Exploiting Point Motion, Shape Deformation, and Semantic Priors for Dynamic 3D Reconstruction in the Wild

Minh Phuoc Vo

CMU-RI-TR-19-73

August, 2019

School of Computer Science
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Srinivasa G. Narasimhan, (Co-chair) (CMU)
Yaser Sheikh, (Co-Chair) (CMU/Facebook)
Michael Kaess, (CMU)
Marc Pollefeys, (ETH/Microsoft)

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

Abstract

With the advent of affordable and high-quality smartphone cameras, any significant events will be massively captured both actively and passively from multiple perspectives. This opens up exciting opportunities for low-cost high-end VFX effects and large scale media analytics. However, automatically organizing large scale visual data and creating a comprehensive 3D scene model is still an unsolved problem. State of the art 3D reconstruction algorithms are mostly applicable to static scenes, mainly due to the lack of triangulation constraints for dynamic objects observed by unsynchronized cameras and the difficulties in finding reliable correspondences across cameras in diverse and dynamic settings.

This thesis aims to provide a computational pipeline for high-quality 3D reconstruction of the dynamic scene captured by multiple unsynchronized video cameras in the wild. The key is to exploit the physics of motion dynamics, shape deformation, scene semantics, and the interplay between them. Toward this end, this thesis makes four enabling technical contributions.

First, this thesis introduces a spatiotemporal bundle adjustment algorithm to accurately estimate a sparse set of 3D trajectories of dynamic objects from multiple unsynchronized mobile video cameras. The lack of triangulation constraint on dynamic points is solved by carefully integrating physics-based motion prior describing how points move over time. This algorithm takes advantage of the unsynchronized video streams to estimate 3D motion reconstruction in the wild at much higher temporal resolution than the input videos.

Second, this thesis presents a simple but powerful self-supervised framework to adapt a generic person appearance descriptor to the unlabeled videos by exploiting motion tracking, mutual exclusion constraints, and multi-view geometry without any manual annotations. The adapted descriptor is strongly discriminative and enables a tracking-by-clustering formulation. This advantage enables a first-of-a-kind accurate and consistent markerless motion tracking of multiple people participating in a complex group activity from mobile cameras in the wild with further application to multi-angle video cutting for intuitive tracking visualization.

Third, this thesis creates a framework for 3D tracking of the rigidly moving objects even in severe occlusions by fusing single-view unstructured tracklets and multi-view semantic structured keypoints reconstruction. No spatial correspondences are needed for the unstructured points. No temporal correspondences are needed for the structured points. The imprecise but accurate 3D structured keypoint is compensated by the sparse but precise 3D unstructured tracks, leading to improvements in both structured keypoints localization and motion tracking of the entire object.

Fourth, this thesis presents a single-shot illumination decomposition method for dense dynamic shape capture of highly textured surfaces illuminated by multiple projectors. The decomposition scheme assumes smooth shape deformation and can accurately recover the illumination image of different projectors and the texture images of the scene from their mixed appearances.

Acknowledgments

I am deeply thankful to my advisors, Yaser Sheikh and Srinivasa Narasimhan, for their constant and dedicated support, advice, and inspiration. I not only learn scientific and ambitious thoughts but also the important art of presentation. I also greatly appreciate them for the endless freedom in pursuing my research. I could not ask for more from them.

I thank Hongdong Li for the insightful discussions on geometry problems. He offers a charming perspective on how a researcher should be. He emerged as my third advisor during his short time at CMU.

During my time at Microsoft Research, Adobe Research, and Facebook Reality Lab, I have had the pleasure of working and discussing with Sudipta Sinha, Kalyan Sunkavalli, and Carsten Stoll. I learned a great deal from their distinctive expertise and philosophy in solving research problems. I also thank Sunil Hadap and Ersin Yumer for showing me different faces of industrial survival and striving.

I thank my thesis committee members, Michael Kaess and Marc Pollefeys, for going through the entire thesis and giving me critical comments and advice.

