Sensing, Measuring, and Modeling Social Signals in Nonverbal Communication

Hanbyul Joo

Thesis Committee:
Yaser Sheikh, *Carnegie Mellon University* (Chair)
Takeo Kanade, *Carnegie Mellon University*
Louis-Philippe Morency, *Carnegie Mellon University*
David Forsyth, *University of Illinois Urbana-Champaign*
Mina Cikara, *Harvard University*

The Robotics Institute
Carnegie Mellon University

Tech Report Number: CMU-RI-TR-19-01

This dissertation is submitted for the degree of
Doctor of Philosophy in Robotics

Jan, 2019

To my wife, Sooyeon Lee, and our children, Ian and Erin.

Acknowledgements

I am immensely grateful to my advisor Yaser Sheikh who has been an ideal mentor and teacher for me. Without his encouragement, guidance, and mentorship, none of this work would be possible. I also greatly appreciate my thesis committee - Takeo Kanade, Louis-Philippe Morency, David Forsyth, and Mina Cikara for their valuable feedback and advice.

I would like to thank Hyun Soo Park who guided me at the beginning of my PhD and has been a mentor throughout the years. I have been fortunate to have many great collaborators in materializing the Panoptic Studio: Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Godisart, Bart Nabbe, Iain Matthews, and Shohei Nobuhara. I also thank all of our current and previous group members: Minh Vo, Aayush Bansal, Varun Ramakrishna, Yair Movshovitz-Attias, Natasha Kholgade Banerjee, Donglai Xiang, Gines Hidalgo, Shih-En Wei, Zhe Cao, Luona Yang, Gaku Nakano, Xiu Li, Raaj Yaadhav, and Haroon Idrees. I have been also fortunate to collaborate with Thabo Beeler, Derek Bradley, and Chenglei Wu during my internship at Disney Research Zurich. I am also grateful to the researchers at Facebook Reality Labs Pittsburgh for their advice and feedback during my internship. I would like to express my great appreciation to Jessica Hodgins, Chris Atkeson, Kris Kitani, Srinivasa Narasimhan, Abhinav Gupta, Simon Lucey, Deva Ramanan, Fernando De la Torre, John O'brien, Nancy Pollard, Jim McCann, and Martial Hebert for many insightful discussions, help, and advice. I would like to thank the Samsung Scholarship for supporting my graduate study.

I would like to especially thank my family - my parents, parents-in-law, my brother, brothers-in-law for their love, support, and encouragement. Last, but not least, I dedicate this thesis to my beloved wife, Sooyeon Lee, and our children, Ian and Erin, for their tireless love, support, and sacrifices.

Abstract

Humans convey their thoughts, emotions, and intentions through a concert of social displays: voice, facial expressions, hand gestures, and body posture, collectively referred to as social signals. Despite advances in machine perception, machines are unable to discern the subtle and momentary nuances that carry so much of the information and context of human communication. The encoding of conveyed information by social signals, particularly in nonverbal communication, is still poorly understood, and thus it is unclear how to teach machines to use such social signals to make them collaborative partners rather than tools that we use. A major obstacle to scientific progress in this direction is the inability to sense and measure the broad spectrum of behavioral cues in groups of interacting individuals, which hinders applying computational methods to model and understand social signals.

In this thesis, we explore new approaches in sensing, measuring, and modeling social signals to ultimately endow machines with the ability to interpret nonverbal communication. This thesis starts by describing our exploration in building a massively multiview sensor system, the *Panoptic Studio*, that can capture a broad spectrum of human social signaling—including voice, social formations, facial expressions, hand gestures, and body postures—among groups of multiple people. Second, leveraging this system equipped with more than 500 synchronized cameras, we then present a method to measure the subtle 3D movements of anatomical keypoints in face-to-face interaction, providing a new opportunity to computationally study social signals. In the last part of this thesis, we present a social signal prediction task to model nonverbal communication in a data-driven manner. We establish a new large-scale corpus from hundreds of participants containing various channels of social signal measurements. Leveraging this dataset, we verify that the social signals are predictive each other with strong correlations.

Table of contents

Chapter 1

Introduction

Along with verbal language, we use many other channels for communication, including facial expression, body gestures, hand motions, and interpersonal proximity, collectively referred to as *nonverbal social signals*. The use of all these channels is important in social interaction, where subtle emotions and intentions are transmitted via the combination of such signals [1]. Endowing machines with such social interaction abilities is an essential goal of Artificial Intelligence (AI) to make machines that can effectively cooperate with humans.

One way to endow machines with such social skills would be to encode all the rules that humans observe during social communication [2, 3]. Unfortunately, nonverbal interaction is still poorly understood despite its importance in social communication [4–6], making it hard to formalize all rules about how to understand and use social signals. An alternative direction of this "symbolic" paradigm [7] is to take the "non-symbolic" (or connectionist) approach [8] to learn the way humans communicate purely from the data without any hand-coded high level representations. Interestingly, we have recently witnessed significant progress in Natural Language Processing (NLP) showing the potential to allow machines to "freely" communicate with humans using written language and speech [9]. This success has been led by data-driven approaches leveraging large-scale language datasets and a powerful modeling tool, deep neural networks [10], to automatically learn the patterns of human verbal communication. Remarkably, these achievements have not made extensive use of the prior knowledge about grammar and the structure of languages that linguists have accumulated over centuries. Motivated by this, this thesis hypothesizes that a similar approach can be an important breakthrough in modeling nonverbal communication.

However, there exists a fundamental difficulty in building a data-driven nonverbal communication model: the data is extremely rare. In the verbal language domain, words contain the full expressive power to record verbal signals by a composition of a handful of discrete symbols. Especially on the Internet, there already exist millions of articles or dialogues which

are readily usable for data-driven methods (either supervised or unsupervised). However, for nonverbal signals, how to “record" or collect these signals is uncertain. Imagine a situation where a group of people are communicating in our daily life. The positions and orientations of individuals, their body gestures, gaze, and facial expressions (among others) are the data we are interested in. Notably, these social signals emitted from all people in the group need to be simultaneously sensed to study their correlation and causality. Although there also exist millions of videos where our daily activities—including social interactions—are captured on the Internet, these raw videos cannot be directly used because we have to first measure the behavioral cues (relative location, face, body pose, and so on) from the raw pixels to focus on investigating the rules of nonverbal interactions.

This thesis explores two scientific questions to computationally study social signals in nonverbal communication: (1) how to measure the “a broad spectrum of behavioral social signals", including facial expressions, hand gestures, and body motions of an interacting group and (2) how to formalize a “Connectionist" model by leveraging these new types of signal measurements aiming to teach machines how to decode and encode nonverbal signals to genuinely interact with humans. As major advancements in science are nearly always preceded by innovations in engineering that enable us to measure the world, the first part of this thesis starts by presenting a novel system and method to sense and measure such signals. As a core contribution, a massively multiview capture system, the *Panoptic Studio* shown in the top of Figure 1.2, is built to capture naturally interacting multiple people in social situations (Chapter 2). Methods to measure the subtle 3D movements of anatomical landmarks including facial expression, hand gestures, and body motions are also presented, leveraging a large number of views of the Panoptic Studio System (Chapter 3, 4, and 5). Our system and method enable us to measure and record the various behavioral cues in face-to-face interaction without using any intrusive devices or artificial markers.

In the last part of this thesis, we computationally study the dynamics among interpersonal social signals by leveraging a large-scale social signal corpus built from our system and measurement method. In this research, we verify that the social signals naturally emerging in an interacting group are highly correlated and predictive each other. To achieve the goal, we first collect a large scale dataset where a broad spectrum of social signals are measured from hundreds of participants in a carefully designed triadic interaction scenario (Chapter 6). Then, we formalize a social signal prediction problem as a data-driven way to model the dynamics and correlations between various channels of social signals exchanged during social interaction (Chapter 7).

1.1 Key Challenges

This thesis aims to computationally model social interaction, focusing on nonverbal social signals. Major challenges in pursuing the direction are addressed.

How To Sense Nonverbal Social Signals: To achieve the goal of this thesis, the signals naturally emerging from interacting people should be obtained first. However, it is challenging to sense such social signals. Here, we use the term "sensing" as a process to record the analog signals (human behaviors) existing in the real world to the digital space (videos in raw pixels) which can be computationally processed afterwards (see Figure 1.1). Ideally, we pursue to sense the social signals as complete as possible, to reconstruct the original signals in the digital space. However, sensing social signals exchanged in social interaction is challenging because of the following reasons. First, during social interactions, strong occlusions emerge functionally (e.g., people systematically face each other while interacting, bodies are occluded by gesticulating limbs). Thus, a monocular view or a few camera views can hardly capture the entire social signals of all the people involved in the communication. Second, subtle motions from faces and hands that play an important role in social interactions should be measured together with large motions from torsos and limbs. This introduces a difficulty because both a large volume and subtle details are sensed together. In existing methods, often face and hand motions are captured at close range [11–19], while torso and limb motions are captured in a sufficiently large working volume where people can freely move, less focusing the subtle details in faces and hands [20–23].

How To Measure Nonverbal Social Signals: Given the sensed data recorded in raw pixel space, we need to measure social signals by extracting human behavioral cues. Here, we use the term "measuring" as a process to compute the *registered* social signals from the raw pixels, which provides correspondences across time and across individuals. As examples of the registered social signals, we consider a set of 3D keypoints extracted from each individual that provide sparse correspondences on anatomical joint locations, and also a 3D mesh structure that provides dense correspondences on the surface of bodies (see Figure 1.1). These registered social signals enable us to computationally investigate human behaviors in social interaction. Importantly, the social signals should be measured non-intrusively to allow people to behave naturally and voluntarily. Thus, popular marker-based motion capture systems (e.g., [24]) are not applicable. Despite the advances in human sensing and markerless motion capture fields in computer vision and graphics, these challenges have not been fully solved yet. Most of the existing methods prior to this thesis focus on the motion of a single subject [20, 25, 26, 21, 22, 27, 28] or exaggerated motions of actors [29, 30].

Importantly, there is no existing system that can track, without markers, the human body, face, and hands of multiple individuals simultaneously. Due the limit of available measurement tools, social signals are studied in limited scenarios, for example: (1) studies only focus on face signals ignoring body motions [31–33]; (2) studies consider situations in a table setup where participants' motions are limited [34, 35, 31, 36, 32, 33]; (3) studies assume a small number of people (mostly dyadic communications) [31, 36, 32, 37, 33, 38].

How To Model Nonverbal Communications: Modeling social signals using computational methods is a largely unexplored area due to the following four reasons. First, the lack of available dataset or measurement technology limits the opportunity to computationally investigate the social signal modeling. Because of this reason, studies on body gestures are relatively rare [38–40], while many researchers focus on facial expressions exploiting existing automatic measurement tools [41] with an existing coding system [42]. Similarly, although we want to investigate the correlation between various behavioral cues (e.g., facial expression and hand gestures), the challenges in simultaneously measuring these signals make it hard to explore this direction.

Second, there is no good way to represent and objectively annotate the semantics embedded in social signals. For example, a "smile" signal, formed by flexing the muscles at the side of the mouth with a contraction of the muscles at the corner of the eyes, may be recognized as "happiness". However, the internal emotion expressed by the signals may not be simply annotated by a discrete status of "smile" or "happiness", because it may have a variety of subtly different meanings depending on the intensity of the facial movements. Importantly, the mapping between the social signal measurements to the semantic space is subjective to the observers [43].

Third, the high complexity of social signals that are located in a continuous and high dimensional space makes the modeling harder. Unlike words, the start and end timing of social signals are ambiguous. The organized motions from torso, face, and hands need to be considered together [44, 45]. A social interaction among multiple people (more than two) is also challenging due to the diversity of interactions. All these challenges make computational social signal modeling difficult.

Finally, it is unclear how to evaluate the performance of the model. For example, a social signal prediction model may generate a "realistic" nonverbal signals, but we are lack of methods to quantify how realistic the output is.

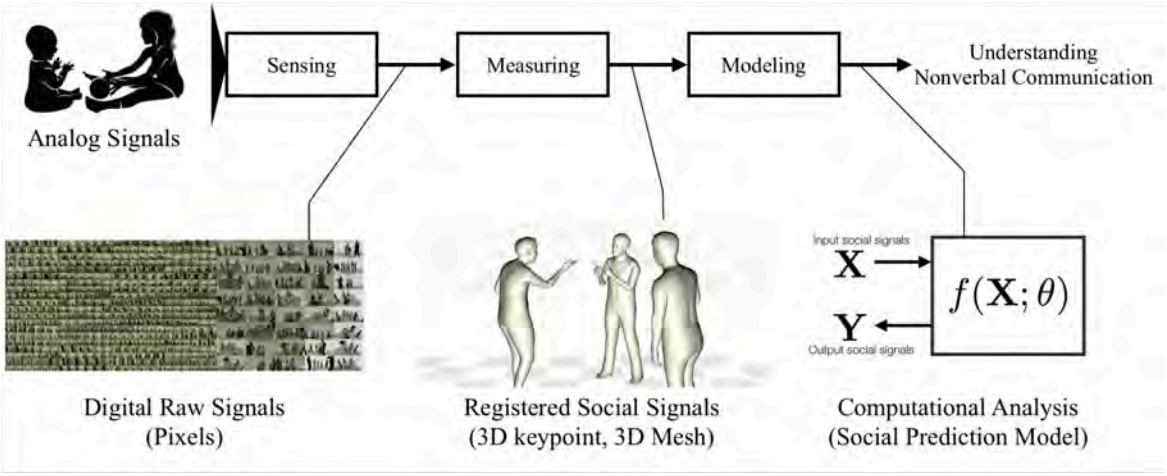


Fig. 1.1 Thesis overview. We present new approaches in sensing, measuring, and modeling social signals to ultimately endow machines with nonverbal communication abilities. In the sensing stage, the analog signals (human behaviors) existing in the real world are recorded to the digital space (videos in raw pixels). In the measuring stage, we extract the *registered* social signals (3D keypoints or 3D meshes) that provide correspondences across time and across individuals, from raw pixels. Finally, in the modeling stage, we build a function to mimic the social behaviors of humans in social interaction leveraging a large scale social signal corpus.

1.2 Thesis Contribution

Thesis statement. *We advocate that a broad spectrum of social signals—including the body gestures, facial expression, and hand motion—of a naturally interacting group can be measured by a system with a sufficient number of cameras. Based on the measurement, we also argue that such naturally emerging social signals are predictive each other, and modeling their dynamics is crucial to enable social Artificial Intelligence.*

This thesis presents a “full-stack" framework to apply computational methods to model nonverbal social communication. The core contribution of thesis spans a new sensor system to capture interacting multiple people (Chapter 2), a novel methodology to measure a broad spectrum of social signals (Chapter 3, 4, and 5), a large-scale social interaction dataset (Chapter 6), and formalizing a predictive model of social signals (Chapter 7). The core contributions of this thesis are summarized below and example results are shown in Figure 1.2.

1.2.1 Panoptic Studio: A Massively Multiview System (Chapter 2)

We built a new sensor system, the Panoptic Studio, specifically designed to overcome the sensing challenges emerging in social situations. The system is composed of 521 synchronized sensors with different types, including 480 VGA cameras, 31 HD cameras, 10 RGB+D sensors, and 23 microphones. The large number of sensors densely cover a large capture volume (5.49m of a diameter) sufficient to have multiple people, enabling to relieve occlusions among interacting people.

We present a novel modularized design architecture to simplify the complexity of the system. The system is composed of repeated modules with the same shape and architecture, and each module is controlled by a separate local machine. This modularized design makes it easy to control and manage the system, and also enables efficient data acquisition by saving data locally. All cameras are accurately synchronized (or temporally aligned) by a hardware clock, and spatially calibrated in a common 3D world space. The core design choices and technical solutions including structural design, architecture, temporal calibration, and spatial calibration are addressed in Chapter 2.

1.2.2 Measuring 3D Social Signals (Chapters 3, 4, 5)

We present the first method to measure a broad spectrum of 3D social signals including subtle motions from body, face, and hands. The core goals in developing the methods are: (1) simultaneously capturing the *total body motion* of all body parts when a group of people are naturally interacting; (2) building a fully automatic method to capture social sequences at scale without human labor; (3) minimizing assumptions about scenes to be applicable in any type of motion and appearance of arbitrary number of people; and (4) avoiding intrusive approaches, such as attaching artificial markers and avoiding a tedious 3D template building step. To this end, we present a method based on “weak” perceptual processes from a large number of views, and robustly measure social signals by satisfying the aforementioned principles. In Chapter 3, a method to reconstruct dense surface movements, which we refer to as a “trajectory stream", is presented by fusing optical flow cues from the large number of views in 3D. The core challenge of the method is to reason about a time varying visibility to fully exploiting the large number of views. In Chapter 4, reconstructing the movement of 3D anatomical landmarks is presented. In this method, we run 2D keypoint detectors for body, face, and hands respectively in each view, and fuse them in 3D to reconstruct 3D locations of anatomical landmarks. By associating them with the trajectory stream, we obtain accurate markerless motion capture results, as well as semantic labeling of the trajectory stream. In Chapter 5, we present a novel 3D deformable human body model for total body

motion capture, which can express the motions from full body including facial expression and hand gestures in a unified parametric space.

1.2.3 A Large-Scale Human Motion Database (Chapter 6)

We build a large-scale dataset by capturing face-to-face interactions of hundreds of participants in our Panoptic Studio. The scenes are captured in a carefully designed triadic negotiation scenario, referred to as *Haggling*, where voluntary social behaviors can naturally emerge. This social scenario makes it easier to model social interaction, by putting all the subjects in the same social situation. The various behavioral cues including facial expressions, body motions, hand gestures, and individual voices, are sensed and measured by our markerless motion capture system.

Along with the social game dataset, we also use our system to capture other motions to build a large-scale 3D human motion database. Our database contributes to invent new computer vision techniques. For example, our dataset enables us to build a popular hand keypoint detector [46] in the OpenPose Library [47], the first deformable 3D human model for total motion capture [48], and the first monocular 3D human total capture method applicable in-the-wild [49]. Importantly, we publicly released our dataset in our database website[1].

1.2.4 Formalizing Social Signal Prediction To Model Nonverbal Interactions (Chapter 7)

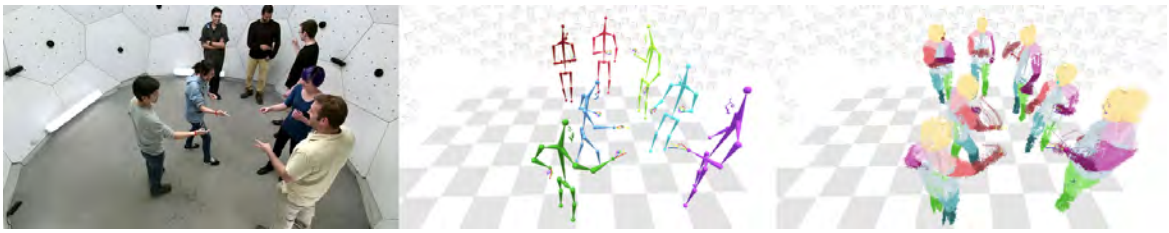
We introduce a *Social Signal Prediction* task as a way to computationally model nonverbal interaction. The objective of this task is to predict the behavior of a target person in a social situation. We hypothesize and verify that a target person's behavior is strongly correlated to the behavioral cues of other individuals. For example, the location and orientation of the target person should be strongly affected by the position of conversational partners (known as Proxemics [50] and F-formation [51]), and the gaze direction, body gestures, and facial expressions of the target person should also be "conditioned" by the behaviors of the conversational partners. In this social signal prediction task, we model the dynamics of social signals exchanged during social interaction, to ultimately teach a robot how to behave in a similar social situation. In Chapter 7, we present several subtasks in modeling a subset of social signals among subject involved in social interaction, and demonstrate their social signals are predictive.

[1] http://domedb.perception.cs.cmu.edu

The relevant publication list for this thesis is as follows:

- (Chapter 2) DIY A Multiview Camera System: Panoptic Studio Teardown, in conjunction with CVPR 2017 (Tutorial Page)
- (Chapters 2 and 4) Panoptic Studio: A Massively Multiview System for Social Motion Capture [52], ICCV 2015
- (Chapters 2 and 4) Panoptic Studio: A Massively Multiview System for Social Interaction Capture [53], TPAMI 2017 (Video)
- (Chapter 3) MAP Visibility Estimation for Large-Scale Dynamic 3D Reconstruction [54], CVPR 2014 (Video)
- (Chapter 4) Hand Keypoint Detection in Single Images using Multiview Bootstrapping [46], CVPR 2017 (Video)
- (Chapter 5) Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies [48], CVPR 2018 (Best Student Paper Award) (Video)
- (Chapter 6) Panoptic Studio Database: http://domedb.perception.cs.cmu.edu

(a) The Panoptic Studio



(b) Measuring 3D Anatomical Landmarks

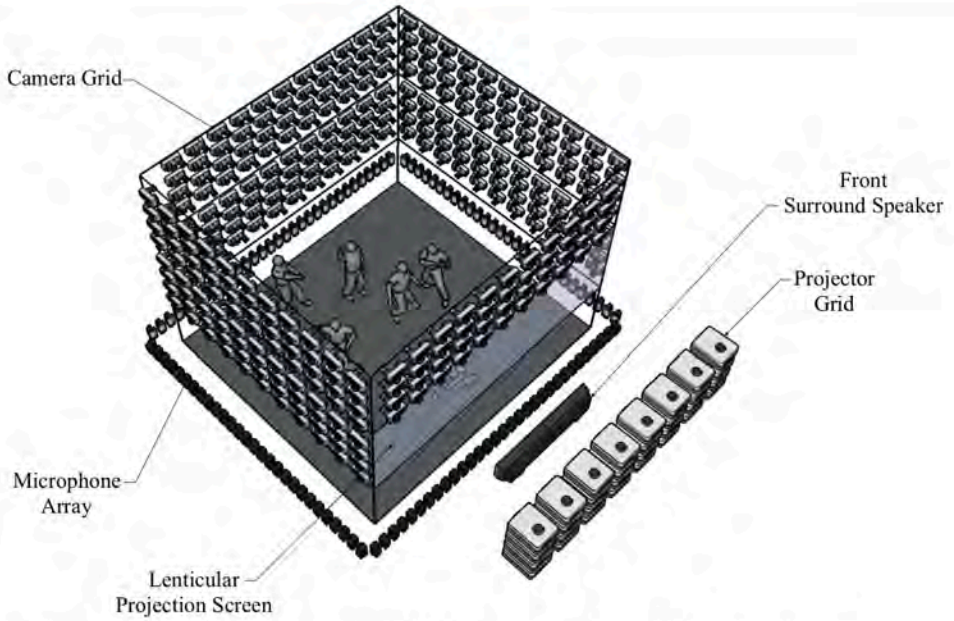
(c) Total Body Motion Capture

(d) Modeling Social Interaction

Fig. 1.2 Thesis contribution. (a) The Panoptic Studio with 521 unique views by 480 VGAs, 31 HDs, and 10 RGB+D sensors; (b) An example scene captured in the Panoptic Studio and measured 3D anatomical landmarks from bodies and hands; (c) A total motion capture result showing the reconstructed 3D mesh models, capturing body, face, hand, and foot motions; (d, left and middle) An example Haggling Game scene and measured 3D social signals; (d, right) an example output of social signal modeling (speaking prediction)

Part I

Sensing Social Signals



The Virtualization Studio - The Initial System Design of the Panoptic Studio
by Sheikh and Kanade, 2009 (http://www.cs.cmu.edu/~virtualized-reality)

Chapter 2

The Panoptic Studio

There are principal challenges in capturing social signals between individuals in a group: (1) social interactions have to be sensed over a volume sufficient to house a dynamic social group, yet subtle details of the motion where important social signals are embedded must be captured; (2) strong occlusions emerge functionally in natural social interactions (e.g., people systematically face each other while interacting, bodies are occluded by gesticulating limbs); (3) human appearance and configuration variation is immense; and (4) social signaling is sensitive to interference—for instance, attaching markers to the face or body, a pre-capture model building stage, or even instructing each individual to assume a canonical body pose during an interaction, primes the nature of subsequent interactions.

The Panoptic Studio is developed to overcome these challenges. The system has a large geodesic sphere structure with a 5.49m diameter, with heterogeneous sensors on its surface including 480 VGA cameras, 31 HD cameras, 10 RGB+D sensors, 23 microphones, and 5 DLP projectors. The core principle and motivation in building this system is to obtain as much measurement as possible to have minimum assumptions about the scenes. We found that the large number of views of the system greatly relieves all the aforementioned sensing challenges, and also enables us to use relatively “weak" perceptual processes from the large number of views rather than a complicated method with strong assumptions about the scenes. To this end, this system enables us to measure a broad spectrum of social signals of naturally interacting groups. This chapter covers the various hardware and low-level software issues in building the Panoptic Studio, including structure design, architectures, synchronization, and spatial calibration.

The Panoptic Studio is the output of several years of collaboration with Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Timothy Godisart, Sean Banerjee, Hyun Soo Park, Iain Matthews, Bart Nabbe, Shohei Nobuhara, Takeo Kanade, and Yaser Sheikh.

2.1 Structural Design

The physical frame of the studio is a variant of a face-transitive solid called a truncated pentagonal hexecontahedron. This particular structure was selected because it has among the largest number of transitive faces of any geodesic dome [55]. The transitivity of the faces enables the modular architecture, and ensures that the structure remains easy to upgrade and customize with different panels of the same configuration. The structure has a diameter of 5.49m and a total height of 4.15m. The floor of the dome is 1.40m below the center of the sphere structure to increase access to the edges. In all, the structure consists of 6 pentagonal panels, 40 hexagonal panels, and 10 trimmed base panels. The interior and exterior view of the Panoptic Studio are shown in Figure 2.1 and a 360° panoramic view of the interior of the Panoptic Studio is shown in Figure 2.2. The structural design is illustrated in the left of Figure 2.3.

Our design was modularized so that each hexagonal panel houses a set of 24 VGA cameras and a HD camera. To determine the placement of the VGA cameras, we initialized their positions by tessellating the hexagon face into 24 triangles and using this initialization to define a 3-neighborhood structure shown in the right of Figure 2.3. Using this neighborhood structure and the initialization we determine the placement of the cameras over the geodesic dome by minimizing the difference in angles between all neighbors of every camera,

$$\{\theta_{ij}\}^* = \arg\min_{\{\theta_{ij}\}} \sum_{p=1}^{P} \sum_{i=1}^{N} \sum_{j \in \mathcal{N}(i)} \sum_{k \in \mathcal{N}(i) \neq j} (r(\theta_{ij}|p) - r(\theta_{ik}|p))^2,$$

where $P = 20$ is the number of panels, $N = 24$ is the number of cameras in each panel, $\mathcal{N}(\cdot)$ is the neighborhood of a camera, $r(\cdot|p)$ is a function transforming the angle on a reference panel to the p-th panel. The cameras sample the span of the vertical axis of the space and sample 48.71° of the horizontal axis. With this distribution, the minimum baseline between any VGA camera and its nearest three neighbors is 21.05cm.

The 31 HD cameras are installed at the center of each hexagonal panel, and 5 projectors are installed at the center of each pentagonal panel[1]. Additionally, a total of 10 Kinect v2 RGB+D sensors are mounted at heights of 1 and 2.6 meters, forming two rings with 5 evenly spaced sensors each.

[1]Note that no sensors are installed on some panels (e.g., ceiling panels occluded by lights).

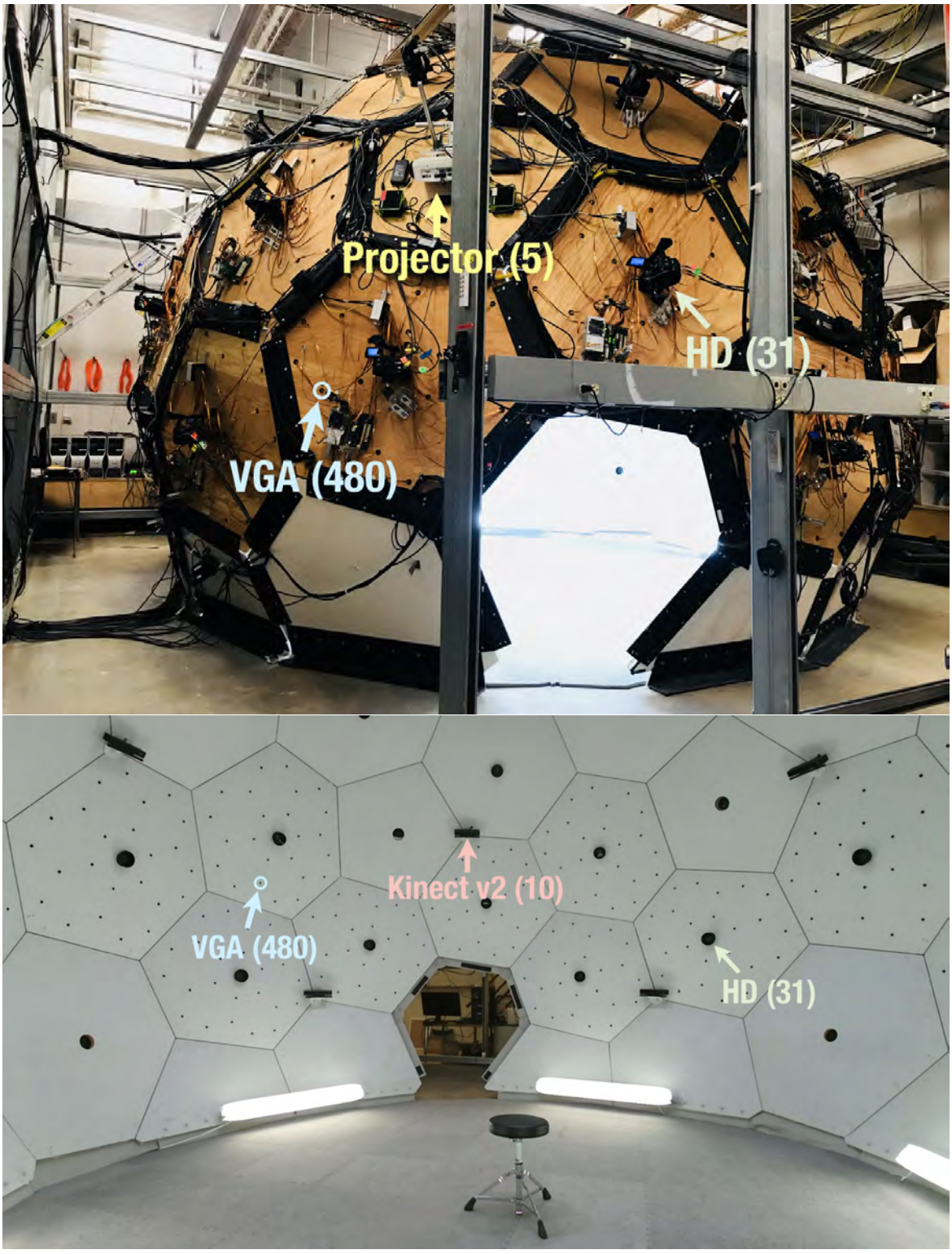


Fig. 2.1 The exterior and interior shape of the Panoptic Studio, equipped with 480 VGA cameras, 31 HD cameras, 10 RGB+D cameras, and 5 DLP Projectors. (Top) The exterior of the Panoptic Studio. (Bottom) The interior of the Panoptic Studio.

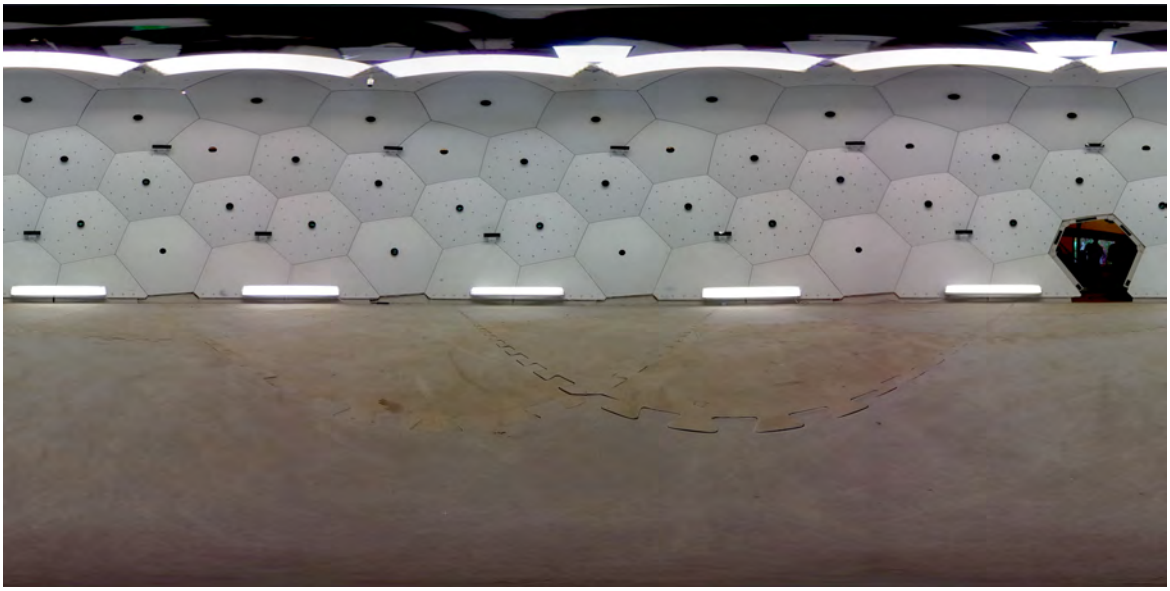


Fig. 2.2 A 360° panoramic photo captured inside the Panoptic Studio. The empty panel is the entrance of the studio.

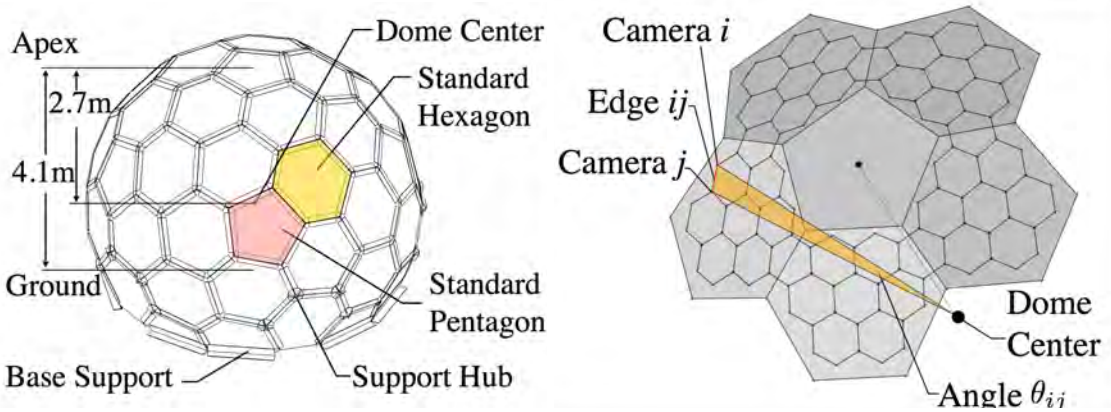


Fig. 2.3 Structural Design of the Panoptic Studio (Left) The system has a modularized design with repeated pentagon and hexagon shapes, to ensure interchangeability (Right) Optimized VGA camera positions to ensure uniform angles with respect to the dome center between each camera and all its neighbors (e.g., Camera i is a neighbor of Camera j).

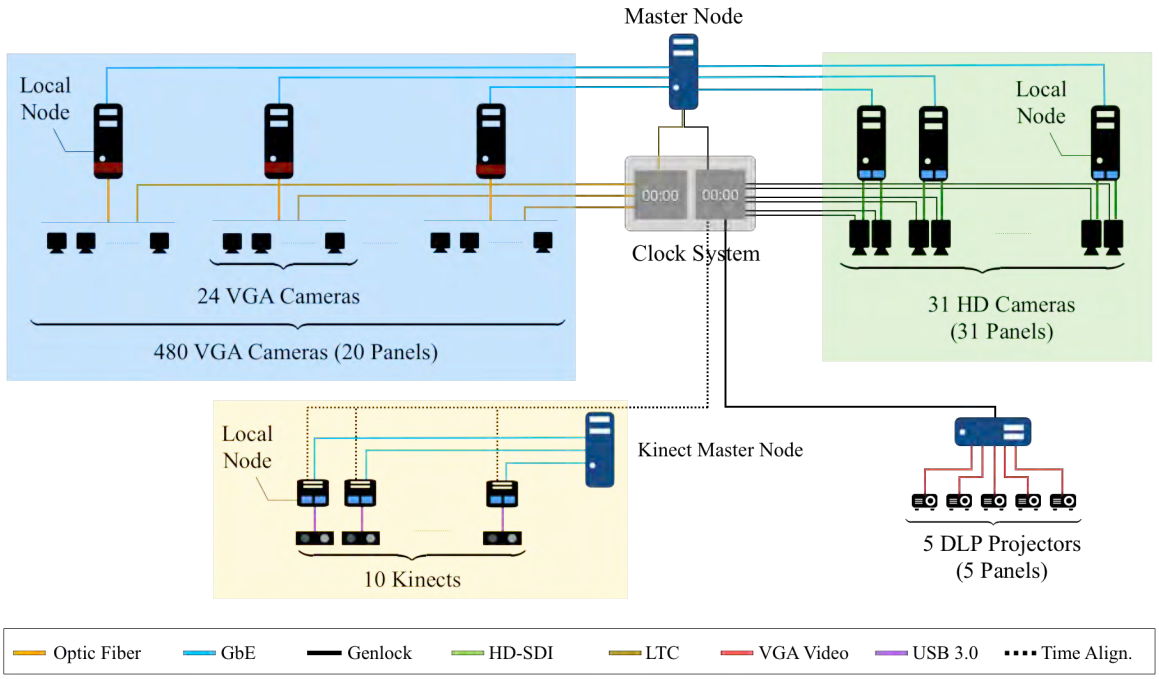


Fig. 2.4 Modularized system architecture. The system is composed of four sub-systems, depending on sensor types. The studio houses 480 VGA cameras, 31 HD cameras, 10 RGB+D sensors, and 5 DLP projectors. The 480 VGA cameras are synchronized by a clock system with 25 Hz, and HD cameras and projectors are synchronized by another clock system with 29.97 Hz. Both clocks are temporally aligned by recording them as a stereo signal. 10 RGB-D sensors are also time-aligned in the common time domain. All the sensors are spatially calibrated to the same coordinate system.

2.2 System Architecture

Figure 2.4 shows the architecture of our system. The panoptic studio is composed of four sub-systems: VGA camera system, HD camera system, RGB-D camera system, and projector system.

VGA Camera System: The 480 cameras are arranged modularly with 24 cameras in each of 20 standard hexagonal panels on the dome. Each module in each panel is managed by a Distributed Module Controller (DMC) that triggers all cameras in the module, consolidates the videos from cameras, and transmits the data to the local machine. Each individual camera has a global shutter CMOS sensor, with a fixed focal length of 4.5mm, capturing VGA (640×480) resolution images at 25Hz. The detailed configuration is shown in Figure 2.5.

Each panel produces an uncompressed video stream at 1.47 Gbps, and thus, the data-rate of the entire set of 480 cameras is approximately 29.4 Gbps. To handle this stream, the system pipeline has been designed with a modularized communication and control structure.

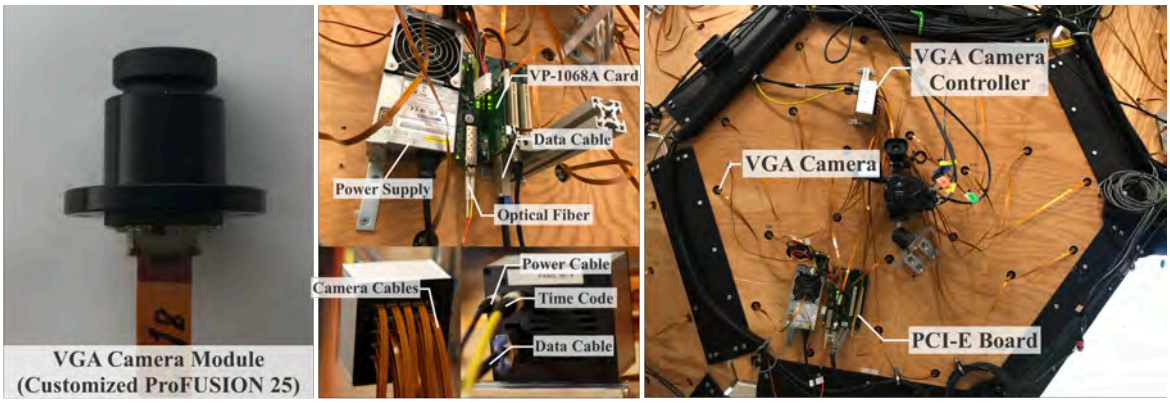


Fig. 2.5 A VGA camera module. We use a customized camera array with 24 VGA cameras as a single module. Each module is controlled by a camera controller that is connected to a small desktop machine. They are attached to the panel. The small desktop machine is connected to a single VGA node machine, to transfer captured data.

