure 6.2(c) illustrates how our mean-shift vector moves random initial points according to the gradient information towards social charges.

6.1.3 Social Charge Temporal Association

We find local maxima of the distribution using a meanshift algorithm in Section 6.1.2. We present a method to track the detected social charges across time based on membership features.

Let $\mathbf{M}_i \in \mathbb{R}^J$ be a *membership feature* associated with each social charge. Each element of the membership feature denotes a probability that the j^{th} member belongs to the i^{th} social charge,

$$\mathbf{M}_i = \frac{1}{\sqrt{\sum_{i=1}^{M} f_i(\mathbf{x},t)^2}} \begin{bmatrix} f_1(\mathbf{x},t) \\ \vdots \\ f_M(\mathbf{x},t) \end{bmatrix}. \tag{6.8}$$

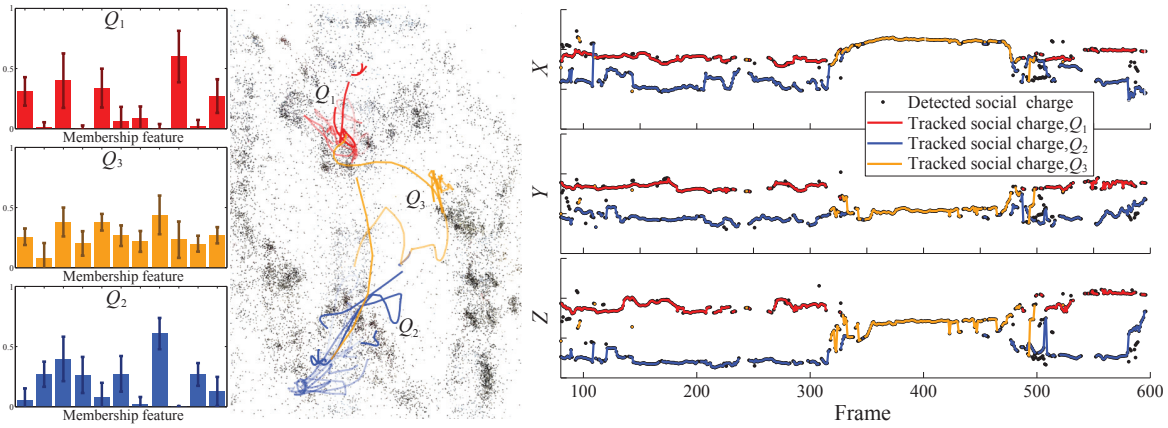
where $f_i(\mathbf{x},t)$ is a function used to construct the social saliency field. $\mathbf{M}_i$ is a normalized descriptor where the magnitude is one and each element in $\mathbf{M}_i$ states relative importance between members for $\mathbf{x}$, i.e., it is inversely proportional to the distance from each gaze ray models. This membership feature enables us to describe a social charge in terms of the participating members.

The membership feature from a social charge remains a similar pattern across time because the same members tend to stay in their social clique as shown in Figure 6.3(a). We compute the membership features of all the detected social charges and cluster the charges using the classic meanshift algorithm [42] based on the features. The meanshift clustering enables us to label each charge across time instances. A set of the charges clustered by the same label forms a trajectory of a single social charge. When multiple charges at the same time instant are labeled in a single cluster, we choose the charge that is close to the center of the feature cluster.

The social charge representation via a membership feature enables us to track a social charge across location and time. A charge may move in 3D as long as the participating members remain the same. It can dissolve and re-emerge as the group disperses and re-unites, respectively. This introduces missing data because of temporary dissolution of the social charge as shown in Figure 6.3(b). Our tracking method can re-associate with the re-emerging charges based on the membership feature clustering because two temporally separated trajectories of the social charge have the same membership feature.

6.2 Results

We evaluate our algorithm quantitatively using a motion capture system to provide ground truth and apply it to real world examples where social interactions frequently occur. We use GoPro HD Hero2 cameras (www.gopro.com) and use the head mounting unit provided by GoPro. We synchronize the cameras using audio signals, e.g., a clap. In the calibration step, we ask people to form pairs, and move back and forth and side to side at least three times to allow the gaze ray model to be accurately estimated. For the initial points of the mean-shift algorithm, we sample



(a) Membership feature (b) Social charge tracking

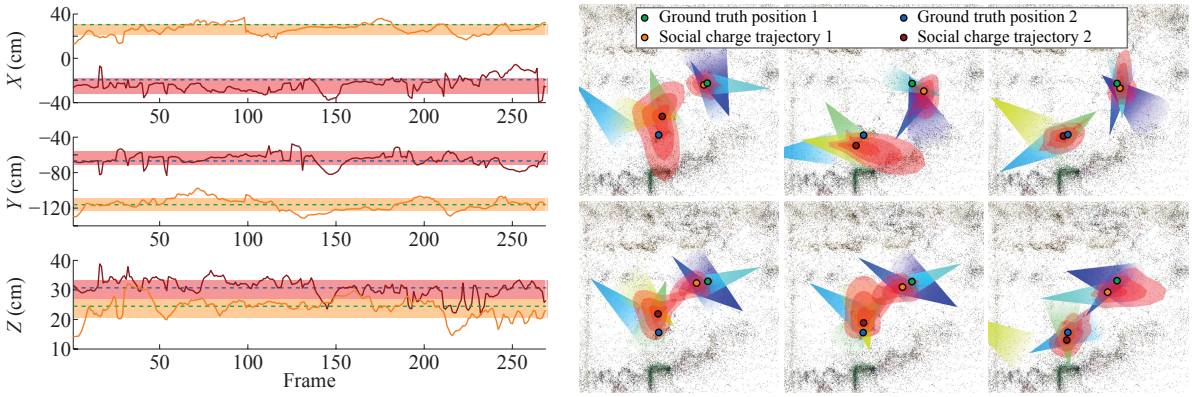
Figure 6.3: (a) The membership feature reflects the participating members in a group. We temporally associate the detected charges based on the membership features. The membership features for Q_1 and Q_2 are complementary because the groups are formed in the same time. (b) The trajectories of the social charges are illustrated. Q_1 and Q_2 dissolve at frame 350 and reappear at frame 500. Our membership based tracking allows us to associate the temporally separated trajectories.

several points on the primary gaze rays. This sampling results in convergences of the mean-shift because the local maxima form around the rays. If the weights of the estimated mode are dominated by only one gaze, we reject the mode, i.e., more than one gaze rays must contribute to estimate a social charge.

6.2.1 Quantitative Evaluation

We compare the 3D social charges estimated by our result with ground truth obtained from a motion capture system (capture volume: 8.3m$\times$17.7m$\times$4.3m). We attached several markers on a camera and reconstructed the camera motion using structure from motion and the motion capture system simultaneously. From the reconstructed camera trajectory, we recovered the similarity transform (scale, orientation, and translation) between two reconstructions. We placed two static markers and asked six people to move freely while looking at the markers. Therefore, the 3D social charges estimated by our algorithm should coincide with the 3D position of the static markers.

The top row in Figure 6.4 shows the trajectories of the social charges (solid lines) overlaid by the static marker positions (dotted lines). The mean error is 10.1cm with 5.73cm standard deviation. The bottom row in Figure 6.4 shows the social charges (orange and red points) with the ground truth positions (green and blue points) and the confidence regions (pink region) where a high value of the saliency field is achieved (region which has higher than 80% of the local maximum value). The ground truth locations are always inside these regions.



(a) Quantitative result (b) Social charge location

Figure 6.4: (a) The solid lines (orange and red) are the trajectories of the social charges and the dotted lines (green and blue) are the ground truth marker positions. The colored bands are one standard deviation wide and are centered at the trajectory means. (b) There are two social charges with six people.

6.2.2 Qualitative Evaluation

We apply our method to reconstruct 3D social charges in three real world scenes: a meeting, a musical, and a party. Figures 6.6, 6.5, and 6.7 show the reconstructed social charges and the projections of 3D social charges onto the image plane (top row). 3D renderings of the social charges (red dots) with the associated confidence region (salient region) are drawn in the middle row and the cone-shaped gaze ray models are also shown. The trajectories of the social charges are shown in the bottom row. The transparency of the trajectories encodes the timing. All results are best seen in the videos from the following project website[2].

Meeting scene: There were 11 people forming two groups: 6 for one group and 5 for the other group as shown in Figure 6.6. The people in each group started to discuss among themselves at the beginning (2 social charges). After a few minutes, all the people faced the presenter in the middle (50th frame: 1 social charge), and then they went back to their group to discuss again (445th frame: 2 social charges) as shown in Figure 6.6.

Musical scene: 7 audience members wore social cameras and watched the song, "Summer Nights" from the musical *Grease*. There were two groups of actors, "the Pink Ladies (women's group)" and "the T-birds (men's group)" and they sang the song alternatingly as shown in Figure 6.5. In the figure, we show the reconstruction of two frames when the pink ladies sang (41st frame) and when the T-birds sang (390th frame).

Party scene: There were 11 people forming 4 groups: 3 sat on couches, 3 talked to each other at the table, 3 played table tennis, and 2 played pool (178th frame: 4 social charges) as shown in Figure 6.7. Then, the whole group moved to watch the table tennis game (710th frame: one social charge).

Croquet scene: There were 6 people played a croquet game with the social cameras as shown in Figure 6.8. The game took 25 minutes. In most cases, social saliency is formed around the active

[2] http://www.cs.cmu.edu/~hyunsoop/gaze_concurrence.html

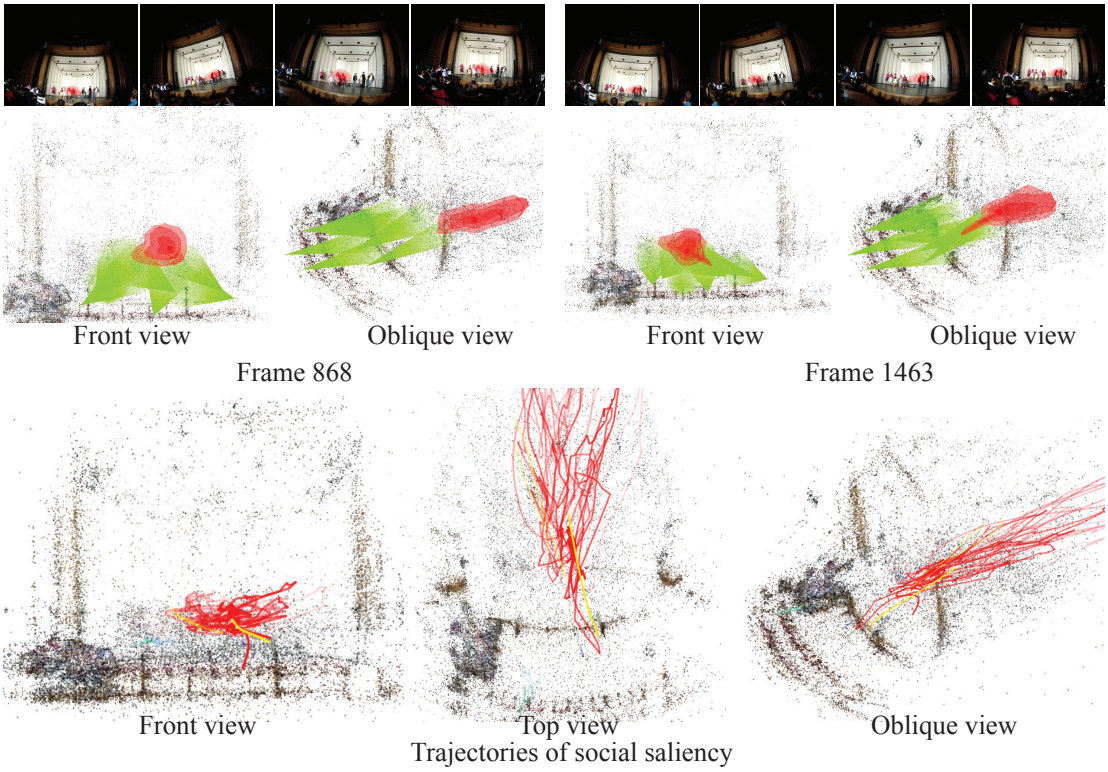


Figure 6.5: We reconstruct social saliency from the audiences of a musical. 7 social cameras were used to capture the scene. There were two groups of actors: the pink ladies and the T-birds. They sang the "Summer Nights" song from *Grease*, alternatingly. Each column corresponds to different time instant. Top row: images with the reprojection of social saliency, middle row: rendering of the 3D social saliency with cone-shaped gaze ray models, bottom row: the trajectories of social saliency. The transparency of trajectories encodes the timing.

player. Depending on the location of the active player, the social saliency moves. Our method correctly evaluates the social charges at the location where people look.

6.3 Summary

Our algorithm constructs a 3D social saliency field of a scene by superimposing the gaze models of members in social groups. This field represents the density of social attention and the modes of the field correspond to social charges (joint attention). We present a novel method to automatically estimate the number, locations, and magnitudes of the social charges via mode-seeking in the social saliency field at each time instant. We find a temporal association via mean-shift clustering on membership features, which enables us to track social charges over time.