I am fortunate to have many terrific colleagues in during my doctoral study. I thank Aayush for the positive energy and inspiring thoughts and Hanbyul for the sincere and constructive feedback. I thank Tomas and Supreeth for the technical hot fixes and Robert for his help with the projector. I thank Dinesh for the fascinating effort on car reconstruction. I also thank other friends for the up and downtime together: Yuxiong, Hyunsoo, Varun, Yair, Natasha, Suren, Jian, Chao, Gines, Shih-En, Zhe, Tiancheng, Shumian, Donglai, Sean, Lin, and Gaku.

Last, but not least, I thank my extended family for their support and encouragement. I forever owe a debt of gratitude to my beloved wife, Phuong, and our children, Michelle and Maxwell, for the time apart. This thesis is dedicated to them.

This thesis is dedicated to my wife, Phuong, and our children, Michelle and Maxwell.

Contents

Chapter 1

Introduction

Our memory is filled with many joyful social events: a birthday party with a group of close friends, a show enacted by children, or our own wedding ceremony. Imagine if we could go back in time and revisit such fine moments of our life. With the advent of affordable and high-quality smartphone cameras and the popularity of social media sharing websites, any significant events can be captured from multiple perspectives. These collections of such massive visual data provide a unique opportunity for rich explorations of the scenes, far exceeding what is possible with a single camera. See Figure 1.1 for an example.

Despite great potentials, automatically organizing such rich dynamic visual data into a single comprehensive 4D (space and time) model that could facilitate content browsing or media analytics is challenging. The video data could be acquired from different locations, angles, zoom settings, with varying focus, different types of cameras (high quality, low quality), and at different times. The dynamics and diversity of this new form of visual data make it difficult to correspond images of the same object in the videos. Additionally, the mathematics of reconstructing moving objects from unsynchronized cameras is not well-understood. Consequently, current techniques on large scale analytics of community visual data are limited to static scenes [75, 78, 188, 196], high fidelity dynamic 3D reconstruction requires elaborate setups and

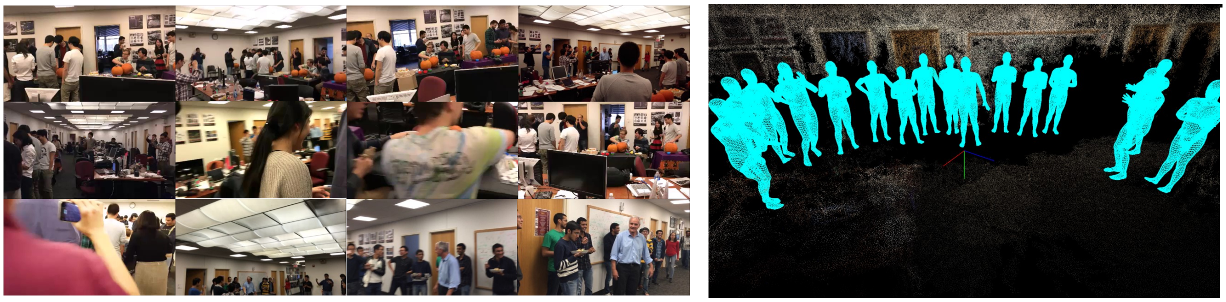


Figure 1.1: A surprise birthday party for Martial Hebert organized by the faculty members and students at CMU. Today, most interesting dynamic social events can be easily recorded from multiple high-quality mobile cameras. Organizing and creating a 4D (space and time) scene virtualization from such imagery opens up new exciting ways for novel visual experiences.

expensive acquisition systems [3, 52, 61, 95, 179], and dynamic event browsing is limited to image-based rendering [19, 138] or predefined multi-angle videos without interactive scene exploration [2, 14].