For each module, the clock generator sends a frame counter, trigger signal, and the pixel clock signal to each DMC associated with a panel. The DMC uses this timing information to initiate and synchronize capture of all cameras within the module. Upon trigger and exposure, each of the 24 camera heads transfers back image data via the camera interconnect to the DMC, which consolidates the image data and timing from all cameras. This composite data is then transferred via optical interconnect to the module node, where it is stored locally. The 20 local nodes for VGA camera system are shown in the left of the Figure 2.7. Each local node for a module has dual purpose: it serves as a distributed RAID storage unit[2] and participates as a multi-core computational node in a cluster. All the local nodes of our system are on a local network on a gigabit switch. The acquisition is controlled via a master node that a system operator can use to control all functions of the studio.

HD Camera System: HD cameras are also modularized and each pair of cameras are connected to a local node machine via SDI cables. We use Cannon XH G1s High Definition camcorders, each of which has 3 CCD sensors and captures scenes at 29.97 Hz with (1920×1080) resolution. Each HD camera is equipped with a Canon 4.5-90 mm f1.6-f3.5 HD resolving lens, which we fit with a wide angle converter (Canon WD-H72). All HD cameras are genlocked and time-coded, driven by an external clock. The details of the HD camera modules are shown in Figure 2.6. The data from a pair of HD cameras is transferred via HD-SDI to a local HD node. Each local node saves the data from two cameras to two

[2]Each node has 3 HDDs integrated as RAID-0 to have sufficient write speed without data loss, totaling 60 HDDs for 20 modules.

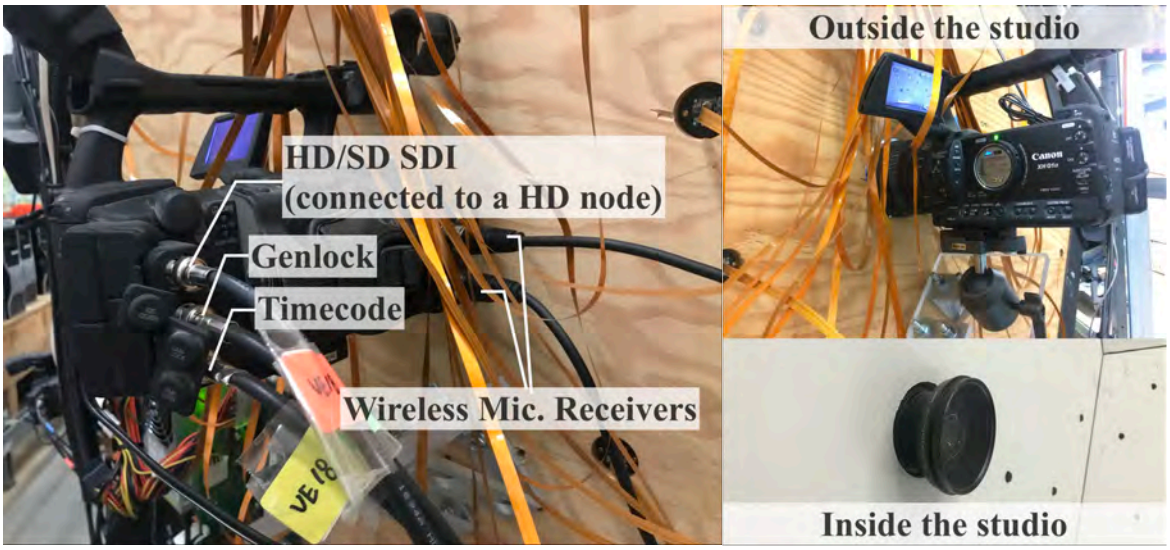


Fig. 2.6 An HD camera module. We use an off-the-shelf HD camcorder with external genlock and timecode signals for the synchronization. Two HD cameras are connected to a single HD machine.

SSD storage units respectively. 16 HD nodes are dedicated for 31 HD cameras, as shown in the right of the Figure 2.7.

RGB+D Sensor System: Each RGB+D sensor is connected to a dedicated capture node that is mounted on the dome exterior. To capture at rates of approximately 30 Hz, the nodes are equipped with two SSD drives each and store color, depth, and infrared frames as well as body and face detections from the Kinect SDK. A separate master node controls and coordinates the 10 capture nodes via the local network. The details of this subsystem are described in the thesis work of Simon [56].

Data Size and Storage System: The data size of a minute of data from entire sensors is about 531 GB. The detailed data size for each sub-system is summarized in Figure 2.8a and an example scene captured by all the sensors are shown in the Figure 2.9. The data from each sensor module is first saved to a local storage system in each connected node. We use different local storage solutions based on the data size generated by each sensor type. For VGA module with 25 cameras, we use 3 HDDs integrated as RAID-0. This solution provides fast writing speed with a sufficient storage size to store more than two hours of data with relatively low cost[3]. For each HD camera, we dedicate a 1TB SSD, which can store more than 2 hours of data. And for Kinect module, we dedicate two 1TB SSDs. All sensor data

[3]However, as known, the RAID-0 system is unstable with frequent RAID failures.

20 VGA Nodes 16 HD Nodes

Fig. 2.7 VGA subsystem is controlled by 20 node machines, and HD subsystem is controlled by 16 node machines. All nodes are connected and controlled by a master node to initiate the capture.

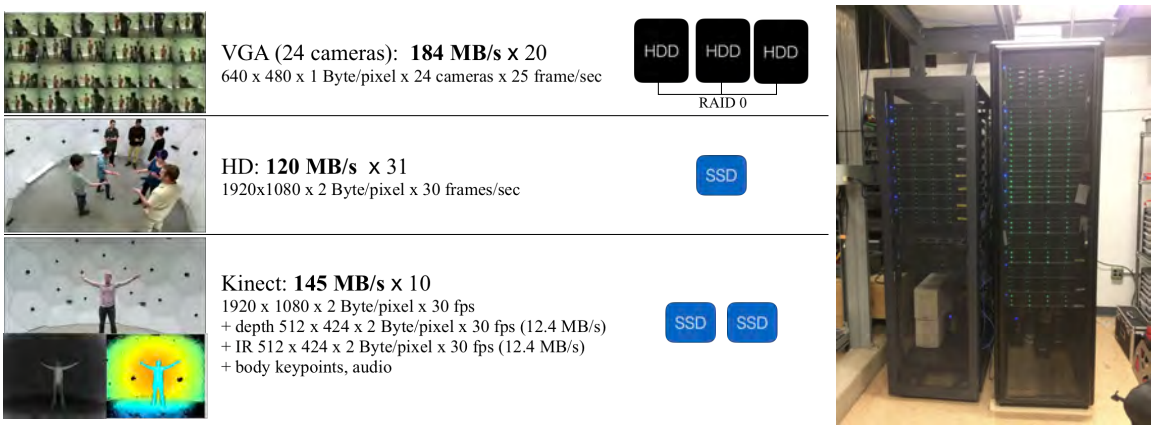
in the local storages are transferred to NAS for long-term storage, shown in Figure 2.8b, by a backup script. Currently, the Panoptic Studio system has about 1.5 PB long-term storage space.

2.3 Temporal Calibration for Heterogeneous Sensors

Synchronizing the cameras is necessary to use geometric constraints (such as triangulation) across multiple views. In our system, we use hardware clocks to trigger cameras at the same time. Because the frame rates of the VGA and HD cameras are different (25 fps and 29.97 fps respectively) we use two separate hardware clocks to achieve shutter-level synchronization among all VGA cameras and among all HD cameras, respectively. To precisely align both time references, we record the timecode signals generated from the two clocks as a single stereo audio signal, which we then decode to obtain a precise alignment at sub-millisecond accuracy, as shown in Figure 2.10.

Time alignment with the Kinect v2 streams (RGB and depth) is achieved with a small hardware modification: each Kinect's microphone array is rewired to instead record an LTC timecode signal[4]. This timecode signal is the same that is produced by the genlock and timecode generator used to synchronize the HD cameras, and is distributed to each Kinect via a distribution amplifier. We process the Kinect audio to decode the LTC timecode, yielding temporal alignment between the recorded Kinect data—which is timestamped by the capture

[4] As a result of this modification, microphone output on the Kinects is therefore discarded. More details about this hardware modification are available upon request.



(a) Data size and local storage for each node

(b) NAS sotrage system

Fig. 2.8 (a) Overall data size the Panoptic Studio generates per each second, and storage solutions for local nodes (b) Network Attached Storage (NAS) system to store the data for long-term.

API for accurate relative timing between color, depth, and audio frames—and the HD video frames. Empirically, we have confirmed the temporal alignment obtained by this method to be of at least millisecond accuracy.

2.4 Spatial Calibration

We use Structure from Motion (SfM) to calibrate all of the 521 cameras. To effectively generate feature points for SfM, five projectors are also installed on the geodesic dome. For calibration, they project a random pattern on a white structure as shown in the Figure 2.11, and multiple scenes (typically three) are captured by moving the structure within the dome. We perform SfM for each scene separately and perform a bundle adjustment by merging all the matches from each scene. We use a VisualSfM software [57] with 1 distortion parameter to produce an initial estimate and a set of candidate correspondences, and subsequently run our own bundle adjustment implementation with 5 distortion parameters for the final refinement. The computation time is about 12 hours with 6 scenes (521 images for each) using a 6 core machine. In this calibration process, we only use the color cameras of Kinects. We additionally calibrate the transformation between the color and depth sensor for each Kinect with a standard checkerboard pattern, placing all cameras in alignment within a global coordinate frame.

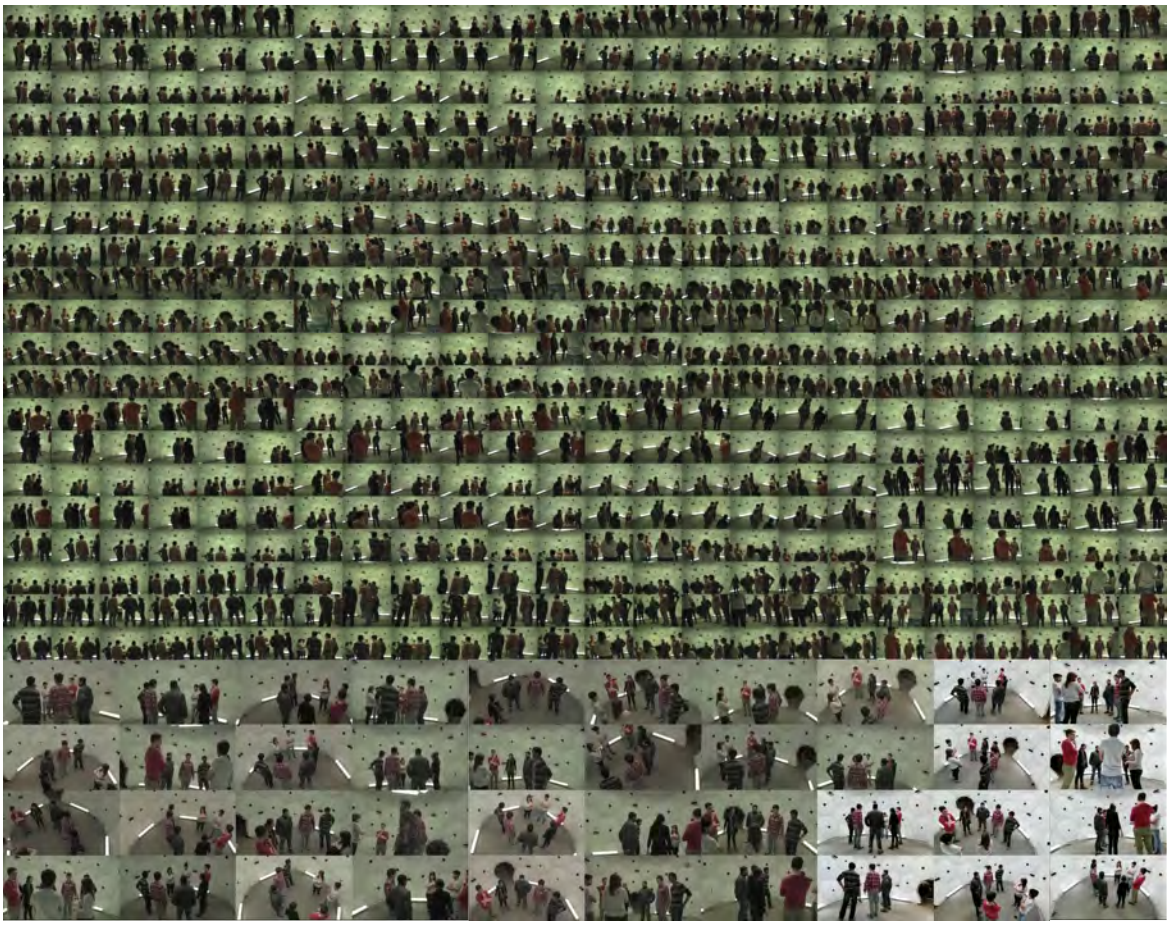


Fig. 2.9 An example scene from 520 camera views of the Panoptic Studio, from 480 VGA cameras, 30 HD cameras, and 10 RGB cameras of the RGB+D sensors.

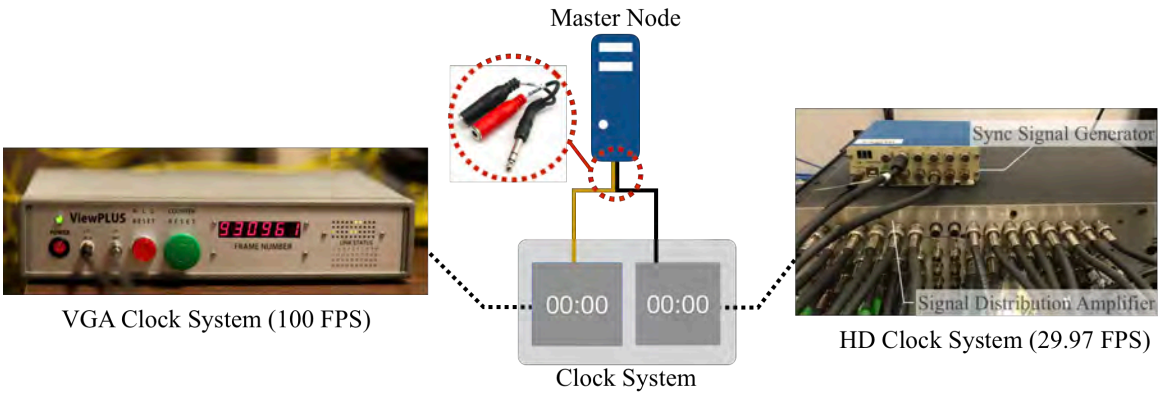


Fig. 2.10 Time alignment between VGA camera system and HD camera system. The VGA clock generates 100 Hz time signals, which is downsampled to 25 Hz in VGA cameras, while the HD clock generates signals in 29.97 Hz. We connect them as a stereo audio signal and record it during the capture, which can be decoded afterward for temporal alignment.

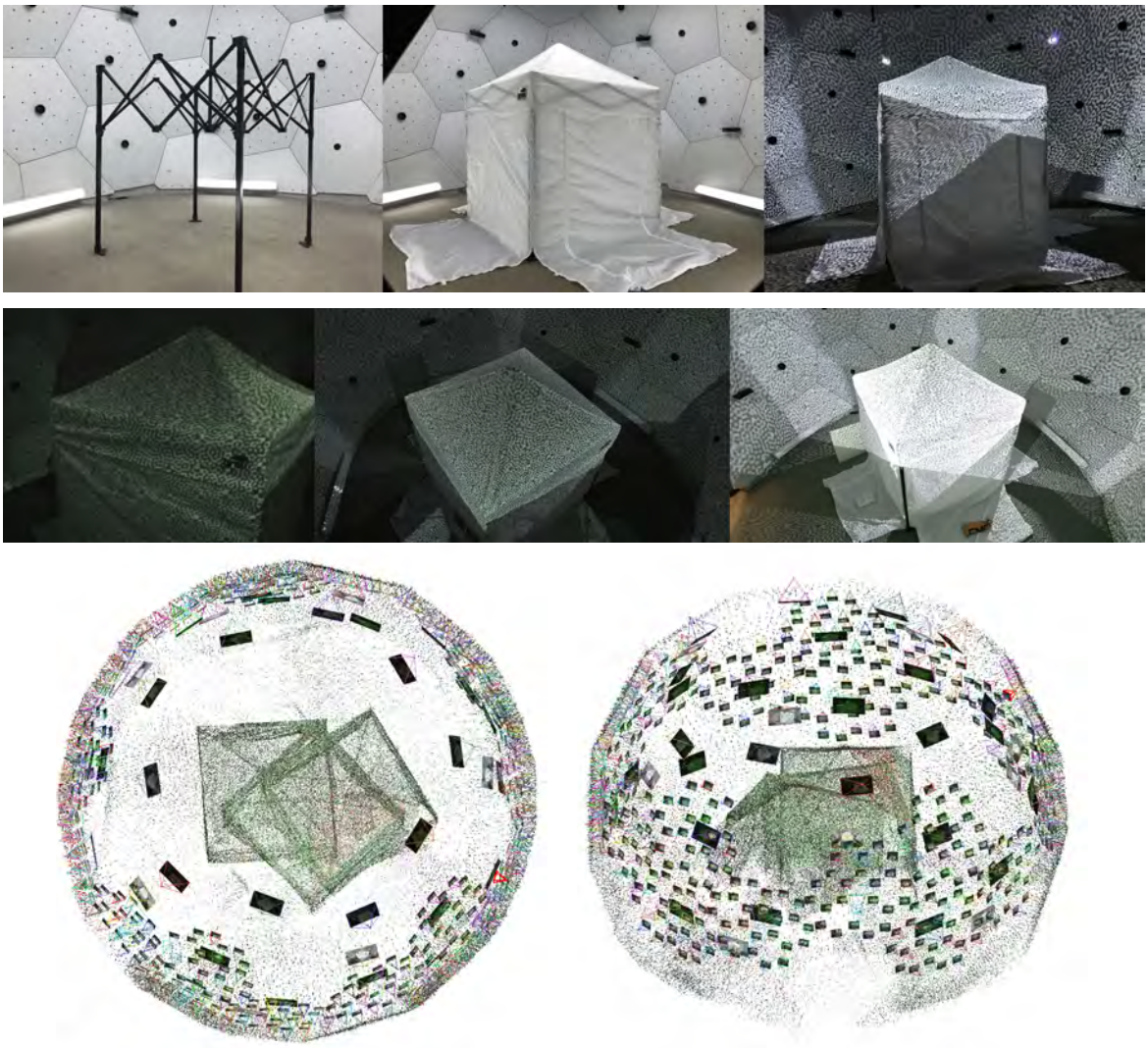


Fig. 2.11 For efficient spatial calibration process, we use a portable folding tent structure by projecting random patterns from 5 DLP projectors. (Top row) the portable tent and projected patterns; (Middle left) a VGA view; (Middle center) an HD view; (Middle right) a Kinect view from its color camera; (Bottom row) Visualizations of 3D camera localization and a reconstructed 3D point cloud.

2.5 Lighting, Audio Recording, and Capture Softwares

Lighting: We use a relatively low-cost solution for lighting by installing fluorescent lamps on the ceiling and floors, as shown in Figure 2.12. The floor lights are important to reduce the shadow issues of the scenes.
Audio Recording: Audio from microphones can be easily recorded and synchronized with cameras by connecting microphones to HD cameras. Each HD camera has two mono microphone inputs, by which the audio signals are saved as a stereo audio file. We installed one microphone on the floor and two microphones on the ceiling. We also use 20 wireless lapel microphones by connecting the wireless receivers to HD cameras, as shown in Figure 2.13. Each lapel microphone is assigned to a subject during the social interaction capture.
Capture Softwares: As an important part of the system, we built several custom software systems to control the capture process. Each subsystem is controlled by a separate capture software. We built a shell script to initiate both VGA and HD capture software along with the sync signal recording together in the master node. For VGA, we built a server-client software system, where a local server is opened and runs at each VGA node to communicate with the master node. In the master node, an operator can check the status of each VGA module, visualize the camera views in real-time streaming, and initiate the capture. The VGA capture tool is shown in the Figure 2.14. For the HD, we use a secure shell (SSH) protocol to automatically initiate the capture software at each machine. Kinect is separately controlled by the Kinect master node.

After the capture, the captured scenes from VGA and HD sensors can be visualized in our viewer, as shown in Figure 2.15, where the viewer loads capture data by remotely connecting all local storages.

2.6 Discussion

We found three limitations in our hardware design which can be reconsidered for follow-up research. The first is the incompatible frame rates among heterogeneous sensors, especially between HD cameras and VGA cameras, which makes it hard to fuse them for 3D reconstruction. Due to this reason, we use each subsystem for different purposes; The VGA system is mainly used to reconstruct 3D body motions, and HD system is used to reconstruct the face and hand motions. In the end, all reconstruction results are combined via an interpolation. A better synchronization may enable much easier solutions to leverage all sensors together. The second issue is that all the camera views mainly focus on the center of the dome and,

Fig. 2.12 We use a relatively low-cost solution for lighting by installing fluorescent lamps on the ceiling and the floor.

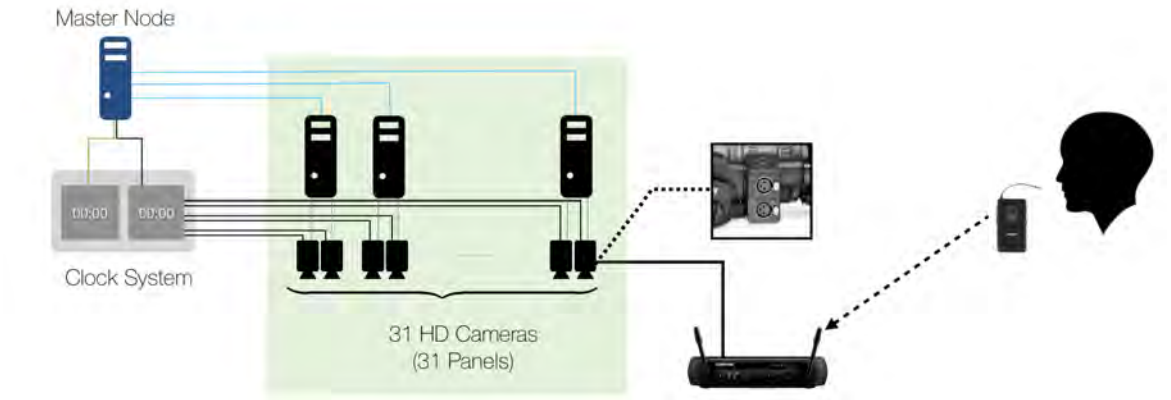


Fig. 2.13 Each HD camera has two mono microphone inputs, by which the audio signals are saved as a stereo audio file. We also use 20 wireless lapel microphones by connecting the wireless receivers to HD cameras.

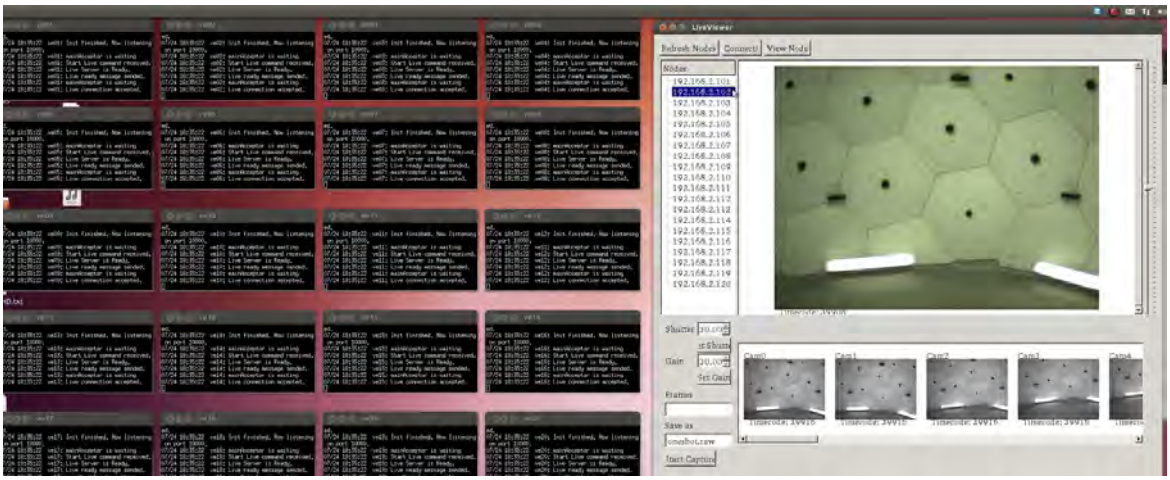


Fig. 2.14 A capture software for VGA sub-system. In the master node, users can check the status of each VGA module (black consoles on the left), visualize the camera views in real-time streaming (the GUI tool on the right), and initiate the capture.

thus, fewer views are available at the edges of the capture volume. Such design is ideal given the assumption that subjects are located at the center of the system, but we observe that sometimes people tend to stand near the walls during social interactions. An alternative direction would be to make cameras focus on random locations so that view coverage can be uniformly spread throughout the working volume. Last, we found that the spatial and temporal resolutions of cameras installed in the studio limit the reconstruction quality. Sensors with higher resolution and faster speed need to be considered in the follow-up research.

Fig. 2.15 A viewer to visualize the captured data from VGA and HD cameras.

Part II

Measuring Social Signals

“Measure what is measurable, and make measurable what is not so."

— Galileo Galilei

Chapter 3

Measuring Dynamic Dense 3D Surface Movements

Important messages during communication are often transmitted by subtle body movements. Accurately measuring such body movements in 3D across time is therefore important in analyzing social interactions. In this chapter, we aim at reconstructing dense 3D trajectories of dynamic 3D object, which we refer to as a *3D trajectory stream*. However, such video-based 3D motion reconstruction is challenging, as natural motion produces a greater occurrence of measurement loss due to occlusion and also causes artifacts in imagery (e.g., motion blur and texture deformation). Utilizing a large number of cameras can address these challenges, because it is likely to (1) narrow the average baseline between nearby cameras, (2) reduce the occurrence of occlusion, and (3) provide robustness to measurement noise due to the surplus views. However, previous approaches are unable to fully leverage the increasing number of views to improve 3D tracking performance (in terms of the average length of reconstructed trajectories, the density of the trajectories, and the accuracy of localization). The principal cause of failure emerges from errors in reasoning about the time-varying *visibility* of dynamic 3D points. Poor visibility reasoning severely affects tracking performance, as an algorithm cannot benefit from an alternate viewpoint if it is unaware that the point is visible in the alternate view. Furthermore, an erroneous conclusion that a point is visible in a camera can bias the reconstruction, often producing a characteristic “jump" artifact where a point assumes the identity of a different location.

In this chapter, we demonstrate a method that precise inference of point visibility allows reconstruction algorithms to fully leverage large numbers of views to produce longer 3D trajectories with higher accuracy. In particular, our core algorithmic contributions in this chapter are: (1) the use of motion consistency as a cue for the visibility of moving points; (2) the use of viewpoint regularity as a prior and a measure for viewpoint proximity; and

(3) a maximum a posteriori (MAP) estimate for visibility estimation by probabilistically incorporating these cues with photometric and geometric consistency. We report empirical performance in reconstructing 3D motion captured by 480 VGA cameras[1] in scenes that contains significant occlusion, large displacement, and changes in the topology of the scene.

3.1 Related Work

Dynamic 3D reconstruction approaches can be broadly categorized in methods that use silhouettes for reconstruction (e.g., [20, 58–61]) and methods that use correspondence for reconstruction (e.g., [62–64]. Silhouette-based approaches typically use visual hulls to produce highly dense reconstruction, but require subsequent processing to estimate 3D trajectories [20, 59]. Surface matching algorithms are used to provide dense correspondences between consecutive frames [65–67]. In these approaches, mesh models in each frame are independently generated using shape-from-silhouette techniques, and sparse matching between key mesh vertexes are performed using various cues such as shape and appearance features. Dense matching is then carried out based on the sparse matches using a regularized cost function based on geodesic distance. The accuracy of motion estimation depends highly on the initial surface and texture, and is limited by the vertex resolution. Silhouette-based methods also require stationary cameras to be able to estimate accurate silhouettes.

In comparison to silhouette-based reconstruction approaches, correspondence-based methods produce sparser reconstructions, but do not require stationary cameras and can directly produce 3D trajectories. Among correspondence-based methods, perhaps the most related approaches are scene flow reconstruction methods, introduced by Vedula et al. [62]. Independently estimated 2D optical flow from multiple calibrated cameras was triangulated to generate the 3D flow, assuming that visibility was given a priori via reconstructed object shape. Several subsequent algorithms also have been proposed to recover both shape (depth) and motion simultaneously [68–70]. The basic assumption in these approaches is brightness constancy (or photometric consistency), which is used to determine the correspondences across views; spatial regularization is used to condition the optimization and reduce the noise. While these approaches represent the target as a 3D point, other approaches use richer 3D representation such as dynamic surfels [63, 64] or meshes [26]. Mesh-based approaches have demonstrated robust results, producing trajectories of longer duration, but at the cost of assuming a fixed topology with a known mesh, and through the use of regularization.

Typically, in previous work, only a small number of cameras are considered. In scene flow approaches, stereo cameras are usually used, and other approaches also use at most

[1] This research was conducted before we build the HD and RGB+D sensors in the Panoptic Studio

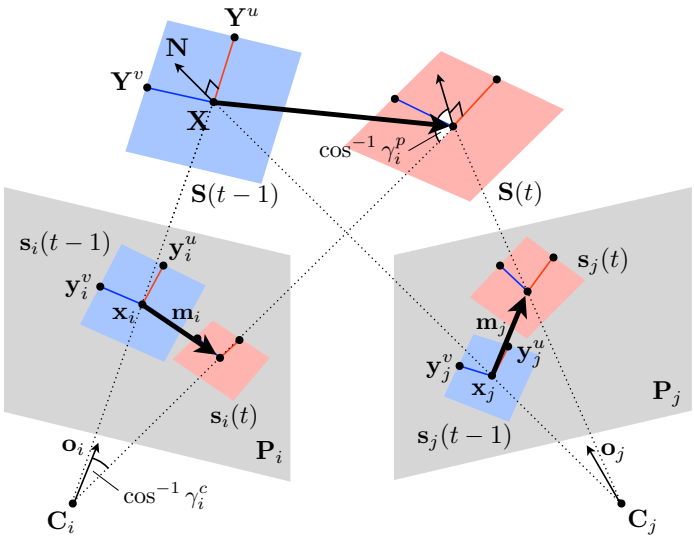


Fig. 3.1 The motion of a patch between time $t-1$ and t is reconstructed from multiple cameras.

10 to 20 cameras (17 by Vedula et al. [62], 22 by Furukawa and Ponce [26], 8 by Huguet and Devernay [70], 7 by Carceroni and Kutalakos [63]). At this scale, information loss due to motion blur, texture deformation, occlusion, and self-occlusion are severe, and therefore necessitate significant spatiotemporal regularization of reconstructions. In most algorithms, precise camera visibility information is not considered, because the noise from a small number of outlier cameras can be ignored. Camera visibility is either assumed to be given by the 3D reconstruction algorithm [62] or handled by a robust estimator [69, 64, 68, 71]. Patch-based methods use photometric consistency to determine visibility by comparing the texture across views [63, 64, 26]. However, these approaches require the texture of the 3D patch, which depends heavily on the accuracy of the recovered patch shape.

3.2 Notation

Our algorithm takes, as input, image sequences from N calibrated and synchronized cameras over F frames and produces, as output, 3D trajectories of P moving points with their instantaneous orientations and associated visibility in each camera frame. Since the method is applied to each point independently, we consider only a single point here to simplify the exposition.

As shown in Figure 3.1, we track a parallelogram patch centered on a target 3D point $\mathbf{X} \in \mathbb{R}^3$, whose extent is defined by two additional points $\mathbf{Y}^u$ and $\mathbf{Y}^v \in \mathbb{R}^3$. The texture

information $\mathbf{Q} \in \mathbb{R}^m$ associated with the patch is defined by a unit vector concatenating normalized intensity values at a fixed number of grid positions on the patch, where m is the number positions in the grid[2].

The patch $\mathbf{S}(t)$ is denoted by the set $\{\mathbf{X}(t), \mathbf{Y}^u(t), \mathbf{Y}^v(t), \mathbf{Q}(t)\}$, which is associated with the camera visibility set $\mathbf{V}(t) = \{\mathbf{v}_1(t), \cdots, \mathbf{v}_N(t)\}$, where $\mathbf{v}_i(t)$ is a binary value representing visibility with respect to the i^{th} camera. A 3D point is projected onto the i^{th} camera associated with a 3×4 projection matrix $\mathbf{P}_i$. The projection matrix is parametrized by a camera center vector $\mathbf{C}_i \in \mathbb{R}^3$ and a 3×3 rotation matrix $\mathbf{R}_i \in SO(3)$. The "look-at" vector $\mathbf{o}_i$ is aligned with the z-axis of the camera, i.e., the third column of $\mathbf{R}_i^\mathsf{T}$.

The 3D patch is projected onto the camera plane to form the projected patch $\mathbf{s}_i(t) = \{\mathbf{x}_i(t), \mathbf{y}_i^u(t), \mathbf{y}_i^v(t), \mathbf{q}_i(t)\}$, where $\mathbf{x}_i(t)$, $\mathbf{y}^u(t)$, and $\mathbf{y}^v(t) \in \mathbb{R}^2$ are the projected points, i.e., $\widehat{\mathbf{x}}_i(t) \cong \mathbf{P}_i\widehat{\mathbf{X}}(t)$, $\widehat{\mathbf{y}}_i^u(t) \cong \mathbf{P}_i\widehat{\mathbf{Y}}^u(t)$, and $\widehat{\mathbf{y}}_i^v(t) \cong \mathbf{P}_i\widehat{\mathbf{Y}}^v(t)$, where $\widehat{\cdot}$ is the homogeneous coordinate representation of each vector. $\mathbf{q}_i \in \mathbb{R}^m$ is the texture information of the projected patch, which is defined by a concatenation of all the intensities from the i^{th}camera, corresponding to the projected grid positions of $\mathbf{S}$, and normalized as in Equation (3.1). Ideally, $\mathbf{Q} = \mathbf{q}_i$ if the 3D patch $\mathbf{S}$ is visible from the i^{th} camera, discounting illumination variation. We denote $\mathbf{m}_i$ as 2D optical flow at $\mathbf{x}_i(t-1)$ in the i^{th} camera, as shown in Figure 3.1.

The relationship between the i^{th} camera and patch can be defined by the co-visibility set $\mathbf{\Gamma}_i = \{\gamma_i^c, \gamma_i^p\}$, where

$$\gamma_i^c = \frac{(\mathbf{X} - \mathbf{C}_i)^\mathsf{T}\mathbf{o}_i}{||\mathbf{X} - \mathbf{C}_i||} \quad and \quad \gamma_i^p = \frac{(\mathbf{C}_i - \mathbf{X})^\mathsf{T}\mathbf{N}}{||\mathbf{C}_i - \mathbf{X}||},$$

γ_i^c encodes the angle cosine of the patch location with respect to the camera "look-at" vector $\mathbf{o}_i$ and γ_i^p encodes the angle cosine of the camera location with respect to the 3D patch normal $\mathbf{N}$.

3.3 Overview

At the initial time instance t_0, a target 3D patch is reconstructed and, over time, the algorithm alternately estimates the patch position and normal and its visibility with respect to all

[2]The texture vector $\mathbf{Q}$ is normalized as follows:

$$\mathbf{Q} = \frac{1}{\sqrt{\sum_{j=1}^m (Q_j - \overline{Q})^2}} \begin{bmatrix} Q_1 - \overline{Q} \\ \vdots \\ Q_m - \overline{Q} \end{bmatrix} \tag{3.1}$$

where $\overline{Q} = \sum_{j=1}^m Q_j/m$ and Q_j is the j^{th} intensity value of the texture.

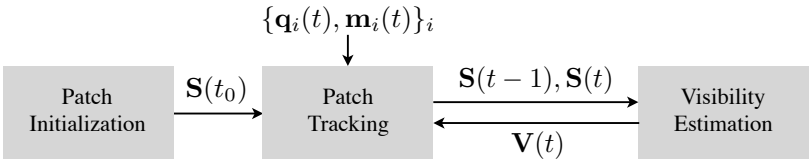


Fig. 3.2 Overview of patch tracking and visibility estimation.

cameras. It should be noted that t_0 can be any arbitrary frame and that the tracking and the visibility computation are performed both forwards and backwards in time from t_0. We consider only forward tracking, from $t-1$ to t to simplify the description. The flow chart of our algorithm is shown in Figure 3.2.

Patch Initialization. Given the images from different cameras at the same time instance t_0, the algorithm reconstructs 3D points by matching features and triangulates them within a RANSAC framework. A 3D patch centered on $\mathbf{X}$ is reconstructed by maximizing the photometric consistency among the cameras where the patch is visible[3]. This initializes $\mathbf{S}(t_0)$ and $\mathbf{V}(t_0)$.

Patch Tracking. Given the previously obtained 3D patch $\mathbf{S}(t-1)$ and visibility $\mathbf{V}(t-1)$, the algorithm estimates the next 3D patch $\mathbf{S}(t)$ based on 2D optical flow in the cameras defined by $\mathbf{V}(t-1)$. For the i^{th} camera in $\mathbf{V}(t-1)$, optical flow [73] is estimated at multiple scales at the points $\mathbf{x}_i(t-1)$, $\mathbf{y}_i^u(t-1)$, and $\mathbf{y}_i^v(t-1)$. To eliminate unreliable flow, a backward-forward consistency check [74] is performed for flow at each scale and only the most reliable flow is retained. The next 3D positions, $\mathbf{X}(t)$, $\mathbf{Y}^u(t)$, and $\mathbf{Y}^v(t)$, are estimated by triangulating optical flow outputs within a RANSAC framework. The RANSAC process is crucial since $\mathbf{V}(t-1)$ may not be valid anymore at time t, due to motion. After RANSAC, the normal is refined by maximizing the photometric consistency among the images that belong to the inliers of RANSAC, as in the patch initialization process.

Visibility Estimation. Based on the reconstructed $\mathbf{S}(t)$ and its motion from $\mathbf{S}(t-1)$, our approach finds the MAP estimate of the current visibility set $\mathbf{V}(t)$ by fusing photometric consistency, motion consistency, and geometric consistency, in conjunction with a Markov Random Field (MRF) prior. Typically, the tracking process is severely affected by false positive cameras where the target is not visible. Poor visibility reasoning at the RANSAC stage can cause a characteristic “jump" error to a different scene point, and also reduces

[3]The cameras that participate in RANSAC are used as an initial visible set, and the reference camera $\mathbf{P}_{\text{ref}}$ is selected as the one closest to the initial 3D point in the iniler set. A 3D patch centered on $\mathbf{X}$ is initialized as a fixed scale square patch (40mm×40mm), with $\mathbf{N}$ parallel to $\mathbf{o}_{ref}$. We refine the patch based on the method described by Furukawa and Ponce [72] and select a new reference camera as the one closest to the current patch normal. The corresponding visibility set is updated by selecting cameras that have higher Normalized Cross Correlation (NCC) score than a threshold compared to $\mathbf{P}_{\text{ref}}$. Within the patch initialization process, the normal refinement and visibility update are iterated.

the normal refinement performance causing frequent local minima during the optimization process. Our precise visibility estimation results in longer trajectories of higher accuracy.

Patch tracking and visibility estimation are interdependent processes. At each time instance, we can iterate these two procedures until convergence; in practice, a single iteration is usually sufficient.

3.4 Visibility Estimation

In this section, we present a method to compute the maximum a posteriori (MAP) estimate of visibility $\mathbf{V}$ using photometric consistency, motion consistency, and geometric consistency, with a proximity prior. These cues are represented using 2D texture $\{\mathbf{q}_i\}_{i=1}^N$, 2D optical flow $\{\mathbf{m}_i\}_{i=1}^N$, and the co-visibility set $\{\mathbf{\Gamma}_i\}_{i=1}^N$. Given these cues and by applying Bayes theorem, the probability of visibility is

$$\begin{aligned} &P(\mathbf{V}|\mathbf{q}_1,\mathbf{m}_1,\mathbf{\Gamma}_1,\cdots,\mathbf{q}_N,\mathbf{m}_N,\mathbf{\Gamma}_N) \\ &\propto P(\mathbf{q}_1,\mathbf{m}_1,\mathbf{\Gamma}_1,\cdots,\mathbf{q}_N,\mathbf{m}_N,\mathbf{\Gamma}_N|\mathbf{V})P(\mathbf{V}). \end{aligned}$$

Given the visibility of each camera, we assume that (1) the cues in that camera are conditionally independent to the cues in other cameras and the visibility of other cameras, (2) that each cue within the same camera is conditionally independent to each other. The probability can be written as

$$\left(\prod_{i=1}^{N} P(\mathbf{q}_i|\mathbf{v}_i)P(\mathbf{m}_i|\mathbf{v}_i)P(\mathbf{\Gamma}_i|\mathbf{v}_i)\right)P(\mathbf{V}). \tag{3.2}$$

The MAP estimate of visibility $\mathbf{V}^*$ can be obtained by maximizing the expression in Equation (3.2), i.e.,

$$\mathbf{V}^* = \underset{\mathbf{V}}{\operatorname{argmax}} \left(\prod_{i=1}^{N} P(\mathbf{q}_i|\mathbf{v}_i)P(\mathbf{m}_i|\mathbf{v}_i)P(\mathbf{\Gamma}_i|\mathbf{v}_i)\right)P(\mathbf{V}),$$

or equivalently,

$$\begin{aligned} \mathbf{V}^* = \underset{\mathbf{V}}{\operatorname{argmax}} \sum_{i=1}^{N} \log P(\mathbf{q}_i|\mathbf{v}_i) + \sum_{i=1}^{N} \log P(\mathbf{m}_i|\mathbf{v}_i) + \\ + \sum_{i=1}^{N} \log P(\mathbf{\Gamma}_i|\mathbf{v}_i) + \log P(\mathbf{V}). \end{aligned} \tag{3.3}$$

We describe the probability of each cue and the prior in the subsequent sub-sections, and compute the MAP estimate by finding the minimum cut of a capacitated graph over cameras [75].