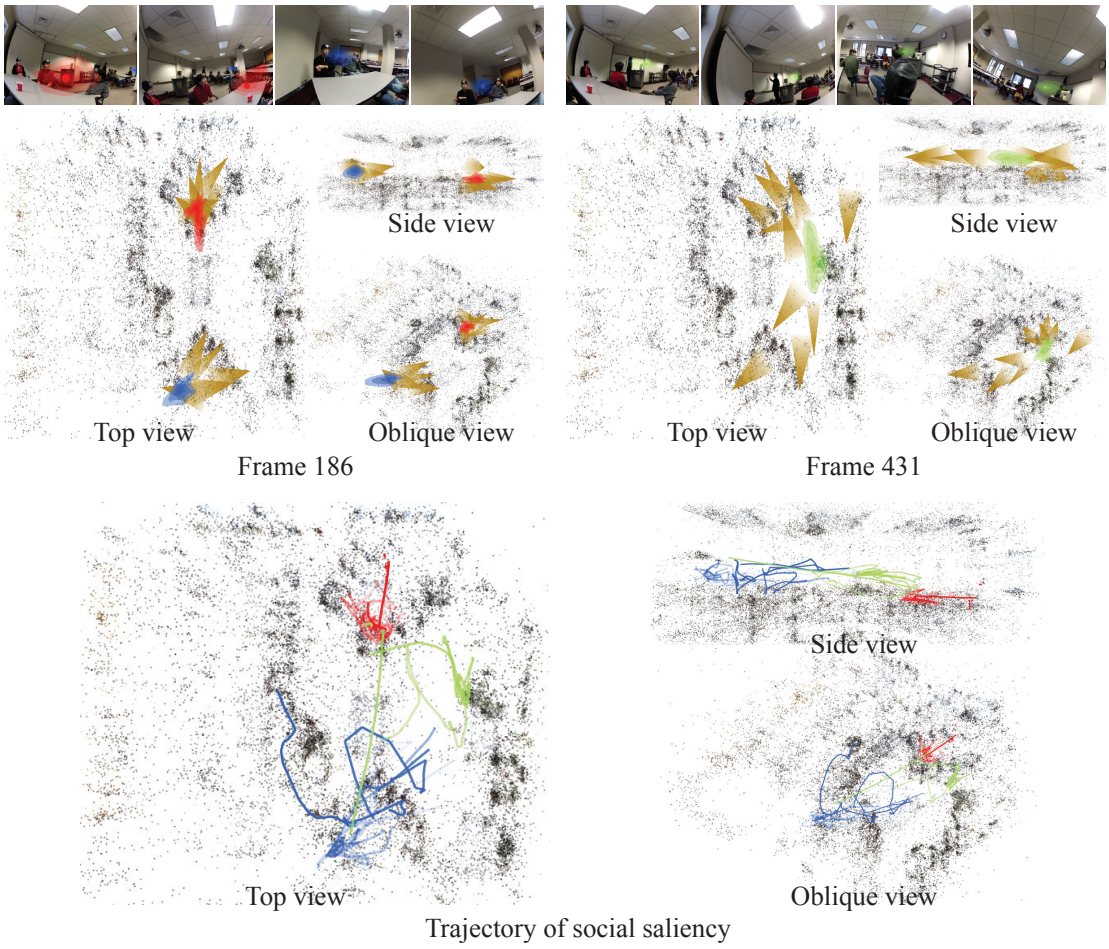


Figure 6.6: We reconstruct social saliency in the meeting scene. 11 people formed two groups; 6 for one group and 5 for the other group. At the beginning, people in the group discussed each other (two social charges) and then faced at the presenter (one social charge). Each column corresponds to different time instant. Top row: images with the reprojection of social saliency, middle row: rendering of the 3D social saliency with cone-shaped gaze ray models, bottom row: the trajectories of social saliency. The transparency of trajectories encodes the timing.

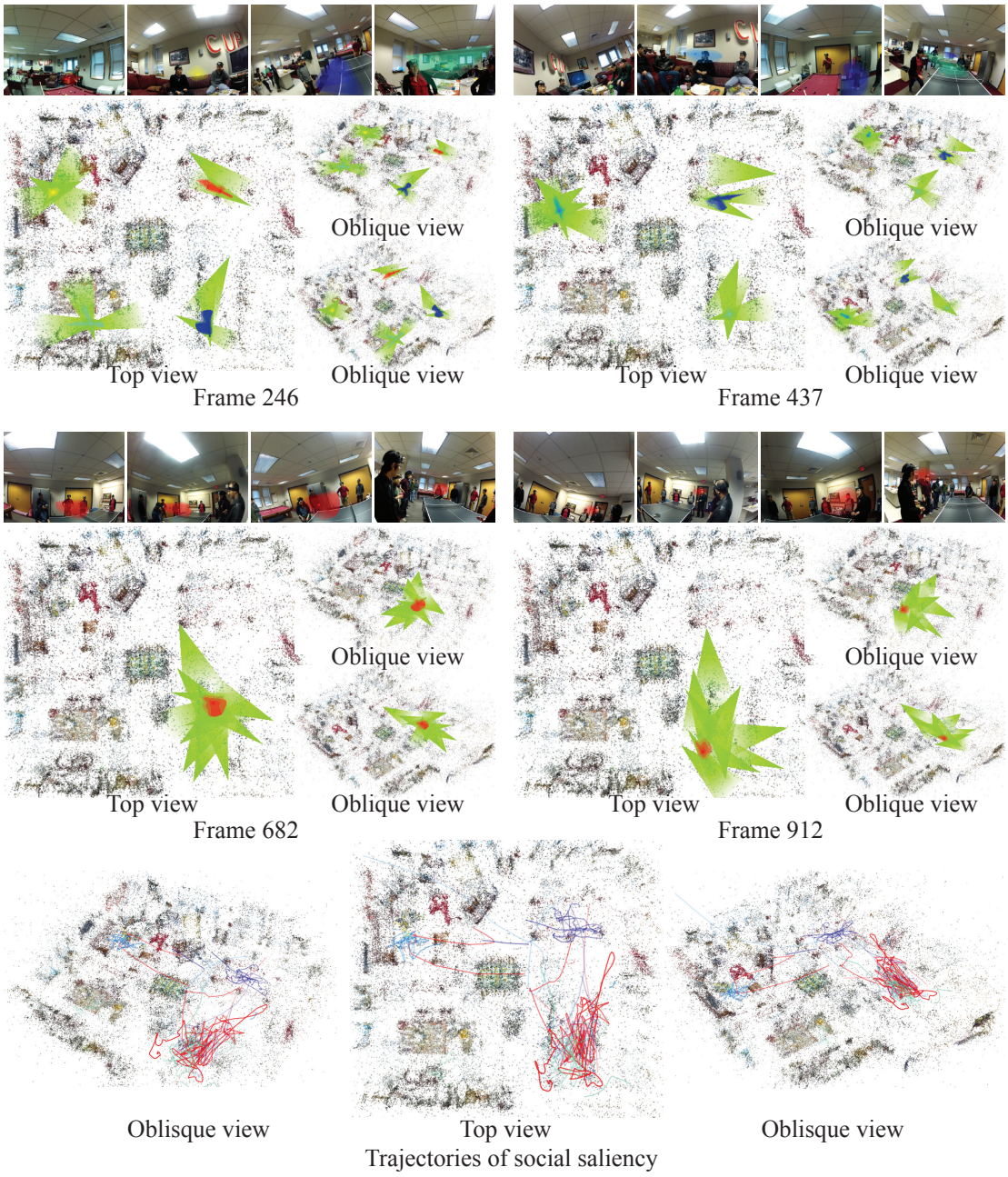


Figure 6.7: We reconstruct social saliency in the party scene. 11 people formed four groups; three sat on the couches, three talked at the table, three played the table tennis, two played the pocket ball (four social charges). Then, they moved to the table tennis to watch the game (one social charge). The first and third rows: images with the reprojection of social saliency, the second and forth rows: rendering of the 3D social saliency with cone-shaped gaze ray models, bottom row: the trajectories of social saliency. The transparency of trajectories encodes the timing.

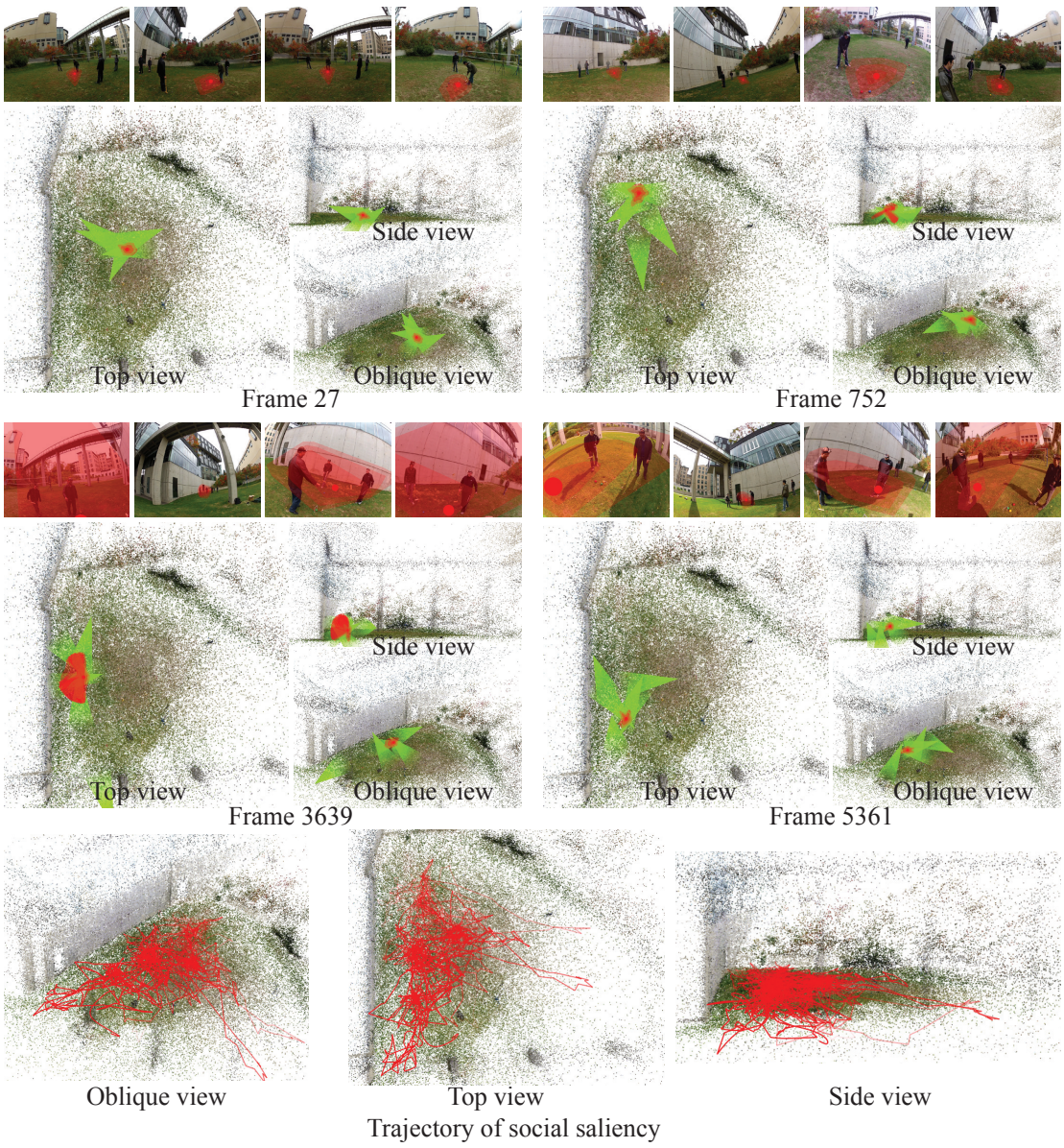


Figure 6.8: We reconstruct social saliency in the croquet scene. 6 people played and one social charge is formed across all time instances. The first and third rows: images with the reprojection of social saliency, the second and forth rows: rendering of the 3D social saliency with cone-shaped gaze ray models, bottom row: the trajectories of social saliency. The transparency of trajectories encodes the timing.

Chapter 7

Social Gaze Behavior Prediction

In this chapter, inspired by Coulomb's law, which describes the electrostatic interaction between charged particles, we present a predictive model to describe the primary gaze behavior of individuals in a social scene. We posit hypothetical social charges described in Chapter 6 that attract the attention of the individuals in the scene, and we analyze the time-varying behavior of these charges (i.e., their emergence, transition, and dissolution). We characterize how information of the time-varying location and charge of multiple moving social charges is combined to induce a *gaze field* analogous to an electric field. Under our model, this field is used to predict a distribution over primary gaze direction at any time and at any location in the scene.

We validate our social charge model on four real world sequences where various human interactions occur, including a social game, office meetings, and an informal party. We evaluate our gaze prediction with ground truth data via a cross validation scheme against a baseline regression algorithm. Finally, we demonstrate the potential of gaze prediction as a prior for head tracking and anomaly detection.

Attentive behavior is an early indicator in the diagnoses of behavioral disorders (e.g., autism [24]). Predictive models of primary gaze behavior will enable anomaly detection and hold the promise of automated diagnoses and monitoring. Such models can also be used within a filtering framework to more effectively track primary gaze direction in a social scene. In augmented reality applications, predictive models of primary gaze behavior will enable the insertion of believable virtual characters into social scenes that respond to the social dynamics of a scene. Finally, such models can also be used in human-robot interaction scenarios to appropriately direct sensors and to limit the extent of the scene that the system needs to process and react to.