Overcoming these challenges unlocks strong potential applications in VR/AR, surveillance, statistical understanding of crowd psychology, and even the chance of reconstructing historic events from the Internet videos. As an example application, since the videos are captured from different perspectives and at slightly different times, combining their information in 3D space effectively re-creates the scene at higher spatial and temporal resolution than what can be obtained from individual videos. Imagine we could playback the moment when your wife entered your wedding ceremony, when we poured the champagne into the stack of champagne glasses, or when our wives threw the bouquet to the crowd at slow motion and from all possible angles. Such magical moments, lying deep in your memory, magically come back to us immersively and realistically. The same system can also be used for surveillance or crowd psychology analytics as the unconstrained visual data has been automatically organized, reconstructed, and tracked in both space and time. As another example, consider a historical tragedy such as The Boston Bombing in 2013. This event was captured by infrastructure cameras and civilian personal cameras from many different angles. If the information was automatically fused from the vast amount of visual data, both the suspects and the explosive devices could be quickly identified. More importantly, the civilian could be evacuated along the save escape routes that were planned according to the reconstructed dynamic scene model. The pattern of people running away from the event could also be studied for better construction of emergency evacuation routes.

The goal of this thesis is to create a set of computational algorithms for reliable and accurate dynamic 3D reconstruction in the unconstrained crowd captured settings. Our key is to exploit the physics of motion dynamics, shape deformation, scene semantics, and their interplay. Toward this goal, we first model the dynamic motion of objects to jointly estimate the spatial and temporal configuration of both the cameras and the dynamic objects. Secondly, we create a self-supervise semantic scene adaptation algorithm to find reliable coarse correspondences across wide-baseline views. Lastly, we connect the motion and semantics using shape deformation prior and improve both motion tracking as well as semantic detections.

Thesis organization: While we mostly focus this thesis on reconstructing event captured from passive mobile sensors (e.g., smartphone camera) because this is the most common mode of visual data nowadays (Chapters 2 3 4), we also briefly describe our efforts on active sensing for dense dynamic capture (Chapter 5). We believe with the recent progress in projector/emitter miniaturization (time-of-flight cameras are already available in Galaxy S10 and Huawei P30 Pro) and smart sensing algorithms [5, 165], smartphones can be massively equipped with active sensing system for dense dynamic reconstruction in the wild.

1.1 Challenges

Creating a dynamic 3D reconstruction by integrating data from multiple imaging sensors in the wild poses two foremost challenges: (1) geometry scene reconstruction with **no geometric triangulation constraint**; (2) **accurate spatiotemporal correspondences** under diverse scene motion and wide-baseline cameras for coherent shape and motion tracking. These challenges are decomposed into the specific problem as following:

Unknown temporal alignment: While dense and accurate dynamic structure estimation might not be needed often for event browsing, precise video sub-frame alignment is essential [19, 127]. Current computational methods for video alignment, either using audio signal [215], video bit rate profile [215], action recognition [24], or image-based interpolation of visual features [38, 63, 221, 239], are unreliable especially for scenes with strong ambient sound such as in a party or relatively fast-moving scenes. Estimating the correct time alignment of multiple cameras is a combinatorial problem. As the number of video increases, even a minute error in the temporal sequencing of the video frames severely hurts the 3D motion reconstruction (as shown later in Chapter 2) and creates annoying playback artifacts degrading the browsing experience.

Lack of geometric constraint: When a moving point is observed from multiple cameras with simultaneously triggered shutters, the dynamic 3D reconstruction problem reduces exactly to the case of static 3D reconstruction. The classic point triangulation constraint [140], and the algorithmic edifice of bundle adjustment [211] built upon it, applies directly. However, no current consumer mechanism exists to ensure that multiple personal cameras, i.e., smartphones, or consumer camcorders, are simultaneously triggered [127]. Thus, in the vast majority of dynamic scenes captured by multiple independent video cameras, no two cameras see the 3D point at the same time instant. This fact trivially invalidates the triangulation constraint and disqualifies the decades of research in photogrammetry [150] and geometry-based vision [94, 144].