3.4.1 Photometric consistency

Photometric consistency has been widely used for reasoning about visibility [76–78, 26, 64]. It measures the correlation between the texture $\mathbf{Q}$ of a 3D patch and the texture $\mathbf{q}_i$ of the corresponding patch in the i^{th} camera. Normalized Cross Correlation (NCC) is one such measure of photometric consistency, which is robust to illumination variation. Since $\mathbf{Q}$ and $\mathbf{q}_i$ are defined as normalized unit vectors by Equation (3.1), $\mathbf{Q}^\mathsf{T}\mathbf{q}_i$ measures the NCC. We model the probability distribution of $\mathbf{q}_i$ using a von Mises-Fisher distribution around $\mathbf{Q}$, i.e., $\mathbf{q}_i \sim \mathscr{V}(\mathbf{Q}, \kappa)$, which is defined by $\mathbf{Q}^\mathsf{T}\mathbf{q}_i$. κ is a concentration parameter that controls the degree of variation of the texture. Lower values of κ allows more variation between $\mathbf{Q}$ and $\mathbf{q}_i$. From the distribution, we can describe the logarithm of the probability of $\mathbf{q}_i$ given $\mathbf{v}_i$ as

$$\log P(\mathbf{q}_i|\mathbf{v}_i) \propto \kappa \mathbf{Q}^\mathsf{T}\mathbf{q}_i. \tag{3.4}$$

3.4.2 Motion Consistency

In dynamic scenes, motion is an informative cue for determining visibility. Given the 3D motion of a patch, the observed optical flow at the i^{th} camera must be consistent with the projected 3D motion of the target patch, if the patch is visible from the camera view. In other words, motion consistency requires that 2D optical flow $\mathbf{m}_i$ must be consistent with the projected displacement of the 3D motion $\mathbf{x}_i(t) - \mathbf{x}_i(t-1)$.

We model the probability distribution of $\mathbf{m}_i$ using a normal distribution around the projected 3D displacement, i.e., $\mathbf{m}_i \sim \mathscr{N}(\mathbf{x}_i(t) - \mathbf{x}_i(t-1), \sigma)$, where σ is the standard deviation capturing the certainty of the 3D motion estimation in pixel units. Therefore, the log likelihood can be written as

$$\log P(\mathbf{m}_i|\mathbf{v}_i) \propto -\frac{\|\mathbf{m}_i - (\mathbf{x}_i(t) - \mathbf{x}_i(t-1))\|^2}{2\sigma^2}. \tag{3.5}$$

Motion consistency is a necessary condition. We now characterize cases when the motion consistency cue is ambiguous. Let $\mathbf{X}(t)$ and $\mathbf{X}'(t)$ be two distinct points in 3D space. Motion

consistency cue is ambiguous if and only if the following two conditions hold:

$$\begin{aligned} \mathbf{P}_i\widehat{\mathbf{X}}(t) &\cong \mathbf{P}_i\widehat{\mathbf{X}}'(t) \\ \mathbf{P}_i\widehat{\mathbf{X}}(t+1) &\cong \mathbf{P}_i\widehat{\mathbf{X}}'(t+1) \end{aligned} \tag{3.6}$$

where $\|\mathbf{X}-\mathbf{C}_i\| > \|\mathbf{X}'-\mathbf{C}_i\|$, i.e., $\mathbf{X}'(t)$ occludes $\mathbf{X}(t)$ for the i^{th} camera. In a static scene, motion does not exist and thus, the motion consistency cue is always ambiguous because $\mathbf{X}(t) = \mathbf{X}(t+1)$ and $\mathbf{X}'(t) = \mathbf{X}'(t+1)$. Another case that occurs in practice is when the occluding patch and the occluded patch lie on a body undergoing global translational motion, under a camera that approaches orthographic projection.

We characterize the set of ambiguous motions where Equation (3.6) holds, assuming that $\mathbf{X}(t)$ and $\mathbf{X}'(t)$ undergo the same affine transform between frames, as

$$\begin{aligned} \mathbf{X}(t+1) &= \mathbf{A}\mathbf{X}(t)+\mathbf{a} \\ \mathbf{X}'(t+1) &= \mathbf{A}\mathbf{X}'(t)+\mathbf{a}, \end{aligned} \tag{3.7}$$

where $\mathbf{A} \in \mathbb{R}^{3\times3}$ and $\mathbf{a} \in \mathbb{R}^3$ represent a 3D affine transform. The motion consistency cue is ambiguous if and only if the following condition holds:

$$\mathbf{X} \in n\mathit{ull}\left([\mathbf{a}]_\times \mathbf{A}\right), \tag{3.8}$$

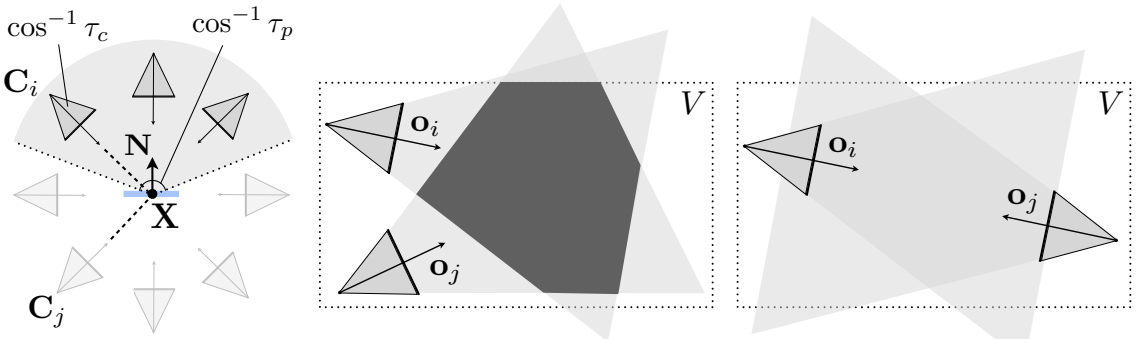
where $n\mathit{ull}(\cdot)$ is the null space of $\cdot$. See the Appendix for a proof. In ideal cases with infinite precision and zero measurement noise, this condition rarely occurs (if there is motion).

3.4.3 Geometric consistency

Oriented patches are only visible from cameras whose “look-at" vector $\mathbf{o}_i$ is in the opposite direction to the patch normal $\mathbf{N}$ and in front of it. We incorporate this geometric cue based on the co-visibility set $\mathbf{\Gamma}_i$ considering the camera position relative to the patch normal direction and the patch position relative to the camera “look-at" vector. The probability of $\mathbf{\Gamma}_i$, given visibility $\mathbf{v}_i$, can be written as

$$P(\mathbf{\Gamma}_i|\mathbf{v}_i) = \begin{cases} \frac{1}{(1-\tau_c)(1-\tau_p)} & \mathit{if}\ \gamma_i^c \geq \tau_c, \mathit{and}\ \gamma_i^p \geq \tau_p \\ 0 & \mathit{otherwise}, \end{cases} \tag{3.9}$$

where $\tau_c < 1$ is the cosine angle representing the field of view of the camera, and $\tau_p < 1$ is a threshold (cosine angle) to determine the angular visibility with respect to the patch normal.



(a) Geometric Consistency (b) $\int_V H(\mathbf{o}_i^\mathsf{T}\mathbf{o}_j)F_i(v)F_j(v)\mathrm{d}v$ (c) $H(\mathbf{o}_i^\mathsf{T}\mathbf{o}_j)=0$

Fig. 3.3 (a) The valid region filtered by γ_p and τ_p is shown as a shaded region. The angle limitation with respect to the $\mathbf{N}$ is computed as $\cos^{-1}\tau_p$. (b) g_s computed by two cameras are shown as a shaded polygon, where $\int_V \mathscr{H}(\mathbf{o}_i^\mathsf{T}\mathbf{o}_j)F_i(v)F_j(v)\mathrm{d}v > 0$. (c) An example where $\mathscr{H}(\mathbf{o}_i^\mathsf{T}\mathbf{o}_j)=0$ is shown. An oriented patch cannot be visible by two cameras facing each other simultaneously.

Figure 3.3(a) shows an example of the cue, where the shaded area represents the valid region according to τ_p.

3.4.4 Visibility Regularization Prior

Under a Markov Random Field prior over camera visibility, we decompose the joint probability of visibility $P(\mathbf{V})$ into pairwise probabilities, i.e.,

$$P(\mathbf{v}_1,\cdots,\mathbf{v}_N) = \prod_{i,j\in\mathscr{G}(i)} P(\mathbf{v}_i,\mathbf{v}_j), \tag{3.10}$$

where $\mathscr{G}(i)$ is the set of adjacent camera indices of the i^{th} camera. This decomposition captures the prior distribution of visibility, representing the prior that two cameras that have similar viewpoints are likely to have consistent visibility. This proximity constraint constitutes prior knowledge that can regularize noisy visibility when both photometric consistency and motion consistency cues are weak (e.g., due to motion blur in an individual camera). We model the log likelihood of the joint probability as follows:

$$\log P(\mathbf{v}_1,\cdots,\mathbf{v}_N) \propto \sum_{i,j\in\mathscr{G}(i)} g_s(\mathbf{v}_i,\mathbf{v}_j), \tag{3.11}$$

where g_s is defined by the cost between two cameras using the overlapping volume of the two camera frustums. This is estimated as follows:

$$g_{\mathrm{s}}(\mathbf{P}_i,\mathbf{P}_j)=\frac{\int_V \mathscr{H}(\mathbf{o}_i^{\mathsf{T}}\mathbf{o}_j)F_i(v)F_j(v)\mathrm{d}v}{\int_V F_i(v)+F_j(v)-F_i(v)F_j(v)\mathrm{d}v}, \tag{3.12}$$

where v is an infinitesimal volume in the working space V (see Figure 3.3(c)). $F_i(v)$ is a binary function defined as

$$F_i(v)=\begin{cases}1 & \mathrm{i}f\ v\ \mathrm{is}\ visible\ from\ the\ i^{th}\ camera\\ 0 & \mathrm{o}therwise.\end{cases} \tag{3.13}$$

$\mathscr{H}$ is a Heaviside step function to take into account a pair of cameras oriented in similar directions. Equation (3.11) captures the ratio between the volume of the intersections of camera frustums and the volume of the union of camera frustums. Figure 3.3(b) illustrates g_{s} where the shaded polygon represents $\int_V \mathscr{H}(\mathbf{o}_i^{\mathsf{T}}\mathbf{o}_j)F_i(v)F_j(v)\mathrm{d}v$, and Figure 3.3(c) shows an example where $\mathscr{H}(\mathbf{o}_i^{\mathsf{T}}\mathbf{o}_j)=0$.

In practice, we discretize the working volume using voxels and count the number of common voxels that are projected inside both cameras. This enables us to reward consistent visibilities in proximal cameras.

3.4.5 MAP Visibility Estimation via Graph Cuts

We incorporate Equations (3.4), (3.5), (3.9), and (3.11) into Equation (3.3) to find the MAP estimate of visibility $\mathbf{V}^*$ and, therefore, Equation (3.3) can be rewritten as:

$$\mathbf{V}^*=\underset{\mathbf{V}}{\operatorname{argmin}}\sum_{i=1}^{N}E_d(\mathbf{v}_i)+\sum_{i,j\in\mathscr{G}(i)}E_s(\mathbf{v}_i,\mathbf{v}_j), \tag{3.14}$$

where E_d encodes photometric consistency, motion consistency, and geometric consistency, and E_s encodes the prior between cameras.

$$E_d(\mathbf{v}_i)=\frac{\|\mathbf{m}_i-(\mathbf{x}_i(t)-\mathbf{x}_i(t-1))\|^2}{2\sigma^2}-\kappa\mathbf{Q}_i{}^{\mathsf{T}}\mathbf{q}_i+\delta(\mathbf{\Gamma}_i)$$

$$E_s(\mathbf{v}_i,\mathbf{v}_j)=\begin{cases}0 & \mathrm{i}f\ \mathbf{v}_i=\mathbf{v}_j\\ g_{\mathrm{s}}(\mathbf{P}_i,\mathbf{P}_j) & \mathrm{o}therwise,\end{cases}$$

where $\delta=\log(1-\tau_c)(1-\tau_p)$ if $\gamma_i^c>\tau_c$ and $\gamma_i^p>\tau_p$, or $\delta=\infty$, otherwise. This minimization problem can be optimally computed via graph cuts [75].

Table 3.1 Summary of the datasets.

Sequence	Frames	Duration	# of points	Av. traj. length
Circ. Movement	250	10.0 sec	10433	404.9 cm
Volleyball	210	8.4 sec	8422	326.4 cm
Bat Swing	200	8.0 sec	3849	224.1 cm
Falling Boxes	160	6.4 sec	17934	164.7 cm
Confetti	200	8.0 sec	10345	103.0 cm
Fluid Motion	200	8.0 sec	3153	123.1 cm

3.5 Results

We evaluate our algorithm on a variety of challenging scenes in the presence of significant occlusion (Circular Movement and Falling Boxes), large displacement (Confetti and Fluid motion), and topological change (Falling boxes and Volleyball). Our visibility estimation enables us to better leverage a large number of cameras in producing accurate and long trajectories. The dataset used in the evaluation is summarized in Table 3.1 and is available on the project website. The sequences were captured at the CMU Panoptic Studio [53] containing 480 cameras capturing 640×480 video at 25 Hz. The cameras are extrinsically and intrinsically calibrated, and are synchronized via an external clock.

3.5.1 Quantitative Evaluation

Visibility Estimation Accuracy. We select an arbitrary patch in the Circular Movement sequence reconstructed at a time instance, and manually generate ground-truth visibility data at each sampled time instance by selecting cameras where the target patch is visible. We compare our visibility estimation method (MAP) against a baseline method based on photometric consistency alone, which is a cue commonly used by previous approaches [63, 64, 26]. Visibility estimation results generated from each method at a time instance are visualized in Figure 3.4. As a criterion, we compute the true positive detection rate between the ground truth data and $\mathbf{V}(t)$ estimated by both methods. The true positive rate from each method is shown in Figure 3.4, demonstrating that our method outperforms the baseline method by a significant margin.

Tracking Accuracy and Length. We evaluate our method considering both tracking accuracy and trajectory length. Inspired by the evaluation criterion proposed by Furukawa and Ponce [26], a test sequence is generated by appending it at the end of itself in reverse order, and the tracking algorithm is performed on the generated sequence. The tracked patches must return back to the original position, if tracking is accurate. In this experiment, the

3D error is defined by the 3D distance between initial and the final locations of the target point. We generate five test sequences using the Circular Movement sequence by changing the duration (10 to 50 frames) from a fixed initial frame. For the evaluation, we count the number of successfully reconstructed trajectories that have less than 2 cm drift error. Figure 3.5a shows a histogram of the number of trajectories using 480 cameras. Our MAP estimate method outperforms the method based on photometric consistency in terms of both number of trajectories and length of trajectories. We also perform experiments with different number of cameras by uniformly sampling cameras to examine its impact on tracking success rate. Figure 3.5b shows how our method leverages a large number of cameras. Note that the number of successfully tracked trajectories increases faster than the method based on photometric consistency.

3.5.2 Qualitative Evaluation

Visibility Boundary. We qualitatively demonstrate the performance of our MAP visibility estimation using the sequence by illustrating cameras in the visibility set in 3D, and showing the projection of the target patch in the images, as shown in Figure 3.6. This result shows a clean visibility boundary, showing the occluded views by the baseball bat.

3D Trajectory Recontruction. We generate an initial patch cloud for a selected time instance, and perform forwards and backwards patch tracking, up to 150 frames, for all the sequences summarized in Table 3.1. Figure 3.7 shows the reconstructed trajectories. The reconstructed time instances are color coded. Note that our method can be applied multiple times to different time instances to increase the density of the trajectories.

Circular movement: Three people rotate around the person at the center (Figure 3.7a). This experiment is used to evaluate our method in terms of visibility reasoning

Volleyball: Two people play volleyball (Figure 3.7b). We demonstrate an event where motion is fast and occlusion is severe. We are able to reconstruct the trajectories of the ball and players.

: A person swings a baseball bat. The reconstructed long trajectories can provide a computational basis for sport analytics, capturing subtle motion (Figure 3.7c).

Falling boxes: A person collides with stacked boxes and the boxes collapse. The scene includes severe occlusion and topological change of the structure (Figure 3.7d).

Confetti: A person throws confetti in the air. 3D reconstruction of such sequences is challenging because of occlusion and appearance changes. Visibility estimation is challenging as the confetti are small and their appearance changes abruptly (Figure 3.7e).

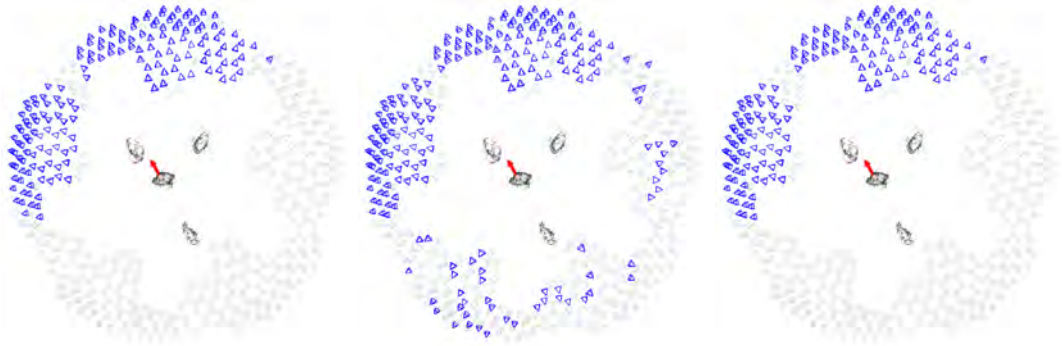
Fluid motion: We generate turbulent flow in a room using a fan and small confetti (Figure 3.7e)[4].

3.6 Summary

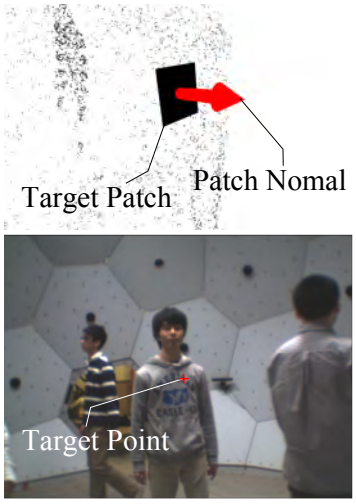
In this chapter, we present a method to reason about the time-varying visibility for 3D trajectory reconstruction to leverage the large number of views in the Panoptic Studio. We address novel cues (motion consistency, geometric consistency, and visibility regularization prior) for visibility estimation, and fuse them with the commonly used photometric consistency cue, within a MAP estimation framework. We demonstrate that our algorithm provides a more accurate visibility and, consequently, produces longer and denser 3D trajectories than a baseline using only photometric consistency. Unlike the photometric consistency cue, The motion consistency cue is complementary to the photometric cue, as it does not require the texture and the explicit 3D shape of the target 3D patch. Although the motion consistency cue can be ambiguous, this ambiguity, in practice, usually occurs for the cameras behind the target patch when the whole object body (including the patch) undergoes pure translation; this case is handled well by the geometric consistency of the patch and camera.

A key benefit of our approach is that it does not use any spatial or temporal regularization over the position of the point—the regularization used in our approach is over visibility. This results in “faithful" reconstruction of 3D point motion that is not biased or smoothed out by prior models of deformation, and this advantage is essential in measuring objective social signals in social situations.

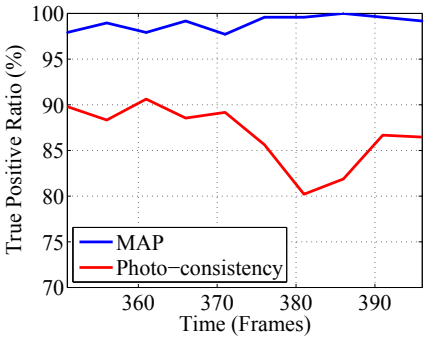
[4]For this result, we turned off geometric consistency by setting $\tau_c = 0$ and $\tau_p = 0$, as the objects are well approximated by planes.



(a) Ground truth (b) Photo-consistency (c) MAP

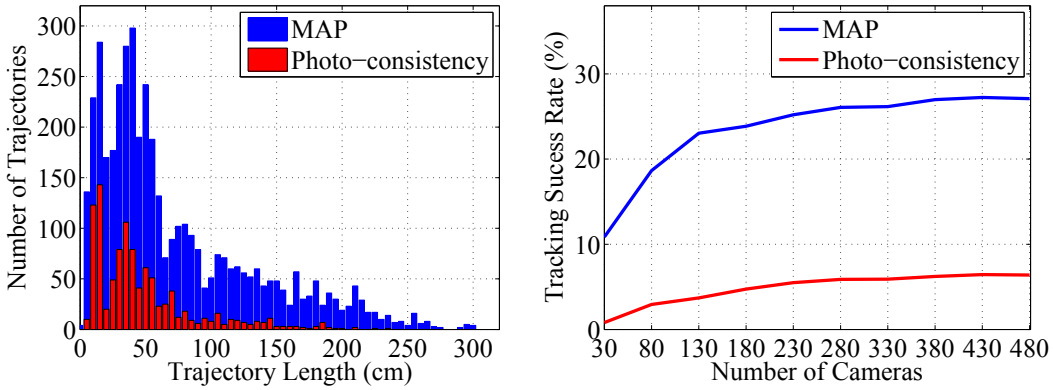


(d) Selected patch



(e) Visibility Accuracy Comparison

Fig. 3.4 The red arrow denotes the normal vector of the selected patch. The pyramid structures represent camera poses, where blue cameras belong to the visible set (we warp the camera positions for better visualization). (a) The selected patch is shown in 3D and 2D image. (b) We manually generate ground truth visibility. (c) Visibility estimated by the baseline. (d) Visibility estimated by our method. (e) We compare accuracy of visibility estimates of both methods.



(a) Tracking Performance (b) Number of cameras

Fig. 3.5 (a) Our MAP estimate outperforms the baseline method in terms of the number of trajectories and the length of trajectories. (b) Our method leverages the large number of views, and shows a faster increasing curve than the baseline method.

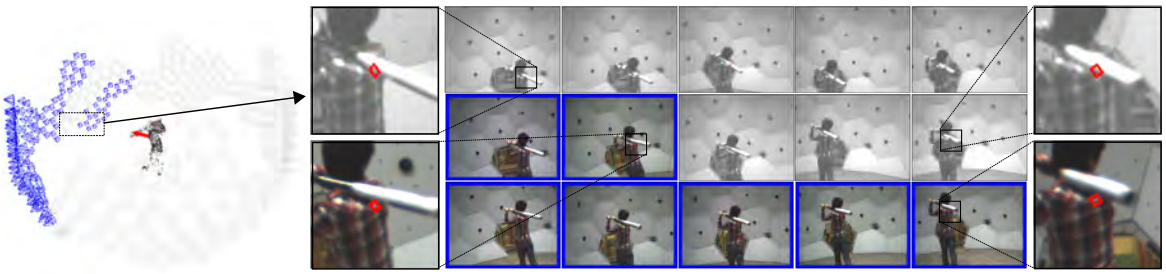
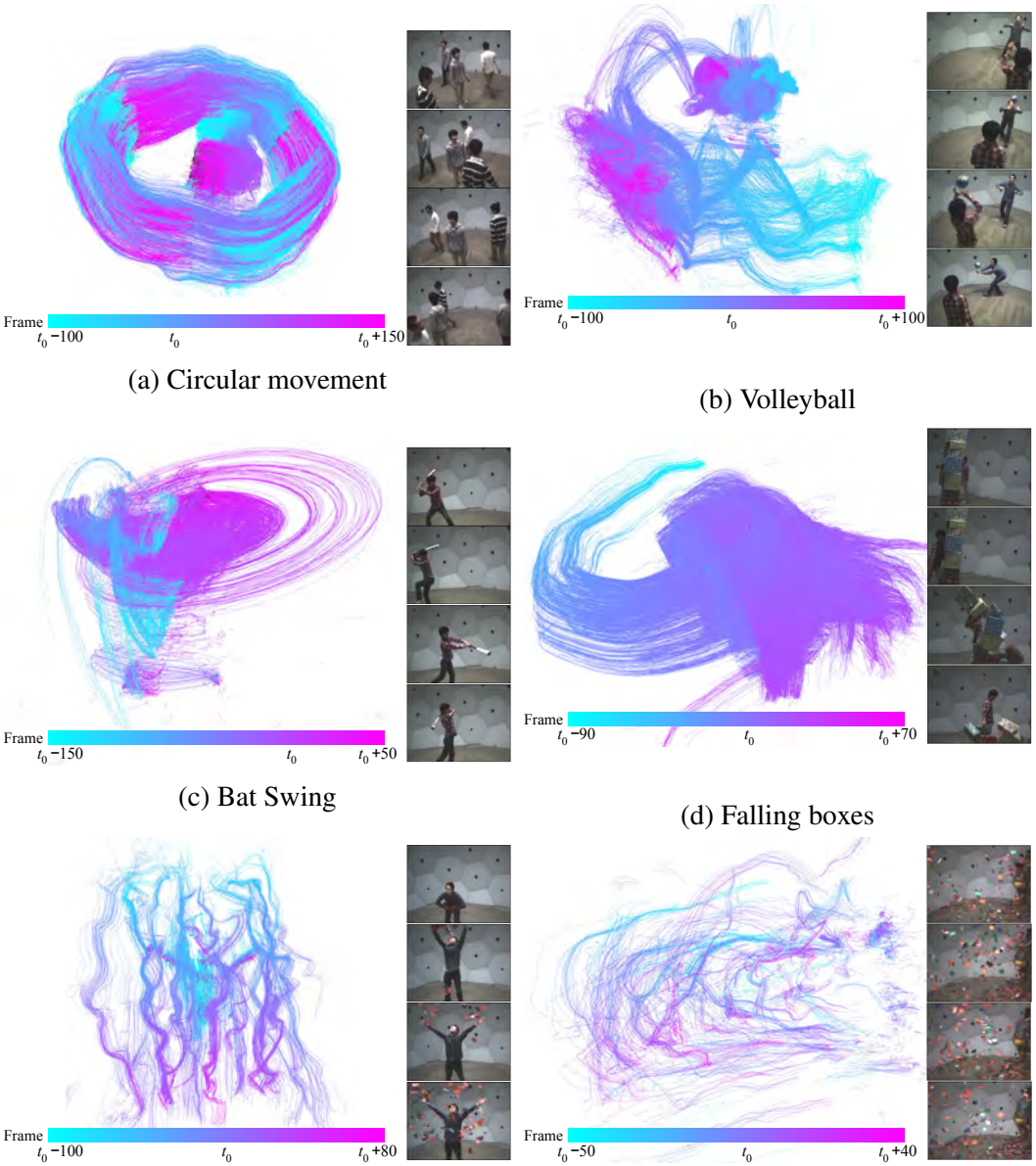


Fig. 3.6 We qualitatively demonstrate the performance of MAP visibility estimation using the sequence. The normal of the selected patch is shown as a red arrow in the 3D view (left) and projected patch is shown as a red polygon in each image (right). The images with a blue boundary are the views that belongs to the visibility set. The bat occludes the patch and its effect can be seen as a “shadow" on the visibility set of cameras (left).



(a) Circular movement

(b) Volleyball

(c) Bat Swing

(d) Falling boxes

(e) Confetti

(f) Fluid motion

Fig. 3.7 We reconstruct 3D trajectories in real world scenes in the presence of significant occlusion, large displacement, and topological change. The color codes the time that trajectory points are reconstructed. Note that each trajectory is individually reconstructed without any spatial or temporal regularization.

Chapter 4

Measuring 3D Motion of Anatomical Landmarks

A set of 3D anatomical landmarks from face, body[1], and hands can approximate human behavioral cues. These "registered" 3D cues directly provide correspondence across time, enabling to study how each body parts are moving over time, and across individuals, enabling to study the correlations among behavioral signals exchanged during social interaction by investigating examples from many scenes. Notably, it is important to measure the signals as complete as possible to avoid losing any important subtlety transmitted in social interaction. Most prior approaches in social signal processing, however, have remained almost entirely focused on the analysis of facial expressions only, despite emerging evidence [81, 82] that facial expressions provide a fundamentally *incomplete* characterization of nonverbal communication. One proximal cause for this singular focus on the face is that capturing natural social interaction presents challenges that current state-of-the-art motion capture systems simply cannot address.

In this chapter, we present a method to measure 3D motion of anatomical landmarks from face, body, and hands by exploiting a large number of cameras in the Panoptic Studio. The organizing principle is that social signal capture should be performed by the consolidation of a large number of "weak" perceptual processes rather than the analysis of a few sophisticated sensors. The large number of views provides robustness to occlusions, provides precision over the capture space, and facilitates the boosting of weak 2D human pose detectors into a strong 3D skeletal tracker without any prior about the scenes and subjects. Our method does not rely on a 3D template model or any subject-specific assumption such as body shape, color, height, and body topology. Yet, our method works robustly in various challenging social

[1] We use the term "body" landmark to refer to the landmarks from torso, arms, and legs detected by human body pose detectors [79, 80]. It does not contain faces and hands.

interaction scenes of arbitrary number of people, producing temporally coherent time-varying body structures. Furthermore, our method is free from error accumulation and, thus, enables capture of long term group interactions (e.g., more than 10 minutes). All these properties are important to enable the study of social interactions at scale (capturing motion from hundreds of participants).

In this chapter, we first focus on body motion capture of interacting multiple people. Some example results reconstructing naturally emerging body postures are shown in Figure 4.1. Once 3D body landmarks are obtained, 3D faces and hands are subsequently reconstructed by running 2D face and hand detector [46] on the candidate regions specified by the body landmarks. To this end, 3D faces and hands are reconstructed in the same coordinate with bodies by triangulating the detected 2D hand and face landmarks across views, providing the 3D anatomical landmarks from body, face, and hands for multiple individuals.

4.1 Related Work

4.1.1 Automated Group Behavior Analysis

Over the last decade, there has been increasing interest in analyzing social interaction among multiple people automatically using multiple camera sensors. Several datasets recording free-standing conversational groups are presented where multiple people (from 5 to 14 subjects) move and behave naturally communicating without restriction [83–85]. Compared to the scene captured in structured environment such as round-table meetings [35], the subjects in such unstructured environments show richer social signals by their body motion. However, due to the unconstrained nature, measuring their motion become much more challenging due to the occlusions by limited view points and limited resolutions to cover large area. Thus all of the previous work rely on coarse level manual annotations (e.g., quantized body/head orientation in every3 seconds), and mainly focus on higher level social understanding from the coarse measurement such as F-formation detection [35] and personality predictions [85, 83]. Non of the previous work try to measure accurate full body motion of every individuals in such challenging scenarios although a lot of social signals are embedded in those subtle cues.

4.1.2 Markerless Motion Capturing Using Multiple View Systems

In computer vision area, there has been a long history in the research to measure the 3D structure and motion of dynamically moving human subjects using multiple camera sensors. Kanade et al. [86] pioneered the use of multi-view sensing systems to “virtualize" reality, using 51 cameras mounted on geodesic dome of 5 meter in diameter. A number of systems

were subsequently proposed to produce realtime virtualizations [87–90]. Vlasic et al. [25] recovered detail by applying multi-view photometric stereo constraints using a system with 1200 lights on a dome and eight cameras. More recently, a multimodal multi-view stereo system fusing 53 RGB cameras and 53 infrared cameras has been proposed to reconstruct high quality 3D virtual characters [91]. These researches mainly focus on reconstructing 3D virtual structure and surface rather than motions by independently processing each frames.

Other methods explicitly tackle the marker-less motion capture problem by producing the motion as the 3D skeletal structures over time. To obtain greater details and improve the robustness given limited number of cameras (usually less than 15), de Aguiar et al. [20], Vlasic et al. [92], and Furukawa and Ponce [26] deformed pre-defined templates of fixed topology to recover the details that were subsampled or occluded in the set of views at a time instant. These methods require to generate a rigged 3D model for each individual and the quality of the template is important to reach high accuracy. The model also should be aligned at the initial frame to be tracked, and usually a predefined pose (such as T-pose) should be performed by all the individuals at the beginning of the capture. In most cases, silhouette is used as an important measurement to deform the template, and, thus, a Chroma key environment of the studio is frequently used for easier background subtraction. The methods in this area fundamentally suffer from topological changes of the scene. Although the method shows promising results in some scenes, the requirement of high quality 3D template specified for each individual limits the practicality of the method especially for the social motion capture because: (1) any subjects should be captured to understand human behavior and reconstructing template for each individual requires to much effort and time; (2) topological changes usually happen in our motion during the interaction (3) instructing people to perform the predefined canonical body pose would even interfere their naturalism.

It should be also noted that the performance of the previous methods are demonstrated by researcher themselves' or actors with exaggerated motions (e.g., fighting, jumping, dance, and so on). The scenes usually have a single person, and few approaches have been proposed to reconstruct two people [29], and three interacting people [30].

In contrast to previous work, our method are free from all the above limitations. Our method does not rely on the predefined 3D template, and can reconstruct the people's motion directly without knowing prior assumptions about the scene and individuals: the scene may have arbitrary number of people; people can be any shape (children to adult, small to tall); people can wear any color of clothes; people can freely leave or participate to the scene without any requiring any initialization; people can perform any natural motion they want free from topological constraints. Our date contains multiple people's natural interactions (up to 8), and people performs any challenging natural motion without constrains (crossing

Fig. 4.1 HD example views showing frequently occurring postures that carry rich social signals, with 3D body pose automatically annotated by our method.

arms, chin on a hand, and so on). This is directly advantageous for social motion capture because it minimize the potential interference on natural social signaling and capture the subjects at scale which is important for social behavior analysis.

4.1.3 Pose Detection Approaches

Depth sensor such as the Kinect [93, 27] is also emerging as a promising sensing modality. The main advantage of this sensor is that it can produce 3D pose from a single view. However, this sensing paradigms directly interfere with social interaction: the Kinect requires people to face the sensor direction to get reasonable measurements. Using multiple Kinects has a potential [29] in that it may produce dense point cloud easily, but how to fuse them for 3D pose estimation has not been explored, and more importantly synchronization among Kinects is inherently challenging in the current system specification.

Over the last few years, single view 2D pose estimation methods have made significant progress [80], by utilizing Convolutional Neural Network framework with large scale of human pose dataset [94]. The state-of-the-art method shows [80] excellent performance in various environments regardless subject's shape, appearance, and scales. It is a natural direction to use the body pose detector in multiple views by fusing the 2D detection results in 3D [95, 96, 79, 23, 97]. Ideally only two views are enough to reconstruct 3D joints from the 2D detector without considering occlusion. Obviously, the problem becomes challenging if occlusions in the scene becomes severe, and more and more views are required to get the desired performance. However, non of the previous work focus on the social scenes as in this paper, and the study about the views and the scene complexity has never been performed.

4.2 Method Overview and Notation

Our algorithm is composed of two major stages. The first stage takes, as input, images from multiple views at a time instance (calibrated and synchronized), and produces 3D body skeletal proposals for multiple human subjects. The second stage further refines the output of the first stage by using a dense 3D patch trajectory stream [98], and produces temporally stable 3D skeletons and an associated set of labeled 3D patch trajectories for each body part, describing subtle surface motions.

In the first stage, a 2D pose detector [80] is computed independently on all 480 VGA views at each time instant t, generating detection score maps for each body joint (see Fig. 4.2b). The 2D score maps for each body joint $j \in \{1, \cdots, J\}$ are combined into a 3D score map $H_j(\mathbf{Z})$ by projecting a grid of voxels $\mathbf{Z} \in \mathbb{R}^3$ onto the 2D score maps and computing an average 3D score at each voxel (subsection 4.3.1).

Our approach then generates several levels of proposals, as shown in Figure 3.2. A set of *node proposals* $\mathbf{N}_j$ for each joint j is generated by non-maxima suppression of the 3D score map $H_j(\mathbf{Z})$, where the k-th node proposal $\mathbf{N}_j^k \in \mathbb{R}^3$ is a putative 3D position of that anatomical landmark. Similarly, the set of *part proposals* is denoted by $\mathbf{P}_{uv}$, where u and v are joints and $(u,v) \in \mathbf{B}$ is the set of body parts or *bones* composing a skeleton hierarchy. The k-th part proposal, $\mathbf{P}_{uv}^k = (\mathbf{N}_u^{k_u}, \mathbf{N}_v^{k_v}) \in \mathbb{R}^6$, is a putative body part connecting two node proposals, $\mathbf{N}_u^{k_u}$ and $\mathbf{N}_v^{k_v}$, where the index k enumerates all possible combinations of k_u and k_v. As the output of the first stage, our algorithm produces *skeletal proposals*; we refer to the k-th proposal as $\mathbf{S}^k = \{\mathbf{P}_{uv}^k\}_{uv \in \mathbf{B}}$. A skeletal proposal is generated by finding an optimal combination of part proposals using a dynamic programming method under the score function defined in subsection 4.3.3. Here, we abuse the notation to have $\mathbf{P}_{uv}^k$ refer to the optimally assigned part u,v of skeleton k (the superscript k is understood to be the optimal mapping, from context). After reconstructing skeletal proposals at each time t independently, we associate skeletons from the same identities across time and generate *skeletal trajectory proposals* $\tilde{\mathbf{S}}^k(t) = \{\tilde{\mathbf{P}}_{uv}^k(t)\}_{uv \in \mathbf{B}}$, where $\tilde{\mathbf{P}}_{uv}^k(t)$ is a *part trajectory proposal*, a moving part across time, with k similarly overloaded to denote the optimal associations determined in each frame t.

In the second stage, we refine the skeletal trajectory proposals generated in the first stage using dense 3D patch trajectories [98]. To produce evidence of the motion of different anatomical landmarks, we compute a set of dense 3D trajectories $\mathbf{F} = \{\mathbf{f}_i\}_{i=1}^{N_F}$, which we refer to as a *3D patch trajectory stream*, by tracking each 3D patch independently. Each patch trajectory f_i is initiated at an arbitrary time (every 20th frame in our results), and tracked for an arbitrary duration (30 frames backward-forward in our results) using the method of Joo et al. [98]. Our method associates a part trajectory $\tilde{\mathbf{P}}_{uv}^k$ with a set of patch trajectories $\mathbf{F}_{uv}^k$

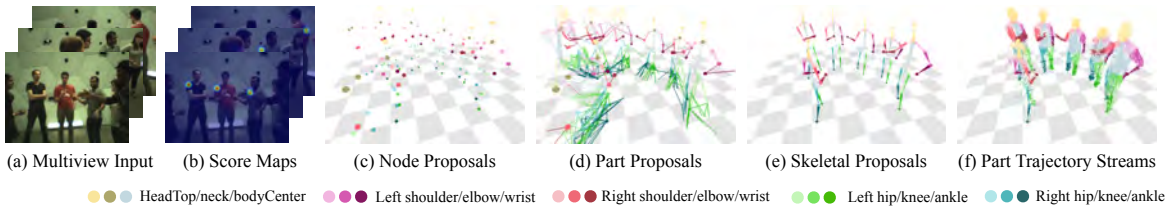


Fig. 4.2 Several levels of proposals generated by our method. (a) Images from upto 480 views. (b) Per-joint detection score maps. (c) Node proposals generated after non-maxima suppression. (d) Part proposals by connecting a pair of node proposals. (e) Skeletal proposals generated by piecing together part proposals. (f) Labeled 3D patch trajectory stream showing associations with each part trajectory. In (c-f), color means joint or part labels shown below the figure.