7.1 Primary Gaze Behavior Prediction

A social *member* is a participant in a social scene in which multiple members interact with each other. Let $\mathbf{p}_j \in \mathbb{R}^3$ represent the center of the eyes of the j^{th} member and $\mathbf{v}_j \in \mathbb{R}^3$ represent the primary gaze direction, i.e., the ray emitted from $\mathbf{p}_j$ oriented towards the neutral gaze direction [64]. The set $\{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^{J}$ is the set of primary gaze directions and locations for the J members in the scene. Note that each $\mathbf{v}_j(t)$ and $\mathbf{p}_j(t)$ is time-varying, as the attention or location of each member can change over time. In this chapter, we predict the primary gaze

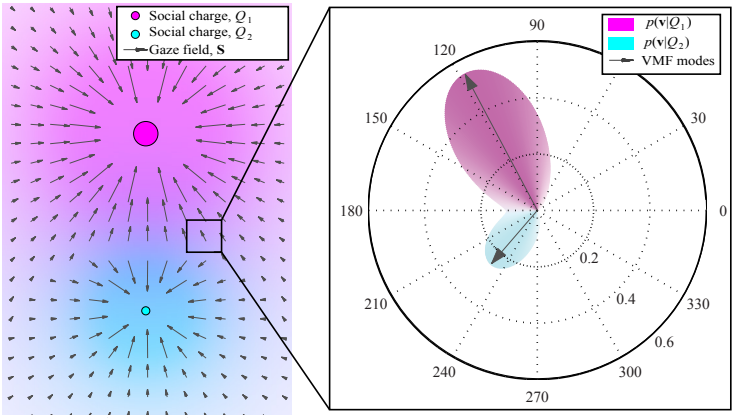


Figure 7.1: We model the relationship between a primary gaze direction and a social charge via a gaze field inspired by Coulomb's law. The two social charges (the purple and cyan points) generate the gaze field on the left figure. The size of the social charges is proportional to their magnitude. In the right figure, we show the probability distribution over gaze direction modeled by a mixture of von Mises-Fisher distributions in Equation (7.7).

direction at any 3D location and time, given the observed gaze behavior of the members. At any 3D location $\mathbf{p}$ in the scene, we can compute the maximum likelihood estimate of the gaze direction $\mathbf{v}$, given $\{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J$ by optimizing the following probability,

$$\mathbf{v}^* = \underset{\mathbf{v}}{\operatorname{argmax}}\, p\left(\mathbf{v}|\mathbf{p}, \{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J\right), \tag{7.1}$$

where $p\left(\mathbf{v}|\mathbf{p}, \{(\mathbf{v}, \mathbf{p}_j)\}_{j=1}^J\right)$ is the probability of the primary gaze direction at $\mathbf{p}$ given the observed primary gaze directions, $\{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J$.

One approach would be to directly regress $\mathbf{v}$ from $\{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J$ [36, 72]. Instead, inspired by Coulomb's law, we generatively model the relationship between primary gaze directions via latent *social charges* that drive attention of members in the scene — we show that this approach demonstrates superior predictive precision in the presence of missing and noisy measurements compared to the direct regression approach.

According to Coulomb's law, the force exerted on an electric charge due to the presence of another electric charge is directed along the line that connects these two charges. We represent a social charge as $Q = (q, \mathbf{r})$ where $q \in \mathbb{R}$ is a measure of social saliency, i.e., how strongly the social charge draws attention, and $\mathbf{r} \in \mathbb{R}^3$ is the 3D location of the charge as shown in Figure 7.1. The decay of the spatial influence of the social charge is modeled as an inverse squared function (as with classic electric field model). A social charge is a quantity that changes over time because the scene includes dynamic human interactions. There may exist multiple social charges, $\{Q_i\}_{i=1}^I$ when multiple social groups are formed, where I is the number of the charges.

The social charge, Q_i is a latent quantity, i.e., it cannot be observed directly, and can only be estimated by its observed influence on the primary gaze direction of the members in the scene. Estimating the social charges given the primary gaze directions of the members is equivalent to

optimizing the following likelihood,

$$\{Q_i^*\}_{i=1}^I = \underset{\{Q_i\}_{i=1}^I}{\operatorname{argmax}} L\left(\{Q_i\}_{i=1}^I | \{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J\right). \tag{7.2}$$

This estimates the optimal $\{Q_i^*\}_{i=1}^I$ such that each observed primary gaze direction is oriented towards one of the social charges.

From these social charges, we can predict the most likely primary gaze direction at $\mathbf{p}$ by maximizing the following probability,

$$\begin{aligned} \mathbf{v}^* &= \underset{\mathbf{v}}{\operatorname{argmax}}\, p\left(\mathbf{v}|\mathbf{p}, \{Q_i^*\}_{i=1}^I, \{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^J\right) \\ &= \underset{\mathbf{v}}{\operatorname{argmax}}\, p\left(\mathbf{v}|\mathbf{p}, \{Q_i^*\}_{i=1}^I\right). \end{aligned} \tag{7.3}$$

Our social charge model assumes that $\mathbf{v}$ is conditionally independent on $\{\mathbf{v}_j, \mathbf{p}_j\}_{j=1}^J$ given $\{Q_i\}_{i=1}^I$. We discuss this assumption in more detail in the discussion.

We will develop a computational representation for the relationship between social charges and primary gaze directions to predict the primary gaze behavior via optimizing Equation (7.3) in Section 7.2. Based on the relationship, we present a method to estimate the latent social charges given primary gaze behaviors of the observers via optimizing Equation (7.2) in Section 7.3.

7.2 Gaze Field Model

In this section, we present a computational model that captures the relationship between time-varying social charges and primary gaze behavior. The charges induce a gaze field (Figure 7.1) that enables us to define a probability of the primary gaze direction given a location and time in Equation (7.3). Comparison between the gaze field and electric field can be found in Table 7.1. The interacting force between two charges, $Q = (q, \mathbf{r})$ and $Q_{\mathbf{x}} = (q_{\mathbf{x}}, \mathbf{x})$, from Coulomb's law

Electrostatics	Gaze behaviors
Electric charge	Social charge, $Q = \{q(t) \in \mathbb{R}_{\geq 0}, \mathbf{r}(t) \in \mathbb{R}^3\}$
Electric field, $\mathbf{E}$	Gaze field, $\mathbf{S}(\mathbf{x}, t) \in \mathbb{R}^3$
$\mathbf{E}_{\text{net}} = \sum_i \mathbf{E}_i$	$\mathbf{S}_{\text{net}} = \max \mathbf{S}_i$

Table 7.1: Analogy between concepts in electric field and gaze field

is:

$$\mathbf{F} = K\frac{qq_{\mathbf{x}}\left(\mathbf{r} - \mathbf{x}\right)}{\|\mathbf{r} - \mathbf{x}\|^3}, \tag{7.4}$$

where K is a normalizing constant. The force between two charges is proportional to their magnitude of charges and inversely proportional to squares of distance. When two charges have opposite polarities, the attractive force applies along the line that connects those two charges.

A point in space that attracts attention of members is represented as a negative charge — the more attractive the point, the higher negative charge. A member in the space is represented as an

infinitesimal positive charge. We posit that a negative social charge, q, exerts an attractive force on a member (with an infinitesimal positive charge), along the line connecting the two charges $(\mathbf{r}-\mathbf{x})/\|\mathbf{r}-\mathbf{x}\|$, and with spatial influence decaying according to an inverse squared function, $\|\mathbf{r}-\mathbf{x}\|^{-2}$, as in Equation (7.4).

Analogous to the electric field, a gaze field is defined by the limiting process,

$$\mathbf{S}(\mathbf{x}) = \lim_{q_{\mathbf{x}} \to 0} \frac{\mathbf{F}}{q_{\mathbf{x}}} = K \frac{q(\mathbf{r}-\mathbf{x})}{\|\mathbf{r}-\mathbf{x}\|^3}, \tag{7.5}$$

where $\mathbf{S}(\mathbf{x})$ is the gaze field evaluating at $\mathbf{x}$, induced by a single social charge, $Q = (q, \mathbf{r})$.

When multiple electric charges exist, the net electric field induced by the charges are the superposition of the electric fields by all charges, i.e., $\mathbf{E}_{\text{net}} = \sum_{i=1}^{I} \mathbf{E}_i$ where $\mathbf{E}_{\text{net}}$ is the net electric field and $\mathbf{E}_i$ is the electric field generated by the i^{th} electric charge. Unlike the electric field, the net gaze field selectively takes one of the gaze fields[1], i.e.,

$$\mathbf{S}(\mathbf{x}) = \underset{\{\mathbf{S}_i(\mathbf{x})\}_{i=1}^{I}}{\operatorname{argmax}} \|\mathbf{S}_i(\mathbf{x})\|, \tag{7.6}$$

where $\mathbf{S}_i(\mathbf{x})$ is the gaze field induced by the i^{th} social charge, $\mathbf{Q}_i$.

To reflect selective gaze behavior, we model the underlying probability distribution of a primary gaze direction using a mixture of von Mises-Fisher distributions,

$$p\left(\mathbf{v}|\mathbf{x}, \{Q_i\}_{i=1}^{I}\right) = \sum_{i=1}^{I} \pi_i \mathcal{V}\left(\mathbf{v} \,\middle|\, \frac{\mathbf{S}_i(\mathbf{x})}{\|\mathbf{S}_i(\mathbf{x})\|}, \kappa\right), \tag{7.7}$$

where $\mathcal{V}$ is a von Mises-Fisher distribution[2] that accounts for eye-in-head motion and κ is a concentration parameter of the distribution. The mixture coefficients, $\pi_i = \|\mathbf{S}_i(\mathbf{x})\| / \sum_{k=1}^{I} \|\mathbf{S}_k(\mathbf{x})\|$, reflect the inverse squared function prior for the charges. Each von Mises-Fisher distribution measures the distance between the primary gaze direction, $\mathbf{v}$, and a unit vector from each gaze field, $\mathbf{S}_i/\|\mathbf{S}_i\|$.

Each social charge may move independently depending on the primary gaze behavior of the participating group. A trajectory of a social charge, $Q(t)$, can be written as

$$Q(t) = \begin{cases} \{q(t), \mathbf{r}(t)\} & t_e \le t \le t_d, \\ \text{undefined} & \text{otherwise}, \end{cases} \tag{7.8}$$

where t_e and t_d are the emergence and dissolution time instances of the social charge. The charge is defined between the emergence and dissolution times, and otherwise the charge does not exist.

Given the gaze field from each charge at each time instant, the net time-varying gaze field can be written as

$$\mathbf{S}(\mathbf{x}, t) = \underset{\{\mathbf{S}_i(\mathbf{x},t)\}_{i=1}^{I}}{\operatorname{argmax}} \|\mathbf{S}_i(\mathbf{x}, t)\|. \tag{7.9}$$

[1] A primary gaze direction is not oriented towards an average location between two social charges but towards one of the charges.

[2] The von Mises-Fisher distribution is the nominal equivalent of the normal distribution over $\mathbb{S}^2$.

7.3 Gaze Field Estimation

In this section, we present a method to estimate the time-varying location and magnitude of the social charges $\{Q_i(t)\}_{i=1}^{I}$, given the primary gaze directions of members, $\{(\mathbf{v}_j(t), \mathbf{p}_j(t))\}_{j=1}^{J}$, in the scene, i.e., maximize Equation (7.2). The data likelihood of Equation (7.2) can be rewritten by exploiting Equation (7.7) as

$$L\left(\{Q_i\}_{i=1}^{I} | \{(\mathbf{p}, \mathbf{v})\}_{j=1}^{J}\right) = \prod_{j=1}^{J} \left(\sum_{i=1}^{I} \pi_i \mathcal{V}\left(\mathbf{v}_j \left| \frac{\mathbf{S}_i(\mathbf{p}_j)}{\|\mathbf{S}_i(\mathbf{p}_j)\|}, \kappa \right.\right) \right). \tag{7.10}$$

Maximizing Equation (7.10) finds the optimal estimates of $\{Q_i\}_{i=1}^{I}$ that explain the observed primary gaze directions, $\{(\mathbf{v}_j, \mathbf{p}_j)\}_{j=1}^{J}$, given the number of social charges.

7.3.1 Expectation Maximization

An Expectation-Maximization (EM) algorithm allows us to solve this optimization problem. In the expectation step, we estimate the membership of each social charge given the social charge locations, i.e.,

$$\gamma_{ij} = \frac{\pi_i \mathcal{V}\left(\mathbf{v}_j \left| \frac{\mathbf{S}_i(\mathbf{p}_j)}{\|\mathbf{S}_i(\mathbf{p}_j)\|}, \kappa \right.\right)}{\sum_{k=1}^{I} \pi_k \mathcal{V}\left(\mathbf{v}_j \left| \frac{\mathbf{S}_k(\mathbf{p}_j)}{\|\mathbf{S}_k(\mathbf{p}_j)\|}, \kappa \right.\right)}, \tag{7.11}$$

where γ_{ij} is the probability that the j^{th} member looks at the i^{th} social charge. This also allows us to compute the gaze $q_i = \sum_{j=1}^{J} \gamma_{ij}$, i.e., how many members focus on the social charge. In the maximization step, we estimate the social charge locations based on the membership, i.e.,

$$Q_i = \underset{\mathbf{r}_i}{\operatorname{argmin}} \sum_{j=1}^{J} \gamma_{ij}^2 d((\mathbf{v}_j, \mathbf{p}_j), \mathbf{r}_i)^2, \tag{7.12}$$

where $d(\cdot, \cdot)$ is a distance between a ray and point defined as follows,

$$d((\mathbf{v}, \mathbf{p}), \mathbf{x}) = \begin{cases} \frac{\|\mathbf{x} - \widehat{\mathbf{x}}\|}{\mathbf{v}^{\mathsf{T}}(\mathbf{x} - \mathbf{p})} & \text{for } \mathbf{v}^{\mathsf{T}}(\mathbf{x} - \mathbf{p}) \geq 0 \\ \infty & \text{otherwise,} \end{cases} \tag{7.13}$$

where $\widehat{\mathbf{x}} = \mathbf{p} + \mathbf{v}^{\mathsf{T}}(\mathbf{x} - \mathbf{p})\mathbf{v}$ is the projection of $\mathbf{x}$ onto the primary gaze direction. Equation (7.12) estimates the optimal location of Q_i, where the primary gaze directions that belong to Q_i intersect. This is equivalent to the triangulation of a 3D point given 2D projections [56].