Determining dense correspondences of dynamic objects from multiple cameras: This is a challenge because many of the dynamic objects are non-rigid, i.e., not only their positions but also their shapes change over time. The unknown shapes and motion models prohibit reliable visual motion prediction, especially for multiple people interacting scenes where visual occlusion or motion blur are frequent. Moreover, some dynamic objects may be textureless which makes pixel-level surface tracking impossible. Currently, solutions based on active sensing are still not practical for mobile devices due to their large energy consumption.

Accurate camera localization in the presence of moving objects: For any interesting dynamic event involved many people, the background is usually dynamic. The static objects such as buildings, walls are mostly occluded by other humans. This makes camera localization challenging because there are few visual features of static objects available to build a reliable model of the environment.

Visual artifacts caused by camera sensors and optics: Most modern consumer-grade cameras employ a rolling shutter sensor. While the estimation of rolling shutter camera poses and its

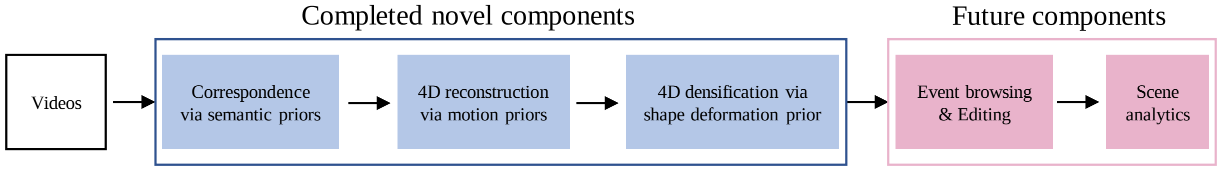

Completed novel components
Future components
Videos
Correspondence via semantic priors
4D reconstruction via motion priors
4D densification via shape deformation prior
Event browsing & Editing
Scene analytics

Figure 1.2: Our proposed pipeline for automatic total 4D reconstruction of a dynamic scene from crowd capture settings. It takes multiple video sequences of an event and produces accurate temporally coherent motion tracking of all objects along with their coarse meshes that can be later fed to a learning-based module for realistic event browsing and scene analytics.

possible degeneracy configuration have been studied [8, 10, 56], most algorithms make strong assumptions about the smoothness of the camera motion [169], which could adversely filter out the high-frequency motion content, e.g. sudden vibration due to footstep. Additionally, accurate self-calibration of rolling shutter scanning speed in unconstrained settings is unexplored. For fast-moving dynamic scenes, improper handling of this temporal parameter could cause large errors in estimating the 3D trajectories of the moving points (as shown later in Chapter. 2). Realistic optics issues such as motion blur, camera defocus, or image overexposure are extra sources of error for visual tracking algorithms.

In general, we must recognize that the challenges are **tightly coupled**. As an example, consider the problem of estimating 3D camera pose. While segmenting out stationary points and using them to estimate camera pose is a strategy that has been used in prior work [168], it ignores evidence from moving points that are often closer to the cameras and therefore provide tighter constraints for precise camera calibration. As another example, consider the multiview time alignment problem. While there are infinitely many dynamic 3D trajectories passing through the camera viewing rays, each of these paths corresponds to a different temporal sequencing of the rays. And yet, the true trajectory that correctly aligns all the cameras must follow a and physically "natural and plausible" path. Lastly, while finding dense and accurate pixel-level correspondences between cameras are hard, coarse region-level correspondence constraints the pixel correspondence search. In return, accurate pixel-level correspondence stabilizes the motion tracking and enables better localization of the region correspondences. These facts serve as the guiding principles of our solution.