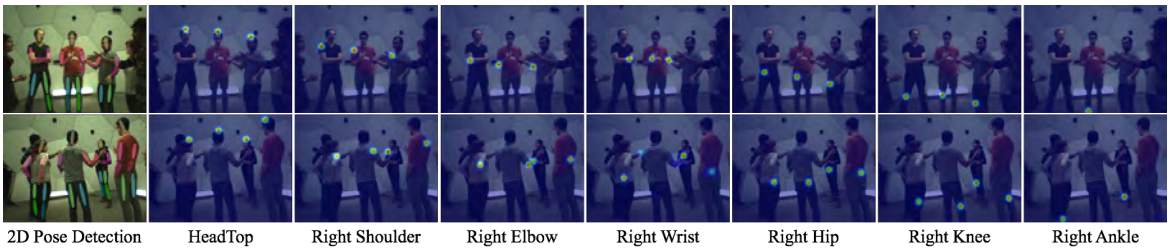


Fig. 4.3 2D pose detections and score maps generated by the method of [80]. (Column 1) Example views out of 480 views with proposals by the pose detector (Column 2-7) Heat maps for each node on each view. Note that the body pose detector distinguishes left-right limbs.

out of **F**, and these trajectories determine rigid transformations, $T(t+1\,|\,t) \in SE(3)$, between any time t to $t+1$ for this part. These labeled 3D trajectories associated to each part provide surface deformation cues and also play a role in refining the quality by reducing motion jitter, filling missing parts, and detecting erroneous parts.

4.3 The First Stage: Skeletal Proposals Generation

Our algorithm integrates 2D pose detections across the many views of our massively multiview system, fusing simple 2D cues to estimate 3D skeletal poses at each time instance. While detections in any single view may be incomplete or inaccurate—typically due to occlusions—we find that aggregating these cues across many views yields very stable results. Our method is simple, yet robust thanks to the large number of views. In contrast, prior marker-less motion capture methods are typically "model-dependent", requiring a 3D template model to constrain shape deformations, a motion model to constrain temporal deformations, and a relatively complex energy function minimization that trades off each of

Table 4.1 Summary of Notation.

Notation	Descriptions
s_i^c	i-th 2D skeleton detection in a camera view c
s_{ij}^c	j-th joint of i-th 2D skeleton in a camera view c
$h_{ij}^c(\mathbf{z})$	2D score map of j-th joint of ith skeleton in view c
$h_j^c(\mathbf{z})$	Merged score map of j-th joint of all skeletons in view c
$H_j(\mathbf{Z})$	3D score map for the j-th joint
$\mathbf{N}_j$	Node proposals for the j-th joint
$\mathbf{P}_{uv}$	Part proposals for the part connecting nodes (u,v)
$\mathbf{S}$	Skeletal proposals connecting multiple part proposals
$\tilde{\mathbf{S}}(t)$	Skeletal trajectory proposals, associated through time
$\tilde{\mathbf{P}}_{uv}(t)$	Part trajectory proposals for the connecting nodes (u,v)
$\mathbf{F}$	3D Patch Trajectory Stream, $\{\mathbf{f}_i\}_{i=1}^{N_F}$
$\mathbf{F}_{uv}$	A subset of $\mathbf{F}$ associated to $\tilde{\mathbf{P}}_{uv}$

these priors (e.g., [21, 26, 97]). Our method in this stage is essentially based on triangulating detections at a single time instance, and, thus, does not suffer from error accumulation or drift. It does not require a 3D template model, prior assumptions about the subject or the motion, or an initial alignment for tracking. In this section, we describe how the proposals are generated and built up from 2D cues.

4.3.1 3D Node Score Map and Node Proposals

A single-view 2D pose detector is computed on all VGA views at each time instant, and is used to generate 2D pose detections and per-joint score maps in each image. Because the first stage of our method is performed at each time independently, we will consider a fixed time instant t, and drop the time variable for clarity. We use the detector of Wei et al. [80] without additional training. The method of [80] requires bounding box proposals for each human body as initialization, thus, we first apply a person detector similar to R-CNN [99], and run the pose detector on the detected person proposals represented as bounding boxes. Each 2D skeleton detection i in a camera view c is denoted by $\mathbf{s}_i^c \in \mathbb{R}^{2\times 15}$, and is composed of 15 anatomical landmarks or *nodes* (3 for the head/torso and 12 for the limbs), also referred to as joints[2]. The position of the j-th node of the i-th person detection is denoted by $\mathbf{s}_{ij}^c \in \mathbb{R}^2$. The

[2]We modify the skeleton hierarchy of [80] to have an explicit torso bone, by taking the center of the two hip nodes as a body center node.

method of [80] also provides a score map representing the per-pixel detection confidence for each node s_{ij}^c, which we denote as $h_{ij}^c(\mathbf{z}) \in [0,1]$, where $\mathbf{z} \in \mathbb{R}^2$ indexes 2D image space. We also compute a merged score map by taking the maximum across all person detections at each pixel, $h_j^c(\mathbf{z}) = \max_i h_{ij}^c(\mathbf{z})$. Merged score maps of example views are shown in Figure 4.3.

To combine 2D node score maps from multiple views into 3D, we generate a 3D score map for each node using a spatial voting method. We first index the 3D working space into a voxel grid (4cm in our implementation), and compute the *node-likelihood* score of each voxel by projecting the center of the voxel to all views and taking the average of the 2D scores at the projected locations. The 3D score map $H_j(\mathbf{Z})$ for a node j at the 3D position $\mathbf{Z} \in \mathbb{R}^3$ is defined as

$$H_j(\mathbf{Z}) = \frac{1}{|V(\mathbf{Z})|} \sum_{c \in V(\mathbf{Z})} h_j^c(\mathscr{P}_c(\mathbf{Z})), \tag{4.1}$$

where $\mathscr{P}_c(\cdot) \in \mathbb{R}^2$ denotes projection into camera c, $V(\mathbf{Z})$ is the set of cameras where the 3D location $\mathbf{Z}$ is visible, and $|V(\mathbf{Z})|$ is the cardinality of the set. Note that the 3D score map for each node is computed separately, producing fifteen 3D score maps at each time instant.

From the 3D score map for each node at each time instance, we perform Non-Maxima Suppression (NMS), and keep all the candidates above a fixed threshold τ (we use τ=0.05). The results are shown in the Figure 3.2c, and the same results color-coded by the node scores are shown in Figure 4.4. Each node proposal, denoted as $\mathbf{N}_j^k$ for the k-th proposal for node j, is a putative candidate for the j-th anatomical landmark of a participant.

4.3.2 Part Proposals

Given the generated node proposals, we infer part proposals by estimating connectivity between each pair of nodes that make up a possible body part. The 2D detector [80] uses appearance information during the inference, and, thus, the result tends to preserve connectivity information (e.g., left knee is connected to the left foot of the same person). Our approach fuses them by voting 2D connectivity into 3D. More specifically, we define a connectivity score between a pair of node proposals by projecting them onto all views, and checking in how many views they are actually connected, i.e., both nodes belong to the same person detection. Formally, the connectivity score of a part $\mathbf{P}_{uv}^k$ between two node proposals $(\mathbf{N}_u^{k_u}, \mathbf{N}_v^{k_v})$, where $(u,v) \in \mathbf{B}$, is defined as

$$\Phi(\mathbf{P}_{uv}^k) = \frac{1}{|V(\mathbf{P}_{uv}^k)|} \sum_{c \in V(\mathbf{P}_{uv}^k)} \max_i \phi_{iuv}^c \left(\mathscr{P}_c(\mathbf{N}_u^{k_u}), \mathscr{P}_c(\mathbf{N}_v^{k_v}) \right),$$

$$\phi_{iuv}^c(\mathbf{z}_u, \mathbf{z}_v) = w_{iuv}^c(\mathbf{z}_u, \mathbf{z}_v) \delta_{iuv}^c(\mathbf{z}_u, \mathbf{z}_v)$$

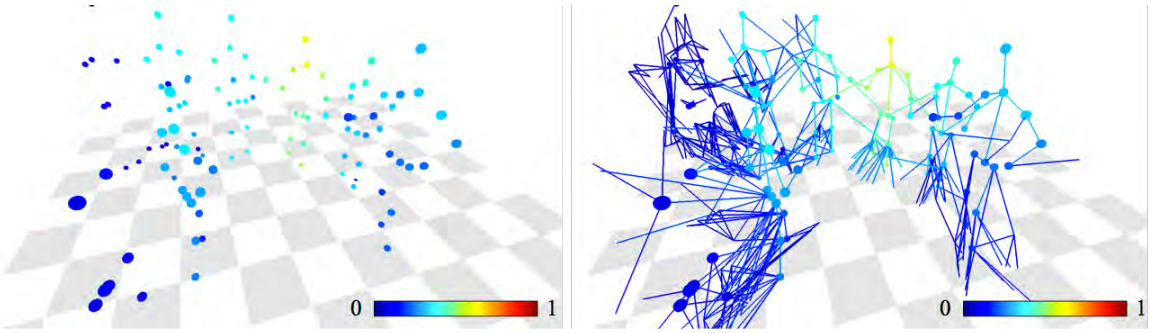


Fig. 4.4 Computed scores for node proposals and part proposals. The color encodes scores.

where

$$w^c_{iuv}(\mathbf{z}_u, \mathbf{z}_v) = \frac{1}{2}\left(h^c_{iu}(\mathbf{z}_u) + h^c_{iv}(\mathbf{z}_v)\right), \text{and}$$
$$\delta^c_{iuv}(\mathbf{z}_u, \mathbf{z}_v) = \begin{cases} 1 & \text{if } h^c_{iu}(\mathbf{z}_u) > \tau \text{ and } h^c_{iv}(\mathbf{z}_v) > \tau \\ 0 & otherwise. \end{cases}$$

Here, $\mathscr{P}_c(\mathbf{N}^{k_u}_u)$ and $\mathscr{P}_c(\mathbf{N}^{k_v}_v)$ are the projections of the two nodes of $\mathbf{P}^k_{uv}$ in view c, and $V(\mathbf{P}^k_{uv})$ is the set of cameras where the 3D part is visible. Intuitively, the part score Φ represents the average connectivity score across all views from all potentially corresponding 2D person detections. Because we do not know the correspondence from 3D parts to 2D person detections, we take the maximum score across all possible detections i in each view. Assuming that the projected part corresponds to the i-th person detection in camera c, the part connectivity score ϕ^c_{iuv} is defined as the average score of the projected nodes, denoted by $w^c_{iuv}(\mathbf{z}_u, \mathbf{z}_v)$. The delta function δ^c_{iuv} additionally ensures that ϕ^c_{iuv} is nonzero only if both projected node locations have a sufficiently high score for the same detection i (i.e., both nodes are detected as part of a single person). An example of computed part scores is shown in Figure 4.4.

4.3.3 Generating Skeletal Proposals by Dynamic Programming

Our method generates skeleton proposals by piecing together the part proposals. Since each skeleton is a tree structure, this can be computed efficiently using Dynamic Programming (DP)—but only for a single person. Therefore, we use DP to greedily find 3D skeletons $\mathbf{S}^k$ which maximize the sum of part scores,

$$\Theta(\mathbf{S}^k) = \max_{(k_1,\cdots,k_J)} \sum_{(u,v)\in\mathbf{B}} \Phi\left(\mathbf{P}^k_{uv}\right).$$

A skeleton $\mathbf{S}^k$ is given by the mapping $k \mapsto (k_1, \cdots, k_J)$, where the J-tuple $(k_1, \cdots, k_J)$ determines the assignment of node proposals $\mathbf{N}_j^{k_j}$ for each joint j in the body. After picking the highest scoring skeleton $\Theta(\mathbf{S}^k)$, the assigned nodes $(k_1, \cdots, k_J)$ are removed from the pool of available node proposals and we run DP again to find the next highest scoring skeleton, and so on until all possible skeletons are found.

One option here would be to threshold the skeleton scores $\Theta(\mathbf{S}^k)$ at some minimum value to determine valid detections. However, we can do better: each 3D skeleton should be supported by 2D detections, and each 2D detection can correspond to only a single 3D skeleton. This observation is important because the voting used to generate 3D node proposals assigns equal score to *all* voxels along the line of sight of each 2D detection (Sect. 4.3.1), and, similarly, the max over detections in the part score $\Phi(\cdot)$ makes $\Theta(\mathbf{S}^k)$ an overestimate.

To avoid this form of double counting, our method places each 3D node $\mathbf{N}_j^k$ in skeleton $\mathbf{S}^k$ in correspondence with the closest 2D joint detection in each view. For each 3D node $\mathbf{N}_j^k$, we create a set of correspondences $\mathscr{C}_j^k$ with elements (c,i) such that the distance $\|\mathscr{P}_c(\mathbf{N}_j^k) - \mathbf{s}_{ij}^c\|_2$ is the minimum across all detections i in view c and smaller than δ=10px. Once a 2D correspondence is established, we remove it from the set of available 2D detections, and, as above, this is performed greedily in order of decreasing skeleton score $\Theta(\mathbf{S}^k)$. Skeletons where the head node has fewer than two correspondences are discarded, i.e., if $|\mathscr{C}_j^k| < 2$ for j the head.

We additionally use the set of correspondences $\mathscr{C}_j^k$ to refine the 3D node locations by minimizing reprojection error. This overcomes the discretization error introduced by the voxel grid resolution. The final 3D node location $\hat{\mathbf{N}}_j^k$ is then

$$\hat{\mathbf{N}}_j^k = \arg\min_{\mathbf{Z}} \sum_{(c,i)\in\mathscr{C}_j^k} \left\| \mathscr{P}_c(\mathbf{Z}) - \mathbf{s}_{ij}^c \right\|_2 .$$

The output of the algorithm described in this section is 3D skeletal proposals reconstructed independently at each time instance. After performing this process on all frames, our method associates skeletons from the same identity across time by simply considering spatial distance of the head node. That is, for a $\mathbf{S}_t^{k_1}$ reconstructed at time t, we find a corresponding skeleton at $\mathbf{S}_{t+1}^{k_1}$ with the closest head node location from $\mathbf{S}_t^{k_1}$ within a threshold. To be somewhat robust to missing skeleton detections, our method associates across a window of time. If there is no corresponding skeleton at time $t+1$, we also consider the next time $t+2$ and find a corresponding skeleton.

This first stage of our method is performed without considering any temporal cues. The advantages of this are that the method can easily handle a varying number of people, there is

no need to impose priors on the motion or skeletons, and the bulk of the computation is easily parallelized across frames. In many cases, we find that the results from the first stage are already sufficient for many applications. However, the results exhibit some jitter—especially for complex scenes with limited views per person—and missed or noisy detections do not benefit from evidence found in adjacent frames. We address these issues in the second stage of our method.

4.4 The Second Stage: Temporal Refinement and Trajectory Stream Labeling

The per-frame skeletal proposals from the first stage can be improved by using temporal coherence. We use motion cues from a *3D patch trajectory stream*: dense 3D point tracks computed by the method of Joo et al. [98]. We find an association between each part trajectory proposal and a subset of the patches in the trajectory stream, and use it to reduce motion jitter, remove false detections, and fill in missing detections. The resulting labeled patch trajectories also capture rich motion information representing the subtle deformations of the surface for each body part (see Fig. 3.2f).

4.4.1 Patch Trajectory Stream Reconstruction

We can only observe surface motions, not the true motion of the underlying skeleton, so it is not immediately apparent how best to enforce temporal consistency in the motion of body parts. Clothing in particular makes the relationship between surface motion and body parts difficult to model. To keep the use of priors and models to a minimum, we therefore choose to measure surface motion independently from the underlying skeletal motion and postpone all decisions about part-to-surface associations.

To represent surface motion, we use the method of [98] to track a dense 3D patch cloud—a set of points with corresponding surface normal and a small spatial extent, representing the surface locally—and estimate the motion of each of these patches. Instead of generating the initial patches to track using SIFT matching and triangulation (as in [98]), we use the depth maps from our 10 RGB+D sensors to generate an initial set of 3D patches. For a single frame, a dense 3D point cloud is first generated from the depth maps, and planar local patches centered on each point are initialized. The size of patches is manually determined by considering image resolution and fixed for entire processes at 6cm$\times$6cm. To find the normal of each patch, we apply Singular Value Decomposition (SVD) to the coordinates of points within a neighborhood (determined by Euclidean distance from the center point with the

patch size as a threshold), and the least principal axis is selected as the normal direction. The sign of the normal is disambiguated by considering camera visibility.

The remainder of the algorithm (3D patch tracking) is as described in [98]. As a brief overview, a patch is represented by a triplet points (the center point, and two orthogonal points on the patch plane), and it is tracked by projecting the triplet points on all views where the target patch is visible. Optical flow tracking is performed in 2D on each point, and the tracked 2D flows are triangulated into 3D. The core idea to fully leverage a large number of views is to reason about the time-varying camera visibility of each patch. The visibility is optimally estimated in a MAP framework that combines photometric consistency, motion consistency, and visibility priors, see [98] for more details. For our results, we initialize a 3D patch cloud every 20th frame, and track them backward and forward for 30 frames in each direction. As output, we obtain a dense 3D patch trajectory stream, $\mathbf{F} = \{\mathbf{f}_i\}_{i=1}^{N_F}$, where each $\mathbf{f}_i(t) \in \mathbb{R}^3$ is the time-varying position of a tracked patch.

4.4.2 Associating Part Trajectory Proposals and Trajectory Stream

Part trajectory proposals $\tilde{\mathbf{P}}_{uv}$ represent the moving body parts of a single person, and are given by the optimal assignment used to generate skeletal trajectory proposals. These part trajectories lack temporal coherence because they are reconstructed independently in each frame. However, the trajectory streams provide evidence of the motion of each limb, and can be used to refine the motion of each body part. We therefore find an association between each part trajectory proposal and a subset of patch trajectories. This can be seen as a semantic labeling of the patch trajectory stream with the corresponding body parts (see Fig.3.2f).

Before performing this association, we first remove erroneous part detections which can readily be identified as outliers. We find that a simple yet robust method is to use the depth maps from the multiple RGB+D sensors. At any time instant, a part can be considered as an outlier if it is *outside* of every surface in the dense point cloud. We simply test this by checking whether a part proposal is in front of the measured depth in any view, and mark it as erroneous if it is. This is a necessary but not sufficient condition because we test this from only the 10 available depth map views. However, we find that this method works well in practice and is efficient to implement. After identifying these outliers, we remove and treat them as missing data. Then, we can assume that this filtered part trajectory only suffers from relatively small jitter and occasionally missing data.

We associate a set of patch trajectories with a filtered part trajectory proposal if they move rigidly and the patch normal is a match. Intuitively, the part should be located inside the body surface, and, thus, a vector from the closest point on the part to the patch center should have a similar direction as the patch normal—their inner product should be positive.

For a part trajectory proposal, we only consider patches for which the normal satisfies this criterion for the entire duration of the patch trajectory. As additional criteria, we compute a measure of rigidity between a patch trajectory and a part trajectory proposal. We define this as the difference between the minimum and maximum distance between them across all frames t in which they overlap:

$$d(\mathbf{f}_i, \tilde{\mathbf{P}}^k_{uv}) = \max_t l\left(\mathbf{f}_i(t), \tilde{\mathbf{P}}^k_{uv}(t)\right) - \min_t l\left(\mathbf{f}_i(t), \tilde{\mathbf{P}}^k_{uv}(t)\right),$$

where $l(\cdot,\cdot)$ is the orthogonal distance between the patch center and the line segment of the body part, i.e.,

$$l\left(\mathbf{f}_i(t), \tilde{\mathbf{P}}^k_{uv}(t)\right) = \min_\alpha \|\alpha \mathbf{N}^{k_u}_u(t) + (1-\alpha)\mathbf{N}^{k_v}_v(t) - \mathbf{f}_i(t)\|_2.$$

Here, the set of time instants t satisfies that both the patch trajectory and part trajectory streams are valid, and only patch trajectories i for which $0 \leq \alpha \leq 1$ at some time t are considered. Intuitively, this cost approximates how rigidly they move together over time, going to zero for completely rigid motion. Each part trajectory $\tilde{\mathbf{P}}^k_{uv}$ is then associated with a set of patch trajectories $\mathbf{F}^k_{uv}$, for which the rigidity cost is less than a threshold, i.e., $\mathbf{F}^k_{uv} = \{\mathbf{f}_i : d(\mathbf{f}_i, \tilde{\mathbf{P}}^k_{uv}) \leq 10\text{cm}\}$. If a patch trajectory is selected by multiple body parts (e.g., a static scene as an extreme case), the trajectory is associated with the body part with minimum $\max_t l(\mathbf{f}_i(t), \tilde{\mathbf{P}}^k_{uv}(t))$ distance. An example of this labeling is shown in Figure 3.2f.

4.4.3 Motion Refinement by Associated Patch Trajectories

From the set of patch trajectories $\mathbf{F}^k_{uv}$ associated to the part trajectory proposal $\tilde{\mathbf{P}}^k_{uv}$, we can compute the rigid transform between subsequent time instances from t to $t+1$, $T(t+1 \mid t)$, and, progressively, to any frame t' by concatenating transformations between subsequent frames, so that $T(t' \mid t)\tilde{\mathbf{P}}^k_{uv}(t)$ represents the propagated part from time t to t'. Using the transformation it is possible to propagate a body part's position to other time instants. Our method uses the propagated parts to reduce jitter and fill in missing holes by averaging multiple part locations propagated from different time instances. For a target time t, we can produce multiple proposals for the same part, including the proposal from the first stage of our method and propagated parts using the estimated transformations, creating a set

$$\{T(t \mid t-n)\tilde{\mathbf{P}}^k_{uv}(t-n), \cdots, \tilde{\mathbf{P}}^k_{uv}(t), \cdots, T(t \mid t+n)\tilde{\mathbf{P}}^k_{uv}(t+n)\}.$$

If there are elements in this set, we take the average. If the set is empty due to consistently severe occlusions, we determine that the part at time t is still missing. In practice, we use $n=1$.

This procedure can also be iterated multiple times (including patch trajectory re-association) to fill in missing parts that are further than n frames from any part proposal. We iterate this refinement until no more missing parts can be filled. After refinement, a node connected to multiple body parts can have different locations corresponding to each of the averaged parts, and we simply take the average to determine the final node locations. It should be noted that our method is different from temporal smoothing (e.g., [97]). Instead, we use an actual measurement of 3D motion rather than impose a motion prior, which prevents over-smoothing even after several iterations.

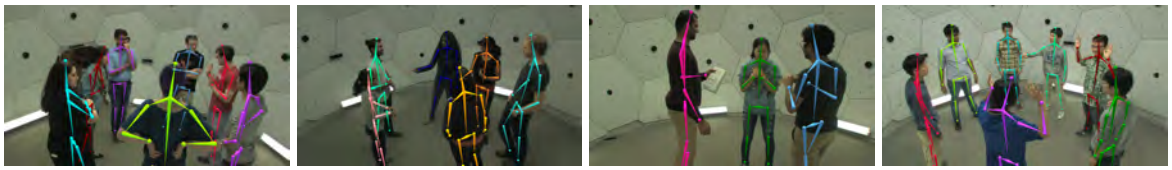
4.5 Face and Hand Captures

In this section, we briefly describe the method to capture 3d motion of faces and hands given reconstructed body landmarks. By projecting 3D landmark locations of head and wrists on each view, approximate regions of hands and face are obtained. We then apply 2d face pose detector and 2D hand detector on each region to find 2D landmark locations. Correspondences of landmarks across views are already given by construction, and final 3D face and hand are reconstructed by triangulation with RANSAC.

We use the hand detector we present in [46]. The hand detector is based on the same Convolutional Neural Network architecture of [80] which we used for 2D body pose detection, but trained on a hand keypoint dataset generated by Multiview Bootstrapping method proposed in [46]. Similarly, we also produce a 2D face detector using the same CNN architecture and a face dataset by Multiview Bootstrapping in the Panoptic Studio. We found that this method shows a comparable performance to the state-of-the-art methods and outperforms them in profile views. Before applying detectors, occluded views are excluded by considering the orthogonal distance from camera rays and body skeletons, where thresholds is used to determine occlusion (15 cm for the metacarpals, 9 cm for the proximal phalanges, and 5 cm for the remaining bones).

For a landmark location, we robustly triangulate it into a 3D location. We use RANSAC [100] on points with confidence above a detection threshold λ. Additionally, we use a 4 pixel reprojection error to accept RANSAC inliers. With this set of inlier views for point, we minimize [101] the reprojection error to obtain the final triangulated position.

To improve robustness specifically for hands, we reconstruct entire fingers simultaneously. We triangulate all landmarks of each finger (4 points) at a time, and use the average reprojection error of all 4 points to determine RANSAC inliers. This procedure is more robust because errors in finger detections are correlated: e.g., if the knuckle is incorrectly localized, then dependent joints in the kinematic chain—the inter-phalangeal joints and finger



(a) Mafia (b) Ultimatum (c) Haggling (d) 007-Bang

Fig. 4.5 Example scenes of social game sequences. The reconstructed 3D skeletons from the 480 VGA views are projected on novel HD views.

Table 4.2 Processing time for one minute of data.

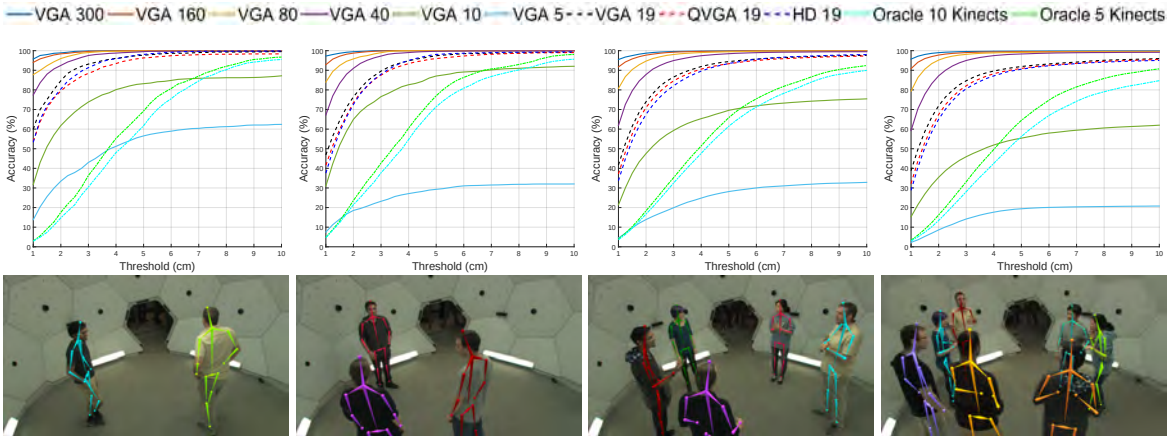
	Procedure	Time
Stage 1	(4.3.1) 2D pose detection (1 GPU)	40 h
	(4.3.1-4.3.2) Node and part proposal recon. (1 GPU)	4 h
	(4.3.3) Skeletal proposal reconstruction by DP	3 m
	(4.3.3) Skeletal proposal optimization	11 m
Stage 2	(4.4.1) Trajectory stream recon. (400 CPU cores)	35 h
	(4.4.2) Trajectory association and refinement	5 m

tip—are unlikely to be correct. This reduces the number of triangulated keypoints (because the entire finger needs to be correct in the same view) but it further reduces the number of false positives, which is more important so that we do not train with incorrect labels. A similar approach is applied for face by grouping each eye, nose, and lip, and apply RANSAC respectively.

4.6 Results

We quantitatively and qualitatively evaluate our method on various sequences captured in the Panoptic Studio. In the quantitative evaluation, we empirically show how the large number of views solves the challenging interaction capture problem; we compare performance using varying number of cameras on the scenes with different number of people. In the qualitative evaluation, we demonstrate the “model-free" advantage of our method by showing compelling motion reconstruction results on subjects of diverse appearance, body shapes, and body sizes.

In this result section, we mainly focus on the performance of body motion capture, yet qualitative results of 3D face and hand captures are shown.



(a) Two people (b) Three people (c) Five people (d) Seven people

Fig. 4.6 Performance evaluation using Probability of Correct Keypoint (PCK) metric for varying number and type of cameras on *160422 ultimatum1*. We use the result of 480 VGA cameras after manually excluding outliers as ground truth. The X-axis of each graph represents thresholds, and the Y-axis represents accuracy by the thresholds. Each graph is generated for scenes with a different number of people. The results demonstrate that more views (rather than higher resolution) are beneficial to improve accuracy, and the distinction is more noticeable if the scene contains more people.

Table 4.3 Quantitative evaluation of the accuracy of our method on the *160422 ultimatum1* sequence.

Skel. #	Node #	Outlier Node #	Node Acc.	Skel. Acc.
81,829	1,227,435	8700	99.29%	93.55%

4.6.1 Processing Time

The time to process one minute of data (1500 frames) of 480 VGA views is summarized in Table 4.2. We use different computing devices for procedures. A machine with Intel i7 3.4GHz processor and 32GB RAM is used for general processing, a GTX Titan X is used for GPU computation, and a cluster server with 400 CPU cores (2.2GHz per processor) is used for trajectory stream reconstruction.

In the first stage, most of the time is spent in running the 2D body pose detector. The detector runs at about 5 frames per second on a single GPU, but due to the large number of views (720K images per minute), processing a minute of video takes about 40 hours. In practice, we use multiple GPUs to process multiple images in parallel. In the second stage, the main computational bottleneck is the trajectory stream generation. Although they are tracked in parallel, the running time is long due to the large number of patches at each time. In our experiments, on average 15K patches are tracked per person.

4.6.2 Performance Analysis 3D Body Motion Capture

We quantitatively evaluate the performance of our method for the *160226 ultimatum1* sequence by varying the number and type of cameras. We choose the ultimatum sequence because it captures varying number of people (from two to seven people) in each time period, which is suitable to study the relation between scene complexity and the number of cameras needed to reach a desired performance. In this experiment, we only evaluate the first stage of our method.

Performance using all VGA cameras: We first quantify the performance of our system when all 480 VGA cameras are used. Due to the absence of ground truth data, we manually annotate the correctness of the reconstructed 3D skeletons by verifying their projections in multiple 2D views. We labeled a 3D joint node as an outlier if the node is projected outside of the corresponding limb or far from the presumable target joint in multiple 2D views. We exclude the period where people come in and out of the system, since at the moment body parts lie on the edge of our system's working volume. The result of the quantitative evaluation for the 15 min of sequence is summarized in Table 4.3. There are 12 sessions in the sequence, and 61 temporally associated skeletal structures are reconstructed. Among about 1.2 million body joints, about 8.7K nodes are determined as outliers or missed (rejected by thresholds of our system), showing 99.29% accuracy in node reconstruction. And, 93.55% out of about 82K 3D skeletons are correctly reconstructed without any incorrect joints. The majority of the failures are caused by insufficient visibility of the target part. An example is the pose holding hands behind one's back near the wall of the system as shown in Figure 4.8 (left). Although the hands are visible from few cameras, they are too close to be detected by the pose detector. Interestingly, our method still reconstructs the hands using the "guessed" 2D locations from 2D pose detector in frontal views, although the accuracy is limited as shown in Figure 4.8.

Comparison with varying number of cameras: To evaluate the impact of the number of views, we perform our method using varying number of cameras. The cameras are uniformly sampled (except the 19 VGA camera case explained later); i.e., we sample the next camera as the one furthest from all the already sampled cameras, and, thus, the selected cameras are always a subset of the set of the larger number of cameras. To quantify the results, we treat the result with 480 VGA cameras as ground truth after excluding the manually annotated outliers. For evaluation, we only use every tenth frame to reduce computation time. As an evaluation metric, we use the PCK (Probability of Correct Keypoint) metric, which is commonly used to evaluate 2D pose detectors [94]. Here, we use 3D distance in physical scale (cm) obtained from calibration data for the threshold of PCK, in contrast to the 2D ratio of torso/head as in 2D pose detection cases [94]. Figure 4.6 shows the PCK accuracy by

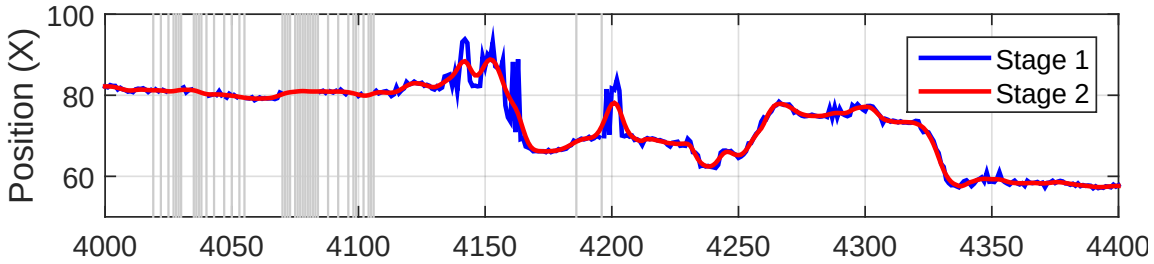


Fig. 4.7 Refinement result by the Stage 2 of our method on the *160226 mafia2* sequence. The most erroneous node is selected. The graph represents the X coordinate of the node across frames after the Stage 1 method (blue) and Stage 2 method (red). The gray regions represent the frames where the part is missing in the stage 1 output. They are recovered via the temporal propagation in the result of Stage 2.

varying the camera number on the scenes with different number of people. In all the results, we find that using a larger number of views is beneficial. If the scene is simpler (e.g., the case with two or three people), we observe that the results with a smaller number of cameras, e.g., 160 cameras, show a similar performance with 480 cameras. However, if the scene becomes more complicated, e.g., seven people, we see clearer gaps according to the camera numbers. This results can be meaningfully used to design a multiple camera system to determine the required number of cameras given a desired group size. For example, assuming that the target scenes have about five people, we forecast that a system with 80 cameras can reach about 94% of accuracy with a 2cm threshold.

Comparison with varying resolutions: As an additional evaluation, we perform a similar experiment for different camera resolutions using the multiple HD cameras installed in our system. Among 31 HD cameras, we use 19 HD cameras installed on the same panels with VGA cameras[3]. To compose similar viewpoints, we choose the closest VGA cameras from the selected 19 HDs. Additionally, we generate 19 QVGA inputs (320×240 resolution) by resizing the selected VGA videos. Because the HD cameras are not perfectly synchronized with VGAs, we interpolate the result from HDs into VGA time domain using the hardware sync data. The performance of a same number of HDs, VGAs, and QVGAs is shown as dashed lines in Figure 4.6. The result shows that the performance differences among them is marginal, although HD views have about 7 time more pixels than VGAs and about 27 times more pixels than QVGAs. The result demonstrates that the pose reconstruction performance of our method is marginally affected by the resolution changes compared to the changes by the number of views. Note that the integral of number of pixels in the 19 HD views are equivalent to about 128 VGA views, and the result clearly shows that it is more advantageous

[3]We have 20 HD cameras installed on the same panels with VGAs, but we missed 1 HD camera due to the hardware failure during the capture.

Table 4.4 Quantitative comparison to [52] on the *150129 007Bang* sequence.

Methods	Frames	Joints	Outliers	Missed	Accuracy
Ours	300	22,500	1	0	99.99%
[52]	300	22,500	1871	2248	80.80%

to have more unique camera views rather than having higher resolutions, given a fixed pixel budget. The main reason underlying this finding is that dealing with occlusions is more crucial in interaction capture scenarios, and, in particular, higher resolution is not beneficial in our method, since 2D joint localization accuracy is still limited by the 2D pose detector.

Comparison to multiple Kinects: We also compare our results with the result of multiple Kinects. Since Kinect with its accompanying SDK is one of the most commonly used sensors for markerless motion capture in various communities, using multiple Kinects can be considered as an option to handle severe occlusions for interaction capture. However, how to fuse multiple Kinect cues is not straightforward, and, thus, we naively fuse them as follows. We first generate 3D skeletal proposals from all individual Kinects, and simply find the best candidate closest to our ground truth data in Euclidean space, assuming that an Oracle chooses the best one given the GT data. This can be considered an upper bound of a naive multiple Kinects method. Since the keypoint locations of the Kinects are not identical to the skeletons of our method, for a fair comparison, we adjust the Kinect skeletons by finding an offset vector from each Kinect node toward our node of GT's skeleton in a person-centric coordinate system. As shown in Figure 4.6, the results of the *Oracle* Kinects is limited, showing less than 80% accuracy at a 5cm threshold.

4.6.3 Refinement by Trajectory Stream

We compare the performance improvement of our refinement method (the second stage) over the output of the first stage. We choose a challenging scene in *160226 mafia2* sequence where the first stage of our method shows failures due to the erroneous 2D pose detection results. To see the performance change, we plot the X coordinate of the most erroneous node (right wrist of a subject) as shown in Figure 4.7. The frames denoted as gray regions are the time when the nodes are missed due to the consistent 2D pose detector failures. It is shown that our refinement method can recover the missing parts and also noticeably reduce the motion jitter for the unstable frames. Note that our refinement method is not just smoothing but based on the temporal transformation measured by a dense trajectory stream. Thus, it does not suffer from over-smoothing, even after several iterations.

4.6.4 Qualitative Evaluation For Body Motion Capture

We apply our method, producing about 3 hours of interaction capture results. Due to the computation time, the second stage of our method is applied on a subset of the dataset; yet the first stage of our method is applied on all the sequences[4]. Example results are shown in Figure 4.9. Our result is fully automatic—given video streams and calibration data, our method generates temporally associated 3D skeletons (and labeled patch trajectory stream of each body part if the second stage is applied) for each individual without any human supervision. Refer to the supplementary videos and the live 3D viewer on our website.

Group interaction capture: Our method produces motion capture results on various social game scenarios performed by multiple people (up to 8 people). The number of subjects in the scenes is automatically determined by our method, and allowed to vary during the capture. The reconstructed results contain motions that frequently occur during communication, such as crossed-arms-on-chest, resting-chin-on-hand, mouth-guard, hands-on-back, hands-on-waist, and so on. In spite of their importance as non-verbal signals transmitting a variety of messages, such motions get little attention by prior work. In particular, severe topological changes and self-occlusions make it hard to apply 3D template-based motion capture approaches. Our method reconstructs the motion of such challenging scenes by fusing 2D pose detection cues and motion cues using a larger number of views, and demonstrates a compelling performance for social interaction capture.

Robustness to appearance, body sizes, and topological changes: Our results demonstrate robustness to subjects of diverse appearance, body types, and sizes. As mentioned, subjects' clothing is not controlled, and the captured sequences contain people with various clothing such as black pants, thick padding jumpers, hoodies, short pants, scarfs, and so on. During the discussions, they also unconsciously adjust their clothing, for example by rolling up sleeves or relocating scarfs. The height of subjects varies from a two-year old toddler to adults more than 190 cm tall. The "model-free" nature of our method enables us to reconstruct their motions without changing any parameter. It demonstrates a major advantage of our system for social behavior studies in that it can be easily applied for captures at scale, without any laborious template generation or initial alignment step. Especially, the toddler scene is challenging to "model-heavy" approaches, since instructing young children to be stationary to generate their template models (e.g., laser scanning) may not be practical.

Other interesting scenes: We also demonstrate the performance of our method on other atypical motion capture scenarios including musical performances (*drummer* and *cellist*) and *dancer* sequences. Motion capture for musical performance is a good application for markerless motion capture, because markers may interfere with their functional movements

[4]Results will be updated in our website, as they are processed.

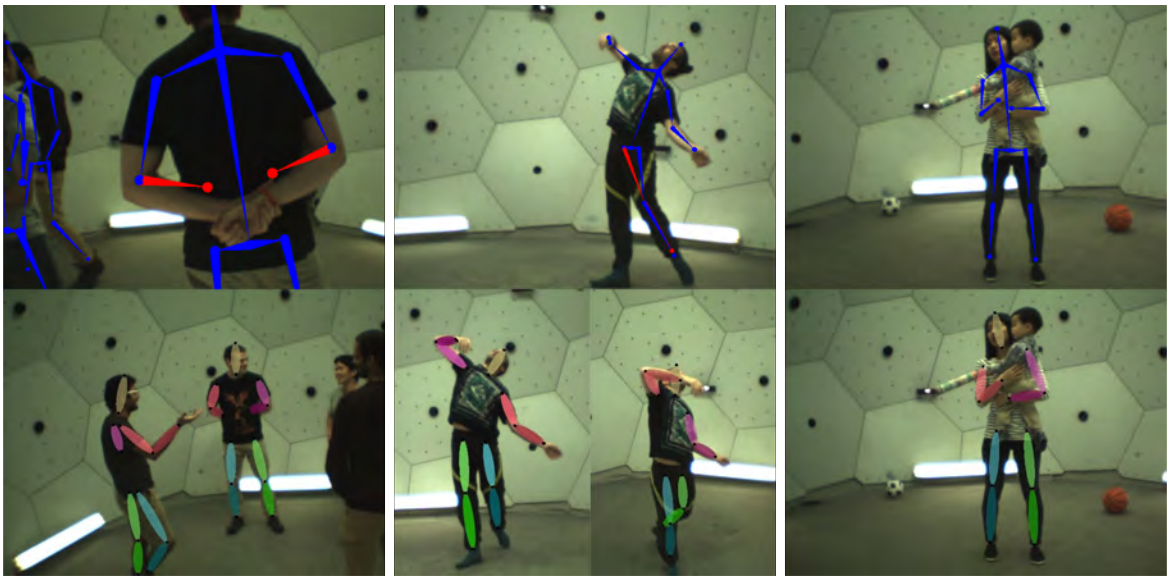


Fig. 4.8 Example failure cases. For each column, the first row shows the projection of reconstructed 3D skeletons on a view where the red colored parts are manually annotated outliers. The second row shows the 2D pose detection results. (Left) The hands are severely occluded and only visible from few cameras where they are too close to be detected by 2D pose detector. (Center) The left/right legs are confused in performing 2D pose detection, which causes failures in our 3D inference. (Right) The toddler is not detected by the pose detector, since he is severely occluded.

during the capture. Although the scenes are challenging due to the severe occlusions by musical instruments, our method shows good performance in reconstructing the performer's subtle motions (e.g., the vibrato motion in the *cellist* sequences).