For a time-varying gaze field, we can modify the expectation and maximization steps in Equation (7.11) and (7.12) as follows:

$$\begin{aligned} \text{E}: \ \gamma_{ij} &= \frac{\int_{t_e}^{t_d} \pi_i \mathcal{V}\left(\mathbf{v}_j \left| \frac{\mathbf{S}_i(\mathbf{p}_j)}{\|\mathbf{S}_i(\mathbf{p}_j)\|}, \kappa \right.\right) \mathrm{d}t}{\sum_{k=1}^{I} \int_{t_e}^{t_d} \pi_k \mathcal{V}\left(\mathbf{v}_j \left| \frac{\mathbf{S}_k(\mathbf{p}_j)}{\|\mathbf{S}_k(\mathbf{p}_j)\|}, \kappa \right.\right) \mathrm{d}t}, \\ \text{M}: \ Q_i &= \underset{\mathbf{r}_i}{\operatorname{argmin}} \int_{t_e}^{t_d} \sum_{j=1}^{J} \left(\gamma_{ij} d\left((\mathbf{v}_j, \mathbf{p}_j), \mathbf{r}_i\right)\right)^2 \mathrm{d}t + \lambda_g \mathbf{G}(\mathbf{r}_i), \end{aligned} \tag{7.14}$$

where $\mathbf{G}(\cdot)$ is a temporal filter[3] that regularizes the temporal coherency of the social charge and λ_g is a weight on the filter term. Note that emergence and dissolution times, t_e and t_d, are the same for all social charges in Equation (7.14). In practice, we split the time windows such that the number of the social charges remains constant for each time window. This EM method requires prior knowledge of the number social charges and a good initialization of $\{Q_i\}_{i=1}^{I}$. We use the social charge estimation described in Chapter 6.

7.4 Results

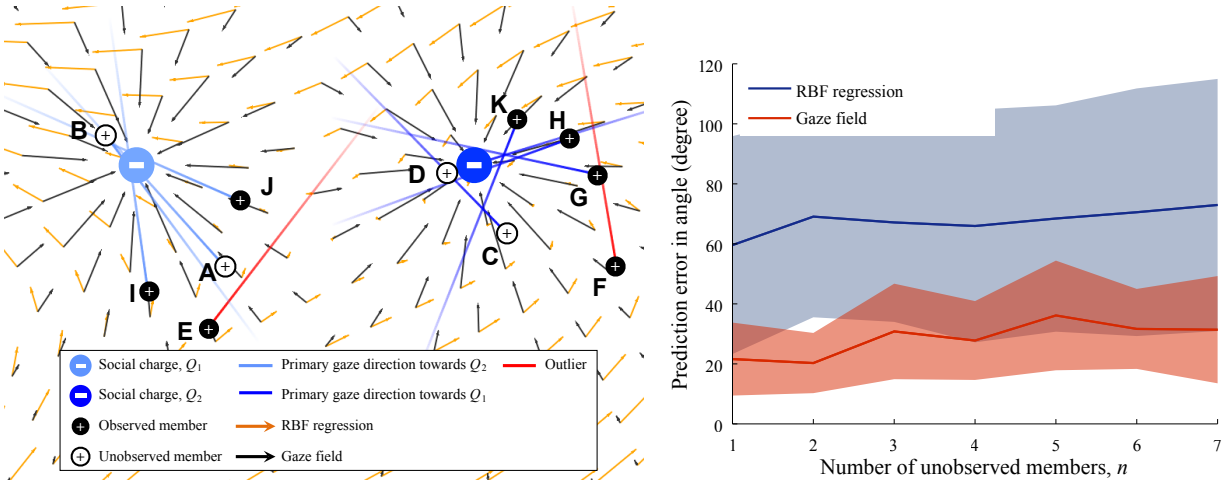
We validate our gaze field model and evaluate the prediction accuracy, quantitatively and qualitatively via four real world sequences capturing various human interactions from scene cameras and social cameras.

7.4.1 Quantitative Evaluation

We validate our gaze prediction via a leave-one-out cross validation on the Meeting sequence in Chapter 6. In the scene, 11 people interact with each other by forming two subgroups. We leave out one of the members and estimate the time-varying social charges from the primary gaze behaviors of the rest of members. Using the estimated social charges, we evaluate the predictive validity of the left-out primary gaze direction. We run this cross validation scheme and measure the angle difference between the predicted gaze direction and the ground truth gaze direction. The mean error is 21.67 degrees with a standard deviation 15.73 degrees. In most cases, our prediction angle error is lower than 30 degrees, which is within the range of eye-in-head motion.

We use a leave-k-out cross validation scheme to compare our gaze prediction against a field generated by Radial Basis Function (RBF) regression. This model was used by Kim et al. [72] to predict players' behaviors in soccer, which directly regresses from the observed directions to the predicted one. We randomly choose k number of members out of 11 members and predict their primary gaze directions using (11-k) number of the primary gaze directions. The orange vector field and dark gray vector field in Figure 7.2(a) are the RBF regression model and a gaze field, respectively. The gaze field outperforms over the RBF regression in three aspects: (1) The gaze field is insensitive to outliers while the RBF regression is often biased by the outliers. For example, prediction at A is highly influenced by the outlier E, which results in inaccurate prediction. (2) The RBF model does not reflect selective gaze behavior. It produces a weighted average vector particularly at extrapolated area (see B, C, and D) that are not necessarily oriented towards a source of attention. (3) The magnitude of the field does not reflect the probability of primary gaze direction. The further from the social charges, the larger vectors that are formed. In Figure 7.2(b), we evaluate prediction error in angle produced by two methods by increasing k. This illustrates that our model produces approximately three times greater average accurate prediction compared to the RBF regression.

[3]For example, if discrete time instances are considered, $\mathbf{G}(\mathbf{r})$ can be $\sum_{t=t_e+1}^{t_d} \|\mathbf{r}(t) - \mathbf{r}(t-1)\|^2$ if one considers minimal displacement of the trajectory, or $\sum_{t=t_e+1}^{t_d-1} \|2\mathbf{r}(t) - \mathbf{r}(t-1) - \mathbf{r}(t+1)\|^2$ if one regularizes acceleration [136].



(a) Gaze field (b) Comparison

Figure 7.2: (a) We compare our model against RBF regression [72]. We estimate the social charges from randomly chosen observed members (E to J) and predict the primary gaze directions of the unobserved members (A to D). The gaze field model shows superior predictive precision. For instance, the outlier E or F does not contribute to estimate the field while the RBF regression model produces inaccurate estimation at A. Also our model is insensitive to the spatial distribution of the observed members while the RBF prediction is not reliable at extrapolated points such as B, C, or D. (b) We evaluate predictive validity using cross validation as the number of members decreases. Our gaze field model produces lower error with less standard deviation.

7.4.2 Qualitative Evaluation

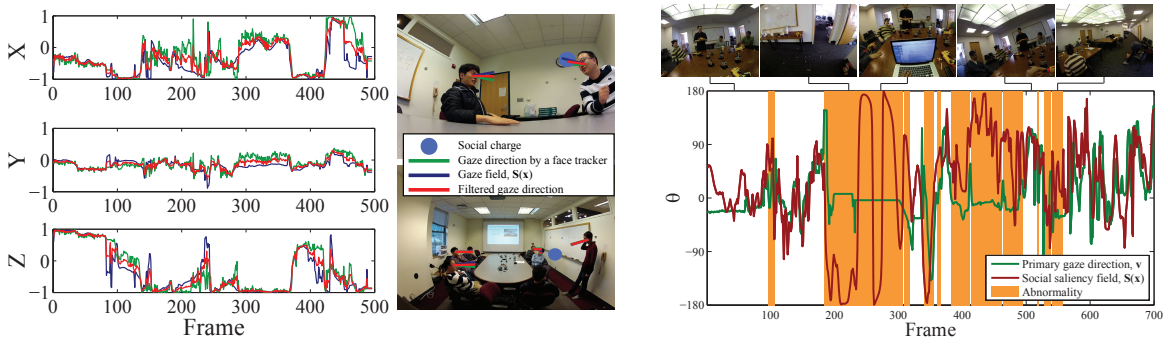
We applied our algorithm on two datasets of Chapter 6. Two sequences (Party and Meeting) are used to estimate the gaze field as shown in Figure 7.4(c) and 7.4(d). These results are best seen in the supplementary video. As a proof-of-concept, we also applied our the social field model as prior within a simple filtering framework for tracking and for anomaly detection.

Tracking: We collected data from a meeting scene where 7 people including a presenter were engaged in a discussion. We instrumented 17 cameras in a meeting scene and calibrated the cameras using structure-from-motion. We used the method described in Section 3.2.2 to reconstruct the primary gaze directions and we generated a gaze field as shown in Figure 7.4(a).

We used the gaze field as a naive prior for tracking 3D facial pose. We estimated social charge motion from other members and fused gaze prediction by the gaze field with the face orientation estimate at each frame from the PittPatt system. We average out these two measurements to get the filtered direction. Figure 7.3(a) shows that the noisy face tracking measurements[4] can be regularized by our gaze prediction.

Anomaly Detection: We captured data of 8 people playing a social game called *Mafia*. GoPro Hero3 Black Edition cameras were mounted on each player and calibrated by structure from motion. While they interrogated each other during the game, the social charge stays in the group. Once a particular player is identified as a mafia, the player no longer stays in the group. In

[4]Face pose tracking from a scene camera is noisy when the face is not directly oriented to the camera.



(a) 3D Face tracking

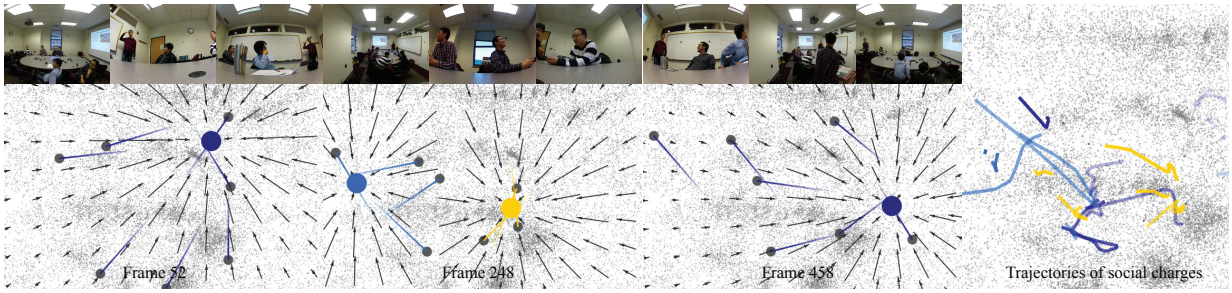
(b) Anomaly detection

Figure 7.3: (a) Our gaze prediction method can be used as a filter for a face tracking task. We exploit social charge motion estimated by other members to regulate the noisy face tracking process. (b) We detect anomalies in the scene based on social attention. A member who is not involved in any common social activity is classified as an outlier.

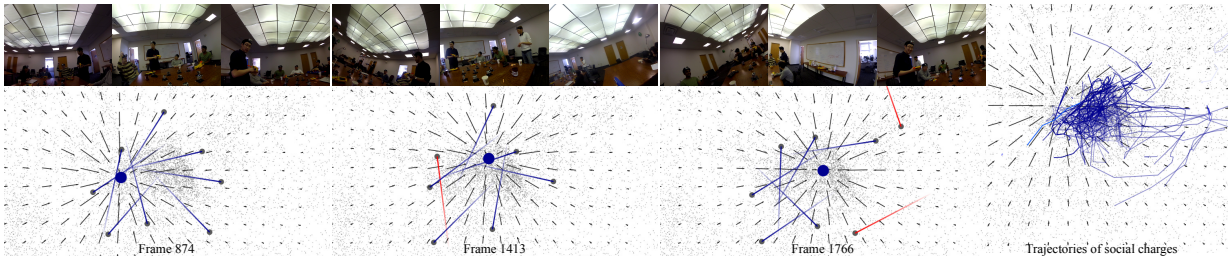
Figure 7.4(b), we estimate the social charge motion. In most cases, the social charge stays near the player who is investigated. Based on the gaze field, we show that we can detect the outliers whose primary gaze direction does not behave in accordance with social attention. This results in the detection of anomalous behavior, as shown in Figure 7.3(b). These outliers are the players who are not pay attention to the game.

7.5 Summary

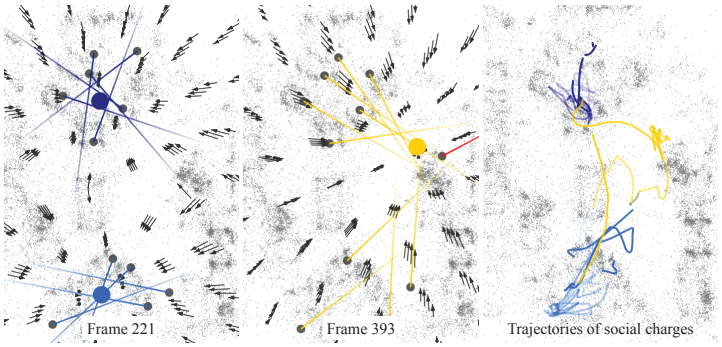
We present the gaze field induced by the motion of social charges as a model to predict primary gaze behavior of people in a social scene. The motion of the charges is estimated from the observed primary gaze behavior of members of a social scene. The net gaze field is created by selecting the maximum of a mixture of von Mises-Fisher distributions, each produced by a different social charge. We evaluate the predictive validity of spatial and temporal forecasting on real sequences and demonstrate that the gaze field model is supported empirically.