1.2 Contributions

Figure 1.2 shows our pipeline for automatic dynamic scene reconstruction from crowd capture data. We tackle the above challenges by exploiting the physics of motion dynamics, shape deformation, scene semantics, and their interplay. The reconstruction evolves progressively from coarse the object skeleton level estimation using semantics priors, to sparse but precise trajectory reconstruction of salient features using motion prior, and finally the combination of sparse trajectories and coarse skeleton to improve both data sources. Using active sensing and shape deformation prior, we also present a novel illumination decomposition approach for dense dynamic shape capture. We believe with the recent progress in projector/emitter miniaturization (time-of-flight cameras are already available in Galaxy S10 and Huawei P30 Pro) and smart sensing algorithms [5, 165], smartphones can be massively equipped with active sensing system for dense dynamic reconstruction in the wild. We state our specific contributions as follow.

1.2.1 Spatiotemporal Bundle Adjustment for Dynamic 3D Reconstruction

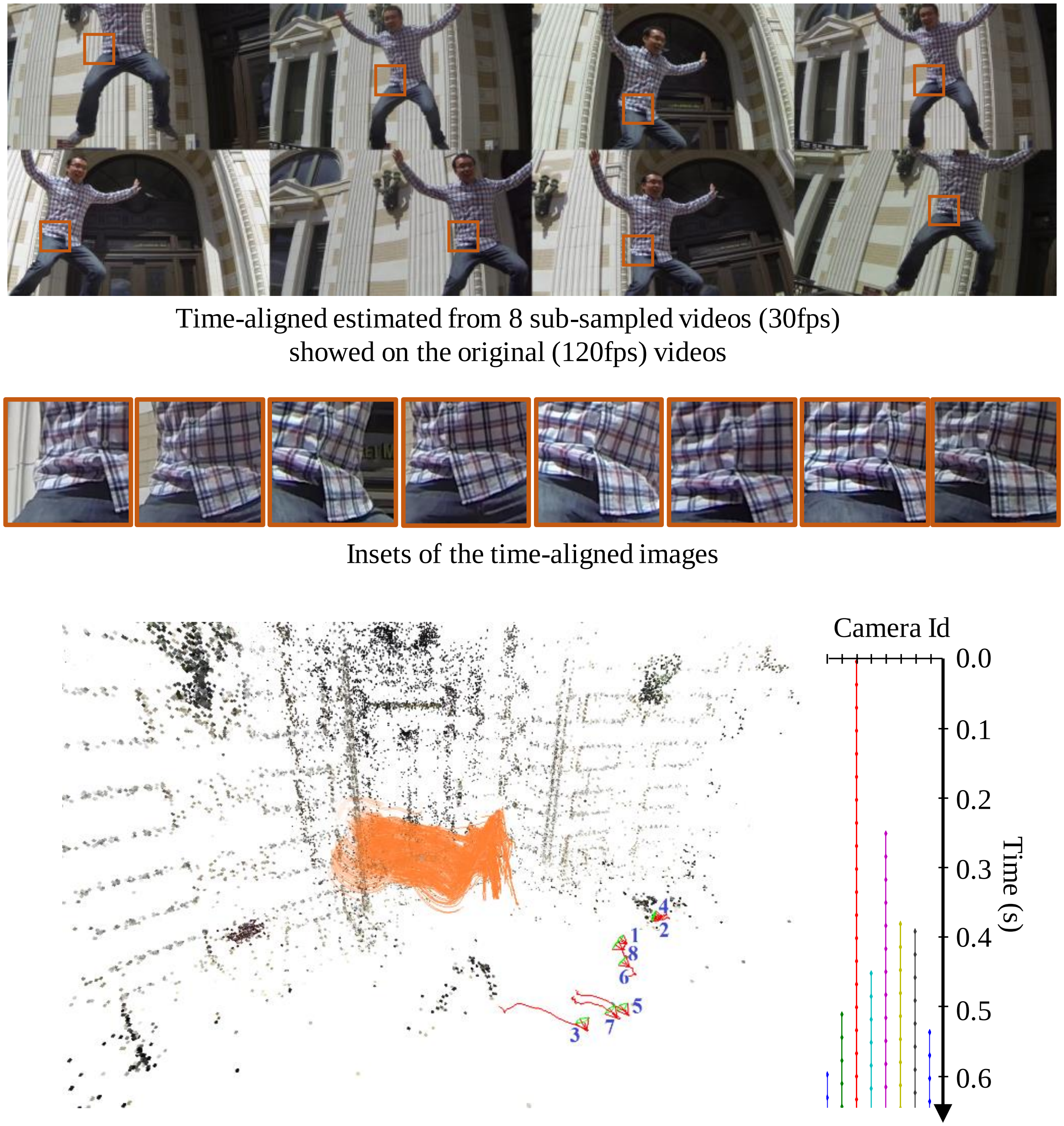
To optimally solve the dynamic 3D reconstruction from unsynchronized video streams, we must first recognize all the constituent sub-problems that exist. The classic problems of point triangulation and camera resectioning in the static case are subsumed. Two new problems arise: reconstructing 3D trajectories of moving points and estimating the temporal alignment of each camera. Second, we must recognize that the sub-problems are tightly coupled and must be solved jointly. Since there is only one observation of the dynamic point at any time instances, the reconstruction of its motion/trajectory is ill-posed. Clearly, there are infinitely many trajectories passing through the camera viewing rays and each of these paths corresponds to a different temporal sequencing of the rays. Yet, the true trajectory must also correctly align all the cameras. Thus, to sufficiently constrain the solution, we recognize that **both the geometry and dynamics of the point have to be considered**. We analyze several physical motion priors, i.e. least kinetic energy, least force, and least action, on a large motion capture corpus of CMU Motion Capture database, and carefully integrate of these priors within the reconstruction pipeline. Our formulation not only allows sub-frame temporal alignment but also fully exploits the asynchronous video streams to recover sparse but accurate 3D trajectories at much high temporal resolution than the framerate of the input videos. As a demonstration, we reconstruct 3D trajectories of dynamic actions captured outdoor at 240fps from 8 hand-held cameras captured at 30fps (see Figure 1.3)[1].