On the other hand, the *dancer* sequences contain fast motion and unusual poses. Due to failures in reconstructing the trajectory stream for the extremely fast movement compared to our relatively low frame rate cameras (25 Hz), we only apply the first stage of our method. Separating reconstruction (Stage 1) from temporal refinement (Stage 2) is advantageous in this case, because the first stage, based on per-frame reconstruction, is not affected by motion magnitude and free from error accumulation. We can optionally apply temporal refinement (Stage 2) based on the quality of trajectory stream to further refine the results. We find that, however, in a few extremely unusual poses our method becomes unstable due to consistent 2D pose detection failures, which will be discussed in subsection 4.6.6.

4.6.5 Qualitative Evaluation For Hand and Face Motion Capture

The reconstruction results of 3D hands and faces are shown in the Figure 4.10 and 4.11. We only use 31 HD views for the reconstruction due to the limited image resolution of VGA

views. Compared to prior work on high-quality face reconstruction where entire cameras mainly focus on a face [11], the cameras in our system have much wider field of views to capture full body motion of multiple people, and, thus, the quality of our face reconstruction is limited and produces only a fixed number of landmark locations (70 points determined by 2D face detector). Yet, our system shows a reasonable performance in capturing facial expressions of interacting people.

Our 3D hand capture is the first in capturing hand gestures of interacting multiple people in a practical level, which is even challenging in state-of-the-art marker based motion capture system. Especially, we also found that our hand capture shows a great performance in capturing hand motion interacting with objects as shown in Figure 4.10, despite limited camera resolutions. Reconstruction results in capturing full body motion of hands, body, and faces of interacting multiple people are shown in Figure 4.11, which is also demonstrated for the first time.

4.6.6 Discussion

A limitation of our method is the dependency on a 2D pose detection method. State-of-the-art pose detectors are weak in detecting unusual poses and closely located people (as shown in Fig. 4.8). We also find that the pose detector sometimes gets confused in distinguishing left-right limbs (as shown in the center of Figure 4.8). Although our method can overcome these issues by fusing cues across views via spatial voting and across time via associating trajectory streams, if the 2D body pose detectors fail consistently, our method is unable to recover. The second limitation is the long computation time to process the large number of views. However, depending on the application, the computation time can be greatly reduced by using a smaller number of cameras with a trade-off in accuracy (see Fig. 4.6).

Fig. 4.9 Our method is performed on various scenes including social interactions of multiple people. (Row 1) Reprojected skeletons on novel HD views; (Row 2) Rendered 3D skeletons in novel 3D views with node trails over time; (Row 3) Labeled 3D trajectories representing articulated non-rigid body parts with same color representations as in Figure 3.2; (Row 4-7) Reprojected skeletons on novel HD views of various scenes.

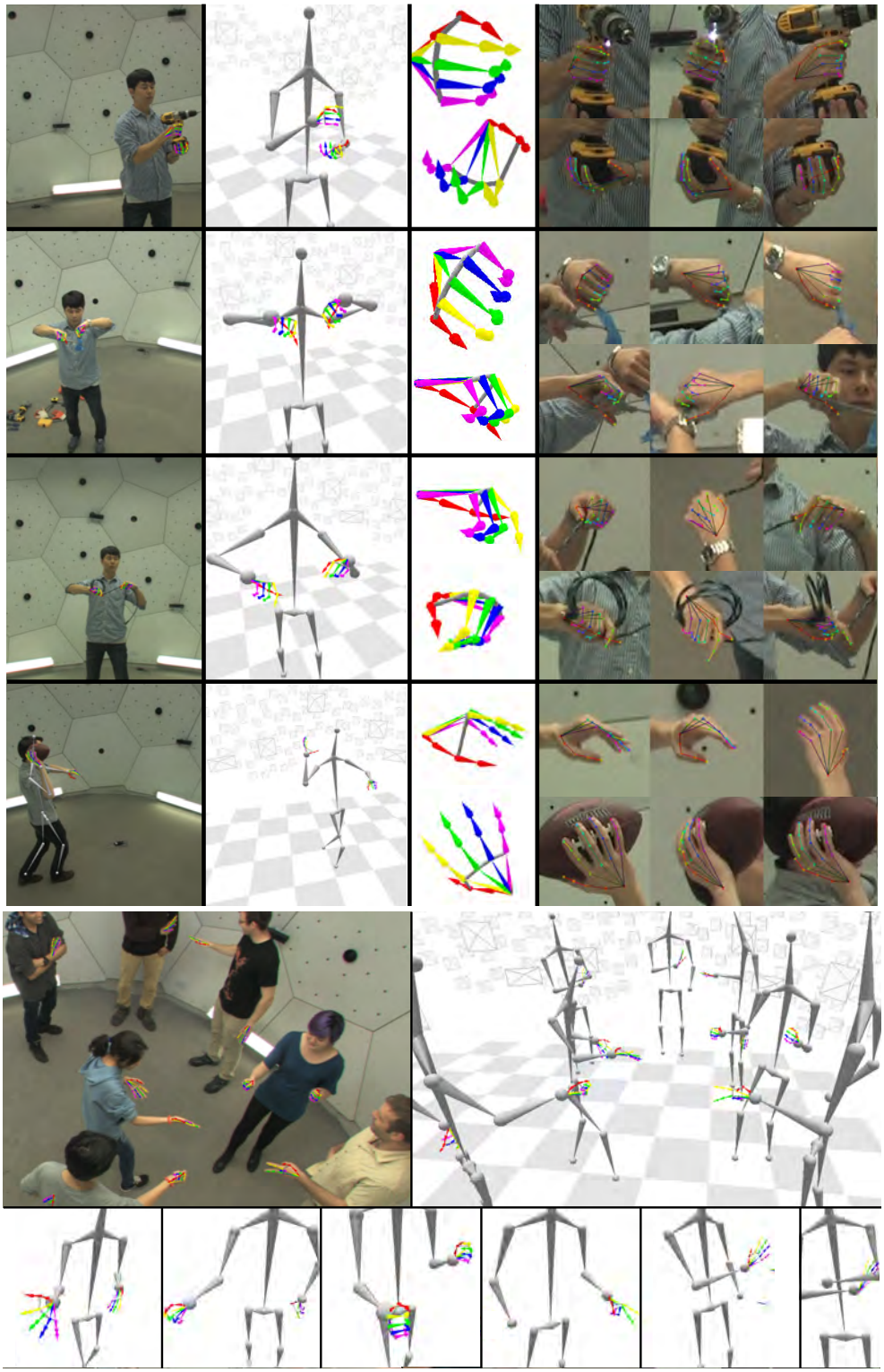


Fig. 4.10 Qualitative results on hand and body motion capture. (First 4 rows) (a) Reprojected triangulation. (b) 3D hands in context. (c) Metric reconstruction. (d) 2D detections from hand pose detector presented in [46]. (Last 4 rows) Hand and body motion capture of interacting multiple people.

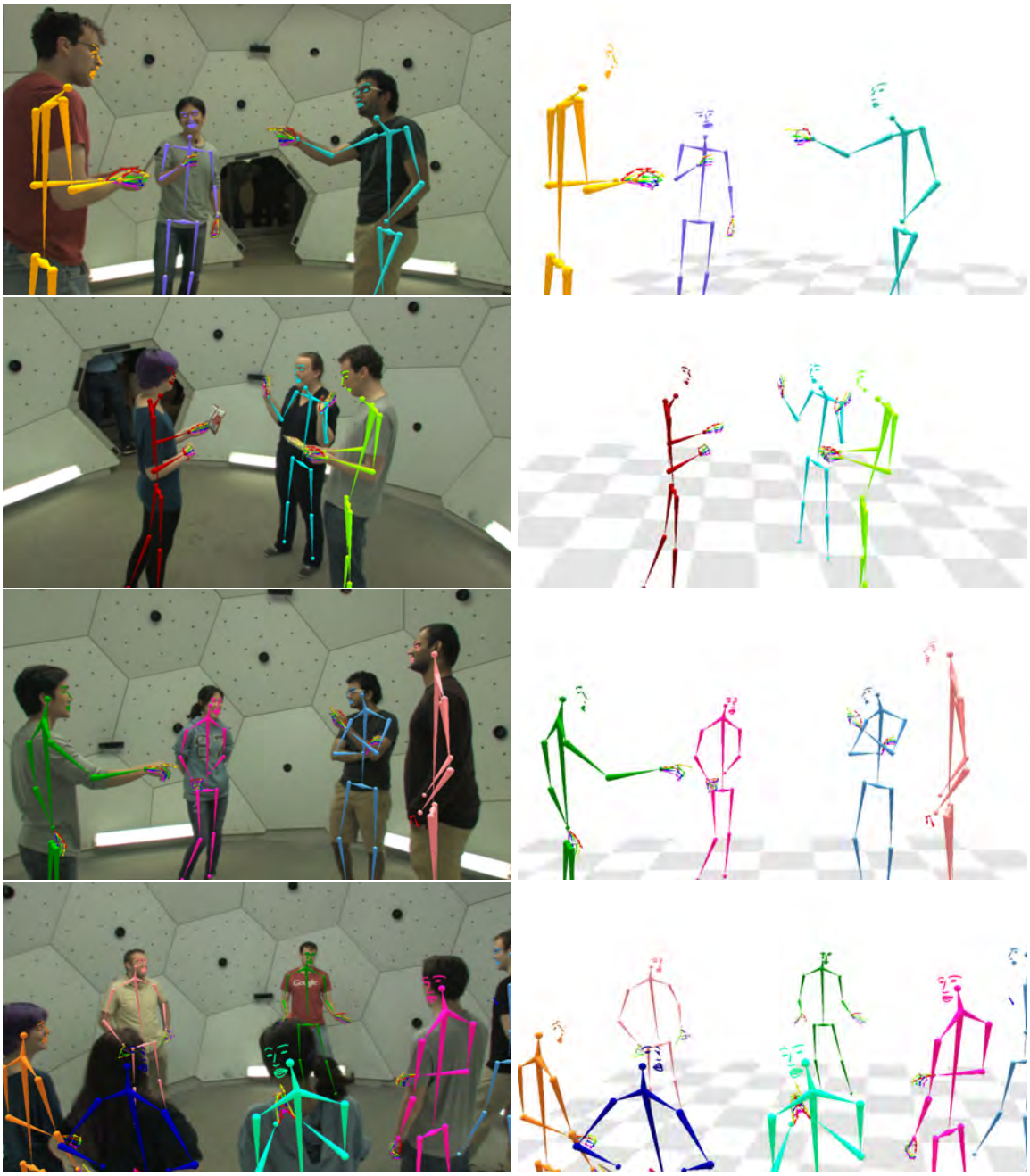


Fig. 4.11 Capturing anatomical landmarks from full body parts (face, body, and hand). (Left) An example scenes overlaid by the projection of reconstructed results (Right) Reconstruction results in 3D views.

Chapter 5

Total Body Motion Capture

We communicate tremendous amounts of information with the subtlest movements. Between a group of interacting individuals, gestures such as a gentle shrug of the shoulders, a quick turn of the head, or an uneasy shifting of weight from foot to foot, all transmit critical information about the attention, emotion, and intention to observers. Notably, these social signals are usually transmitted by the organized motion of the whole body: with facial expressions, hand gestures, and body posture. These rich signals layer upon goal-directed activity in constructing the behavior of humans, and are therefore crucial for the machine perception of human activity.

However, there are no existing systems that can track, without markers, the human body, face, and hands simultaneously in a single mesh structure. Current markerless motion capture systems focus at a particular scale or on a particular part. Each area has its own preferred capture configuration: (1) torso and limb motions are captured in a sufficiently large working volume where people can freely move [20–23]; (2) facial motion is captured at close range, mostly frontal, and assuming little global head motion [11–15]; (3) finger motion is also captured at very close distances from hands, where the hand regions are dominant in the sensor measurements [16–19]. These configurations make it difficult to concurrently analyze the broad spectrum of social signaling.

To overcome this sensing challenge, we present a novel generative body deformation model that has the ability to express the motion of each principal body part. In particular, we describe a procedure to build an initial body model, named “Frank", by seamlessly consolidating available part template models [102, 103] into a single skeleton hierarchy. To fit this model to data, we leverage keypoint detection (e.g., faces [41], bodies [80, 104, 105], and hands [46]) in multiple views to obtain 3D keypoints which are robust to multiple people and object interactions. We fit the “Frank” model to a capture of 70 people, and learn a new deformation model, named “Adam", capable of additionally capturing variations of hair

and clothing with a simplified parameterization. We present a method to capture the total body motion of multiple people with the 3D deformable model. Finally, we demonstrate the performance of our method on various sequences of social behavior and person-object interactions, where the combination of face, limb, and finger motion emerges naturally.

The output generated by this method provides a dense correspondence across individuals from the surface of bodies. Although our current reconstruction framework is largely dependent to the measurement of sparse anatomical landmarks, it provides a way to enable dense social signal measurement by defining thousands of corresponding 3D keypoints for each individual. Moreover, the motion capture output of total motion capture is in the same form (joint rotation angles) as in widely used motion capture tools [24], which can be directly used for many graphics applications such as 3D animation and virtual reality. As another key advantage especially for social interaction modeling, the total motion capture output allows us to decouple the body motion cues from the identity specific cues (shape).

5.1 Related Work

Marker-based motion capture systems that track retro-reflective markers [24, 106] are the most widely used method to capture human body motion. However, in addition to a laborious process of attaching markers on subjects, these methods still suffer from major limitations including: (1) a necessity of sparsity in marker density for reliable tracking, which limits the spatial resolution of motion measurements [107]; (2) a limitation in automatically handling occluded markers which requires expensive manual clean-up; and (3) markers on the faces, bodies, and hands hinder participants from engaging in natural social interaction. Due to these limitations, capturing the total body motion of interacting people is still a challenging problem even in state-of-the-art motion capture systems [24].

Markerless motion capture methods have been explored over the past two decades to achieve the same goal of motion capture systems, but they tend to implicitly admit that their performance is inferior to their marker-based counterpart, advocating their "markerless" nature as the major advantage. Most markerless motion capture methods largely focus on the motion of the torso and limbs. The standard pipeline is based on a multiview camera setup and tracking with a 3D template model [30, 108–112, 92, 113, 22, 20, 23]. In this approach, motion capture is performed by aligning a 3D template model to the measurements, which can include colors, textures, silhouettes, point clouds, and keypoints. Recent methods exploit a generative deformable body model [114, 102, 115] to express both shape and body variations of humans. Since these body models often assume minimum clothing for subjects, explicit modeling for clothing is needed to capture clothed subjects [116, 117]. Recent advances in

2D keypoint detection [105, 104, 80] make it possible to reliably reconstruct 3D keypoints in a multiview setup, where a 3D model can be fitted [23, 52, 53]. A specific strength of learning-based detectors is that they can provide a “guess" for occluded parts, based on the spatial human body configurations learned from a large-scale 2D pose dataset. Note that we differentiate markerless motion capture approaches, producing motion parameters as output, from multiview performance capture approaches [25, 26] which aim to obtain detailed surface shapes by free-form mesh deformations. With the introduction of commodity depth sensors, single-view depth-based body motion capture also became a popular direction [27, 93]. More recently, a collection of approaches aims to reconstruct 3D skeletons directly from monocular images, either by fitting 2D keypoint detections with a prior on human pose [118, 119] or getting even closer to direct regression methods [120–122].

In all earlier work, face and hand motion captures are often considered as separate research domains. Facial scanning and performance capture has been greatly advanced over the last decade. There exist multiview methods showing excellent performance on high-quality facial scanning [11, 12] and facial motion capture [13–15]. Recently, lightweight systems based on a single camera show compelling performance by leveraging a morphable 3D face model on 2D measurements [123, 41, 124, 125, 103, 126, 127]. Most of these methods are based on a deformable 3D face rig such as the method of Cao et al. [103]. Hand motion capture is mostly led by single depth-sensor based methods [16, 128, 17, 129–134, 18, 19, 135], with few exceptions based on multi-view systems [136, 133, 137]. Recently, 2D hand keypoint detection and the use of it to obtain 3D hand keypoints in a multiview setup are introduced by Simon et al. [46]. Notably, a generative 3D model that can express body and hands was also introduced by Romero et al. [137].

In contrast, this paper presents the first approach for “total” markerless motion capture of multiple interacting people, producing a parameterized representation that jointly captures the time-varying body pose, hand pose, and facial expressions of each of the interacting participants.

5.2 Frank Model

The motivation for building the Frank[1] body model is to leverage existing part models: SMPL [102] for the body, FaceWarehouse [103] for the face, and an artist-defined hand rig (shown in Fig. 5.1). Each of these capture shape and motion details at an appropriate scale for the corresponding part. This choice is not driven merely by the free availability of the component models: note that due to the trade-off between image resolution and field of view

[1] Frank is an homage to a certain *Modern Prometheus*.

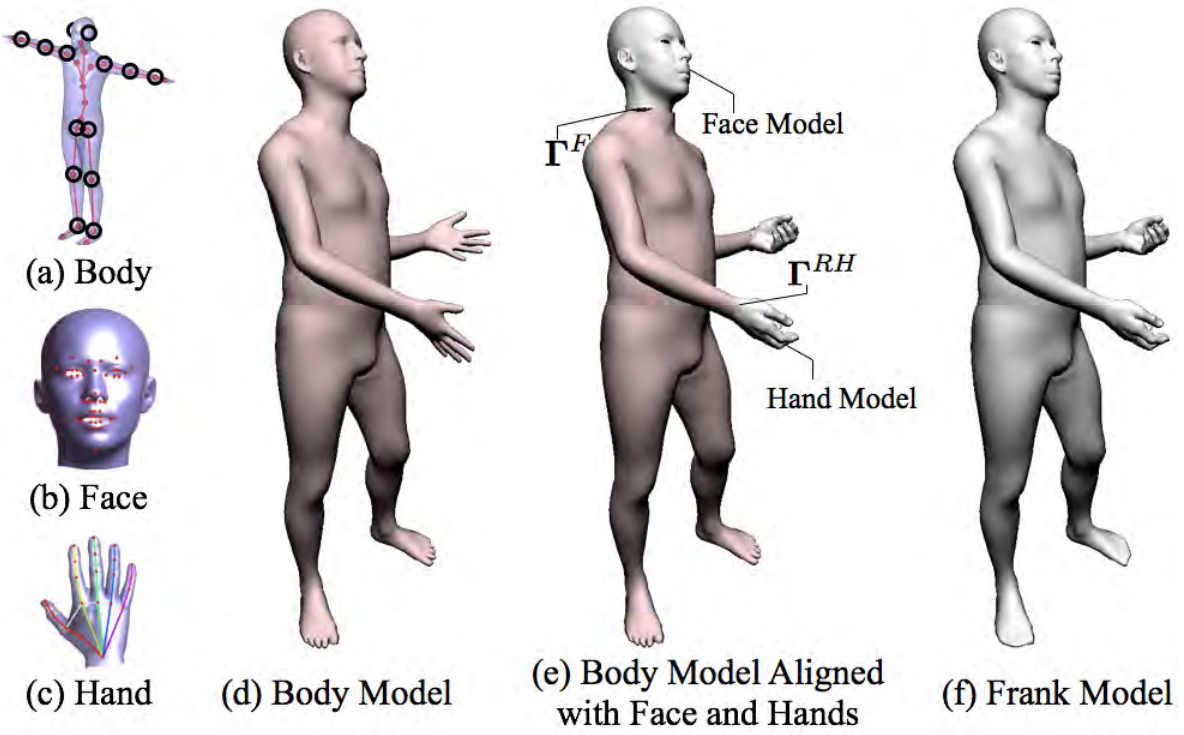


Fig. 5.1 Part models and the Frank model. (a) The body model [102]; (b) the face model [103]; and (c) a hand rig. In (a-c), the red dots have corresponding 3D keypoints reconstructed by detectors; (d) Body only model; (e) Face and hand models substitute the corresponding parts of the body model. Alignments are ensured by $\mathbf{\Gamma}$s; and (f) The blending matrix $\mathbf{C}$ is applied to produce a seamless mesh.

of today's 3D scanning systems, scans used to build detailed face models will generally be captured using a different system than that used for the rest of the body. For our model, we merge all transform bones into a single skeletal hierarchy but keep the native parameterization of each component part to express identity and motion variations. As the final output, the Frank model produces motion parameters capturing the total body motion of humans, and generates a seamless mesh by blending the vertices of the component meshes.

5.2.1 Stitching Part Models

The Frank model M^U is parameterized by motion parameters $\boldsymbol{\theta}^U$, shape (or identity) parameters $\boldsymbol{\phi}^U$, and a global translation parameter $\mathbf{t}^U$,

$$\mathbf{V}^U = M^U(\boldsymbol{\theta}^U, \boldsymbol{\phi}^U, \mathbf{t}^U), \tag{5.1}$$

where $\mathbf{V}^U$ is a seamless mesh expressing the motion and shape of the target subject. The motion and shape parameters of the model are a union of the part models' parameters:

$$\boldsymbol{\theta}^U = \{\boldsymbol{\theta}^B, \boldsymbol{\theta}^F, \boldsymbol{\theta}^{LH}, \boldsymbol{\theta}^{RH}\}, \tag{5.2}$$

$$\boldsymbol{\phi}^U = \{\boldsymbol{\phi}^B, \boldsymbol{\phi}^F, \boldsymbol{\phi}^{LH}, \boldsymbol{\phi}^{RH}\}, \tag{5.3}$$

where the superscripts represent each part model: B for the body model, F for the face model, LH for the left hand model, and RH for the right hand model. Each of the component part models maps from a subset of the above parameters to a set of vertices, respectively, $\mathbf{V}^B \in \mathbb{R}^{N^B \times 3}$, $\mathbf{V}^F \in \mathbb{R}^{N^F \times 3}$, $\mathbf{V}^{LH} \in \mathbb{R}^{N^H \times 3}$, and $\mathbf{V}^{RH} \in \mathbb{R}^{N^H \times 3}$, where the number of vertices of each mesh part is N^B=6890, N^F=11510, and N^H=2068. The final mesh of the Frank model, $\mathbf{V}^U \in \mathbb{R}^{N^U \times 3}$, is defined by linearly blending them with a matrix $\mathbf{C} \in \mathbb{R}^{N^U \times (N^B + N^F + 2N^H)}$:

$$\mathbf{V}^U = \mathbf{C} \left[\begin{array}{cccc} (\mathbf{V}^B)^T & (\mathbf{V}^F)^T & (\mathbf{V}^{LH})^T & (\mathbf{V}^{RH})^T \end{array} \right]^T, \tag{5.4}$$

where T denotes the transpose of a matrix. Note that $\mathbf{V}^U$ has fewer vertices than the sum of part models because there are redundant parts in the body model (e.g., face and hands of the body model). In particular, our final mesh has N^U=18540 vertices. Fig. 5.1 (e) shows the part models that are aligned, and (f) shows the final mesh topology of the Frank model after applying the the blending matrix $\mathbf{C}$ at the mean shape in the rest pose. The blending matrix $\mathbf{C}$ is a very sparse matrix; most rows have a single column set to one with zeros elsewhere and simply copy the vertex locations from the corresponding part models with minimal interpolation at the seams.

In the Frank model, all parts are rigidly linked by a single skeletal hierarchy, which is crucial as an output of motion capture. This unification is achieved by substituting the hands and face branches of the SMPL body skeleton with the corresponding skeletal hierarchies of the detailed part models. All parameters of the Frank model are jointly optimized for motion tracking and identity fitting. The parameterization of each of the part models is detailed in the following sections.

5.2.2 Body Model

For the body, we use the SMPL model [102] with minor modifications. In this section, we summarize the salient aspects of the model in our notation. The body model, M^B, is defined as follows,

$$\mathbf{V}^B = M^B(\boldsymbol{\theta}^B, \boldsymbol{\phi}^B, \boldsymbol{t}^B), \tag{5.5}$$

with $\mathbf{V}^B = \{\mathbf{v}_i^B\}_{i=1}^{N^B}$. The model uses a template mesh of N^B=6890 vertices, where we denote the i-th vertex as $\mathbf{v}_i^B \in \mathbb{R}^3$. The vertices of this template mesh are first displaced by a set of blendshapes describing the *identity* or body shape. Given the vertices in the rest pose, the posed mesh vertices are obtained by linear blend skinning (LBS) using transformation matrices $\mathbf{T}_j^B \in \mathrm{SE}(3)$ for each of the J joints,

$$\mathbf{v}_i^B = \mathbf{I}_{3\times 4} \cdot \sum_{j=1}^{J^B} w_{i,j}^B \mathbf{T}_j^B \begin{pmatrix} \mathbf{v}_i^{B0} + \sum_{k=1}^{K_b} \mathbf{b}_i^k \phi_k^B \\ 1 \end{pmatrix}, \tag{5.6}$$

where $\mathbf{b}_i^k \in \mathbb{R}^3$ is the i-th vertex of the k-th blendshape, ϕ_k^B is the k-th shape coefficient in $\boldsymbol{\phi}^B \in \mathbb{R}^{K_b}$ with K_b=10 the number of identity body shape coefficients, and $\mathbf{v}_i^{B0}$ is the i-th vertex of the mean shape. The transformation matrices $\mathbf{T}_j^B$ encode the transform for each joint j from the rest pose to the posed mesh in world coordinates, which is constructed by traversing the skeleton hierarchy from the root joint with pose parameter $\boldsymbol{\theta}^B$ (see [102]). The j-th pose parameter θ_j^B is the angle-axis representation of the relative rotation of joint j with respect to its parent joints. $w_{i,j}^B$ is the weight with which transform $\mathbf{T}_j^B$ affects vertex i, with $\sum_{j=1}^{J^B} w_{i,j}^B$=1 and $\mathbf{I}_{3\times 4}$ is the 3×4 truncated identity matrix to transform from homogeneous coordinates to a 3 dimensional vector. We use J^B=21 with $\boldsymbol{\theta}^B \in \mathbb{R}^{21\times 3}$, ignoring the last joint of each hand of the SMPL model. For simplicity, we do not use the pose-dependent blendshapes[2].

5.2.3 Face Model

As a face model, we build a generative PCA model from the FaceWarehouse dataset [103]. Specifically, the face part model, M^F, is defined as follows,

$$\mathbf{V}^F = M^F(\boldsymbol{\theta}^F, \boldsymbol{\phi}^F, \mathbf{T}^F), \tag{5.7}$$

with $\mathbf{V}^F = \{\mathbf{v}_i^F\}_{i=1}^{N^F}$, where the i-th vertex is $\mathbf{v}_i^F \in \mathbb{R}^3$, and N^F=11510. The vertices are represented by combining shape and expression subspaces:

$$\hat{\mathbf{v}}_i^F = \mathbf{v}_i^{F0} + \sum_{k=1}^{K_f} \mathbf{f}_i^k \phi_k^F + \sum_{s=1}^{K_e} \mathbf{e}_i^s \theta_s^F \tag{5.8}$$

where, as before, $\mathbf{v}_i^{F0}$ denotes i-th vertex of the mean shape, and ϕ_k^F and θ_s^F are the k-th face identity (shape) and s-th facial expression (pose) parameters respectively. Here, $\mathbf{f}_i^k \in \mathbb{R}^3$ is

[2]For our target sequences, the modeling error between the SMPL model [102] and the 3D surface measurements is dominated by clothing artifacts, which the pose-blendshapes were not trained on.

the i-th vertex of the k-th identity blendshape ($K_f = 150$), and $\mathbf{e}_i^s \in \mathbb{R}^3$ is the i-th vertex of the s-th expression blendshape ($K_e = 200$).

Finally, a transformation $\mathbf{T}^F$ brings the face vertices into world coordinates. To ensure that the face vertices transform in accordance to the rest of the body, we assume that the mean face $\mathbf{v}_i^{F0}$ is aligned with the body mean shape as shown in Fig. 5.1, which is manually done in building the model. This way, we can apply the transformation of the body model's head joint $\mathbf{T}^B_{j=F}(\boldsymbol{\theta}^B)$ as a global transformation for the face model in Eq. 5.9. However, to keep the face in alignment with the body, an additional transform matrix $\boldsymbol{\Gamma}^F \in \mathrm{SE}(3)$ is required to compensate for displacements in the root location of the face joint due to body shape changes in Eq. 5.6.

Finally, each face vertex position is given by:

$$\mathbf{v}_i^F = \mathbf{I}_{3\times 4} \cdot \mathbf{T}^B_{j=F} \cdot \boldsymbol{\Gamma}^F \begin{pmatrix} \hat{\mathbf{v}}_i^F \\ 1 \end{pmatrix}, \tag{5.9}$$

where the transform $\boldsymbol{\Gamma}^F$, which is directly determined by the body shape parameters $\boldsymbol{\phi}^B$, aligns the face model with the body model.

5.2.4 Hand Model

We use an artist-rigged hand mesh. Our hand model has J^H=16 joints and the mesh is again deformed via linear blend skinning. The hand model has a fixed shape, but we introduce scaling parameters for each bone to allow for different finger sizes. The transform for the j-th joint is parameterized by the Euler angle rotation with respect to its parent, $\boldsymbol{\theta}_j^H \in \mathbb{R}^3$, and an additional anisotropic scaling factor along each axis, $\boldsymbol{\phi}_j^H \in \mathbb{R}^3$. Specifically, the linear transform for the j-th joint in the bone's local reference frame becomes $\mathrm{eul}(\boldsymbol{\theta}_j^H) \cdot \mathrm{diag}(\boldsymbol{s}_j^H)$, where $\mathrm{eul}(\boldsymbol{\theta}_j^H)$ converts from an Euler angle representation to a 3×3 rotation matrix and $\mathrm{diag}(\boldsymbol{\phi}_j^H)$ is the 3×3 diagonal matrix with the X,Y,Z scaling factors ϕ_j^H on the diagonal. The vertices of the hand in world coordinates are given by LBS with weights $w_{i,j}^H$:

$$\mathbf{v}_i^H = \mathbf{I}_{3\times 4} \cdot \mathbf{T}^B_{j=H} \cdot \boldsymbol{\Gamma}^H \cdot \sum_{j=1}^{J} w_{i,j}^H \mathbf{T}_j^H \begin{pmatrix} \mathbf{v}_i^{H0} \\ 1 \end{pmatrix}. \tag{5.10}$$

where $\mathbf{v}_i^{H0}$ denotes i-th vertex of the mean shape , $\mathbf{T}_j^H$ is each bone's composed transform (with all parents in the hierarchy), $\mathbf{T}^B_{j=H} \in \mathrm{SE}(3)$ is the transformation of the corresponding hand joint in the body model, and $\boldsymbol{\Gamma}^H$ is the transformation that aligns the hand model to the

body model. As with the face, this transform depends on the shape parameters of the body model.

5.3 Motion Capture with Frank

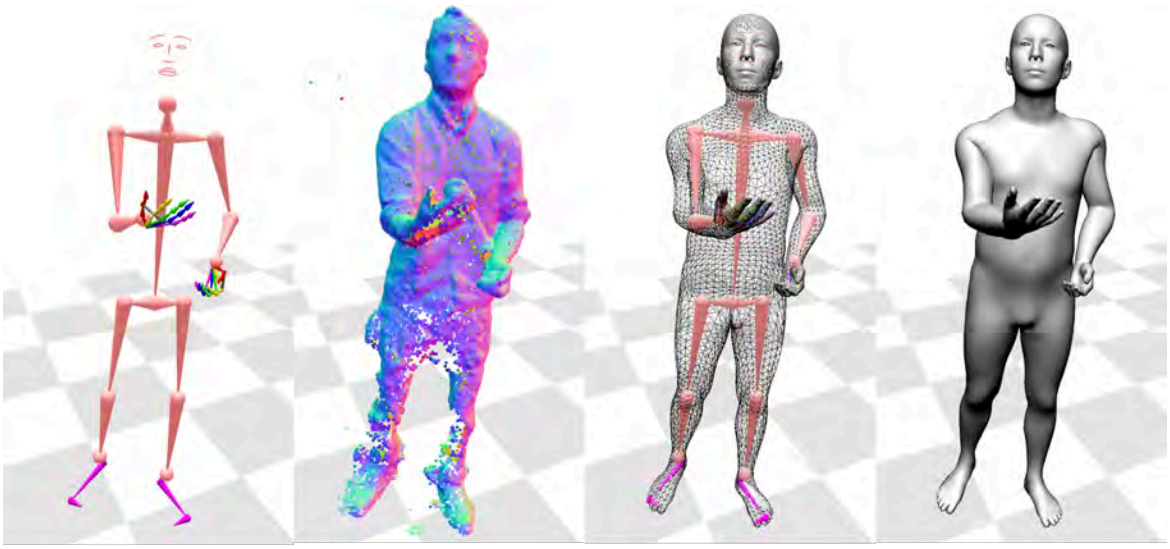
We fit the Frank model to data to capture the total body motion, including the major limbs, the face, and fingers. Our motion capture method relies heavily on fitting mesh correspondences to 3D keypoints, which are obtained by triangulation of 2D keypoint detections across multiple camera views. To capture shape information we also use point clouds generated by multiview stereo reconstructions. Model fitting is performed by an optimization framework to minimize distances between corresponded model joints and surface points and 3D keypoint detections, and iterative closest point (ICP) to the 3D point cloud. Note that more details are provided in the supplementary material.

5.3.1 3D Measurements

We incorporate two types of measurements in our framework as shown in Fig. 5.2: (1) corresponded 3D keypoints, which map to known joints or surface points on the mesh models (see Fig. 5.1), and (2) uncorresponded 3D points from multiview stereo reconstruction, which we match using ICP.

3D Body, Face, and Hand Keypoints: We use the OpenPose detector [47] in each available view, which produces 2D keypoints on the body with the method of Cao et al. [104], and hand and face keypoints using the method of Simon et al. [46]. 3D body skeletons are obtained from the 2D detections using the method of [53], which uses known camera calibration parameters for reconstruction. The 3D hand keypoints are obtained by triangulating 2D hand pose detections, following the method of [46], and similarly for the facial keypoints. Note that subsets of 3D keypoints can be entirely missing if there are not enough 2D detections for triangulation, which can happen in challenging scenes with inter-occlusions or motion blur.

3D Feet Keypoints: An important cue missing from the OpenPose detector is keypoints on the feet. For motion capture, this is an essential feature to accurately determine the orientation of the feet. We therefore train a keypoint detector for the tip of the big toe, the tip of the little toe, and the ball of the foot. We annotate these 3 keypoints per foot in each of around 5000 person instances of the COCO dataset, and use the architecture of Wei et al. [80] with a bounding box around the feet determined by the 3D body detections. We also apply multiview bootstrapping in the Panoptic Studio to improve the quality, as described by Simon et al. [46].



(a) 3D Key Points (b) 3D Point Cloud by MVS (c) Frankenstein Model Fitting Result

Fig. 5.2 Fitting Frank: The optimization takes, as input, (a) 3D keypoints, and (b) point clouds, and produces (c) a fitted skeleton and mesh as output.

3D Point Clouds: We use the commercial software RealityCapture [138] to obtain 3D point clouds from the multiview images, with associated point normals.

5.3.2 Objective Function

We initially fit every frame in the sequence independently. For clarity, we drop the time index from the notation and describe the process for a single frame, which optimizes the following cost function:

$$E\left(\boldsymbol{\theta}^U, \boldsymbol{\phi}^U, \boldsymbol{t}^U\right) = E_{\text{keypoints}} + E_{\text{icp}} + E_{\text{seam}} + E_{\text{prior}} \tag{5.11}$$

We use Levenberg-Marquardt with the Ceres Solver library [101] with multiple stages to avoid local minima. See the supplementary material for the details.

Anatomical Keypoint Cost: The term $E_{\text{keypoints}}$ matches 3D keypoint detections, which are in direct corresponce to our mesh models. This term includes joints (or end effectors) in the body and hands, and also contains points corresponding to the surface of the mesh (e.g., facial keypoints and the tips of fingers and toes). Both of these types of correspondence are expressed as combinations of vertices via a regression matrix $\mathbf{J} \in \mathbb{R}^{C \times N^U}$, where C denotes the number of correspondences and N^U is the number of vertices in the model. Let $\mathscr{D}$ denote

the set of available detections in a particular frame. The cost is then:

$$E_{\text{keypoints}} = \lambda_{\text{keypoints}} \sum_{i \in \mathscr{D}} ||\mathbf{J}_i \mathbf{V} - \mathbf{y}_i^T||^2, \tag{5.12}$$

where $\mathbf{J}_i$ indexes a row in the correspondence regression matrix and represents an interpolated position using a small number of vertices, and $\mathbf{y}_i \in \mathbb{R}^{3\times 1}$ is the 3D detection. The $\lambda_{\text{keypoints}}$ is a relative weight for this term.

ICP Cost: The 3D point cloud measurements are not a priori in correspondence with the model meshes. We therefore establish their correspondence to the mesh using Iterative Closest Point (ICP) during each solver iteration. We find the closest 3D point in the point cloud to each of the mesh vertices, and compute the point-to-plane residual, i.e., the distance along the normal direction,

$$E_{\text{icp}} = \lambda_{\text{icp}} \sum_{\mathbf{v}_j \in \mathbf{V}^U} \mathbf{n}(\mathbf{x}_{j^*})^T (\mathbf{x}_{j^*} - \mathbf{v}_j), \tag{5.13}$$

where $\mathbf{x}_{i^*}$ is the closest 3D point to j-th vertex $\mathbf{v}_j$, $\mathbf{n}(\cdot) \in \mathbb{R}^3$ represents the point's normal, and λ_{icp} is a relative weight for this term.

Seam Constraints: The part models composing the Frank model are rigidly linked by the skeletal hierarchy. However, the independent surface parameterizations of each of the part models may introduce discontinuities at the boundary between parts (e.g., a fat arm with a thin wrist). To avoid this artifact, we encourage the vertices around the seam parts to be close by penalizing differences between the last two rings of vertices around the seam of each part, and the corresponding closest point in the body model in the rest pose expressed as barycentric coordinates.

Prior Cost: Depending on the number of measurements available in a particular frame, the set of parameters of M^U may not be determined uniquely (e.g., the width of the fingers). More importantly, the 3D point clouds are noisy and cannot be well explained by the model due to hair and clothing, which are not captured by the SMPL and FaceWarehouse meshes, resulting in erroneous correspondences during ICP. Additionally, the joint locations of the models are not necessarily consistent with the annotation criteria used to train the 2D detectors. We are therefore forced to set priors over model parameters to avoid the model from overfitting to these sources of noise, $E_{\text{prior}} = E^F_{\text{prior}} + E^B_{\text{prior}} + E^H_{\text{prior}}$. The prior for each part is defined by corresponding shape and pose priors, for which we use zero-mean standard normal priors for each parameter except for scaling factors, which are encouraged to be close to 1.

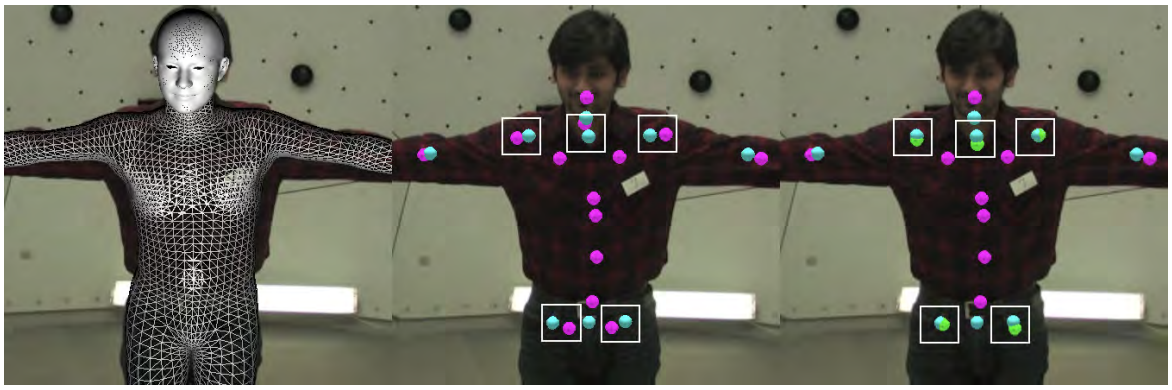


Fig. 5.3 Regressing detection target 3D positions. (Left) The template model is aligned with target object; (Mid.) The torso joints of the template model (magenta) have discrepancy from the joint definitions of 3D keypoint detection (cyan); (Right) The newly regressed target locations (green) are more consistent with 3D keypoint detections.

5.4 Creating Adam

We derive a new model, Adam, enabling total body motion capture with a simpler parameterization than the part-based Frank model. In particular, this new model has a single joint hierarchy and a common parameterization for all shape degrees of freedom, tying together the face, hand, and body shapes and avoiding the need for separate part parameterizations or seam constraints. To build the model, it is necessary to align the reconstructed meshes with all body parts (face, body, and hands) of diverse subjects where the model can learn the variations. To do this, we leverage our Frank model and apply it on a dataset of 70 subjects where each of them performs a short range of motion in a multiview camera system. We select 5 frames for each person in different poses, resulting in 350 meshes, and reconstruct them with our Frank model, producing aligned meshes with joint locations to build Adam. Because we derive the model from clothed people, the blendshapes explain variations of clothing at a coarse level.