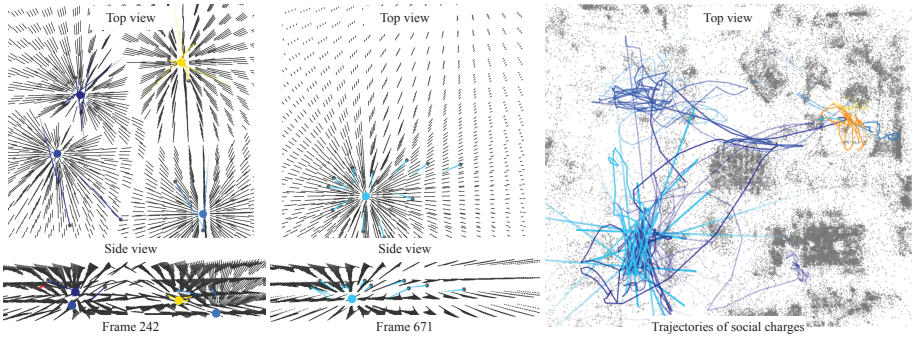
(a) Scene cameras



(b) Social cameras



(c) Meeting scene



(d) Party scene

Figure 7.4: We estimate a gaze field from both scene cameras and social cameras. (a) A social charge is formed at the presenter and splits into two subgroups at frame 248 in the meeting scene. The gaze field reflects the selective gaze behavior. (b) 8 members in the scene play the social game called mafia with social cameras. Our method can correctly detect anomalies (the red rays) based on social attention.

Chapter 8

Discussion

8.1 Summary

For artificial agents to behave in a socially acceptable manner, it is necessary for them to be equipped with computational social cognition—the ability to understand social scenes by perceiving, modeling, and predicting social signals. Developing such social cognition is challenging as the core attributes of social interaction such as attention, intent, and emotion are latent quantities and because social behaviors are interdependent. These challenges preclude the application of existing computational frameworks, e.g., structure from motion [133], activity recognition [168], and human affordance identification [51]. In this thesis, we present a representation of social signals in a unified 3D coordinate system and develop a relational/predictive model of social behaviors from social cameras.

8.1.1 Social Signal Reconstruction

We present a method to reconstruct social signals in a unified 3D coordinate system from social cameras. This 3D reconstruction provides a computational basis for analyzing social behaviors. We focus on three social signals that are frequently transmitted during social interactions: primary gaze direction, human body motion, and general scene motion. Gaze is a key social signal that conveys one's attention. We model the gaze using the primary gaze direction that emanates from the center of the eyes and aligned with the head orientation. Based on the model, we demonstrate two approaches to estimate primary gaze direction by leveraging ego- and exo-motion of social cameras. As human body motion, such as a gesture, is a strong signal that encodes the intent in social interactions, we present a method to reconstruct human body motion modeled by a set of 3D articulated trajectories, i.e., the distance between two adjacent joints remains constant across time. Spatial and temporal constraints are simultaneously applied for the articulated trajectories and we show that reconstructing human body motion is equivalent to solving a binary quadratic programming problem. We further relax the articulation constraint to reconstruct general scene motion without a spatial prior. We model a trajectory using a linear combination of trajectory basis vectors, that results in a least squares system for the trajectory parameters. This allows us to represent topology independent scene motion.

8.1.2 Social Behavior Understanding

Gaze is a prominent social signal that exhibits one's attention [35] and we study the relationship between joint attention and gaze behaviors. We model joint attention using hypothetical social charges that form where the gaze rays of members in a social group intersect in 3D. Using the primary gaze direction represented in a unified 3D coordinate system, we present a method to estimate the number, locations, and magnitudes of social charges. We construct a social saliency field by superimposing the gaze models in 3D and estimate social charges in that field via mode-seeking. A membership feature that encodes the members who pay attention to each social charge is used to find temporal association between different time instant, which allows us to reconstruct temporally consistent social charges. Based on this social charge representation, we build a relational model of attentive social behaviors, which enables us to predict gaze direction of social members. Inspired by the study of electric fields, we model the relationship between gaze behaviors using a gradient field induced by social charges. This gradient field aligns with the gaze direction and encodes its likelihood. Given multiple social charges, we model the likelihood of the gaze direction using a mixture of von Mises-Fisher distributions and predict gaze direction at any location and time that maximizes the likelihood. This prediction can be used for social anomaly detection that finds members who do not pay attention to a social interaction and an estimation filter to efficiently track gaze behavior in social scenes.

8.2 Limitation

Our approach to develop computational social cognition has the following limitataions:

Manual correspondence: In our analysis on human body motion and general scene motion, the algorithms assume that the correspondences of moving points are given. We manually specified point correspondences across images for our experiments. From a practical stand point, this is undesirable. However, as camera optics and sensors improve, and more sophisticated point correspondence methods are developed, the ability to automatically obtain correspondences will likely become achievable. Also as the number of social cameras increases, denser camera placement will make standard feature matching frameworks applicable due to reduced baseline as demonstrated by Joo et al. [67]. In conjunction, feature descriptors that are more robust to viewpoint changes such as HOG (Histogram of Oriented Gradients) [32], can facilitate matching across wide baseline. The anatomic annotation of joints of human body structure can exploit a semantic labeling framework based on articulated deformable parts models [166].

Limited precision of gaze pose estimation: Our gaze representation is mainly driven by the head orientation. Eye-in-head motion is encoded in the form of a probabilistic function in the model given primary gaze direction but we do not explicitly measure the eye-in-head motion due to sensor limitation. For precise gaze estimation, eye-in-head motion must be measured using an additional eye facing camera integrated in head-mounted devices [62, 77, 137]. Furthermore, a pair of the eye facing cameras will allow us to estimate the point of regard in 3D. These enhanced sensors can be complementary to our social cameras, which will capture higher frequency shifts in attention.

Limited expressivity of social charge model: When we estimate social charges, the esti-

mation becomes poorly conditioned if people's gaze rays are almost parallel such as the musical scene (Figure 6.5). The confidence region is stretched along the direction of the primary gaze rays. This is the case where the point of regard is very far away while people look at the point from almost the same vantage point. For such a scene, gaze directions from different points of views can help to localize the social charges precisely. Also, the principal assumption in the gaze field model is the conditional independence of gaze behavior between two observers given the behavior of the social charges. In practice, the gaze behavior of each observer in the scene is known to have a degree of influence on the gaze behavior of other observers [35, 114].

8.3 Future Work

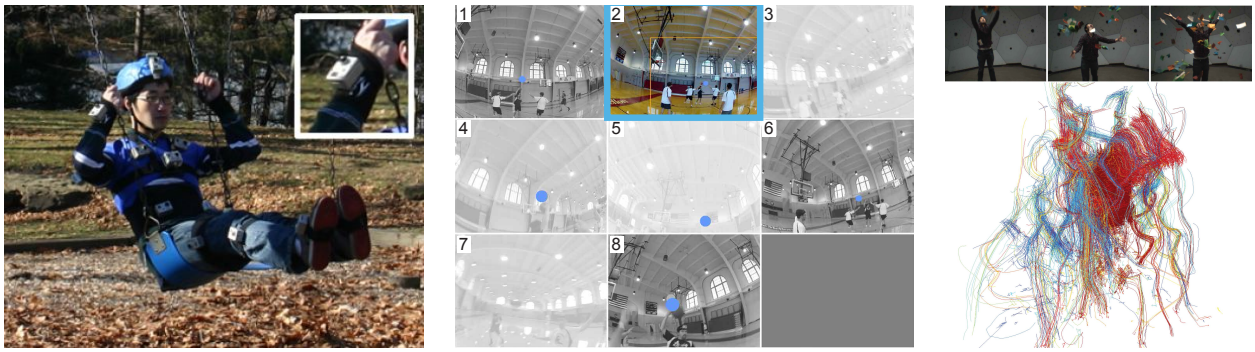
Our thesis takes a first step towards developing computational social cognition for artificial agents. This opens a number of new problems in computer vision, robotics, graphics, and artificial intelligence and we consider 5 future research tasks:

A) To reconstruct subtle social signals or intent of interactions: *Subtle* social signals such as an instant cynical smile, small nod, or finger gesture often convey an important intent or message during human interactions while machines are blind to them. We aim to reconstruct such subtle signals in detail from social cameras as demonstrated by dense motion capture in the Panoptic Studio [128]. The main challenge of subtle motion capture is to establish correspondences of moving points across cameras because baselines between these cameras are usually larger than cameras in the Panoptic Studio. We are interested in studying the relationship between reconstruction quality or ambiguity and camera motion, which will characterize desired camera motion. When desired camera motion is not provided, reconstructing subtle motion can benefit from data captured in the Panoptic Studio. We also plan on investigating spatiotemporal features tailored for human interactions.

B) To reason about the relationship of general social signals: In conjunction with primary gaze direction, other social signals, such as facial expressions and body gestures, provide a coherent context of social interactions. A relational analysis on such signals will allow us to build a richer predictive model beyond joint attention and to infer different social attributes, i.e., intent and emotion. Also understanding a correlation between multimodal social signals by the same social member is a key component of behavioral analysis. For instance, a hand gesture and speech are often correlated during conversation. This analysis will build a new connection between different social signals and enable artificial agents to classify socially important signals.

C) To predict social affordance: Social affordance of space is another relationship of social signals with respect to environments. Some scene structures such as chair, table, or sofa are strongly related to human interactions and the spatial arrangement of such structures characterizes social space. For example, human interactions are more likely to take place at a sofa in a lobby than a corner of wall or under the table. By leveraging the gaze field model based on social charges, we will study how the social interactions are spatially related to 3D scene structure and measure its social affordance. This will provide a richer predictive model of the human interactions. Architect can benefit from this social affordance because it will quantify how the space is social-friendly.

D) To model information dynamics: As gaze direction serves as a channel of communica-



(a) Motion capture (b) Video editing (c) Dense reconstruction

Figure 8.1: (a) We exploit body-mounted cameras to capture motion of a subject. (b) Social charges are used to define the content of footage of social cameras. Based on the content, we present a method to create a representative video. Blue points are social charges and colored image is the selected camera. (c) A large number cameras are used to reconstruct dense motion trajectories in 3D.

tion, information flows along the gaze direction, i.e., humans transmit and receive information well in the particular direction that aligns with their gaze. This states that a social charge has directionality: its influence varies in direction. The directional representation of social charges will allow us to understand information dynamics, which studies a transient behavior of information between social members across time.

E) To build a multi-agent system that reacts according to social interactions: Social cognition enables machines to use social signals in their tasks. We will design a multi-agent system that fully exploits computational social cognition. The predictive model of gaze behaviors provides how humans will behave at the location and time. Using this predictive model, we plan on creating an automatic broadcasting/capturing system as demonstrated by Arev et al. [8] using small navigational robot platforms such as manipulators, wheeled mobile robots, or quadroters for sports, search and rescue, and medical scenes. Sources of attention will be estimated by measuring gaze directions of social members and the paths of the robots will be planned based on the predictive model.

8.4 Broad Impact

We have exploited our work on computational social cognition for different domains of research including computer vision and graphics.

A) Motion capture from body-mounted cameras [129]: We use body-mounted cameras (social cameras) to reconstruct the motion of a subject as shown in Figure 8.1(a). Outward-looking cameras are attached to the limbs of the subject, and the joint angles and root pose are estimated through non-linear optimization. The optimization objective function incorporates terms for image matching error and temporal continuity of motion. Structure from motion is used to estimate the skeleton structure and to provide initialization for the non-linear optimiza-

tion procedure. This work demonstrates a new way of reconstructing social signals from social cameras.

B) Automatic editing of footage from social cameras [8]: We leverage gaze behaviors encoded in social cameras for editing their footage to create a coherent video as shown in Figure 8.1(b). We use social charges to determine where the important "content" in a scene is taking place, and use it in conjunction with cinematographic guidelines to select which cameras to cut to and to determine the timing of those cuts. A trellis graph formulation is used to optimize an objective function that maximizes coverage of the important content in the scene, while respecting cinematographic guidelines such as the 180-degree rule and avoiding jump cuts. This work validates our predictive model of social charges by showing that the social charges can approximate the content of videos by social cameras in real-world scenes.

C) Large-scale dynamic scene reconstruction [67]: We present an algorithm to reconstruct the 3D motion of an event from a large number of videos as shown in Figure 8.1(c). We demonstrate that the key problem of large-scale dynamic reconstruction is the time-varying visibility of each 3D point. Appearance, motion, and normal cues are used to model the likelihoods of visibility of each camera and an optimal estimate of visibility is obtained by graph cuts in Markov Random Field. This work shows an ideal framework to analyze social interactions fully automatically as subtle social signals can be reconstructed by exploiting a large number of cameras.

Appendices

Appendix A

Social Camera Pose Estimation

Social cameras are embedded in social scenes and capture social interactions. They are ideal sensors to capture the social scene as they follow the gaze direction of the camera holders or wearers. In this section, we review a framework to reconstruct 3D social camera poses using structure from motion. The reconstructed social cameras enable us to represent social behaviors in a unified coordinate system and analyze the relationship between the social behaviors.

A.1 3D Geometry of Point and Camera

A 3D point is projected onto a camera plane to form the 2D projection as shown in Figure A.1(a). The projection can be written as,

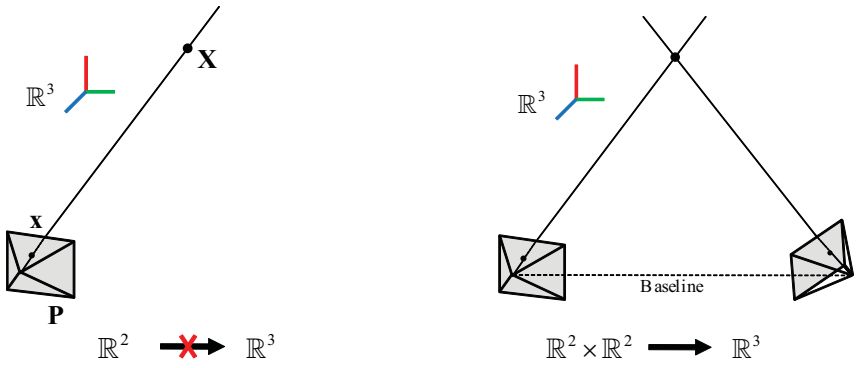
$$\lambda \begin{bmatrix} \mathbf{x} \\ 1 \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{X} \\ 1 \end{bmatrix}, \tag{A.1}$$

where λ is a scalar, $\mathbf{X} \in \mathbb{R}^3$ is a 3D point, $\mathbf{x} \in \mathbb{R}^2$ is the projected 2D point, and $\mathbf{P} \in \mathbb{R}^{3\times4}$ is a camera projection matrix. The camera projection matrix can be written as $\mathbf{P} = \mathbf{K}\mathbf{R} \begin{bmatrix} \mathbf{I}_3 - \mathbf{C} \end{bmatrix}$ where $\mathbf{I}_3$ is a 3 by 3 identity matrix, $\mathbf{R} \in \mathrm{SO}(3)$ is a camera rotation matrix, and $\mathbf{C} \in \mathbb{R}^3$ is a 3D camera center vector. $\mathbf{R}$ and $\mathbf{C}$ are called camera extrinsic parameters. $\mathbf{K}$ is a matrix of camera intrinsic parameters written as,

$$\mathbf{K} = \begin{bmatrix} f_x & 0 & p_x \\ 0 & f_y & p_y \\ 0 & 0 & 1 \end{bmatrix}, \tag{A.2}$$

where f_x and f_y are the focal lengths of the camera, and p_x and p_y are the image origin coordinates.

By the projection, one dimensional information is lost; there exist an infinite number of 3D points that satisfy the image measurement, $\mathbf{x}$. Any 3D point on the line between $\mathbf{x}$ and $\mathbf{X}$ projects into the $\mathbf{x}$ as shown in Figure A.1(a). Therefore, given a single 2D image measurement, estimating the 3D point is impossible without prior assumptions about the scene. Structure from motion exploits multiple images to reconstruct 3D camera poses and points as shown in Figure A.1(b).



(a) Single image (b) Structure from motion

Figure A.1: (a) Given $\mathbf{x}$, estimating $\mathbf{X}$ from a single image is fundamentally ambiguous because there are infinite number of 3D points that project to $\mathbf{x}$. (b) From two views, the 3D point can be triangulated without ambiguity.

When a scene is static, a point correspondence $\mathbf{x}_1 \leftrightarrow \mathbf{x}_2$ from image 1 and 2, respectively, satisfies the following epipolar constraint:

$$\begin{bmatrix} \mathbf{x}_1^{\mathsf{T}} & 1 \end{bmatrix} \mathbf{E} \begin{bmatrix} \mathbf{x}_2 \\ 1 \end{bmatrix} = 0, \tag{A.3}$$

where $\mathbf{E} \in \mathbb{R}^{3\times3}$ is a rank 2 essential matrix [79]. This essential matrix encodes the 3D relative transform (rotation and translation) between two images. The camera poses can be extracted from the essential matrix in 3D and then, the 3D points can be reconstructed by triangulation. The detailed descriptions of the reconstruction can be found in [56].

The 3D reconstructed cameras and points can be refined by bundle adjustment [145] that minimizes reprojection error,

$$\underset{\{\mathbf{P}_i\}_{i=1,\cdots,F},\{\mathbf{X}_i\}_{j=1,\cdots,P}}{\operatorname{argmin}} \sum_{i=1}^{F}\sum_{j=1}^{P} v_{i,j} \left(\left(\frac{\mathbf{P}_i^1 \mathbf{X}_j}{\mathbf{P}_i^3 \mathbf{X}_j} - \mathbf{x}_{i,j}^1 \right)^2 + \left(\frac{\mathbf{P}_i^2 \mathbf{X}_j}{\mathbf{P}_i^3 \mathbf{X}_j} - \mathbf{x}_{i,j}^2 \right)^2 \right), \tag{A.4}$$

where $\mathbf{P}^k$ represents the k^{th} row of $\mathbf{P}$ and $\mathbf{x}^1$ and $\mathbf{x}^2$ are the first and second elements of $\mathbf{x}$. F and P are the number of cameras and points, respectively. $v_{i,j}$ is a binary variable for visibility, i.e., if $\mathbf{X}_j$ is visible to $\mathbf{P}_i$, then $v_{i,j} = 1$, and otherwise zero.

A.2 Pose Estimation in Practice

Given all images taken by social cameras, we apply structure from motion to estimate the social camera poses. We extract SIFT keypoints [81] and find matches between all possible pairs of images using the essential matrix. The RANSAC [40] based matching enables us to automatically obtain scene correspondences of static points. These correspondences are used to estimate camera poses using structure from motion with incremental bundle adjustment [133] to the image collection. From the first image pair, relative camera pose is estimated from the essential matrix,

and then the static correspondences are triangulated. To estimate an additional camera pose, we compare the keypoints registered in 3D with new keypoints observed by the target camera and apply a perspective-n-point algorithm [90] to estimate the camera pose. If there are unregistered keypoints which are also visible from any of the registered cameras, their 3D locations are estimated through triangulation. This procedure is repeated until no image remains. Camera poses and static structures are also refined by sparse bundle adjustment [80] at each time a new camera is registered.

Appendix B

Proof of Theorems

B.1 Coordinate Independence

To prove Result 1 in Section 5.1, we need to show that the transformed trajectory basis, $\mathbf{S}(\mathbf{\Theta})$, span the same space spanned by the original trajectory basis vectors where $\mathbf{S}(\cdot)$ is a similarity transformation, i.e., $\mathrm{col}(\mathbf{S}(\mathbf{\Theta})) = \mathrm{col}(\mathbf{\Theta})$ where $\mathrm{col}(\mathbf{\Theta})$ is a space spanned by the column space of $\mathbf{\Theta}$.

Proof. (*i*) scale: $\mathrm{col}(s\mathbf{\Theta}) = \mathrm{col}(\mathbf{\Theta})$ where s is a scalar.
(*ii*) translation: translation is spanned by the DC component of $\mathbf{\Theta}_{\mathrm{DCT}}$.
(*iii*) rotation: without loss of generality, the trajectory basis can be rearranged as $\bar{\mathbf{\Theta}} = \mathrm{blkdiag}\{\boldsymbol{\theta}, \boldsymbol{\theta}, \boldsymbol{\theta}\}$ where $\boldsymbol{\theta} \in \mathbb{R}^{F \times K}$ is the DCT trajectory basis for each trajectory. The rotated trajectory basis, $(\mathbf{R} \otimes \mathbf{I}_F)\bar{\mathbf{\Theta}}$ span the original trajectory basis vectors $\bar{\mathbf{\Theta}}$ because,

$$
\begin{aligned}
& \mathrm{col}\left((\mathbf{R} \otimes \mathbf{I}_F)\bar{\mathbf{\Theta}}\right) \\
= \; & \mathrm{col}\left(\begin{bmatrix} R_{11}\mathbf{I}_F & R_{12}\mathbf{I}_F & R_{13}\mathbf{I}_F \\ R_{21}\mathbf{I}_F & R_{22}\mathbf{I}_F & R_{23}\mathbf{I}_F \\ R_{31}\mathbf{I}_F & R_{32}\mathbf{I}_F & R_{33}\mathbf{I}_F \end{bmatrix} \begin{bmatrix} \boldsymbol{\theta} & & \\ & \boldsymbol{\theta} & \\ & & \boldsymbol{\theta} \end{bmatrix}\right) \\
= \; & \mathrm{col}\left(\begin{bmatrix} R_{11}\boldsymbol{\theta} & R_{12}\boldsymbol{\theta} & R_{13}\boldsymbol{\theta} \\ R_{21}\boldsymbol{\theta} & R_{22}\boldsymbol{\theta} & R_{23}\boldsymbol{\theta} \\ R_{31}\boldsymbol{\theta} & R_{32}\boldsymbol{\theta} & R_{33}\boldsymbol{\theta} \end{bmatrix}\right) \\
= \; & \mathrm{col}\left(\begin{bmatrix} \boldsymbol{\theta} & & \\ & \boldsymbol{\theta} & \\ & & \boldsymbol{\theta} \end{bmatrix} \begin{bmatrix} R_{11}\mathbf{I}_K & R_{12}\mathbf{I}_K & R_{13}\mathbf{I}_K \\ R_{21}\mathbf{I}_K & R_{22}\mathbf{I}_K & R_{23}\mathbf{I}_K \\ R_{31}\mathbf{I}_K & R_{32}\mathbf{I}_K & R_{33}\mathbf{I}_K \end{bmatrix}\right) \\
= \; & \mathrm{col}\left(\bar{\mathbf{\Theta}}(\mathbf{R} \otimes \mathbf{I}_K)\right) \\
= \; & \mathrm{col}(\bar{\mathbf{\Theta}})
\end{aligned}
$$

where $\otimes$ is the Kronecker product, $\mathbf{R}$ is a 3×3 rotation matrix and, $\mathbf{I}_K$ is a $K \times K$ identity matrix. □

B.2 Unsolvable Systems

Proof. (*i*) If $\mathbf{X}, \mathbf{C} \in \mathrm{col}(\mathbf{\Theta})$, $\mathbf{X} = \mathbf{\Theta}\boldsymbol{\beta}_{\mathbf{X}}$ and $\mathbf{C} = \mathbf{\Theta}\boldsymbol{\beta}_{\mathbf{C}}$. Then,

$$
\begin{aligned}
& \mathrm{null}(\mathbf{Q}\mathbf{\Theta}) \\
= \; & \mathrm{null}\left(\begin{bmatrix} [\mathbf{X}_1 - \mathbf{C}_1]_\times & & \\ & \ddots & \\ & & [\mathbf{X}_F - \mathbf{C}_F]_\times \end{bmatrix} \begin{bmatrix} \mathbf{\Phi}_1 \\ \vdots \\ \mathbf{\Phi}_F \end{bmatrix}\right) \\
= \; & \mathrm{null}\left(\begin{bmatrix} [\mathbf{\Phi}_1(\boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}})]_\times & & \\ & \ddots & \\ & & [\mathbf{\Phi}_F(\boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}})]_\times \end{bmatrix} \begin{bmatrix} \mathbf{\Phi}_1 \\ \vdots \\ \mathbf{\Phi}_F \end{bmatrix}\right) \\
= \; & \mathrm{null}\left(\begin{bmatrix} [\mathbf{\Phi}_1(\boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}})]_\times \mathbf{\Phi}_1 \\ \vdots \\ [\mathbf{\Phi}_F(\boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}})]_\times \mathbf{\Phi}_F \end{bmatrix}\right) \\
\ni \; & \boldsymbol{\beta}_{\mathbf{X}} - \boldsymbol{\beta}_{\mathbf{C}},
\end{aligned} \tag{B.1}
$$

where $\mathbf{\Theta} = \begin{bmatrix} \mathbf{\Phi}_1^\mathsf{T} & \cdots & \mathbf{\Phi}_F^\mathsf{T} \end{bmatrix}^\mathsf{T}$. Since there exists a null space of $\mathbf{Q}\mathbf{\Theta}$, $\mathrm{rank}(\mathbf{Q}\mathbf{\Theta}) < 3K$.

(*ii*) Let us consider two cases where $c \neq 1$ and $c = 1$.

When $c \neq 1$, by plugging $\mathbf{X} = c\mathbf{C} + \mathbf{1} \otimes \mathbf{d}$ into the Equation (5.12), it becomes,

$$
\begin{aligned}
[(c-1)\mathbf{C}_i + \mathbf{d}]_\times \widehat{\mathbf{X}}_i &= [c\mathbf{C}_i + \mathbf{d}]_\times \mathbf{C}_i \\
&= [\mathbf{d}]_\times \mathbf{C}_i.
\end{aligned} \tag{B.2}
$$

From Equation (B.2), $\widehat{\mathbf{X}}_i = \alpha\mathbf{C}_i + (1-\alpha)\mathbf{d}/(1-c)$ where α is a scalar. When $\mathbf{C} \in \mathrm{col}(\mathbf{\Theta})$, it is the case where the first condition (*i*) holds, where the system is unsolvable. When $\mathbf{C} \notin \mathrm{col}(\mathbf{\Theta})$, $\alpha = 0$ because any component of $\mathbf{C}$ that cannot be expressed by the trajectory basis vectors results in the residual error of Equation (5.3). Only $\mathbf{1} \otimes \mathbf{d}/(1-c)$ nullifies the residual error of Equation (5.7) but it is still a trivial solution (i.e., a reconstructed trajectory, $\widehat{\mathbf{X}} = \mathbf{1} \otimes \mathbf{d}/(1-c)$, is simply a stationary point even though the point undergoes motion.).