5.4.1 Fitting Clothes and Hair

The Frank model captures the shape variability of human bodies and faces, but does not account for clothing or hair, since it keeps the original model space of part models ([102] and [103]). To learn a new set of linear blendshapes that better capture the rough geometry of clothed people and also roughly model hair, we need the meshes to match the geometry of the source data more accurately. For this purpose, we deform the meshes outside of the shape-space along each the normal direction of each vertex. For each vertex $\mathbf{v}_i$ in the Frank

model, the deformed mesh vertex $\tilde{\mathbf{v}}_i$ is represented as:

$$\tilde{\mathbf{v}}_i = \mathbf{v}_i + \mathbf{n}(\mathbf{v}_i)\delta_i, \tag{5.14}$$

where $\delta_i \in \mathbb{R}$ is a scalar displacement meant to compensate for the discrepancy between the Frank model vertices and the 3D point cloud, along the normal direction at each vertex. We pose the problem as a linear system,

$$\begin{pmatrix} \mathbf{N}^T \\ (\mathbf{WLN})^T \end{pmatrix} \Delta = \begin{pmatrix} (\mathbf{P} - \mathbf{V}^U)^T \\ \mathbf{0} \end{pmatrix}, \tag{5.15}$$

where $\Delta \in \mathbb{R}^{N^U}$ contains the stacked per-vertex displacements, $\mathbf{V}^U$ are the vertices in the Frank model, $\mathbf{P} \in \mathbb{R}^{N^U \times 3}$ are corresponding point cloud points, $\mathbf{N} \in \mathbb{R}^{N^U \times 3}$ contains the mesh vertex normals, and $\mathbf{L} \in \mathbb{R}^{N^U \times N^U}$ is the Laplace-Beltrami operator to regularize the deformation. We also use a diagonal weight matrix $\mathbf{W} \in \mathbb{R}^{N^U \times N^U}$ to avoid large deformations where the 3D point cloud has lower resolution than the original mesh, such as details in the face and hands.

5.4.2 Detection Target Regression

There exists an important discrepancy between the joint locations of the LBS model (i.e., the 3D centers of rotation for bone deformation) and the location of the keypoint detections (which come from manually annotated guesses of where the anatomical joints are in 2D images). This is shown in Fig. 5.3. This difference has the effect of pulling the model towards a bad fit even while achieving a low keypoint cost, $E_{\text{keypoints}}$, especially for shoulders and hips. We alleviate this problem by computing a new regression function, $\hat{\mathbf{J}}^A \in \mathbb{R}^{J^A \times N^U}$, which relates the vertices in the body model to the expected location of 3D keypoint detections. However, to be able to learn these regressors, we require instances of the fitted model vertices as well as the 3D keypoint detections.

Therefore, we first fit the Frank model (with additional shape variations) using the original joint locations as detection targets, and obtain aligned meshes across all subjects. Based on these outputs, we can build the regression matrix using the locations of 3D keypoint measurements as targets instead of Frank model's joint locations. Similar to the joint regression in SMPL [102], we first select a subset of vertices in the proximity of each detection target, and estimate a fixed, sparse linear combination of these vertices that approximates the location of the 3D keypoint across all fitted meshes. This optimization is

posed as an L1-regularized least-squares problem with non-negative constraints, where we additionally impose that the vertex weights sum to one, resulting in an interpolation.

The results are shown in Fig. 5.3. Note that this new regressor is used only for the optimization in Eq. (5.12), whereas the original joint regressor from SMPL [102], $\mathbf{J}^A$, is used for LBS. However, we also add rows to the joint regression matrix to account for the additional finger joints, which we solve for in the same way. The resulting matrix is $\mathbf{J}^A \in \mathbb{R}^{J^A \times N^U}$ where N^U is the number of vertices of Adam (the same as Frank) and $J^A = 61$ is the number of joints in Adam model including 21 body joints and 20 finger joints (including 5 finger tips) for each hand.

5.4.3 Building the Shape Deformation Space

After model fitting with Δ displacement, we warp each frame's surface to the rest pose, applying the inverse of the LBS transform. With the fitted surfaces warped to this canonical pose, we do PCA analysis to build a joint linear shape space that captures shape variations across the entire body. As in Section 5.2.3, we separate the expression basis for the face and retain the expression basis from the FaceWarehouse model, as our MVS point clouds are of too low resolution to fit facial expressions.

The Adam model is parameterized as:

$$M^A(\boldsymbol{\theta}^A, \boldsymbol{\phi}^A, \boldsymbol{t}^A) = \mathbf{V}^A \tag{5.16}$$

with $\mathbf{V}^A = \{\mathbf{v}_i^A\}_{i=1}^{N^A}$ and N^A=18540 which is equal to the vertices in Frank, N^U. As in SMPL, the vertices of this template mesh are first displaced by a set of blendshapes in the rest pose, $\hat{\mathbf{v}}_i^A = \mathbf{v}_i^{A0} + \sum_{k=1}^{K_A} \mathbf{s}_i^k \phi_k^A$, where $\mathbf{s}_i^k \in \mathbb{R}^3$ is the i-th vertex of the k-th blendshape, ϕ_k^A is the k-th shape coefficients of $\boldsymbol{\phi}^A \in \mathbb{R}^{K_b}$, and $K_A = 40$ is the number of identity coefficients, $\mathbf{v}^{A0}$ is the mean shape and $\mathbf{v}_i^{A0}$ is its i-th vertex. Note that these blendshapes now capture variation across the face, hands, and body. These are then posed using LBS as in Eq. (5.6) after obtaining joint locations by the joint regressor matrix $\mathbf{J}^A$.

5.4.4 Tracking with Adam

The cost function to capture total body motion using Adam is similar to Eqn. 5.11 without the seam term:

$$E\left(\boldsymbol{\theta}^A, \boldsymbol{\phi}^A, \boldsymbol{t}^A\right) = E_{\text{keypoints}} + E_{\text{icp}} + E_{\text{prior}}. \tag{5.17}$$

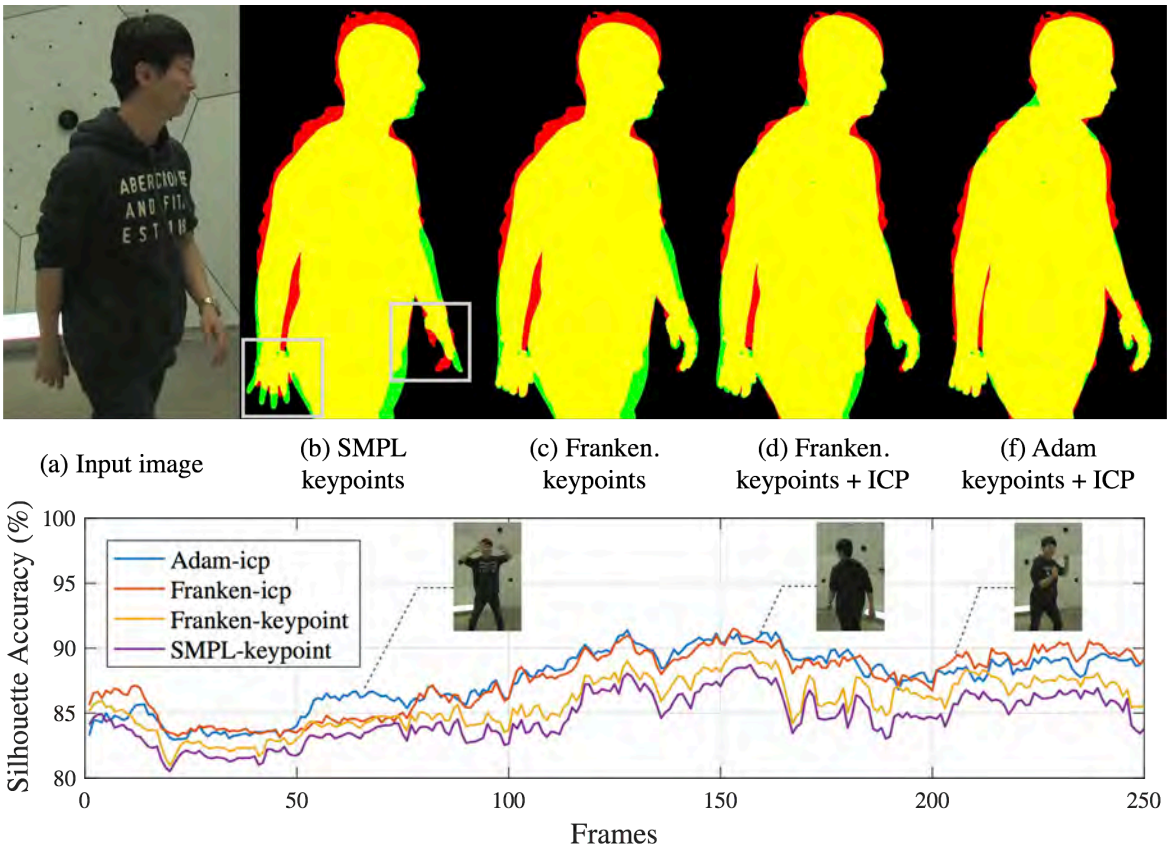


Fig. 5.4 (Top) The silhouette from different methods overlayed with ground-truth. The ground truth is drawn in the red channel and the rendered silhouette masks from each model are drawn in the green channel. Thus, the correctly overlapped region is shown as yellow color. (Bottom) Silhouette accuracy compared to the ground truth silhouette.

However, Adam is much more amenable to optimization than Frank: it has a single set of unified shape and pose parameters for all parts, and does not require seam constraints between disparate models.

5.5 Results

We perform total motion capture using our two models, Frank and Adam, on various challenging sequences. For experiments, we use the dataset captured in the CMU Panoptic Studio [52, 53]. We use 140 VGA cameras to reconstruct 3D body keypoints, 480 VGA cameras for feet, and 31 HD cameras for faces and hands keypoints, and 3D point clouds. We compare the fits produced by our models with the simplified[3] SMPL model [102].

[3] In all our comparison, we disabled the pose-dependent blendshapes of SMPL, and thus here SMPL model means the body part of Frank.

Table 5.1 Accuracy of Silhouettes from different models

	SMPL[102]	Frank	Frank ICP	Adam ICP
Mean	84.79%	85.91%	87.68%	87.74%
Std.	4.55	4.57	4.53	4.18

5.5.1 Quantitative Evaluation

We evaluate how well each model can match a moving person by measuring overlap with the ground truth silhouette across 5 different viewpoints for a 10 second sequence. To obtain the ground truth silhouette, we run a background subtraction algorithm using a Gaussian model for the color of each pixel, with post-processing by morphological transforms to remove noise. As an evaluation metric, we compute the percentage of overlap compared to the union between the GT silhouettes and the rendered forground masks after fitting each model. Here, we compare the fitting results of 3 different models: SMPL, our Frank, and our Adam models. The results are shown in Fig. 5.4 and Table 5.1. We first compare accuracy between SMPL and Frank model by using only 3D keypoints as measurement cues. The major source of improvement of Frank over SMPL is in the articulated hand model (by construction, the body is almost identical). Including ICP term as cues provides better accuracy. Finally in the comparison between our two models, they show almost similar performance. Ideally we expect Adam to outperform Frank because it has more expressive power for hair and clothing, and it shows better performance for certain body shapes (frame 50-75 in Fig. 5.4). However, Adam sometimes produces artifacts showing lower accuracy: it tends to generate thinner legs, mainly due to poor 3D point cloud reconstructions in the training data[4]. However, Adam is simpler for total body motion capture and has potential to be improved once a large dataset is available with a more optimized capture setup.

5.5.2 Qualitative Results

We run our method on sequences where face and hand motions naturally occur. The sequences include short range of motion for 70 people used to build Adam, social interactions of multiple people, a furniture building sequence with dexterous hand motions, musical performances (cello and guitars), and commonly observable daily motions such as typing. Most of these sequences are rarely demonstrated in previous markerless motion capture methods since capturing subtle details is key to achieve realism. Example results are shown in Figure 5.5 but are best seen in the accompanying videos. Here, we also qualitatively compare our models (in silver color for Frank, and gold for Adam) with SMPL (without pose-blendshapes, in

[4]Due to dark clothing combined with fewer camera views of the legs.

pink) [102]. Note that total body motion capture based on our models produces more realism by capturing subtle details from the hands and faces.

5.6 Discussion

We present the first markerless method to capture total body motion including facial expression, body motion from torso and limbs, and hand gestures at a distance. To achieve this result, we present two types of models, Frank and Adam, which can express motion in each of the parts. Our reconstruction results show compelling and realistic results, even when using only sparse 3D keypoint detections to drive the models. As a current limitation of our system, Adam lacks expressive power in surface details due to the limited number of subjects in training. However, the major value of Adam model over Frank lies in its simpler representation to capture total body motion, which can be useful for other applications.

There are two interesting points our paper raises. First, markerless hand motion capture, often considered too challenging compared to body and face captures, shows better localization quality in our results. Body joints are located inside the body and are hard to localize for clothed subjects, and the accuracy of face reconstruction greatly decreases once the face is not facing any camera. However, hands are often bare and the hand keypoint detector [46] provides guessed measurements with high confidence even in self-occlusions, which can be fused in multiple views. Second, our results show a potential that markerless motion capture can eventually outperform its marker-based counterpart. Marker-based methods strongly suffer from occlusions, making it hard to capture both body and hands together, while our method can still exploit measurements for occluded parts by learning-based keypoint detectors.

Fig. 5.5 Total body reconstruction results on various human body motions. For each example scene, the fitting results from three different models are shown by different colors (pink for SMPL [102], silver for Frank, and gold for Adam).

Part III

Modeling Social Signals

"We respond to gestures with an extreme alertness and, one might almost say, in accordance with an elaborate and secret code that is written nowhere, known by none, and understood by all"

— Edward Sapir [139]

Chapter 6

A Large-Scale Social Interaction Corpus

Availability of a large-scale dataset is essential to computationally investigate the nonverbal communication in a data-driven manner. Despite existing datasets that provide measurements for human motion and behaviors [34, 35, 83–85, 140], there is no dataset that satisfies the following core requirements for understanding nonverbal human behaviors: (1) capturing 3D full body motion with a broad spectrum of nonverbal cues (including face, body, and hands); (2) capturing signals of naturally interacting groups (more than two people to include attention switching); and (3) collecting the data at scale. The limited availability of the dataset motivates us to build a new dataset that contains social interactions among hundreds of interacting groups with a broad spectrum of 3D body motion measurements. The key properties of our dataset are as follows:

- Our dataset contains naturally interacting multiple people in a negotiation game scenario, where the game is carefully designed to induce natural and spontaneous interaction
- Participants are randomly recruited
- No behavioral restriction is instructed to participants during the capture
- A broad spectrum of social signals, including the motion from faces, bodies, and hands, are measured using our markerless motion capture (Chapters 4 and 5)
- Our dataset contains synchronized multiple modalities, including RGB videos from over 500 views, depth maps from 10 RGB+D sensors, and sound from 23 microphones
- Voice signals of individuals are recorded via wireless and wired microphones, and speaking status and timing of each subject are manually annotated

- 3D point clouds are provided by fusing the depth maps from 10 RGB+D sensors

Our dataset provides a new opportunity to investigate the dynamics of various interpersonal nonverbal behavioral cues emerging in social situations. Our dataset is captured under a university-approved IRB protocol[1] and publicly released for research purposes, with agreements from all participants[2].

6.1 Related Work

Recognizing emotions from social signals is one of the most popular directions as a way to analyze nonverbal behaviors. Most prior approaches in this domain focus on deciphering the semantic meanings of social signals, which are often annotated into a few discrete categories [141] or a set of latent dimensions[142–144] (see a recent survey [145]). A lot of datasets have been presented to tackle this problem [146–151]. Recognizing emotions from body gestures is a less investigated field. Almost all databases in this domain are collected from actors where the actors are instructed to behave certain emotions [38–40]. Several approaches demonstrate that not just facial expression but the context of the scene including body gestures is important in emotion perception [45, 44]. The dataset collected for this direction contains a single subject only, and thus they are not applicable for modeling nonverbal communication.

A few datasets contain the scenes of socially interacting groups [152–155]. The interactions in these datasets are often in a table setup, where participants' motions are limited and only upper-body of the participants are captured. Importantly, these datasets do no provide accurate 3D motion capture measurements. There are datasets that capture free-standing conversational groups (e.g., cocktail party) [83–85, 156], but these databases also contain coarse signal annotations only such as location and orientations of torso, targeting to study social formation or category-level recognition tasks (e.g., speaker, personality, or role recognition).

Datasets providing accurate 3D body motion capture exist in computer vision and graphics [157, 140, 158]. However, they often contain single subjects and do not include communicative behaviors. The motion capture data capturing casual social communication is extremely rare.

[1] IRBSTUDY2015_00000478
[2] http://domedb.perception.cs.cmu.edu

Fig. 6.1 Before starting the social game capture, participants are instructed the game rules and also spent time to be accustomed to the Panoptic Studio environment, as shown in these photos. We follow a common and strict protocol during all captures to avoid any potential bias.

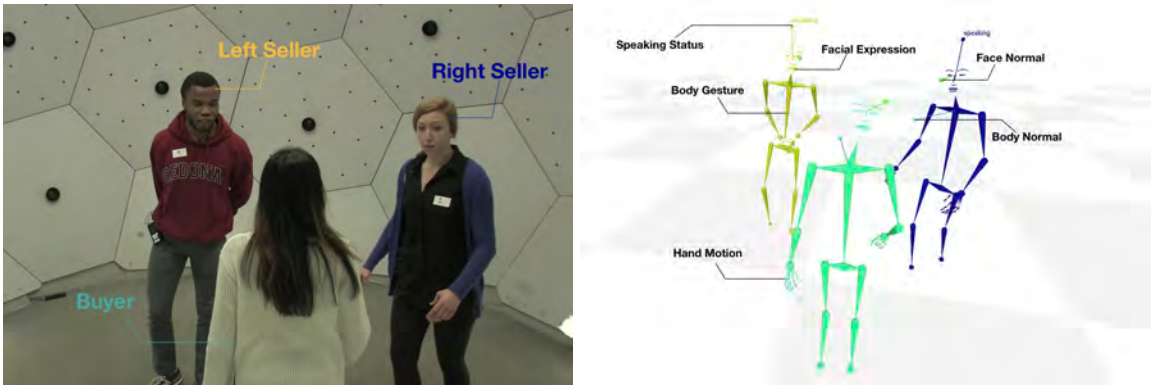


Fig. 6.2 An example of the haggling sequence. (Left) an example scene showing two sellers and one buyer. (Right) Reconstructed 3D social signals showing the 3D body, 3D face, and 3D hand motion. 3D normal direction from faces and bodies are also computed, and speaking status of each individual, manually annotated, is also visualized here.

Items	Seller 1	Seller 2
Phone	Light weight Medium storage	Medium weight Large storage
Laptop	Light weight Medium speed	Medium weight Fast speed
Tablet PC	Large storage Medium speed	Medium storage Fast speed
Speaker	High quality audio Wired	Medium quality audio Wireless

Table 6.1 Examples of items assigned to sellers in our Haggling games.

6.2 The Haggling Game Protocol

To evoke natural interactions, we involved participants in a social game named the *Haggling* game. We invent this game to simulate a haggling situation among two sellers and a buyer. The triadic interaction is chosen to include interesting social behaviors such as turn taking and attention changes, which are missing in previous dyadic interaction datasets [155]. During the game, two sellers are promoting their own comparable products for selling, and a buyer makes a decision about which product he/she buys between the two. The game lasts for a minute, and the seller who has sold his/her product is awarded $5. To maximize the influence of each seller's behavior on the buyer's decision-making, the items assigned to sellers are similar products with slightly different properties. Example items are shown in Table 6.1.

For every capture, we follow the protocol described below. We randomly recruited participants using CMU Participation Pool[3]. Over the 8 days of captures, 122 subjects participated and 180 haggling sequences were captured (about 3 hours of data). The participants arrived at the lab for the capture first sign on the IRB consent form with an agreement to publicly release the data for research purposes only[4] A unique identification number is assigned to each participant, and participants are also equipped with a wireless microphone. Then, all subjects are informed of the rules of the Haggling game by watching a pre-recorded presentation video together. Notably, they are not instructed about how to behave during the game, nor is their clothing or appearance controlled. All motions in the sequences are spontaneous social behaviors based on the informed game rules. After introducing the game rules, participants are asked to spend time inside the studio so that they can be accustomed to the interior view of the Panoptic Studio. Before starting the capture, groups and roles are randomly assigned, and participants line up based on their orders. We provide descriptions

[3] https://cbdr.cmu.edu/

[4] .

about the items written in small cards to sellers 1 minute before the game, and the sellers return the card before entering the studio. With a starting signal, participants in a group enter the studio and start the haggling game immediately. The positions and orientations of the groups in the system are also spontaneously decided. During the capture, their all social signals including voice, positions, orientations, and body motions are recorded. We send a signal by ringing a bell 10 seconds before the end of the game, and send the same alarm at the end of the game. After the capture, the buyer annotates the decision between the two items in the prepared result sheet. The captured sequences contain a lot of voluntary social behaviors of diverse people in a common social context. Example scenes are shown in Figure 6.2 and 6.3.

6.3 Measured Social Signals in Our Corpus

We use our Panoptic Studio System to reconstruct 3D social signals (Chapters 2, 4, and 5. As a key advantage, our method does not require attaching markers on the subject's body, and no behavior restrictions nor initialization poses are needed from the subjects. As output, the system captures 3D body motion, 3D face motion, and 3D hand motion for each individual. From these measurements, we additionally compute the body orientation and face orientation by finding the 3D normal directions of torso and face. Finally, we fit our Adam model [48] to reconstruct both body shape and joint angle parameters, which is the similar form as in a motion capture system. Example visualizations of these social signal measurements are shown in Figure 6.2, 6.4, and 6.5. The voice data of each individual is also recorded by wireless microphones assigned to each individual. From the audio signal, we manually annotate a binary speaking label describing whether the target subject is speaking (labeled as 1) or not speaking (labeled as 0).

6.4 Panoptic Studio Database

Our entire dataset contains various types of human motions in diverse scenarios, containing more than 13 hours of sequences. Many researchers and artists in diverse fields including computer science, business, psychology, and visual arts also exploit our system for various applications. We publicly release various such sequences, including several other social games described below. Categories and detailed information on our dataset are shown in the Table 6.2.

Fig. 6.3 Example scenes of haggling sequences with social signal measurements. For each example, HD images overlaid by the projections of 3D anatomical keypoints (from bodies, faces, and hands) are shown, along with a 3D view of the social signal measurements (top right).

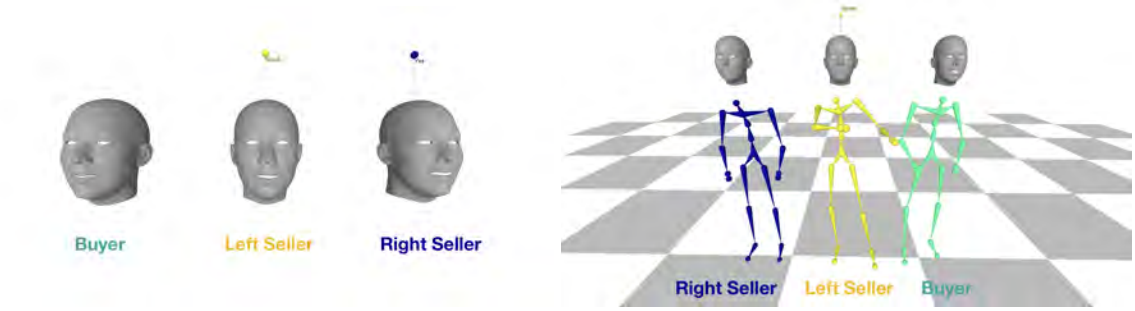


Fig. 6.4 Visualizing part of social signals. (Left) Visualizing only face and speaking annotation where face mesh is reconstructed by the method described in Chapter 5, (Right) Visualizing face mesh, body skeletons, and speaking annotation excluding social formation. As a benefit of our measurement outputs, we can include (or exclude) specific channels of social signals among all measured signals.

Ultimatum. Ultimatum is a bargaining game that was first experimentally studied by Güth et al. [159] and has subsequently become among the most studied games in experimental economics [160]. The game consists of two bargainers who are given a certain amount of money to split ($10 in our experiment). One bargainer, referred to as the proposer, suggests a split of the money, and the other bargainer, referred to as the responder, either accepts the split (and both receive money accordingly) or rejects the split (and neither receive anything). Unlike researchers in experimental economics and game theory, we are interested in evoking interactions rather than predicting outcomes of the game, and we therefore make several adjustments to the usual set up of the game. First, we organized participants into teams of proposers and responders (e.g., two proposers and two responders, or four proposers and one responder). Second, we introduced a one minute, face-to-face discussion phase where the participants discuss what they should do (including both inter- and intra-team discussion). One the discussion phase is over, the proposers suggested a split, which the responders either accept or reject without discussion. Third, we did not control for prior acquaintance. Before each experiment, the subjects were introduced to the game informally, with oral instructions explaining the rules. The proposer(s) entered the eventspace first, followed by the responder(s).

Mafia. Mafia is a game created by Dimitry Davidoff [161] that involves both conflict and cooperation, and produces dynamically changing alliances and rivalries within a group of people. Within the group, two individuals (usually) are secretly assigned the roles of “Mafia" and the rest are assigned roles or ordinary Villagers. The goal of the Villagers is to determine who among them is Mafia via discussion. It is a turn-based game that involves the Villagers choosing to “execute" one player every turn—their best consensus guess at who the mafia

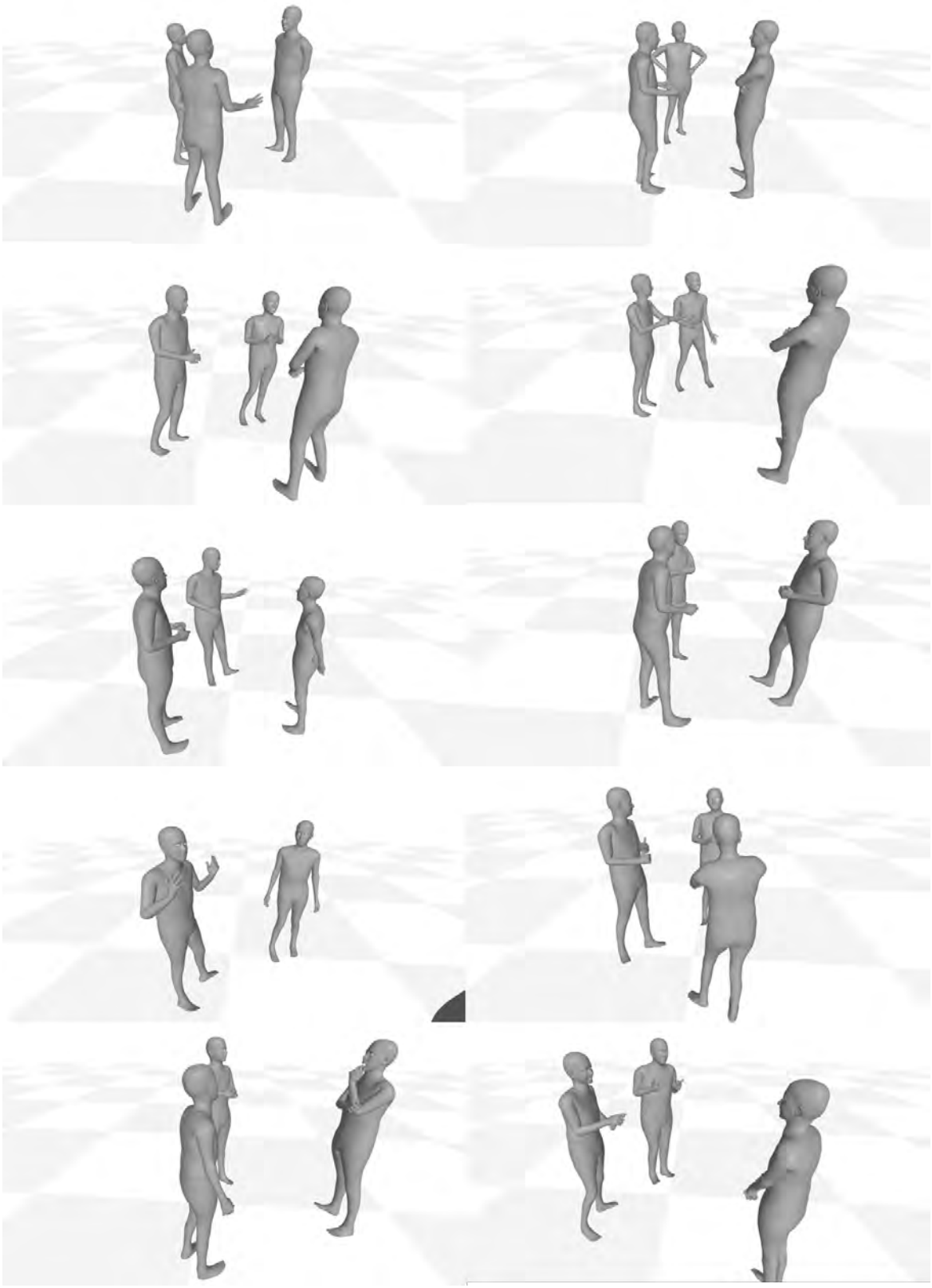


Fig. 6.5 Example scenes of Haggling sequences with Adam model fitting results. Motions from faces, bodies, and hands are captured in a mesh structure.

players are—following by the Mafia secretly choosing to secretly “execute" a Villager of their choice. The game is notable in that it requires some players to engage in outright deception, and requires other players to try to infer this via the interaction alone. In our capture, we involved eight players in the studio. One of them is determined as an operator, and two Mafias and five Villagers are secretly assigned via selecting a lottery. During the game, we gave them approximately a minute to discuss before iterating on each turn. A large number of interesting phenomena were observed, including subtle motion and gestures to suspect or deceive the other group. Participants were compensated $10 for their participation.

Categories	Number of Sequences	Total Duration	Num. of people per scene
Haggling	22	232m 10s	3
Range of Motion	9	117m	1
Ultimatum	6	57m 20s	2-7
Mafia	7	83m	3-8
Dance	14	55m 5s	1-2
Musical Instruments	20	73m	1-3
Pose	33	120m	1
Toddler	12	26m	1-3
Hands	7	34m	1-3
Others	16	20m	3-15
Total	176	13h 37m 35s	1-15

Table 6.2 Statistic of Panoptic Studio Dataset

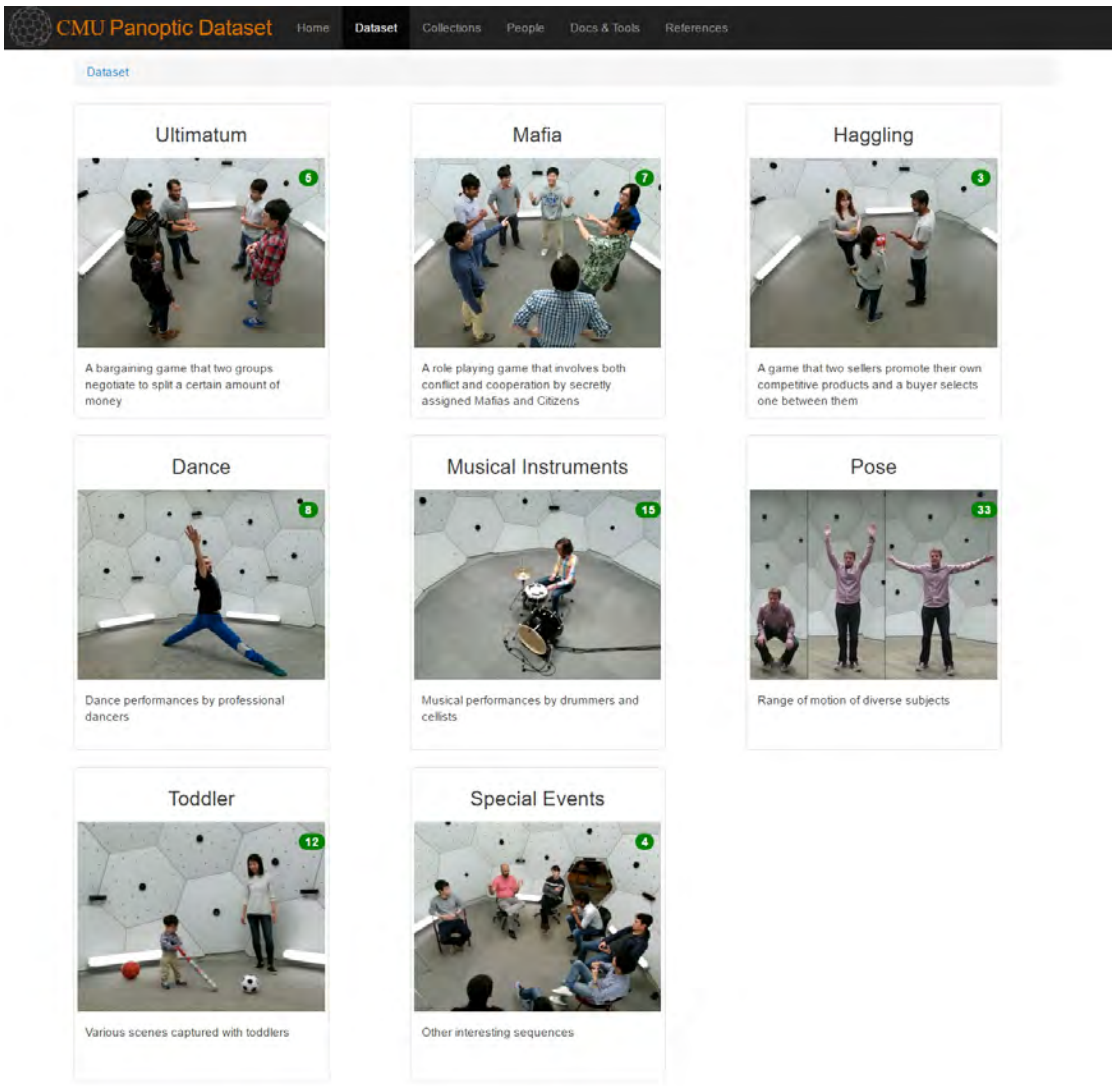


Fig. 6.6 The Panoptic Studio Dataset website. We have collected a large number of sequences in various situations. All the 521 sensor measurements as well as reconstructed 3D motion capture results are available.

Chapter 7

Nonverbal Social Signal Prediction

Consider how humans communicate—we use language, voice, facial expressions, and body gestures to convey our thoughts, emotions, and intentions. Such social signals that encode the internal messages of an individual are then sensed, decoded, and finally interpreted by communication partners (see Figure 7.1). One way to computationally model human communication is by finding the mapping between the signals and the conveyed semantics, as in affective computing field [162–164] to automatically recognize human emotions. However, how to represent the internal status, either by discrete categories [141] or a set of latent dimensions [142–144], is still controversial due to the fact that we have no way to directly observe the internal status of our mind, and, more importantly, mapping the exposed signals and the status is subjective to observers without an objective way to measure it [43].

In our research, we focus on the dynamics between social signals [165] that a subject receives and sends, rather than the mapping between the social signals and their semantics. The advantages of this approach are: (1) we can tackle the problem by investigating the objectively measurable social signal data and (2) it enables us to model subtle details of social communication by considering the original continuous and high-dimensional signal space. By finding the patterns and correlations among social signals, we can computationally model the dynamics of social communications, to ultimately teach a robot how to behave in a social communication. Although similar ideas have been used to model human communication skills in many different fields, including psychology (e.g., study on mimicry) [166], virtual agents [167, 168], human-robot interaction [169], most of them are studied in a limited range of nonverbal signals, often focusing on faces.

In this chapter, we aim to model the dynamics (or spatiotemporal correlations) among social signals in a data-driven way. In particular, we take into account a broad spectrum of social signals between individuals, including facial expressions, body gestures, body proximity, and body orientations. While the correlations among a few types of signals of

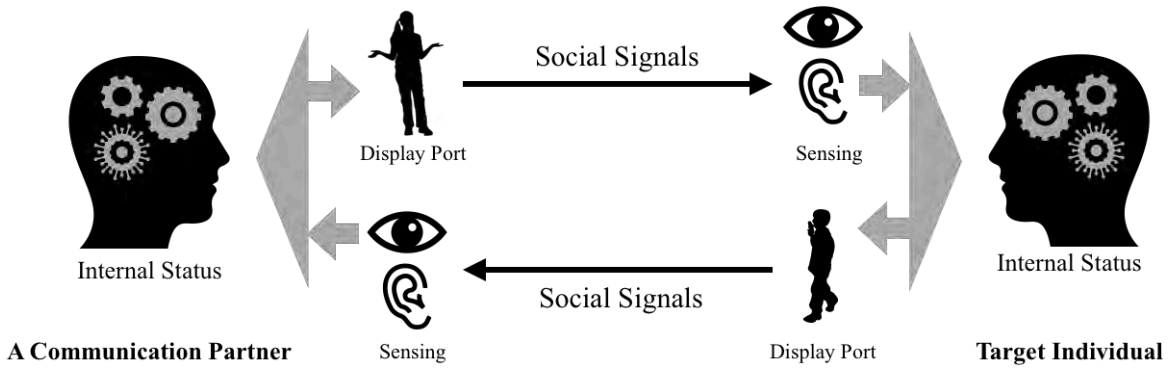


Fig. 7.1 Humans communicate by transmitting signals using several "display ports" such as voice, body motion, and facial expressions. Such signals are then sensed, decoded, and interpreted by others. We hypothesize that there exists a strong link between the signals exchanged during the social interaction, and aim to computational model the dynamics of them.

an individual have been already studied (e.g., between speech and facial expressions [170, 171], speech and gaze [172], speech and body [173, 174]), there is no existing work that computationally studies the dynamics of various channels of social signals (e.g., both body gestures and facial expressions) transmitted among individuals. The main obstacle in pursuing the direction is the lack of available dataset or measurement technique. The sensing and measuring techniques presented in this thesis based on the Panoptic Studio (Chapters 2, 3, 4, 5) allow us to enable this study for the first time, and we leverage the Haggling sequences (described in Chapter 6) captured from hundreds of participants for this direction.

We first formulate a social signal prediction task to model the dynamics of social communication as a function of input and output social signals (see Figure 7.2). In this task, we consider a target individual who receives a set of social signals from others and emits response signals as output. We hypothesize that the social behavior of the target person can be learned by finding the patterns between these signal flows, by which machines know how to behave given the signals humans produce. Importantly, we aim to include as many channels of signals as possible to study the correlation between various channels of social signals. However, directly investigating the dynamics of all spectrum social signals provided by our measurement method is challenging due to the high complexity of the motion space, requiring a much larger scale of data. We thus simplify the problem by focusing on predicting lower dimensional, yet important, output signals—speaking status and social formations—emitted by the target person, while still considering broader channels including body motion and facial expressions as input. We found that this approach still provides an important opportunity to computationally study various channels of interpersonal social signals. In this chapter, we explore various issues in pursuing this research direction by discussing

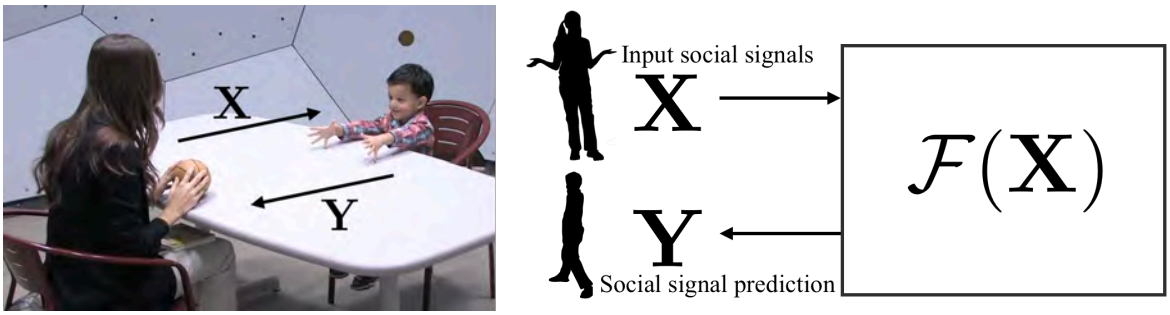


Fig. 7.2 We aim to learn the correlation between the input signals **X** an individual receives and the output social signals **Y** the individual emits. The goal of social signal prediction is to regress a function for these in a data-driven manner.

problem formulation, data representation, and evaluation metric. Results on our baseline social signal prediction models, implemented by neural networks, demonstrate the strong predictive property among social signals and also allow us to quantitatively compare the relative influence among them. To this end, we further discuss the future research direction to ultimately mimic the social behavior of humans.