When $c = 1$, $\mathbf{d}/(1-c)$ term in $\widehat{\mathbf{X}}_i = \alpha\mathbf{C}_i + (1-\alpha)\mathbf{d}/(1-c)$ is indeterminate. It is the case where the camera moves exactly the same way the point moves with some offset and

$\mathrm{rank}(\mathbf{Q}\boldsymbol{\Theta}) = 2K$ because from Equation (5.12) and $\mathbf{X} = \mathbf{C} + \mathbf{1} \otimes \mathbf{d}$,

$$
\begin{aligned}
& \mathrm{rank}(\mathbf{Q}\boldsymbol{\Theta}) \\
= \; & \mathrm{rank}\left(\begin{bmatrix} [\mathbf{d}]_\times & & \\ & \ddots & \\ & & [\mathbf{d}]_\times \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_1 \\ \vdots \\ \boldsymbol{\Phi}_F \end{bmatrix} \right) \\
= \; & \mathrm{rank}\left(\begin{bmatrix} \mathbf{0} & -d_3\theta_1 & d_2\theta_1 \\ d_3\theta_1 & \mathbf{0} & -d_1\theta_1 \\ \vdots & \vdots & \vdots \\ \mathbf{0} & -d_3\theta_F & d_2\theta_F \\ d_3\theta_F & \mathbf{0} & -d_1\theta_F \end{bmatrix} \right) \\
= \; & \mathrm{rank}\left(\begin{bmatrix} \mathbf{0} & -d_3\theta_1 & d_2\theta_1 \\ \vdots & \vdots & \vdots \\ \mathbf{0} & -d_3\theta_F & d_2\theta_F \end{bmatrix} \right) + \mathrm{rank}\left(\begin{bmatrix} d_3\theta_1 & \mathbf{0} & -d_1\theta_1 \\ \vdots & \vdots & \vdots \\ d_3\theta_F & \mathbf{0} & -d_1\theta_F \end{bmatrix} \right) \\
= \; & 2K,
\end{aligned}
$$

where $\mathbf{d} = \begin{bmatrix} d_1 & d_2 & d_3 \end{bmatrix}^\mathsf{T}$ and $\boldsymbol{\Phi}_i = \mathrm{blkdiag}\{\theta_i, \theta_i, \theta_i\}$. The trajectory basis vectors for each coordinate ($x, y,$ and z) are the same. Since the rank of the system is $2K$, the system is unsolvable. □

B.3 Reconstructability

Proof. From the triangle inequality, a square root of the objective function of Equation (5.15) is bounded by (when $\left\|\boldsymbol{\Theta}^\perp \boldsymbol{\beta}_\mathbf{X}^\perp\right\| \to 0$),

$$
\begin{aligned}
& \left\| \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{X} - (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{C} - \mathbf{A}\boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{X}^\perp - (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{C}^\perp \right\| && \text{(B.3)} \\
\leq \; & \left\| \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{X} - (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{C} \right\| + \left\| \mathbf{A}\boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{X}^\perp \right\| + \left\| (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{C}^\perp \right\| \\
\leq \; & \left\| \boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{C}^\perp \right\| \left(\frac{\left\| \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{X} - (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{C} \right\|}{\left\| \boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{C}^\perp \right\|} + \frac{\|\mathbf{A}\|}{\eta} + \|\mathbf{I}-\mathbf{A}\| \right), && \text{(B.4)}
\end{aligned}
$$

or when $\left\|\boldsymbol{\Theta}^\perp \boldsymbol{\beta}_\mathbf{C}^\perp\right\| \to \infty$,

$$
\leq \; \left\| \boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{X}^\perp \right\| \left(\frac{\left\| \boldsymbol{\Theta}\widehat{\boldsymbol{\beta}} - \mathbf{A}\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{X} - (\mathbf{I}-\mathbf{A})\,\boldsymbol{\Theta}\boldsymbol{\beta}_\mathbf{C} \right\|}{\left\| \boldsymbol{\Theta}^\perp\boldsymbol{\beta}_\mathbf{X}^\perp \right\|} + \|\mathbf{A}\| + \|\mathbf{I}-\mathbf{A}\|\,\eta \right). \qquad \text{(B.5)}
$$

As η approaches infinity, $\|\mathbf{A}\|/\eta$ in Equation (B.4) becomes zero or $\|\mathbf{I}-\mathbf{A}\|\,\eta$ in Equation (B.5) becomes infinity. In order to minimize either Equation (B.4) or Equation (B.5), $\mathbf{A} = \mathbf{I}$ because it leaves the last term zero and $\widehat{\boldsymbol{\beta}} = \boldsymbol{\beta}_\mathbf{X}$ because it cancels the first term. This causes the minimum of Equation (B.4) or Equation (B.5) to become zero, which upper-bounds the minimum of Equation (B.3). Thus, as η approaches infinity, $\widehat{\boldsymbol{\beta}}$ approaches $\boldsymbol{\beta}_\mathbf{X}$. □

Bibliography

[1] A. Agarwal and B. Triggs. Recovering 3D human pose from monocular images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006.

[2] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Nonrigid structure from motion in trajectory space. In *Advances in Neural Information Processing Systems*, 2008.

[3] I. Akhter, Y. Sheikh, and S. Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[4] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[5] K. Allemand, K. Fukuda, T. M. Liebling, and E. Steiner. A polynomial case of unconstrained zero-one quadratic optimization. *Mathematical Programming*, 2001.

[6] M. Andriluka, S. Roth, and B. Schiele. Monocular 3D pose estimation and tracking by detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[7] D. E. Angelaki and B. J. M. Hess. Control of eye orientation: where does the brain's role end and the muscle's begin? *European Journal of Neuroscience*, 2004.

[8] I. Arev, H. S. Park, Y. Sheikh, J. Hodgins, and A. Shamir. Automatic Editing of Footage from Multiple Social Cameras . *ACM Transactions on Graphics (SIGGRAPH)*, 2014.

[9] S. Avidan and A. Shashua. Trajectory triangulation: 3D reconstruction of moving points from a monocular image sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.

[10] P. Ballard and G. C. Stockman. Controlling a computer via facial aspect. *IEEE Transactions on Systems, Man and Cybernetics*, 1995.

[11] F. Barahona. A solvable case of quadratic 0-1 programming. *Discrete Applied Mathematics*, 1986.

[12] C. Barron and I. A. Kakadiaris. Estimating anthropometry and pose from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[13] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. I. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *Proceedings of the IEEE Conference on Computer*

Vision and Pattern Recognition, 2008.

[14] L. Bazzani, D. Tosato, M. Cristani, M. Farenzena, G. Pagetti, G. Menegaz, and V. Murino. Social interactions by visual focus of attention in a three-dimensional environment. *Expert Systems*, 2011.

[15] M. Bergtholdt, J. Kappes, S. Schmidt, and C. Schnörr. A study of parts-based object class detection using complete graphs. *International Journal of Computer Vision*, 2010.

[16] M. Bernhard, E. Stavrakis, and M. Wimmer. An empirical pipeline to derive gaze prediction heuristics for 3D action games. *ACM Transactions on Applied Perception*, 2010.

[17] N. Bilton. Behind the google goggles, virtual reality. *The New York Times*, Febrary 2012.

[18] E. Birmingham and A. Kingstone. Human social attention. *Brain Research*, 2009.

[19] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. *ACM Transactions on Graphics (SIGGRAPH)*, 1999.

[20] M. Brand. Morphable 3D models from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[21] M. Brand. A direct method for 3D factorization of nonrigid motion observed in 2D. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[22] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1999.

[23] M. A. Brubaker and D. J. Fleet. The kneed walker for human pose tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[24] K. Chawarska, S. Macari, and F. Shic. Decreased spontaneous attention to social scenes in 6-month-old infants later diagnosed with autism spectrum disorders. *Biological Psychiatry*, 2013.

[25] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995.

[26] W. Choi, K. Shahid, and S. Savarese. Learning context for collective activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[27] K. Choo and D. J. Fleet. People tracking using hybrid monte carlo filtering. In *Proceedings of the International Conference on Computer Vision*, 2005.

[28] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002.

[29] M. Cristani, 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. In *Proceedings of the British Machine Vision Conference*, 2011.

[30] M. Cristani, G. Paggetti, A. Vinciarelli, L. Bazzani, G. Menegaz, and V. Murino. Towards computational proxemics: Inferring social relations from interpersonal distances. In *Proceedings of IEEE International Conference on Social Computing*, 2011.

[31] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure-from-motion factorization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[32] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[33] A. Del Bue. A factorization approach to structure from motion with shape priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.

[34] A. Del Bue, X. Llad, and L. Agapito. Non-rigid metric shape and motion recovery from uncalibrated images using priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[35] N. Emery. The eyes have it: the neuroethology, function and evolution of social gaze. *Neuroscience and Biobehavioral Reviews*, 2000.

[36] A. Fathi, J. K. Hodgins, and J. M. Rehg. Social interactions: A first-person perspective. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[37] O. Faugeras, Q. Luong, and T. Papadopoulou. *The Geometry of Multiple Images: The Laws That Govern The Formation of Images of A Scene and Some of Their Applications*. MIT Press, 2001.

[38] J. Fayad, L. Agapito, and A. Del Bue. Piecewise quadratic reconstruction of non-rigid surface from monocular sequences. In *Proceedings of the European Conference on Computer Vision*, 2010.

[39] J.-A. Ferrez, K. Fukuda, and T. M. Liebling. Solving the fixed rank convex quadratic maximization in binary variables by a parallel zonotope construction algorithm. *European Journal of Operations Research*, 2004.

[40] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981.

[41] C. K. Friesen and A. Kingstone. The eyes have it! reflexive orienting is triggered by nonpredictive gaze. *Psychonomic Bulletin and Review*, 1998.

[42] K. Fukunaga and L. D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 1975.

[43] M. R. Garey and D. S. Johnson. *Computer and Interactability: A guide to the theory of NP-Completeness*. Freeman, 1979.

[44] R. Gayle, W. Moss, M. C. Lin, and D. Manocha. Multi-robot coordination using generalized social potential fields. In *Proceedings of the International Conference on Robotics and Automation*, 2009.

[45] A. H. Gee and R. Cipolla. Determining the gaze of faces in images. *Image and Vision Computing*, 1994.

[46] P. F. U. Gotardo and A. M. Martinez. Computing smooth time-trajectories for camera and deformable shape in structure from motion with occlusion. *IEEE Transactions on Pattern*

Analysis and Machine Intelligence, 2011.

[47] C. W. J. Granger. Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, 1969.

[48] S. Gu. Polynomial time solvable algorithms to binary quadratic programming problems with q being a tri-diagonal or five-diagonal matrix. In *Proceedings of the International Conference on Wireless Communications and Signal Processing*, 2010.

[49] E. D. Guestrin and M. Eizenman. General theory of remote gaze estimation using the pupil center and corneal reflection. *IEEE Transactions on Biomedical Engineering*, 2006.

[50] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis. Understanding videos, constructing plots learning a visually grounded storyline model from annotated videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[51] A. Gupta, S. Satkin, A. A. Efros, and M. Hebert. From scene geometry to human workspace. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[52] E. Hall. *The hidden dimension*. Doubleday New York, 1966.

[53] M. Hamidi and J. Pearl. Comparison of the cosine and fourier transforms of markov-i signal. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1976.

[54] R. Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

[55] R. Hartley and R. Vidal. Perspective nonrigid shape and motion recovery. In *Proceedings of the European Conference on Computer Vision*, 2008.

[56] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[57] D. Helbing and P. Molnár. Social force model for pedestrian dynamics. *Physics Review E*, 1995.

[58] J. M. Henderson. Human gaze control during real-world scene perception. *Trends in Cognitive Sciences*.

[59] C. Hennessey and P. Lawrence. 3D point-of-gaze estimation on a volumetric display. In *Symposium on Eye tracking research & applications*, 2008.

[60] N. R. Howe, M. E. Leventon, and W. T. Freeman. Bayesian reconstruction of 3D human motion from single-camera video. In *Advances in Neural Information Processing Systems*, 1999.

[61] Z. L. Husz, A. M. Wallace, and P. R. Green. Evaluation of a hierarchical partitioned particle filter with action primitives. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop*, 2007.

[62] Y. Ishiguro, A. Mujibiya, T. Miyaki, and J. Rekimoto. Aided eyes: Eye activity sensing for daily life. In *Proceedings of the International Conference on Pervasive Computing*, 2010.

[63] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene

analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998.

[64] R. S. Jampel and D. X. Shi. The primary position of the eyes, the resetting saccade, and the transverse visual head plane. head movements around the cervical joints. *Investigative Ophthalmology and Vision Science*, 1992.

[65] G. Jansson, S. S. Bergstorm, and W. Epstein. *Perceiving Events and Objects*. Lawrence Erlbaum, 1994.

[66] A. Johansson, D. Helbing, and P. K. Shukla. Specification of a microscopic pedestrian model by evolutionary adjustment to video tracking data. *Advances in Complex Systems*, 2008.

[67] H. Joo, H. S. Park, and Y. Sheikh. Optimal visibility estimation for large-scale dynamic 3D reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014.

[68] J. Y. Kaminski and M. Teicher. A general framework for trajectory triangulation. *Journal of Mathematical Imaging and Vision*, 2004.

[69] A. Kendon. *Conducting Interaction: Patterns of Behavior in Focused Encounters*. Cambridge University Press, 1990.

[70] A. Kendon. Spacing and orientation in co-present interaction. In *Development of Multimodal Interfaces: active Listening and Synchrony*, 2010.

[71] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings of the International Conference on Robotics and Automation*, 1985.

[72] K. Kim, M. Grundmann, A. Shamir, I. Matthews, J. Hodgins, and I. Essa. Motion fields to predict play evolution in dynamic sport scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[73] E. M. Klier, H. Wang, A. G. Constantin, and J. D. Crawford. Midbrain control of three-dimensional head orientation. *Science*, 2002.

[74] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 1985.

[75] O. Komogortsev and J. Khan. Perceptual attention focus prediction for multiple viewers in case of multimedia perceptual compression with feedback delay. In *Proceedings of the Eye Tracking Research and Applications*, 2006.