7.1 Related Work

Psychology: Due to its importance in social communication, nonverbal cues have received significant attention in psychology. Work in this area is often categorized into diverse subfields including Proxemics, Kinesics, Vocalics, and Haptics. In our work, we focus only on Proxemics and Kinesics, which are closely related to visual cues. Hall first introduced the concept of Proxemics to describe how humans use their personal space during communications [50], and Kendon studied the spatial formations and orientations established when multiple people communicate in a common space (named F-formation) [51]. Facial expression, in particular, has received lots of attention by researchers since the pioneering work of Charles Darwin [175]. Ekman studied the relation between emotions and facial expressions, and presented the Facial Action Code System (FACS) by introducing a system to describe facial expressions using combinations of atomic units (Action Units) [42]. Since then, this system remains as the de-facto standard to annotate and measure facial expressions, and has had a broad impact across many fields. Compared to the face, body gestures remain relatively unexplored even though research has substantiated the importance of body language in communication [176, 1, 81, 82]. In spite of the efforts of many researchers in diverse fields, little progress has been made in understanding nonverbal communications,

and the approaches proposed several decades ago are still the most widely used methods available [1].

Social Signal Processing: There has been increasing interest in studying nonverbal communication using computational methods [177, 178]. Analyzing facial expression has been a core focus in the vision community [179, 41, 180]. Many other methods to automatically detect social cues from photos and videos have also been proposed, including F-formation detection [181], recognizing proxemics from photos [182], detecting attention [183], recognizing emotions by body pose [184], and detecting social saliency [185]. The affective computing field has been growing rapidly, where computer vision and other sensor measurements are used with machine learning techniques to understand human emotion, social behavior, and roles [162–164].

Forecasting human motion:

Predicting or forecasting human motion is an emerging area in computer vision and machine learning. Researchers propose approaches for predicting a pedestrian's future trajectory [186] or forecasting human interaction in dyadic situations [187]. More recently, deep neural network is used to predict future 3D poses from motion capture data [188–190], but they focus on periodic motions such as walk cycles. Recent work attempts to forecast human body motion in the 2D image domain [191, 192]. A few efforts address trajectory prediction in social situations [193–195].

Measuring Nonverbal Signals in Imagery: Detecting human bodies and keypoints in images has advanced greatly in computer vision. There exist publicly available 2D face keypoint detectors [196], body pose detectors [104, 80, 105], and hand pose detectors [46]. 3D motion can be obtained by markerless motion capture in a multiview setup [21, 30, 23, 53, 48], by RGB-D cameras [93, 27], or even by monocular cameras [197, 119, 198–201]. Recently, methods to capture both body and hands have also been introduced [137, 48].

7.2 Social Signal Prediction

The objective of *Social Signal Prediction* is to predict the behavioral cues of a target person in a social situation by using the cues from communication partners as input. We hypothesize that target person's behavior is correlated to the behavioral cues of other individuals. For example, the location and orientation of the target person should be strongly affected by the position of conversational partners (known as Proxemics [50] and F-formation [51]), and the gaze direction, body gestures, and facial expressions of the target person should also be "conditioned" by the behaviors of the conversational partners. In the social signal prediction task, we model this conditional distribution among interacting subjects, to ultimately teach a

robot how to behave in a similar social situation driven by the behavior of communication partners. There exist cases where the correlation of the social signals among subjects is strong, such as hand-shaking or greeting (waving hands or bowing). But in most of the cases, the correlation is implicit—there exist no specific rules on how to behave given other people's behavior, which makes it hard to manually define the rules. In our approach, we tackle this problem in a data-driven manner, by automatically learning the conditional distributions using a large scale multimodal social signals corpus.

We first conceptually formulate the social signal prediction problem here, and a specific implementation focusing on the Haggling scenario is described in the next section. Let us denote "all types of signals" that the target person received in a social situation at time t as $\mathbf{X}(t)$. Thus $\mathbf{X}(t)$ includes the social signals from other individuals—body gestures, facial expression, body position, voice tones, verbal languages—and also other contextual factors such as the space where the conversation is performed or other visible objects which may affect the behavior of the target person (e.g., some sounds or objects may attract the attention of the person). We divide the input signal $\mathbf{X}(t)$ into two parts, the signals from the conversational partners, $\mathbf{X}_c(t)$, and signals from other sources (e.g., objects, environment, and other human subjects not interacting with the target person), $\mathbf{X}_e(t)$[1]. Thus,

$$\mathbf{X}(t) = \{\mathbf{X}_c(t), \mathbf{X}_e(t)\}. \tag{7.1}$$

The $\mathbf{X}_c(t)$ may contain the social signals from multiple people and we denote the signals from each subject separately:

$$\mathbf{X}_c(t) = \{\mathbf{X}_c^i(t)\}_{i=1}^N, \tag{7.2}$$

where $\mathbf{X}_c^i(t)$ are the signals from the i-th conversational partner in the social interaction and N is the total number of partners. We also denote the signals emitted by the target person at time t in the social situation as $\mathbf{Y}(t)$. Then, the goal of social signal prediction is to find a function $\mathscr{F}$ which takes $\mathbf{X}$ as input and produces $\mathbf{Y}$ as output to mimic the behavior of the target person in the social situation:

$$\mathbf{Y}(t+1) = \mathscr{F}\left(\mathbf{X}(t_0 : t), \mathbf{Y}(t_0 : t)\right), \tag{7.3}$$

where $t_0 : t$ represents a range of time from t_0 to t affecting the current behavior of the target person. Note that we define the function $\mathscr{F}$ to take the history of the target person's own signals $\mathbf{Y}(t_0 : t)$, and the function predicts the immediate future motion (or response) of the target individual. Intuitively, this formulation models the human behavior as a function

[1] Note that the $\mathbf{X}_e(t)$ can be ignored in our Panoptic Studio setup where the other factors (e.g., furniture) are not presented.

that represents the dynamics among social signals the target person receiving and emitting. The function can be defined for a specific individual, representing the personal behavior of the target person encoding characteristic, gender, and culture of the target. Based on that, different individuals may behave differently. If the function is regressed by the data from many people, then we hypothesize that the function produces more general and common social behaviors, where the individual specific behaviors are averaged out.

Previous approaches can be considered as subsets of this model. For example, conversational agents (or chatbots) using natural language only can be represented as:

$$\mathbf{Y}_v(t+1) = \mathscr{F}\left(\mathbf{X}_v(t_0:t)\right), \tag{7.4}$$

where $\mathbf{Y}_v$ and $\mathbf{X}_v$ represents only verbal signals. The human motion forecasting studied in computer vision and graphics [189, 190] can be considered as:

$$\mathbf{Y}_n(t+1) = \mathscr{F}\left(\mathbf{Y}_n(t_0:t)\right), \tag{7.5}$$

where $\mathbf{Y}_n$ represents nonverbal body motion. Note that in this task, there is no social interaction modeling, and the prediction is only for an individual using individual's own previous signals.

In our work, we focus on nonverbal social signal prediction in a triadic social interaction in the Haggling game scenario:

$$\mathbf{Y}(t_0:t) = \mathscr{F}\left(\mathbf{X}_c^1(t_0:t), \mathbf{X}_c^2(t_0:t)\right), \tag{7.6}$$

where we predict the social signals of the target person given the signals of the two other people during the same window of time. In particular, we consider diverse input and output social signals to investigate their dynamics and correlations. The details of our approach are described in the following section.

7.3 Social Signal Prediction in Haggling Scenario

We use our Haggling scenario as an example problem of social signal prediction to computationally model triadic interaction. In this section, we specifically define the input and output signals used in our modeling, and then present two social signal predicting problems, predicting speaking status and predicting social formation, by describing their problem definitions and implementation details. Note that we focus on estimating the target person's

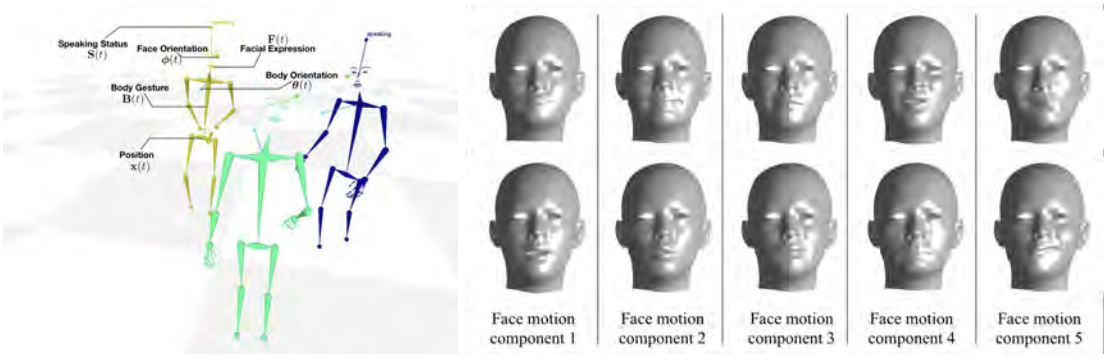


Fig. 7.3 Social signal measurements used for social signal prediction. (Left) Reconstructed 3D social signals showing the body, face, global position, face orientation, body orientation, and speaking status. (Right) The five face motion components (showing -0.3 on the top and 0.3 on the bottom) used in our social signal modeling.

concurrent signals by taking other individuals' signals as input as defined in Equation 7.6 to simplify the problem, rather than forecasting the future signals.

7.3.1 Notation

Our measurement method in the Panoptic Studio reconstructs 3D body motion $\mathbf{B}(t)$ and 3D face motion $\mathbf{F}(t)$ for each individual at each time t[2]. We also denote the global position of the body as $\mathbf{X}(t)$. From these measurements, we additionally compute the body orientation $\boldsymbol{\theta}(t)$ and face orientation $\boldsymbol{\phi}(t)$ by finding the 3D normal direction of torso and face, respectively. We describe the details below, and the left of Figure 7.3 shows an example visualization.

Body Motion: We follow the body motion representation of the work of Holden et al. [202], representing a body gesture at a frame as a 73-dimensional vector, $\mathbf{B}(t) \in \mathbb{R}^{73}$. This representation is based on the skeletal structure of CMU Mocap dataset [157] with 21 joints (63 dimensions), along with the projection of the root joint (the center of the hip joints) on the floor plane (3 dimensions), the relative body locations and orientations represented by the velocity values of the root (3 dimensions), and footstep signals (4 dimensions). The orientations are computed only on the x-z plane with respect to the y-axis, and the location and orientation represent the changes from the previous frame rather than the absolute values, following the previous work [190, 202]. In particular, the first 63 dimensions of $\mathbf{B}(t)$ represents the body motion in the person-centric coordinate, where the root joint is at the

[2]We do not use the hand motion measurement due to the occasional failures in challenging hand motions (e.g., when both hands are close to each other), making it hard to train our model. We, however, believe this cue plays an important role in social interaction, which needs to be considered in future direction.

origin and torso is facing the z direction. We perform a retargeting process to convert our original 3D motion data from the Panoptic Studio, where the skeleton definition is the same as COCO dataset [203] in the global coordinate, to this body motion representation with a fixed body scale. Thus, in our final motion representation, individual specific cues such as heights or lengths of limbs are removed and only motion cues are kept.

Face Motion: For the face motion signal, we use the initial 5 dimensions of the facial expression parameters of Adam model (described in Chapter 5), because we found the remaining dimensions have an almost negligible impact on our reconstruction quality. Note that the face expression parameters in our Adam model (originally from the work of [103]) are sorted by their influence by construction and the initial components have more impact in expressing facial motion. To this end, face motion at a time instance is represented by a 5-dimensional vector, $\mathbf{F}(t) \in \mathbb{R}^5$, as shown in the right of Figure 7.3. Here, we also do not include individual-specific information (the face shape parameters varying for individuals) and only motion cues are kept.

Position and Orientation: For the global position $\mathbf{x}(t)$ of each individual, we use the coordinate of the root joint of the body, ignoring the values in y axis, and thus $\mathbf{x}(t) \in \mathbb{R}^2$. We use a 2D unit vector to represent body orientations $\boldsymbol{\theta}(t) \in \mathbb{R}^2$ and face orientation $\boldsymbol{\phi}(t) \in \mathbb{R}^2$, defined on the x-z plane ignoring the values in y axis. Note that we use unit vectors rather than angle representation, because the angle representation has a discontinuity issue when wrapping around 2ϕ and -2ϕ. In contrast to the relative location and orientation represented in the part of body motion $\mathbf{B}(t)$, these $\mathbf{x}(t)$, $\boldsymbol{\theta}(t)$, and $\boldsymbol{\phi}(t)$ represent the values in the global coordinate, which are used to model social formation. In summary, the status of an individual at a frame in social formation prediction is represented by a 6-dimensional vector, $[\mathbf{x}(t)^\top, \boldsymbol{\theta}(t)^\top, \boldsymbol{\phi}(t)^\top]^\top \in \mathbb{R}^6$.

Speaking Status: The voice data $\mathbf{V}(t)$ of each individual is also recorded by wireless microphones assigned to each individual. From the audio signal, we manually annotate a binary speaking label $\mathbf{S}(t) \in \{0, 1\}$ describing whether the target subject is speaking (labelled as 1) or not speaking (labelled as 0) at time t.

By leveraging these various behavioral cues measured in the Haggling scenes, we model the dynamics of these signals in a triadic interaction. The objective of our direction is to regress the function defined in Equation 7.6. To further constrain the problem we assume that the target person is the seller positioned on the left side of the buyer, and as input we

use the social signals of the buyer ($\mathbf{X}^1$) and the other seller ($\mathbf{X}^2$). Based on our social signal measurements, the input and output of the function are represented as,

$$\begin{aligned}\mathbf{Y} &= [\mathbf{x}^0, \boldsymbol{\theta}^0, \boldsymbol{\phi}^0, \mathbf{B}^0, \mathbf{F}^0, \mathbf{S}^0],\\ \mathbf{X}^1 &= [\mathbf{x}^1, \boldsymbol{\theta}^1, \boldsymbol{\phi}^1, \mathbf{B}^1, \mathbf{F}^1, \mathbf{S}^1],\\ \mathbf{X}^2 &= [\mathbf{x}^2, \boldsymbol{\theta}^2, \boldsymbol{\phi}^2, \mathbf{B}^2, \mathbf{F}^2, \mathbf{S}^2],\end{aligned} \tag{7.7}$$

where we use the superscript 0 to denote the social signals of the target subject (the output of social signal prediction).

7.3.2 Predicting Speaking

We predict whether the target subject is currently speaking or not, denoted by $\mathbf{S}^0$. This is a binary classification task and can be reliably trained by Cross Entropy loss function. We first study the correlation between the speaking signal of the target person, $\mathbf{S}^0$, and the person's own social signals, either body motion $\mathbf{B}^0$ or facial motion $\mathbf{F}^0$, or both. We expect this correlation is stronger than the link across individuals. Formally, a function $\mathscr{F}_{B0\rightarrow S0}$ takes the target person's own body motion $\mathbf{B}^0(t_0 : t)$ to predict the speaking signal:

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{B0\rightarrow S0}\left(\mathbf{B}^0(t_0 : t)\right), \tag{7.8}$$

and similarly,

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{F0\rightarrow S0}\left(\mathbf{F}^0(t_0 : t)\right), \tag{7.9}$$

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{(F0,B0)\rightarrow S0}\left(\mathbf{F}^0(t_0 : t), \mathbf{B}^0(t_0 : t)\right), \tag{7.10}$$

where $\mathscr{F}_{F0\rightarrow S0}$ takes the target person's own face motion, and $\mathscr{F}_{(F0,B0)\rightarrow S0}$ takes both face and body cues.

We compare the performance of these functions with the functions that takes the signals from a communication partner, the other seller:

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{B2\rightarrow S0}\left(\mathbf{B}^2(t_0 : t)\right), \tag{7.11}$$

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{F2\rightarrow S0}\left(\mathbf{F}^2(t_0 : t)\right), \tag{7.12}$$

$$\mathbf{S}^0(t_0 : t) = \mathscr{F}_{(F2,B2)\rightarrow S0}\left(\mathbf{F}^2(t_0 : t), \mathbf{B}^2(t_0 : t)\right), \tag{7.13}$$

where the functions use body cues, face cues, and both cues, respectively.

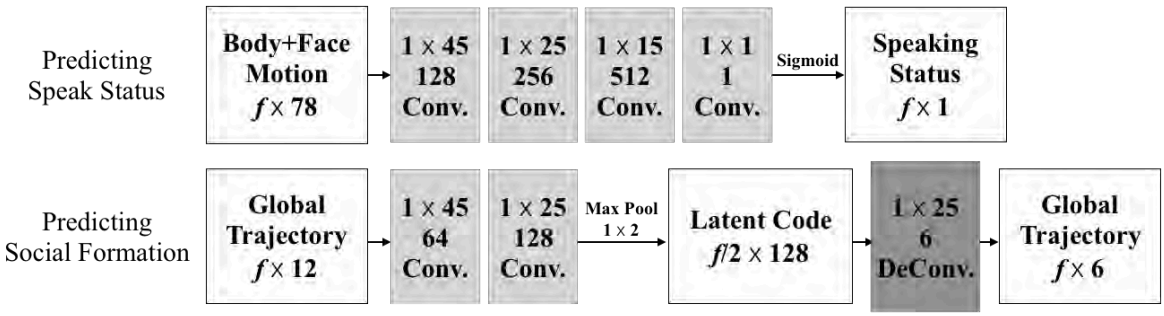


Fig. 7.4 Network Architectures. We use fully convolutional networks for both predicting speaking status and social formation problems. The models can be applied to the input of arbitrary length, but we use the input data of a fixed size $f = 120$ for the efficiency in training.

This framework enables us to quantitatively investigate the link among social signals across individuals. For example, we may easily hypothesize that there exists a strong correlation between the signals from the same individual (e.g., speaking and facial motion of the target person), while the correlation between the signals across different individuals (e.g., speaking of the target person and body motion of another person) may be considered as weak. By comparing their performances, we verify there still exists strong links among these signals exchanged across subjects.

Implementation Details. Our neural network is composed of four 1-D convolutional layers, as shown in the top of the Figure 7.4. The first three layers output 128, 256, and 512 dimensional features respectively with ReLU activation functions, and the last layer has 1×1 convolutions with a sigmoid activation layer. Dropout [204] is also applied for the second and third layers with a retention probability of 0.25. Our model does not require a fixed window size for the input (since it is fully convolutional), but we separate input data into small clips with a fixed size (denoted by f) for the efficiency in training. During testing time, our models can be applied to the input of arbitrary length. We use $f = 120$ (4 seconds) for the input window size for training, and we use an arbitrary length of input data for testing. The feature dimension of the input of our network is the concatenation of face motion and body motion (78 dimensions). If fewer cues are used (e.g., face only or body only), we mask out the unused channels as their average values computed in training set by keeping the same network structure. We use an adaptive gradient descent algorithm, AmsGrad algorithm [205], implemented in PyTorch [206], along with the l^1 regularization loss with the weight of 0.001.

7.3.3 Predicting Social Formations

We predict the location and orientations of the target person, denoted by $\mathbf{Y}_p = [\mathbf{x}^{0\top}, \boldsymbol{\theta}^{0\top}, \boldsymbol{\phi}^{0\top}]^\top$, given the same channels of cues from the communication partners. This problem is strongly related to Proxemics [50] and F-formation [51], illustrating how humans use their space in social communications. Formally,

$$\mathbf{Y}_p(t_0 : t) = \mathscr{F}_p\left(\mathbf{X}_p^1(t_0 : t), \mathbf{X}_p^2(t_0 : t)\right), \tag{7.14}$$

where $\mathbf{Y}_p$, $\mathbf{X}_p^1$, and $\mathbf{X}_p^2$ contain global location and orientation signals $[\mathbf{x}^{i\top}, \boldsymbol{\theta}^{i\top}, \boldsymbol{\phi}^{i\top}]^\top$ (where $i = 0, 1$, or 2) for the target subject and others. Note that we only consider the positions and orientations on the ground plane (in 2D), ignoring the height of the subjects, and thus $\mathbf{Y}_p(t), \mathbf{X}_p^i(t) \in \mathbb{R}^6$. This prediction problem is intended to see whether the machine can learn how to build a social formation to interact with humans [207].

Implementation Details. Our neural network has an autoencoder structure, where the encoder is composed of two 1-D convolutional layers followed by a max pooling layer with stride 2, and the decoder is composed of a single 1-D transposed convolution layer. The network is shown in the bottom of the Figure 7.4. The output feature dimensions are 64, 128, and 6 respectively. Dropout [204] is also applied in front of all layers with a retention probability of 0.25. Similar to the speaking status prediction, our model does not require a fixed window size for the input, but we separate input data into small clips with a fixed size ($f = 120$, or 4 seconds) for the efficiency in training. The input is the concatenation of the cues of other two communication partners (12 dimensions) with a fixed order (buyer and then the right seller), and the output of our network is the position and orientations of the target individual, the left seller (6 dimensions). Similar to the previous prediction task, if fewer cues are used (e.g., position only), we mask out the unused channels as their average values computed in training set by keeping the same network structure. We use an adaptive gradient descent algorithm, AmsGrad algorithm [205], implemented in PyTorch [206], along with the l^1 regularization loss with the weight of 0.1.

7.4 Results

In this section, we show experimental results for two prediction tasks, predicting speaking status and social formations, from different input sources. The core direction in performing this experiments is to explore the correlations of diverse behavioral channels measured in

genuine social communications. We leverage the availability of a broad spectrum of social signal measurements of the Haggling dataset (described in Chapter 6).

7.4.1 Pre-processing Haggling Data

Given the measurement data of the Haggling games, we first manually annotate the start and end time of the game, where the start time is decided when the social formation is built and the end time is defined when the social formation is broken. We crop out the motion dataset based on this start and end time, so that we ignore the time while subjects enter and exit the capture space. For each haggling game scene, we also annotate the players' roles in the game, buyer, left-seller, and right-seller, where the left and right are determined in the buyer's viewpoint. In our experiment, we specify that the left seller is our target person and predict the social behavior of these subjects. As described in our method section, we re-target the motion data to a standardized skeleton size to remove size variation from the body skeletons similar to [202]. We also synthesize footstep signals and decouple the body motion from global translation and orientation using the method of [202]. For face motion, we fit the face part of our Adam model (described in Chapter 5) on the 3D keypoints of individual's face, and use the first five facial expression parameters, as described in Section 7.3.1. Finally, we divide the dataset into 140 training sets and 40 test sets. However, since there exist sequences where the reconstruction errors are severe for some frames, we select only 79 training sets and 28 testing sets which are manually verified to be error free. We additionally divide all training set into slices with 120 frames to train our models. We standardize all input data so that they have zero mean and unit variance.

7.4.2 Speaking Status Prediction

We predict whether the target person is currently speaking or not by observing other channels of social signals in the scene.

A result on intra-personal signals. First, we investigate the performance when the target individual's own social signals are used as input. Three different input sources—facial expressions, body gestures, and both of them—are used to train neural network models respectively. In particular, we use the same neural network architecture for this experiment by keeping the input dimension and network size as the same to make the comparison as fair as possible. As described in the Section 7.3.2, to train each network we mask out the unused channels in the input as their average values computed in the training set. The prediction accuracies from these input signals are shown in the first column of Table 7.1.

Input Signal Types/Sources	Self signal	Other seller's signal	Random person's signal
Face+Body	89.13%	77.97%	49.65%
Face	**89.16**%	**80.21**%	49.64%
Body	76.69%	71.29%	50.22%

Table 7.1 Speaking status classification accuracy using different social signal sources as input

Input Signal Types/Sources	Self signal	Other seller's signal
Face+Body (original)	89.13%	77.97%
After Removing Body	85.11%	75.26 %
After Removing Face	54.33%	67.64 %

Table 7.2 An ablation study after removing certain channels in the input for the networks trained with Face+Body input. The first row is the output of the original network (the same as in Table 7.1), and next rows are testing performance after masking out body or face parts in the input data without any retraining.

As demonstrated in our result, the social signals from the target individual show strong correlations with the speaking. For example, the facial cue of the target person shows the strongest correlation (about 90% accuracy) with the target person's own speaking status, presumably due to the strong correlation between the lip motion and speaking. The body motion also shows a strong correlation with more than 76% prediction accuracy. The result with both body and face signals, shown in the first column of Table 7.1, is similar to the case that only face cue is used, and by applying an ablation study we found that this is because the network dominantly uses the face cues over the body cues for the prediction, as shown in the first column of Table 7.2. More specifically about this ablation study, given the trained model which takes both face and body as input, we mask out the face part or body part in the input data during the "testing" time and compute their performances. As shown in the Table 7.2, the accuracy after removing the body part is similar to the original performance, meaning that the trained network is less dependent to the body cues, while there exists much larger drop if the face part is removed.

A result on inter-personal signals. A more interesting experiment is investigating the performance by using the other seller's social signals as input to predict the target person's speaking status. Similarly, three different input sources are considered, and the results are shown in the second column of the Table 7.1. The result clearly shows that there exists a strong link between interpersonal social signals. The other seller's facial motion shows a strong predictive power for the target person's speaking status, where the accuracy is higher than the case of using the target person's own body signals as input, presumably due to the

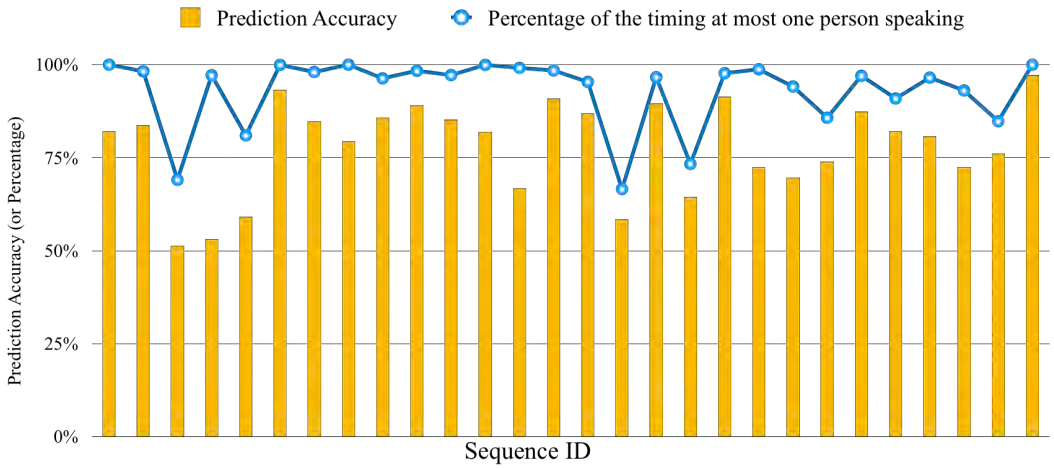


Fig. 7.5 Comparison between the performance of speaking status prediction and turn-taking status for each sequence. Each column shows the prediction performance (yellow bar) where the other seller's face and body signals are used as input. The blue curve represents how well the turn-taking rule is satisfied, which is defined by counting the percentage of the timing where at most one person is speaking.

turn-taking property in social communication. For example, we can assume that the target person is not speaking, when the other seller is speaking. We can further investigate this by checking how well the turn-taking rule is satisfied during each social game scene, along with its predicting performance. As a way to measure the turn-taking status, we consider the percentage of the timing at which at most one person speaks, which defined by:

$$\frac{\sum_t \delta\left(\mathbf{S}^0(t)+\mathbf{S}^1(t)<2\right)}{T}, \tag{7.15}$$

where T is the total time of a Haggling game, $\mathbf{S}$ is the speaking status for sellers, and δ is a function that returns 1 if the condition satisfies and returns 0 otherwise. In this measurement, 100% means that there is no time that both sellers are speaking at the same time, where the turn-taking rules are perfectly satisfied. We compute this measurement to check the turn-taking status for each testing sequence as the blue curve in Figure 7.5. In this figure, we also plot the speaking prediction accuracy for each testing sequence by using the other seller's both face and body signals as input, which is shown as yellow bars. As shown in the figure, the prediction performance shows a very similar pattern to this turn-taking status, and this means that this implicit social "rule" is a source of linking the social signals across individuals. Example qualitative results are shown in the Figure 7.7.

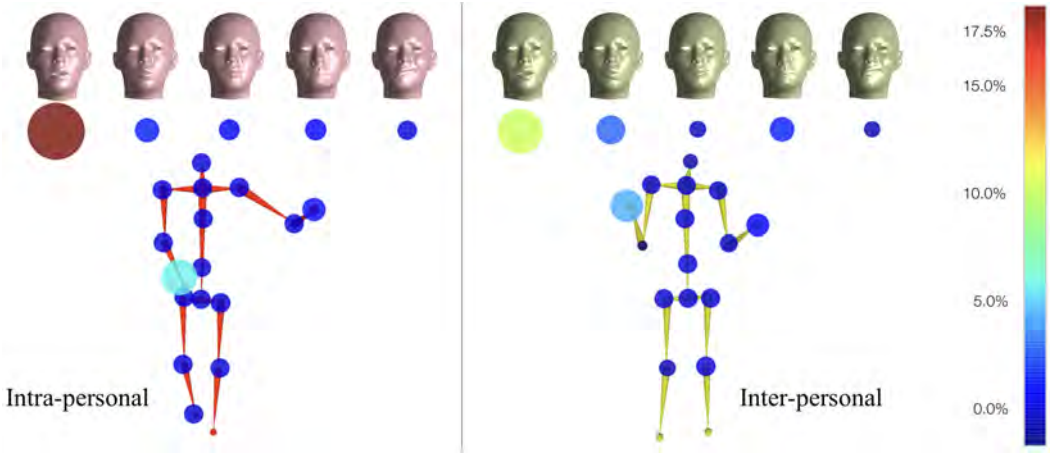


Fig. 7.6 The result of an ablation study by comparing the prediction performances after removing each channel of the social signal input using the trained networks. We use the networks trained by using both body and face cues as input (the first row of the Table 7.1), and the performance drops after removing each part, compared to the original performances, are shown by colors and circle sizes. The left figure is the result by using the target person's own signals, and the right figure is by using the other seller's signals. The colorbar on the right shows the frame drops in percentage from the original performances.

A result on random signals. As a comparison, we also perform an experiment by using social signals from a random individual, where the individual is randomly selected in the testing set without any social link to our target individual. As shown in the third column of the Table 7.2, the classification performance using a random person's motion as input shows about the chance level (50%) with no predictive property.

An ablation study to verify the influence of each part. As another test, we perform an ablation study by comparing the prediction performance after removing every single channel in the trained network. For this test, we use the network trained with both body and face cues (the first row of the Table 7.1). We mask out a certain channel (e.g., a face motion component or a body joint part) in the input during testing time and check the performance drop from the original output. The result is shown in Figure 7.6, where the colors and the sizes of the circles represent the amount of performance decrease. This result shows that the first component of the face motion, which is corresponding to the mouth opening motion as shown in Figure 7.3, has the strongest predictive power for speaking status. As another interesting result, the result shows that the right hand has stronger predictive power than the left hand.

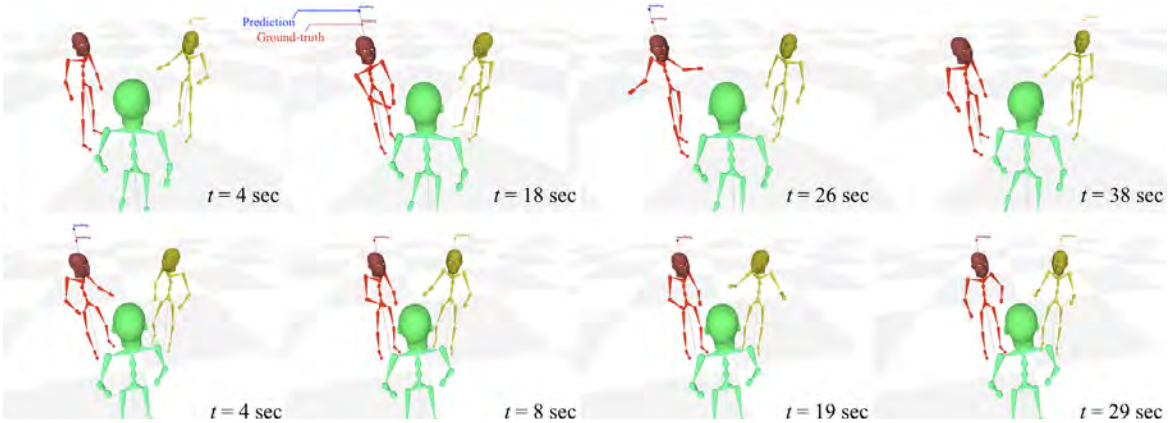


Fig. 7.7 Qualitative results of the speaking prediction of the target person (red) by using the other seller's (yellow) face and body motions as input. The speaking prediction output is shown as the blue "speaking" label above the target person's head, while the ground truth speaking status is shown as the red label. The prediction is accurate, if both blue and red labels are shown, or not shown. Examples scenes of two haggling games (top and bottom) are shown, where the sequence on the top has high accuracy (89%) and the sequence on the bottom has low accuracy (58%). In the haggling game shown on the top, both sellers follow the turn-taking rule almost always, while in the sequence shown on the bottom both sellers frequently speak at the same time.

7.4.3 Social Formation Prediction (Position and Orientation)

We predict the position and orientation of the target person, the "left seller", by using the signals of communication partners. In this test, we explore the prediction accuracy by considering combinations of difference sources: using body position, body orientation, and face orientations. Table 7.3 shows the results. By using all signals, we obtain the best performance. Intuitively, we can imagine that the target person's location can be estimated by triangulating the face normal direction of the other two subjects, which presumably learned from our network. The prediction performance using only position cues shows the worst, but still a reasonable, performance among them. Example qualitative results are shown in Figure 7.8.

We also introduce a baseline. The baseline method (denoted as "Mirroring" in Table 7.3) predicts the location of the target seller to mirror that of the other seller w.r.t the buyer's body orientation, and estimate body orientation as the average between the two input subjects. The face orientation is chosen to always face the buyer. This baseline has large errors, with poor prediction results when the buyer is directly facing the other sellers, making the target person be too close to the other seller.

Types	Position	Body Orientation	Face Orientation
PosOnly	29.83 (13.38)	0.26 (0.12)	0.33 (0.13)
Pos+face	25.23 (9.74)	0.23 (0.09)	0.30 (0.12)
Pos+body	26.57 (10.24)	0.22 (0.08)	0.30 (0.10)
Pose+face+body	**24.59** (10.23)	**0.21** (0.06)	**0.29** (0.09)
Mirroring (baseline)	50.03 (20.84)	0.40 (0.17)	0.52 (0.14)

Table 7.3 Social Formation Prediction Errors (cm). Average position error between our estimation and ground-truth are reported in centimeters. The body orientation and face orientation are computed between the distance of estimated facial/body normal direction and GTs.

Fig. 7.8 Qualitative results of the social formation prediction of a haggling game, visualized from the view at the top. The target person is shown as red spheres. The cues from other people (yellow and cyan spheres) are used as input for the prediction, and the prediction output is shown as the blue cube. The red lines represent body orientations, and the green lines represent face orientations.

7.4.4 Revisiting Proxemics

Our dataset has the measurement of fully spontaneous motions (including the position and orientation of groups) of interacting people, and enables us to revisit the well-known proxemics theory [50]. We first compute the average distance between a pair of subjects: (1) buyer and right sellers (B-RS), (2) buyer and left seller (B-LS), and (3) left seller and right seller (LS-RS). The results are shown in Table 7.4. We found that the result approximately follows the social distance categories defined in the Hall's categorization [50]. The distances among sellers are within the close phase of social distance ranges (from 120 *cm* to 210 *cm*) and the average distance among sellers and buyers are within the far phase of social distance (from 210 *cm* to 370 *cm*) in [50]. To analyze the shape of the social formation, we plot the average formation of games in a person-centric coordinate by a buyer. The results are shown in the Figure 7.9, showing that the formation is often similar to isosceles triangles with relatively far distances between a buyer and two sellers than the distance between sellers.

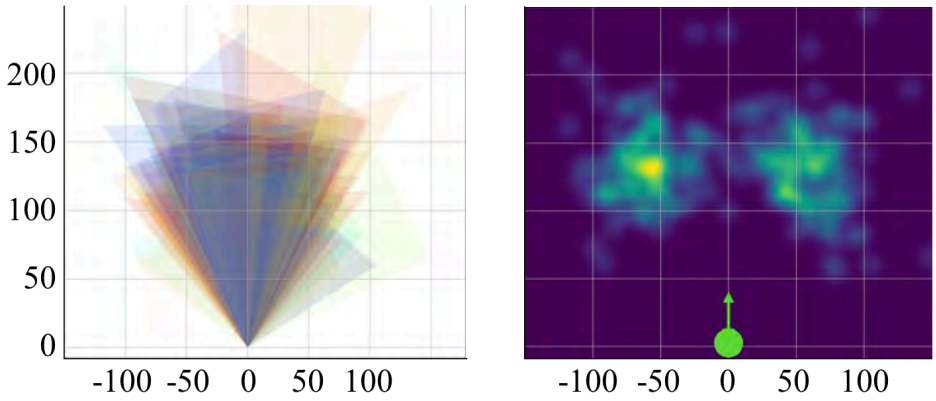


Fig. 7.9 Visualizing social formations in the haggling sequences as triangles (left) and a heat map (right). The formation is normalized w.r.t the buyer's location, and the green circle on the right shows the buyer location (origin) and orientation (z-axis).

	Avg. dist.	Std.	Min	Max
B-RS	148.11	27.26	99.03	265.52
B-LS	151.45	29.62	104.24	284.85
LS-RS	124.13	24.05	77.70	206.26

Table 7.4 Average distances (cm) between subjects. B, RS, and LS denote buyer, right seller, and left seller respectively.

7.4.5 Verifying The Bias of Buyer's Body Orientation Toward Winner

As output of haggling games, the decisions of buyers are known, making winners and losers between two sellers. Equipped with the body location and orientation measurements of all individuals, we can quantitatively verify whether there exists any bias in the body orientations of buyers toward winners. To check this, we first normalize the orientation cues of the buyers with respect to both sellers as shown on the right of the Figure 7.10, where the orientation of a buyer with respect to two sellers is represented by a value around 0.0 and 1.0 at a time. In this representation, 0.0 means that the buyer's body is fully facing the winner, and 1.0 means that buyer's body orientation is fully facing the loser. We plot a histogram of the buyers' body orientations in this normalized orientation space from entire haggling sequences, which is shown on the left of the Figure 7.10. This result shows that the most frequent orientation direction (around 0.41) is less than 0.5 (facing the exact center between two sellers), meaning that buyers' body orientations are slightly biased toward the winners' location. This is an interesting discovery, enabled by the measurement our system produces.

Fig. 7.10 We can quantitatively verify whether there exists any bias in the body orientations of buyers toward winners. A histogram of the buyers' body orientation in a normalized orientation space is shown on the left by using entire haggling sequences. In this normalized orientation space, 0.0 means that the buyer's body is fully facing the winner, and 1.0 means that buyer's body orientation is fully facing the loser (as shown on the right figure).

7.5 Discussion

We present a data-driven social signal prediction framework, which allows us to investigate the dynamics and correlations among interpersonal social signals. We formalize the social signal prediction framework, and describe subtasks with various channels of input and output social signals. To build the models, we leverage our Haggling dataset collected from hundreds of participants by using the sensing and measurement techniques presented in this thesis. In particular, the various channels of nonverbal social signals in the Haggling dataset provide a crucial opportunity to computationally study their dynamics. We demonstrate a clear evidence that the social signals emerging in genuine interactions are predictive each other, by studying two prediction problems, predicting speaking and predicting social formations.

We believe the approach described in this chapter is an important direction to endow machines with the nonverbal communication ability. There are still unexplored issues, and two important questions are discussed in this section.

7.5.1 Predicting More Complicated Signals

We may tackle predicting more complicated social signals using the social signal prediction framework presented in this chapter. For example, we predict body gestures of the target individual by using communication partners' body signals as input:

$$\mathbf{B}^0(t_0:t) = \mathscr{F}_{(B1,B2)\rightarrow B0}\left(\mathbf{B}^1(t_0:t), \mathbf{B}^2(t_0:t)\right). \tag{7.16}$$

We have tested this approach by implementing a neural network. For the implementation, we follow the work of [202] to first learn a human motion manifold space, and then build another neural network to find the mapping between the input signals (body motions from other individuals) and the output body motion (the target individual's body motion). We found that this approach produces a certain level of human-like body motions with dynamic movements on legs and upper bodies, but it was not sufficient to verify a strong correlation across interacting individuals. Two core reasons can be considered about the limited performance: (1) the metric used for training models as loss function and also for the evaluation metric, the Euclidean (or L2) distance, may not be ideal to effectively extract the subtle correlations among high dimensional social signals, and (2) the quantity of our dataset is not sufficient to model the correlation among high dimensional signal spaces. We further discuss these issues in the following subsections.

Types	Avg. Joint Errors (cm)	Std.
Mean Pose	**7.83**	2.33
Prediction by our method	8.72	2.00

Table 7.5 Social Body Gesture Prediction Errors (cm). The error is computed by averaging L2 distance of all joints using the Ground Truth body motion.