[76] H.-J. Lee and Z. Chen. Determination of 3D human body postures from a single view. *Computer Vision, Graphics, and Image Processing*, 1985.

[77] D. Li, J. Babcock, and D. J. Parkhurst. openEyes: a low-cost head-mounted eye-tracking solution. In *Symposium on Eye-Tracking Research and Application*, 2006.

[78] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[79] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 1981.

[80] M. I. A. Lourakis and A. A. Argyros. SBA: A software package for generic sparse bundle

adjustment. *ACM Transactions on Mathematical Software*, 2009.

[81] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004.

[82] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[83] M. Marin-Jimenez, A. Zisserman, and V. Ferrari. "here's looking at you, kid." detecting people looking at each other in videos. In *Proceedings of the British Machine Vision Conference*, 2011.

[84] S. Marks, B. Wünsche, and J. Windsor. Enhancing virtual environment-based surgical teamwork training with non-verbal communication. In *International Conference on Computer Graphics Theory and Applications*, 2009.

[85] P. Marshall, Y. Rogers, and N. Pantidi. Using F-formations to analyse spatial patterns of interaction in physical environments. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*, 2011.

[86] R. Mehran, A. Oyama, and M. Shah. Abnormal crowd behavior detection using social force model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[87] P. Merz and B. Freisleben. Greedy and local search heuristics for unconstrained binary quadratic programming. *Journal of Heuristics*, 2002.

[88] H. Misslisch, D. Tweed, and T. Vilis. Neural constraints on eye motion in human eye-head saccades. *Journal of Neurophysiology*, 1998.

[89] J. L. Moreno and H. H. Jennings. Statistics of social configurations. *Sociometry*, 1938.

[90] F. Moreno-Noguer, V.Lepetit, and P.Fua. EPnP: Efficient perspective-n-point camera pose estimation. In *Proceedings of the International Conference on Computer Vision*, 2007.

[91] S. M. Munn and J. B. Pelz. 3D point-of-regard, position and head orientation from a portable monocular video-based eye tracker. In *Symposium on Eye-Tracking Research and Application*, 2008.

[92] H. Murphy and A. T. Duchowski. Gaze-contingent level of detail rendering. In *Eurographics*, 2001.

[93] R. R. Murphy. Human-robot interaction in rescue robotics. *IEEE Transactions on Systems, Man and Cybernetics*, 2004.

[94] E. Murphy-Chutorian and M. M. Trivedi. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009.

[95] B. Noris, K. Benmachiche, and A. G. Billard. Calibration-free eye gaze direction detection with gaussian processes. In *International Conference on Computer Vision Theory and Applications*, 2006.

[96] N. Oliver, B. Rosario, and A. Pentland. Graphical models for recognising human interactions. In *Proceedings of Neural Information Processing Systems*, 1998.

[97] S. Olsen and A. Bartoli. Using priors for improving generalization in non-rigid structure-

from-motion. In *Proceedings of British Machine Vision Conference*, 2007.

[98] C. Olsson, A. P. Eriksson, and F. Kahl. Solving large scale binary quadratic problems: Spectral methods vs. semidefinite programming. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[99] J. Östlund, A. Varol, D. T. Ngo, and P. Fua. Laplacian meshes for monocular 3D shape recovery. In *Proceedings of the European Conference on Computer Vision*, 2012.

[100] K. E. Ozden, K. Cornelis, L. V. Eycken, and L. V. Gool. Reconstructing 3D trajectories of independently moving objects using generic constraints. *Computer Vision and Image Understanding*, 2004.

[101] M. Paladini, A. Del Bue, M. Stosic, M. Dodig, J. Xavier, and L. Agapito. Factorization for non-rigid and articulated structure using metric projections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[102] V. Parameswaran and R. Chellappa. View independent human body pose estimation from a single perspective image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[103] H. S. Park and Y. Sheikh. 3D reconstruction of a smooth articulated trajectory from a monocular image sequence. In *Proceedings of the International Conference on Computer Vision*, 2011.

[104] H. S. Park, T. Shiratori, I. Matthews, and Y. Sheikh. 3D reconstruction of a moving point from a series of 2D projections. In *Proceedings of the European Conference on Computer Vision*, 2010.

[105] H. S. Park, E. Jain, and Y. Shiekh. 3D social saliency from head-mounted cameras. In *Proceedings of Neural Information Processing Systems*, 2012.

[106] H. S. Park, E. Jain, and Y. Shiekh. Predicting primary gaze behavior using social saliency fields. In *Proceedings of International Conference on Computer Vision*, 2013.

[107] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool. You'll never walk alone: Modeling social behavior for multi-target tracking. In *Proceedings of the International Conference on Computer Vision*, 2009.

[108] A. S. Pentland. To signal is human. *American Scientist*, 2010.

[109] R. J. Peters and L. Itti. Applying computational tools to predict gaze direction in interactive visual environments. *ACM Transactions on Applied Perception*, 2007.

[110] P. Peursum, S. Venkatesh, and G. West. A study on smoothing for particle-filtered 3D human body tracking. *International Journal of Computer Vision*, 2010.

[111] J. C. Picard and P. M. Ratliff. Minimal cost cut equivalent networks. *Management Science*, 1973.

[112] F. Pirri, M. Pizzoli, and A. Rudi. A general method for the point of regard estimation in 3D space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[113] S. Poljak, F. Rendl, and H. Wolkowicz. A recipe for semidefinite relaxation for (0,1)-

quadratic programming. *Journal of Global Optimization*, 1995.

[114] M. I. Posner. Orienting of attention. *Quarterly Journal of Experimental Psychology*, 1980.

[115] K. Prabhakar and J. M. Rehg. Categorizing turn-taking interactions. In *Proceedings of the European Conference on Computer Vision*, 2012.

[116] K. Prabhakar, S. Oh, P. Wang, G. D. Abowd, and J. M. Rehg. Temporal causality for the analysis of visual events. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[117] R. Rae and H. J. Ritter. Recognition of human head orientation based on artificial neural networks. *IEEE Transactions on Neural Networks*, 1998.

[118] R. Raghavendra, A. Del Bue, M. Cristani, and V. Murino. Abnormal crowd behavior detection by social force optimization. In *Human Behavior Understanding*, 2011.

[119] V. Ramakrishna, T. Kanade, and Y. Sheikh. Reconstructing 3D human pose from 2D image landmarks. In *Proceedings of the European Conference on Computer Vision*, 2012.

[120] V. Ramanathan, B. Yao, and L. Fei-Fei. Social role discovery in human events. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[121] J. H. Reif and H. Wang. Social potential field: a distributed behavioral control for autonomous robots. *Robotics and Autonomous Systems*, 1999.

[122] N. Robertson and I. Reid. Estimating gaze direction from low-resolution faces in video. In *Proceedings of the European Conference on Computer Vision*, 2006.

[123] M. S. Ryoo and J. K. Aggarwal. Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities. In *Proceedings of the International Conference on Computer Vision*, 2009.

[124] M. Salzmann and P. Fua. *Deformable Surface 3D Reconstruction from a Single Viewpoint*. Morgan-Claypool, 2010.

[125] J. M. Saragih, S. Lucey, and J. F. Cohn. Deformable model fitting by regularized landmark mean-shift. *International Journal of Computer Vision*, 91:200–215, 2011.

[126] H. Schneiderman and T. Kanade. Probabilistic modeling of local appearance and spatial relationships for object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1998.

[127] A. Shashua and L. Wolf. Homography tensors: On algebraic entities that represent three views of static or moving planar points. In *Proceedings of the European Conference on Computer Vision*, 2000.

[128] Y. Sheikh, S. Nobuhara, H. Joo, H. Liu, L. Tan, L. Gui, M. Vo, B. Nabbe, I. Matthews, and T. Kanade. The panoptic studio. *Technical Report CMU-RI-TR-14-04, Robotics Institute, Carnegie Mellon University*, 2014.

[129] T. Shiratori, H. S. Park, L. Sigal, Y. Sheikh, and J. K. Hodgins. Motion capture from body-mounted cameras. *ACM Transactions on Graphics (SIGGRAPH)*, 2011.

[130] H. Sidenbladh, M. J. Black, and D. J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *Proceedings of the European Conference on Computer Vision*, 2000.

[131] D. J. Simons and C. F. Chabris. Gorillas in our midst: Sustained inattentional blindness for dynamic events. *Perception*.

[132] K. Smith, S. Ba, J.-M. Odobez, and D. Gatica-Perez. Tracking the visual focus of attention for a varying number of wandering people. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[133] N. Snavely, S. M. Seitz, and R. Szeliski. Photo tourism: Exploring photo collections in 3D. *ACM Transactions on Graphics (SIGGRAPH)*, 2006.

[134] M. Sophie, G. Nathalie, and P. Denis. Gaze prediction improvement by adding a face feature to a saliency model. *Recent Advances in Signal Processing*, 2009.

[135] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel. From gaze to focus of attention. In *International Conference on Visual Information and Information Systems*, 1999.

[136] G. Strang. The discrete cosine transform. *SIAM Review*, 1999.

[137] K. Takemura, Y. Kohashi, T. Suenaga, J. Takamatsu, and T. Ogasawara. Estimating 3D point-of-regard and visualizing gaze trajectories under natural head movements. In *Symposium on Eye-Tracking Research and Application*, 2010.

[138] C. Taylor. Reconstruction of articulated objects from point correspondences in a single uncalibrated image. *Computer Vision and Image Understanding*, 2000.

[139] J. Taylor, A. D. Jepson, and K. N. Kutulakos. Non-rigid structure from locally-rigid motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010.

[140] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision*, 1992.

[141] L. Torresani and C. Bregler. Space-time tracking. In *Proceedings of the European Conference on Computer Vision*, 2002.

[142] L. Torresani, D. Yang, G. Alexander, and C. Bregler. Tracking and modeling non-rigid objects with rank constraints. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[143] L. Torresani, A. Hertzmann, and C. Bregler. Learning non-rigid 3D shape from 2D motion. In *Advances in Neural Information Processing Systems*, 2003.

[144] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: Estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2008.

[145] B. Triggs, P. F. McLauchlan, R. I. Hartley, and A. W. Fitzgibbon. Bundle adjustment - a modern synthesis. In *Proceedings of the International Workshop on Vision Algorithms: Theory and Practice*, 2000.

[146] R. Urtasun, D. J. Fleet, and P. Fua. Temporal motion models for monocular and multiview 3-D human body tracking. *Computer Vision and Image Understanding*, 2006.

[147] J. Valmadre and S. Lucey. Deterministic 3D human pose estimation using rigid structure. In *Proceedings of the European Conference on Computer Vision*, 2010.

[148] J. Valmadre and S. Lucey. General trajectory prior for non-rigid reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[149] R. Vidal and D. Abretske. Nonrigid shape and motion from multiple perspective views. In *Proceedings of the European Conference on Computer Vision*, 2006.

[150] A. Vinciarelli, M. Pantic, and H. Bourlard. Social signal processing: Survey of an emerging domain. *Image and Vision Computing*, 2009.

[151] M. Vondrak, L. Sigal, and O. C. Jenkins. Physic simulation for probabilistic motion tracking. In *Proceedings of the European Conference on Computer Vision*, 2008.

[152] J. W. Wallis, T. R. Miller, C. A. Lerner, and E. C. Kleerup. Three-dimensional display in nuclear medicine. *IEEE Transactions on Medical Imaging*, 1989.

[153] J.-G. Wang and E. Sung. Study on eye gaze estimation. *IEEE Transactions on Systems, Man and Cybernetics*, 2002.

[154] P. Wang, G. D. Abowd, and J. M. Rehg. Quasi-periodic event analysis for social game retrieval. In *Proceedings of the International Conference on Computer Vision*, 2009.

[155] X. Wang, V. H. Ablavsky, B. Shitrit, H. Beny, and P. Fua. Take your eyes off the ball: Improving ball-tracking by focusing on team play. *Computer Vision and Image Understanding*, 2013.

[156] X. Wei and J. Chai. Modeling 3D human poses from uncalibrated monocular images. In *Proceedings of the International Conference on Computer Vision*, 2009.

[157] X. Wei and J. Chai. Videomocap: Modeling physically realistic human motion from monocular video sequences. *ACM Transactions on Graphics (SIGGRAPH)*, 2010.

[158] G. Welch and E. Foxlin. Motion tracking: no silver bullet, but a respectable arsenal. *IEEE Computer Graphics and Applications*, 2002.

[159] Y. Wexler and A. Shashua. On the synthesis of dynamic scenes from reference views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2000.

[160] L. Wolf and A. Shashua. On projection matrices $\mathcal{P}^k \rightarrow \mathcal{P}^2, k = 3, \cdots, 6$, and their applications in computer vision. *International Journal of Computer Vision*, 2002.

[161] J. Xiao and T. Kanade. Non-rigid shape and motion recovery: Degenerate deformations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2004.

[162] J. Xiao, J. Chai, and T. Kanade. A closed-form solution to non-rigid shape and motion recovery. *International Journal of Computer Vision*, 2006.

[163] X. Xiong and F. De la Torre. Supervised descent method and its applications to face alignment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[164] J. Yan and M. Pollefeys. A factorization-based approach to articulated motion recovery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[165] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern*

Recognition, 2010.

[166] Y. Yang and D. Ramanan. Articulated human detection with flexible mixtures of parts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[167] Y. Yang, S. Baker, A. Kannan, and D. Ramanan. Recognizing proxemics in personal photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

[168] B. Yao and L. Fei-Fei. Action recognition with exemplar based 2.5D graph matching. In *Proceedings of the European Conference on Computer Vision*, 2012.

[169] Y. Zhou, S. Yan, and T. S. Huang. Pair-activity classification by bi-trajectories analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2008.