7.5.2 Evaluating Social Signal Prediction

The output of social signal prediction should satisfy the following two requirements: (1) the predicted signals should be within a feasible human motion space showing realistic human motions, and (2) the predicted signals should follow the social rules, responding to the behaviors of communication partners. However, it is challenging to evaluate these requirements, because there is no objective metric to quantify "realistic" or "social" properties in behaviors. Notably, the L2 distance between the predicted signals and ground-truth may not be a good metric because it does not consider such properties. For example, although several human behaviors can be still acceptable given the same input in our social signal prediction task, this metric penalizes them if they are different from the single ground-truth motion, regardless their quality. Due to the reason, the L2 metric often favors the output close to the mean of the data distribution, although it is qualitatively far from the expected output. The similar issue has been discussed in human motion forecasting field [188–190, 208]. In body gesture prediction trial described above subsection, we found a similar issue as shown in the Table 7.5, where the mean pose of the training set outperforms our prediction result, although the mean pose is is far from the realistic motion with only a static posture, while our result shows more human-like motion with reasonable dynamics.

7.5.3 Modeling More Diverse Social Interaction

In this chapter, we demonstrate social signal modeling in a triadic scene, focusing on the Haggling scenario we defined. As an important future direction, more general social scenarios (e.g., polyadic interaction among an arbitrary number of individuals) needs to be considered. This direction requires a way to build a larger social interaction database in more diverse social scenarios, which may not be handled by our studio setup. The video sequences capturing daily social interactions, millions of which are already available on the Internet, can be an important source for this direction, although obtaining 3D signals from these in-the-wild monocular videos is a challenging problem. In our recent work, we show a promising result in this direction by presenting a monocular total capture method [49]. How

to model social interaction among an arbitrary number of communication partners is another issue, since our current approach assumes a fixed number of communication partners.

Chapter 8

Discussion

8.1 Summary

The ultimate objective of this thesis is to endow robots with nonverbal communication skills, to make them genuinely interact with humans. This thesis argues that a solution to achieve the goal is by exploiting computational methods and machine learning techniques, leveraging a large-data data containing the social signals that humans use in actual social communications. This thesis directly tackles the fundamental hurdle, the lack of available such dataset, by building the Panoptic Studio, one of the most complicated sensor systems to sense human motions (Chapter 2). Leveraging the sensor system, this thesis presents the state-of-the-art measurement method to capture a broad spectrum of social signals of interacting individuals, including the motions from faces, bodies, and hands (Chapters 3, 4, and 5). Our system and reconstruction method enable us to build a large-scale 3D motion capture corpus capturing various behavioral cues of interacting people, for the first time in the history (Chapter 6). The final part of this thesis computationally verifies that social signals are highly correlated and predictive each other, and introduces a social signal prediction task as a way to modeling nonverbal communication in a data-driven manner (Chapter 7).

This thesis demonstrates that the face, body, and hands motion of freely interacting groups can be markerlessly captured with a sufficient amount of sensors by applying ML-based measurement techniques for each view and consolidating them together. This is a crucial step showing the potential that the markerless motion capture approaches can outperform the popular marker-based motion capture method in the near future, which struggles in handling occlusions causing broken trajectories.

As another important contribution, this thesis also demonstrates that the data captured in the lab environment can be used as a valuable source for the computer vision and machine learning algorithms applied in-the-wild. For example, our system enables to build the first

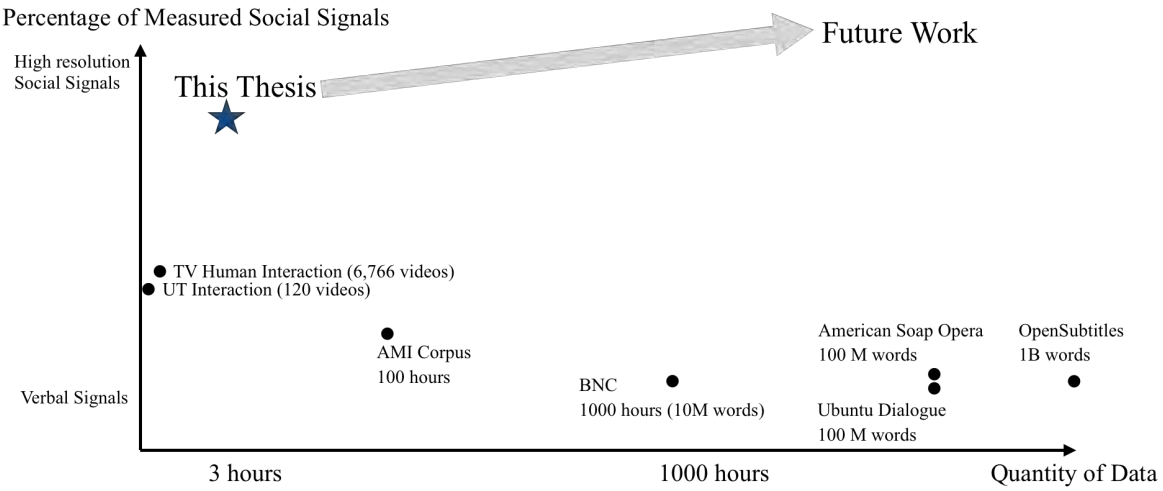


Fig. 8.1 As an important contribution, this thesis enables to measure much higher dimensional social signals exchanged in genuine social interaction compared to the existing work. Collecting higher fidelity and more quantity of data is interesting future directions.

widely used 2D hand keypoint detector [46]. Our system also enables us to build the first 3D deformable human model with the expressive power for face, body, and hands. Our dataset containing paired 3D motion annotation for each RGB image also allows us to build the first monocular total capture system [49].

8.2 Future Work

This thesis takes the first step toward the ambitious goal of building "social Artificial Intelligence". As an important contribution, this thesis enables to measure much higher dimensional social signals exchanged in genuine social interaction compared to the existing work (as illustrated in Figure 8.1), and demonstrates how to use the data to computationally model nonverbal communication. We believe that there will be immense opportunities in future direction by extending the direction this thesis presents. Three key future directions are summarized below.

Measuring Higher Fidelity Social Signals. Although this thesis presents the state-of-the-art sensing and reconstruction techniques to measure social signals, the quality of the output is still not close to the level of human perception. For example, eye gaze, which plays an important role in social communication [209–211], and subtle details in the facial expressions, hairs, and clothing are missing. These are mainly caused by limited camera resolution of the system where the cameras are focusing the large working volumes. In

particular, the cameras were chosen several years ago and already outdated. Although it is known that high-resolution images are extremely beneficial to reconstruct high-fidelity social signals [11, 13], using many high-resolution (e.g., 4K) and high-speed cameras to cover the large working volume directly introduces additional difficulties in handling and processing the larger data size. Current hand reconstruction is also vulnerable if both hands are overlapped each other, since the hand detector assumes that only a single hand is visible in a predicted bounding box. Tackling these issues is interesting future directions with many application domains, including robotics, HCI, virtual reality, and medical applications.

Measuring 3D Social Signals In-the-Wild. The Internet is a golden trove of social communication data, and these data should be utilized to model more general nonverbal communications. However, obtaining 3D signals from these in-the-wild monocular videos is a challenging problem requiring to solve the ill-posed depth ambiguity. This research may require 3D annotations for each 2D image, which cannot be reliably annotated by human annotators. We already show a promising result in this direction by presenting a monocular total capture method [49] leveraging our Panoptic Studio database providing paired 3D skeleton annotations for images. However, this is yet only applicable to the scenes with a single person only. Capturing social signals of more complicated scenes with multiple people with better quality should be considered as a future direction to collect social signal data at scale.

Modeling Long-term Social Behavior in A Living Environment. This thesis only considers a short-term behavior by collecting database in a short social game scenario. However, interpersonal social behaviors may change over time (e.g., our behavior would be changing if we see the same person for a long time). As a future work, a long-term social behavior analysis should be also considered. This requires to capture the behaviors of the same person for a long time, and as a way, a similar multiview setup as in our Panoptic Studio needs to be installed in a living environment where humans are actually living (e.g., A camera system in a home environment). By computationally analyzing an individual's behavior in social situations for a long-term, a subject-specific model which predicts the future signals of the particular person can be trained, which can be an important property in building robots co-existing to our environment (e.g., robotic pets). As human reactions responding to the same input signals vary across people, the individual-specific prediction model can produce a distinctive output reflecting the target individual's characteristics, attending to each individual's unique needs and goals. There are many novel scientific questions we can consider from the measurements in this new scenario including: (1) Does social familiarity

influence interpersonal social behaviors; (2) How does scene affordance (e.g., furniture or TV) affect social behaviors (e.g., social formation); (3) Can long-term observations of the same person increase the social signal prediction accuracy of the particular person.

References

[1] N.-J. Moore, H. M. III, and D. W. Stacks, "Nonverbal communication: Studies and applications," Oxford University Press, 2013.

[2] J. Cassell, C. Pelachaud, N. Badler, M. Steedman, B. Achorn, T. Becket, B. Douville, S. Prevost, and M. Stone, "Animated conversation: rule-based generation of facial expression, gesture & spoken intonation for multiple conversational agents," in *Annual Conference on Computer Graphics and Interactive Techniques*, 1994.

[3] J. Cassell, J. Sullivan, E. Churchill, and S. Prevost, "Embodied conversational agents," MIT press, 2000.

[4] A. Mehrabian and S. R. Ferris, "Inference of attitudes from nonverbal communication in two channels.," in *Journal of consulting psychology*, 1967.

[5] A. Mehrabian, "Silent messages: Implicit communication of emotions and attitudes," Wadsworth Pub Co, 1981.

[6] R. Birdwhistell, "Kinesics and context: Essays on body motion communication," in *University of Pennsylvania Press*, 1970.

[7] A. Newell and H. A. Simon, "Computer science as empirical inquiry: Symbols and search," ACM, 1976.

[8] D. E. Rumelhart and J. L. McClelland, "Parallel distributed processing: explorations in the microstructure of cognition," MIT Press, 1986.

[9] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," in *IEEE Computational intelligenCe magazine*, 2018.

[10] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," in *Nature*, 2015.

[11] T. Beeler, B. Bickel, P. Beardsley, B. Sumner, and M. Gross, "High-quality single-shot capture of facial geometry," in *ACM Transactions on Graphics*, 2010.

[12] A. Ghosh, G. Fyffe, B. Tunwattanapong, J. Busch, X. Yu, and P. Debevec, "Multiview face capture using polarized spherical gradient illumination," in *ACM Transactions on Graphics*, 2011.

[13] T. Beeler, F. Hahn, D. Bradley, B. Bickel, P. Beardsley, C. Gotsman, and M. Gross, "High-quality passive facial performance capture using anchor frames," in *ACM Transactions on Graphics*, 2011.

[14] D. Bradley, W. Heidrich, T. Popa, and A. Sheffer, "High resolution passive facial performance capture," in *ACM Transactions on Graphics*, 2010.

[15] L. Valgaerts, C. Wu, A. Bruhn, H.-P. Seidel, and C. Theobalt, "Lightweight binocular facial performance capture under uncontrolled lighting.," in *ACM Transactions on Graphics*, 2012.

[16] I. Oikonomidis, N. Kyriazis, and A. A. Argyros, "Tracking the articulated motion of two strongly interacting hands," in *Conference on Computer Vision and Pattern Recognition*, 2012.

[17] J. J. Tompson, A. Jain, Y. LeCun, and C. Bregler, "Joint training of a convolutional network and a graphical model for human pose estimation," in *Conference on Neural Information Processing Systems*, 2014.

[18] S. Sridhar, F. Mueller, A. Oulasvirta, and C. Theobalt, "Fast and robust hand tracking using detection-guided optimization," in *Conference on Computer Vision and Pattern Recognition*, 2015.

[19] D. Tzionas, L. Ballan, A. Srikantha, P. Aponte, M. Pollefeys, and J. Gall, "Capturing hands in action using discriminative salient points and physics simulation," in *International Journal of Computer Vision*, 2016.

[20] E. de Aguiar, C. Stoll, C. Theobalt, N. Ahmed, H.-P. Seidel, and S. Thrun, "Performance capture from sparse multi-view video," in *ACM Transactions on Graphics*, 2008.

[21] J. Gall, C. Stoll, E. De Aguiar, C. Theobalt, B. Rosenhahn, and H.-P. Seidel, "Motion capture using joint skeleton tracking and surface estimation," in *Conference on Computer Vision and Pattern Recognition*, 2009.

[22] C. Stoll, N. Hasler, J. Gall, H.-P. Seidel, and C. Theobalt, "Fast articulated motion tracking using a sums of gaussians body model," in *International Conference on Computer Vision*, 2011.

[23] A. Elhayek, E. Aguiar, A. Jain, J. Tompson, L. Pishchulin, M. Andriluka, C. Bregler, B. Schiele, and C. Theobalt, "Efficient convnet-based marker-less motion capture in general scenes with a low number of cameras," in *Conference on Computer Vision and Pattern Recognition*, 2015.

[24] "Vicon motion systems." www.vicon.com, 1984.

[25] D. Vlasic, P. Peers, I. Baran, P. Debevec, J. Popović, S. Rusinkiewicz, and W. Matusik, "Dynamic shape capture using multi-view photometric stereo," in *SIGGRAPH*, 2009.

[26] Y. Furukawa and J. Ponce, "Dense 3d motion capture from synchronized video streams," in *Conference on Computer Vision and Pattern Recognition*, 2008.

[27] A. Baak, M. M, G. Bharaj, H.-p. Seidel, and C. Theobalt, "A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera," in *International Conference on Computer Vision*, 2011.

[28] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore, "Real-time human pose recognition in parts from single depth images," in *Communications of the ACM*, 2013.

[29] G. Ye, Y. Liu, N. Hasler, X. Ji, Q. Dai, and C. Theobalt, "Performance capture of interacting characters with handheld kinects," in *European Conference on Computer Vision*, 2012.

[30] Y. Liu, J. Gall, C. Stoll, Q. Dai, H.-P. Seidel, and C. Theobalt, "Markerless motion capture of multiple characters using multiview image segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[31] D. S. Messinger, M. H. Mahoor, S.-M. Chow, and J. F. Cohn, "Automated measurement of facial expression in infant–mother interaction: A pilot study," in *Infancy*, 2009.

[32] G. Lucas, G. Stratou, S. Lieblich, and J. Gratch, "Trust me: multimodal signals of trustworthiness," in *International Conference on Multimodal Interaction*, 2016.

[33] G. McKeown, M. Valstar, R. Cowie, M. Pantic, and M. Schroder, "The semaine database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent," in *IEEE Transactions on Affective Computing*, 2012.

[34] J. Carletta, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, W. Kraaij, M. Kronenthal, *et al.*, "The ami meeting corpus: A pre-announcement," in *International Workshop on Machine Learning for Multimodal Interaction*, 2005.

[35] B. Lepri, R. Subramanian, K. Kalimeri, J. Staiano, F. Pianesi, and N. Sebe, "Connecting meeting behavior with extraversion—a systematic study," in *IEEE Transactions on Affective Computing*, 2012.

[36] B. Nojavanasghari, T. Baltrušaitis, C. E. Hughes, and L.-P. Morency, "Emoreact: a multimodal approach and dataset for recognizing emotional responses in children," in *International Conference on Multimodal Interaction*, 2016.

[37] C. Katsimerou, J. Albeda, A. Huldtgren, I. Heynderickx, and J. A. Redi, "Crowdsourcing empathetic intelligence: the case of the annotation of emma database for emotion and mood recognition," in *ACM Transactions on Intelligent Systems and Technology*, 2016.

[38] H. Gunes and M. Piccardi, "A bimodal face and body gesture database for automatic analysis of human nonverbal affective behavior," in *International Conference on Pattern Recognition*, 2006.

[39] T. Bänziger, M. Mortillaro, and K. R. Scherer, "Introducing the geneva multimodal expression corpus for experimental research on emotion perception.," in *Emotion*, 2012.

[40] P. R. De Silva and N. Bianchi-Berthouze, "Modeling human affective postures: an information theoretic characterization of posture features," in *Computer Animation and Virtual Worlds*, 2004.

[41] F. De la Torre, W.-S. Chu, X. Xiong, F. Vicente, X. Ding, and J. F. Cohn, "Intraface," in *Automatic Face and Gesture Recognition*, 2015.

[42] P. Ekman and W. V. Friesen, "Facial action coding system," Consulting Psychologists Press, 1977.

[43] S. Steidl, M. Levit, A. Batliner, E. Noth, and H. Niemann, "" of all things the measure is man" automatic classification of emotions and inter-labeler consistency," in *International Conference on Acoustics, Speech, and Signal Processing*, 2005.

[44] H. Aviezer, R. R. Hassin, J. Ryan, C. Grady, J. Susskind, A. Anderson, M. Moscovitch, and S. Bentin, "Angry, disgusted, or afraid? studies on the malleability of emotion perception," in *Psychological science*, 2008.

[45] L. F. Barrett, B. Mesquita, and M. Gendron, "Context in emotion perception," in *Current Directions in Psychological Science*, 2011.

[46] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[47] G. Hidalgo, Z. Cao, T. Simon, S.-E. Wei, H. Joo, and Y. Sheikh, "Openpose library." https://github.com/CMU-Perceptual-Computing-Lab/openpose, 2017.

[48] H. Joo, T. Simon, and Y. Sheikh, "Total capture: A 3d deformation model for tracking faces, hands, and bodies," in *Conference on Computer Vision and Pattern Recognition*, 2018.

[49] D. Xiang, H. Joo, and Y. Sheikh, "Monocular total capture: Posing face, body, and hands in the wild," in *Technical Report*, 2018.

[50] E. T. Hall, "The hidden dimension," Doubleday & Co, 1966.

[51] A. Kendon, "Spatial organization in social encounters: The f-formation system," in *Conducting interaction: Patterns of behavior in focused encounters*, Cambridge University Press, 1990.

[52] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, and Y. Sheikh, "Panoptic studio: A massively multiview system for social motion capture," in *International Conference on Computer Vision*, 2015.

[53] H. Joo, T. Simon, X. Li, H. Liu, L. Tan, L. Gui, S. Banerjee, T. Godisart, B. Nabbe, I. Matthews, *et al.*, "Panoptic studio: A massively multiview system for social interaction capture," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[54] H. Joo, H. S. Park, and Y. Sheikh, "Map visibility estimation for large-scale dynamic 3d reconstruction," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[55] R. Williams, "The geometrical foundation of natural structure: A source book of design," in *Dover Publications*, 1979.

[56] T. Simon, "Measuring human motion in social interactions," in *Doctoral Thesis, The Robotics Institute, Carnegie Mellon University*, 2017.

[57] C. Wu, "Visualsfm: A visual structure from motion system." http://ccwu.me/vsfm/, 2011.

[58] E. de Aguiar, C. Theobalt, C. Stoll, and H.-P. Seidel, "Marker-less deformable mesh tracking for human shape and motion capture," in *Conference on Computer Vision and Pattern Recognition*, 2007.

[59] J. Starck and A. Hilton, "Surface capture for performance-based animation," in *CGA*, 2007.

[60] A. Zaharescu, E. Boyer, and R. Horaud, "Topology-adaptive mesh deformation for surface evolution, morphing, and multiview reconstruction," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011.

[61] C. Budd, P. Huang, M. Klaudiny, and A. Hilton, "Topology-adaptive mesh deformation for surface evolution, morphing, and multiview reconstruction," in *International Journal of Computer Vision*, 2013.

[62] S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade, "Three-dimensional scene flow.," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005.

[63] R. Carceroni and K. Kutalakos, "Multi-view scene capture by surfel sampling: From video streams to non-rigid 3D motion, shape and reflectance," in *International Journal of Computer Vision*, 2002.

[64] F. Devernay, D. Mateus, and M. Guilbert, "Multi-Camera Scene Flow by Tracking 3-D Points and Surfels," in *Conference on Computer Vision and Pattern Recognition*, 2006.

[65] T. Tung and T. Matsuyama, "Dynamic surface matching by geodesic mapping for 3D animation transfer," in *Conference on Computer Vision and Pattern Recognition*, 2010.

[66] J. Starck and A. Hilton, "Spherical matching for temporal correspondence of non-rigid surfaces," in *International Conference on Computer Vision*, 2005.

[67] K. Varanasi and A. Zaharescu, "Temporal surface tracking using mesh evolution," in *European Conference on Computer Vision*, 2008.

[68] T. Basha, Y. Moses, and N. Kiryati, "Multi-view Scene Flow Estimation: A View Centered Variational Approach," in *International Journal of Computer Vision*, 2012.

[69] C. Vogel, K. Schindler, and S. Roth, "3D scene flow estimation with a rigid motion prior," in *International Conference on Computer Vision*, 2011.

[70] F. Huguet and F. Devernay, "A variational method for scene flow estimation from stereo sequences," in *International Conference on Computer Vision*, 2007.

[71] J. Quiroga, F. Devernay, and J. Crowley, "Scene flow by tracking in intensity and depth data," in *CVPR Workshops*, 2012.

[72] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[73] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *International Joint Conferences on Artificial Intelligence*, 1981.

[74] N. Sundaram, T. Brox, and K. Keutzer, "Dense point trajectories by gpu-accelerated large displacement optical flow," in *European Conference on Computer Vision*, 2010.

[75] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2001.

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

[77] J.-m. Frahm, P. Fite-georgel, D. Gallup, T. Johnson, R. Raguram, C. Wu, Y.-h. Jen, E. Dunn, S. Lazebnik, and M. Pollefeys, "Building rome on a cloudless day," in *European Conference on Computer Vision*, 2010.

[78] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski, "Towards internet-scale multiview stereo," in *Conference on Computer Vision and Pattern Recognition*, 2010.

[79] V. Belagiannis, S. Amin, and M. Andriluka, "3D pictorial structures for multiple human pose estimation," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[80] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[81] H. K. M. Meeren, C. C. R. J. van Heijnsbergen, and B. de Gelder, "Rapid perceptual integration of facial expression and emotional body language," in *National Academy of Sciences of the United States of America*, 2005.

[82] H. Aviezer, Y. Trope, and A. Todorov, "Body cues, not facial expressions, discriminate between intense positive and negative emotions.," in *Science*, 2012.

[83] G. Zen, B. Lepri, E. Ricci, and O. Lanz, "Space speaks: towards socially and personality aware visual surveillance," in *ACM International Workshop on Multimodal Pervasive Video Analysis*, 2010.

[84] 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 *British Machine Vision Conference*, 2011.

[85] X. Alameda-Pineda, J. Staiano, R. Subramanian, L. Batrinca, E. Ricci, B. Lepri, O. Lanz, and N. Sebe, "Salsa: A novel dataset for multimodal group behavior analysis," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

[86] T. Kanade, P. Rander, and P. Narayanan, "Virtualized reality: Constructing virtual worlds from real scenes," in *IEEE Multimedia*, 1997.

[87] W. Matusik, C. Buehler, R. Raskar, S. J. Gortler, and L. McMillan, "Image-based visual hulls," in *SIGGRAPH*, 2000.

[88] T. Matsuyama and T. Takai, "Generation, visualization, and editing of 3d video," in *International Symposium on 3D Data Processing, Visualization and Transmission*, 2002.

[89] M. Gross, S. Würmlin, M. Naef, E. Lamboray, C. Spagno, A. Kunz, E. Koller-Meier, T. Svoboda, L. Van Gool, S. Lang, K. Strehlke, A. V. Moere, and O. Staadt, "Blue-c: A spatially immersive display and 3d video portal for telepresence," in *SIGGRAPH*, 2003.

[90] B. Petit, J.-D. Lesage, E. Boyer, and B. Raffin, "Virtualization Gate," in *SIGGRAPH Emerging Technologies*, 2009.

[91] A. Collet, M. Chuang, P. Sweeney, D. Gillett, D. Evseev, H. Calabrese, D.and Hoppe, K. A., and S. Sullivan, "High-quality streamable free-viewpoint video," in *ACM Transactions on Graphics*, 2015.

[92] D. Vlasic, I. Baran, W. Matusik, and J. Popović, "Articulated mesh animation from multi-view silhouettes," in *ACM Transactions on Graphics*, 2008.

[93] J. Shotton, A. Fitzgibbon, M. Cook, and T. Sharp, "Real-time human pose recognition in parts from single depth images," in *Conference on Computer Vision and Pattern Recognition*, 2011.

[94] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[95] M. Burenius, J. Sullivan, and S. Carlsson, "3D pictorial structures for multiple view articulated pose estimation," in *Conference on Computer Vision and Pattern Recognition*, 2013.

[96] S. Amin, M. Andriluka, M. Rohrbach, and B. Schiele, "Multi-view pictorial structures for 3d human pose estimation.," in *British Machine Vision Conference*, 2013.

[97] A. Elhayek *et al.*, "Marconi-convnet-based marker-less motion capture in outdoor and indoor scenes," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.

[98] H. Joo, H. S. Park, and Y. Sheikh, "Map visibility estimation for large-scale dynamic 3d reconstruction," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[99] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Conference on Computer Vision and Pattern Recognition*, 2014.

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

[101] S. Agarwal, K. Mierle, and Others, "Ceres solver." http://ceres-solver.org, 2012.

[102] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black, "Smpl: A skinned multi-person linear model," in *ACM Transactions on Graphics*, 2015.

[103] C. Cao, Y. Weng, S. Zhou, Y. Tong, and K. Zhou, "Facewarehouse: A 3d facial expression database for visual computing," in *IEEE Transactions on Visualization and Computer Graphics*, 2014.

[104] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[105] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*, 2016.

[106] H. Woltring, "New possibilities for human motion studies by real-time light spot position measurement," in *Biotelemetry*, 1973.

[107] S. I. Park and J. K. Hodgins, "Capturing and animating skin deformation in human motion," in *ACM Transactions on Graphics*, 2006.

[108] D. Gavrila and L. Davis, "Tracking of humans in action: A 3-D model-based approach," in *ARPA Image Understanding Workshop*, 1996.

[109] K. M. Cheung, S. Baker, and T. Kanade, "Shape-from-silhouette across time part i: Theory and algorithms," in *International Journal of Computer Vision*, 2005.

[110] C. Bregler, J. Malik, and K. Pullen, "Twist based acquisition and tracking of animal and human kinematics," in *International Journal of Computer Vision*, 2004.

[111] R. Kehl and L. V. Gool, "Markerless tracking of complex human motions from multiple views," in *Computer Vision and Image Understanding*, 2006.

[112] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. P. Andriacchi, "Markerless Motion Capture through Visual Hull, Articulated ICP and Subject Specific Model Generation," in *International Journal of Computer Vision*, 2010.

[113] T. Brox, B. Rosenhahn, J. Gall, and D. Cremers, "Combined region and motion-based 3D tracking of rigid and articulated objects.," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2010.

[114] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis, "Scape: shape completion and animation of people," in *ACM Transactions on Graphics*, 2005.

[115] G. Pons-Moll, J. Romero, N. Mahmood, and M. J. Black, "Dyna: A model of dynamic human shape in motion," in *ACM Transactions on Graphics*, 2015.

[116] C. Zhang, S. Pujades, M. Black, and G. Pons-Moll, "Detailed, accurate, human shape estimation from clothed 3d scan sequences," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[117] G. Pons-Moll, S. Pujades, S. Hu, and M. J. Black, "Clothcap: Seamless 4d clothing capture and retargeting," in *ACM Transactions on Graphics*, 2017.

[118] X. Zhou, S. Leonardos, X. Hu, and K. Daniilidis, "3d shape estimation from 2d landmarks: A convex relaxation approach," in *Conference on Computer Vision and Pattern Recognition*, 2015.

[119] F. Bogo, A. Kanazawa, C. Lassner, P. V. Gehler, J. Romero, and M. J. Black, "Keep it SMPL: automatic estimation of 3d human pose and shape from a single image," in *European Conference on Computer Vision*, 2016.

[120] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei, "Deep kinematic pose regression," in *European Conference on Computer Vision Workshop*, 2016.

[121] D. Mehta, S. Sridhar, O. Sotnychenko, H. Rhodin, M. Shafiei, H. Seidel, W. Xu, D. Casas, and C. Theobalt, "Vnect: Real-time 3d human pose estimation with a single RGB camera," in *ACM Transactions on Graphics*, 2017.

[122] D. Tome, C. Russell, and L. Agapito, "Lifting from the deep: Convolutional 3d pose estimation from a single image," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[123] P. Garrido, L. Valgaerts, C. Wu, and C. Theobalt, "Reconstructing detailed dynamic face geometry from monocular video," in *ACM Transactions on Graphics*, 2013.

[124] H. Li, J. Yu, Y. Ye, and C. Bregler, "Realtime facial animation with on-the-fly correctives.," in *ACM Transactions on Graphics*, 2013.

[125] J. Thies, M. Zollhofer, M. Stamminger, C. Theobalt, and M. Nießner, "Face2face: Real-time face capture and reenactment of rgb videos," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[126] C. Cao, D. Bradley, K. Zhou, and T. Beeler, "Real-time high-fidelity facial performance capture," in *ACM Transactions on Graphics*, 2015.

[127] C. Wu, D. Bradley, M. Gross, and T. Beeler, "An anatomically-constrained local deformation model for monocular face capture," in *ACM Transactions on Graphics*, 2016.

[128] D. Tang, H. Jin Chang, A. Tejani, and T.-K. Kim, "Latent regression forest: Structured estimation of 3D articulated hand posture," in *Conference on Computer Vision and Pattern Recognition*, 2014.

[129] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun, "Hand pose estimation and hand shape classification using multi-layered randomized decision forests," in *European Conference on Computer Vision*, 2012.

[130] C. Xu and L. Cheng, "Efficient hand pose estimation from a single depth image," in *International Conference on Computer Vision*, 2013.

[131] X. Sun, Y. Wei, S. Liang, X. Tang, and J. Sun, "Cascaded hand pose regression," in *Conference on Computer Vision and Pattern Recognition*, 2015.

[132] C. Wan, A. Yao, and L. Van Gool, "Direction matters: hand pose estimation from local surface normals," in *European Conference on Computer Vision*, 2016.

[133] S. Sridhar, A. Oulasvirta, and C. Theobalt, "Interactive markerless articulated hand motion tracking using RGB and depth data," in *International Conference on Computer Vision*, 2013.

[134] T. Sharp, C. Keskin, D. Robertson, J. Taylor, J. Shotton, D. Kim, C. Rhemann, I. Leichter, A. Vinnikov, Y. Wei, *et al.*, "Accurate, robust, and flexible real-time hand tracking," in *ACM CHI Conference on Human Factors in Computing Systems*, 2015.

[135] Q. Ye, S. Yuan, and T.-K. Kim, "Spatial attention deep net with partial pso for hierarchical hybrid hand pose estimation," in *European Conference on Computer Vision*, 2016.

[136] L. Ballan, A. Taneja, J. Gall, L. Van Gool, and M. Pollefeys, "Motion capture of hands in action using discriminative salient points," in *European Conference on Computer Vision*, 2012.

[137] J. Romero, D. Tzionas, and M. J. Black, "Embodied hands: Modeling and capturing hands and bodies together," in *ACM Transactions on Graphics*, 2017.

[138] "Realitycapture software." www.capturingreality.com/, 2016.

[139] E. Sapir, "The unconscious patterning of behavior in society," in *Selected Writings of Edward Sapir in Language, Culture, and Personality*, 1949.

[140] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, "Human3.6m: Large scale datasets and predictive methods for 3D human sensing in natural environments," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2014.

[141] P. Ekman and W. V. Friesen, "The repertoire of nonverbal behavior: Categories, origins, usage, and coding," in *Semiotica*, 1969.

[142] C. E. Osgood, "The nature and measurement of meaning.," in *Psychological bulletin*, 1952.

[143] J. A. Russell, "Affective space is bipolar," in *Journal of personality and social psychology*, 1979.

[144] R. Plutchik, "The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice," in *American scientist*, 2001.

[145] F. Noroozi, C. A. Corneanu, D. Kamińska, T. Sapiński, S. Escalera, and G. Anbarjafari, "Survey on emotional body gesture recognition," in *arXiv preprint arXiv:1801.07481*, 2018.

[146] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *CVPR Workshops*, 2010.

[147] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker, "Multi-pie," in *Image and Vision Computing*, 2010.

[148] A. Dhall, R. Goecke, J. Joshi, M. Wagner, and T. Gedeon, "Emotion recognition in the wild challenge 2013," in *International Conference on Multimodal Interaction*, 2013.

[149] A. Mollahosseini, B. Hasani, M. J. Salvador, H. Abdollahi, D. Chan, and M. H. Mahoor, "Facial expression recognition from world wild web," in *CVPR Workshops*, 2016.

[150] C. Fabian Benitez-Quiroz, R. Srinivasan, and A. M. Martinez, "Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[151] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," in *arXiv*, 2017.

[152] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos, *et al.*, "The ami meeting corpus," in *International Conference on Methods and Techniques in Behavioral Research*, vol. 88, 2005.

[153] M. Zancanaro, B. Lepri, and F. Pianesi, "Automatic detection of group functional roles in face to face interactions," in *International Conference on Multimodal Interfaces*, 2006.

[154] B. Lepri, R. Subramanian, K. Kalimeri, J. Staiano, F. Pianesi, and N. Sebe, "Connecting meeting behavior with extraversion—a systematic study," in *IEEE Transactions on Affective Computing*, 2012.

[155] J. Rehg, G. Abowd, A. Rozga, M. Romero, M. Clements, S. Sclaroff, I. Essa, O. Ousley, Y. Li, C. Kim, *et al.*, "Decoding children's social behavior," in *Conference on Computer Vision and Pattern Recognition*, 2013.

[156] M. Farenzena, A. Tavano, L. Bazzani, D. Tosato, G. Paggetti, G. Menegaz, V. Murino, and M. Cristani, "Social interactions by visual focus of attention in a three-dimensional environment," in *Workshop on Pattern Recognition and Artificial Intelligence for Human Behaviour Analysis*, 2009.

[157] R. Gross and J. Shi, "The cmu motion of body (mobo) database," 2001.

[158] L. Sigal, A. O. Balan, and M. J. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion," in *International booktitle of computer vision*, 2010.

[159] W. Güth, R. Schmittberger, and B. Schwarze, "An experimental analysis of ultimatum bargaining," in *Journal of Economic Behavior and Organization*, 1982.

[160] A. Chaudhuri, "Experiments in economics: Playing fair with money," in *Routledge*, 2009.

[161] F. Haffner, "Questions to dimitry davidoff about the creation of mafia," in *The Basics of Game Theory and Associated Games*, 1999.

[162] R. W. Picard and R. Picard, *Affective computing*. MIT press, 1997.

[163] R. W. Picard, "Affective computing: challenges," in *International Journal of Human-Computer Studies*, 2003.

[164] S. Poria, E. Cambria, R. Bajpai, and A. Hussain, "A review of affective computing: From unimodal analysis to multimodal fusion," in *Information Fusion*, 2017.

[165] A. Pentland, "Social dynamics: Signals and behavior," in *International Conference on Developmental Learning*, 2004.

[166] F. J. Bernieri, J. S. Reznick, and R. Rosenthal, "Synchrony, pseudosynchrony, and dissynchrony: Measuring the entrainment process in mother-infant interactions.," in *Journal of personality and social psychology*, 1988.

[167] L.-P. Morency, I. de Kok, and J. Gratch, "Predicting listener backchannels: A probabilistic multimodal approach," in *International Workshop on Intelligent Virtual Agents*, 2008.

[168] L.-P. Morency, "Modeling human communication dynamics," in *IEEE Signal Processing Magazine*, 2010.

[169] C.-M. Huang and B. Mutlu, "Learning-based modeling of multimodal behaviors for humanlike robots," in *International Conference on Human-robot Interaction*, 2014.

[170] A. Kendon, "Gesticulation and speech: Two aspects of the process of utterance," in *The relationship of verbal and nonverbal communication*, 1980.

[171] D. McNeill, *Hand and mind: What gestures reveal about thought*. University of Chicago press, 1992.

[172] Z. M. Griffin, "Gaze durations during speech reflect word selection and phonological encoding," in *Cognition*, 2001.

[173] S. Levine, P. Krähenbühl, S. Thrun, and V. Koltun, "Gesture controllers," in *ACM Transactions on Graphics*, 2010.

[174] S. Marsella, Y. Xu, M. Lhommet, A. Feng, S. Scherer, and A. Shapiro, "Virtual character performance from speech," in *ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2013.

[175] C. Darwin, *The expression of the emotions in man and animals*. John Murray, 1872.

[176] B. De Gelder, "Why bodies? twelve reasons for including bodily expressions in affective neuroscience," in *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 2009.

[177] A. Vinciarelli, M. Pantic, and H. Bourlard, "Social signal processing: Survey of an emerging domain," in *Image and vision computing*, 2009.

[178] A. Vinciarelli, M. Pantic, D. Heylen, C. Pelachaud, I. Poggi, F. D'Errico, and M. Schroeder, "Bridging the gap between social animal and unsocial machine: A survey of social signal processing," in *IEEE Transactions on Affective Computing*, 2012.

[179] W.-S. Chu, F. De la Torre, and J. F. Cohn, "Selective transfer machine for personalized facial action unit detection," in *Conference on Computer Vision and Pattern Recognition*, 2013.

[180] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," in *Image and Vision Computing*, 2009.

[181] F. Setti, C. Russell, C. Bassetti, and M. Cristani, "F-formation detection: Individuating free-standing conversational groups in images," in *PloS one*, 2015.

[182] Y. Yang, S. Baker, A. Kannan, and D. Ramanan, "Recognizing proxemics in personal photos," in *Conference on Computer Vision and Pattern Recognition*, 2012.

[183] A. Fathi, J. K. Hodgins, and J. M. Rehg, "Social interactions: A first-person perspective," in *Conference on Computer Vision and Pattern Recognition*, 2012.

[184] K. Schindler, L. Van Gool, and B. de Gelder, "Recognizing emotions expressed by body pose: A biologically inspired neural model," in *Neural networks*, 2008.

[185] H. S. Park, E. Jain, and Y. Sheikh, "3d social saliency from head-mounted cameras," in *Conference on Neural Information Processing Systems*, 2012.

[186] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *European Conference on Computer Vision*, 2012.

[187] D.-A. Huang and K. M. Kitani, "Action-reaction: Forecasting the dynamics of human interaction," in *European Conference on Computer Vision*, 2014.

[188] V. Mnih, H. Larochelle, and G. E. Hinton, "Conditional restricted boltzmann machines for structured output prediction," in *arXiv*, 2012.

[189] K. Fragkiadaki, S. Levine, P. Felsen, and J. Malik, "Recurrent network models for human dynamics," in *International Conference on Computer Vision*, 2015.

[190] A. Jain, A. R. Zamir, S. Savarese, and A. Saxena, "Structural-rnn: Deep learning on spatio-temporal graphs," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[191] J. Walker, C. Doersch, A. Gupta, and M. Hebert, "An uncertain future: Forecasting from static images using variational autoencoders," in *European Conference on Computer Vision*, 2016.

[192] R. Villegas, J. Yang, Y. Zou, S. Sohn, X. Lin, and H. Lee, "Learning to generate long-term future via hierarchical prediction," in *arXiv*, 2017.

[193] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," in *Physical review E*.

[194] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Conference on Computer Vision and Pattern Recognition*, 2016.

[195] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Conference on Computer Vision and Pattern Recognition*, 2018.

[196] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "Openface: an open source facial behavior analysis toolkit," in *Winter Conference on Applications of Computer Vision*, 2016.

[197] V. Ramakrishna, T. Kanade, and Y. Sheikh, "Reconstructing 3d human pose from 2d image landmarks," in *Conference on Computer Vision and Pattern Recognition*, 2012.

[198] J. Martinez, R. Hossain, J. Romero, and J. J. Little, "A simple yet effective baseline for 3d human pose estimation," in *International Conference on Computer Vision*, 2017.

[199] X. Zhou, Q. Huang, X. Sun, X. Xue, and Y. Wei, "Towards 3d human pose estimation in the wild: a weakly-supervised approach," in *International Conference on Computer Vision*, 2017.

[200] F. Moreno-noguer, "3d human pose estimation from a single image via distance matrix regression," in *Conference on Computer Vision and Pattern Recognition*, 2017.

[201] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, "Monocular 3d human pose estimation in the wild using improved cnn supervision," in *International Conference on 3D Vision*, 2017.

[202] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," in *ACM Transactions on Graphics*, 2016.

[203] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision*, 2014.

[204] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," in *The Journal of Machine Learning Research*, 2014.

[205] S. J. Reddi, S. Kale, and S. Kumar, "On the convergence of adam and beyond," in *International Conference on Learning Representations*, 2018.

[206] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *International Conference on Learning Representations*, 2017.

[207] M. Vázquez, E. J. Carter, B. McDorman, J. Forlizzi, A. Steinfeld, and S. E. Hudson, "Towards robot autonomy in group conversations: Understanding the effects of body orientation and gaze," in *International Conference on Human-Robot Interaction*, 2017.

[208] Y. Zhou, Z. Li, S. Xiao, C. He, Z. Huang, and H. Li, "Auto-conditioned recurrent networks for extended complex human motion synthesis," in *International Conference on Learning Representations*, 2018.

[209] K. Rayner, "Eye movements in reading and information processing: 20 years of research.," in *Psychological bulletin*, 1998.

[210] C. K. Friesen and A. Kingstone, "The eyes have it! reflexive orienting is triggered by nonpredictive gaze," in *Psychonomic bulletin & review*, 1998.

[211] P. Ricciardelli, E. Bricolo, S. M. Aglioti, and L. Chelazzi, "My eyes want to look where your eyes are looking: Exploring the tendency to imitate another individual's gaze," in *Neuroreport*, 2002.