1.2.2 Self-supervised Scene-adaptive Multiview Human Association

To address the problem of people association across multiple viewpoints and time instances, we propose to use **self-adaptive learning of strong appearance descriptor specifically for the domain videos for people matching**. We combine motion tracking, mutual exclusion constraints, and multi-view geometry in a multi-task learning framework to automatically adapt a generic person appearance descriptor to the domain videos. A discriminative person descriptor enables the use of clustering for tracking individual persons. Since the association is solved globally, there is no tracking drift. As a bi-product of the association, the videos are also temporally aligned (up to the speed of the observed motion).

To further demonstrate the impact of the improved descriptor, we use our association to drive a complete pipeline for 3D human tracking to estimate spatially stable and temporally coherent 3D skeleton for each tracked person. Compared to the baseline, our method shows significant improvement (5-10X) in 3D skeleton reconstruction, stability, minimizing tracking noise. We believe, for the first time, stable and long duration 3D human tracking is demonstrated

[1] See link for our 960fps motion reconstruction from the original 120fps videos and other results

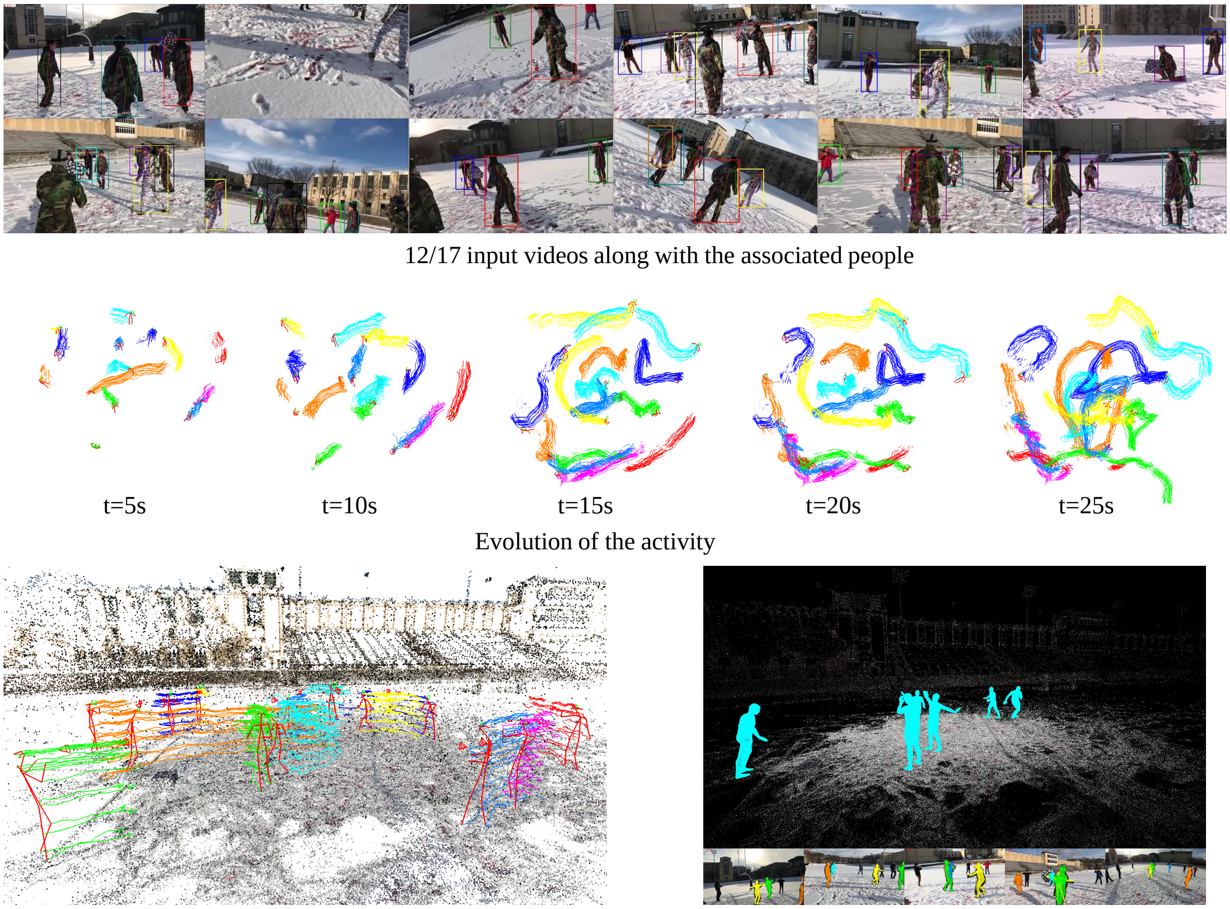


Time-aligned estimated from 8 sub-sampled videos (30fps)
showed on the original (120fps) videos

Insets of the time-aligned images

4D trajectory reconstruction (240fps)

Figure 1.3: The aligned images, estimated from the temporally down-sampled video at 30 fps, are shown for the original video captured at 120fps. As shown in the inset of aligned images, the shadow cast by the folding cloth is well temporally aligned across images. Our motion prior based approach produces plausible reconstruction for the entire course of the action even with relatively low frame-rate cameras (30fps).



12/17 input videos along with the associated people

t=5s

t=10s

t=15s

t=20s

t=25s

Evolution of the activity

Human aware 4D reconstruction

3D mesh fitting to the actors and their 2D reprojections

Figure 1.4: 3D tracks of complex group activity from hand-held smartphone and head-mounted GoPro cameras. A total of 14 people, each associated with a bounding box of unique color, are tracked over time. Our method gives smooth and clean trajectories despite strong occlusion, similar people appearance, and complex motion pattern. The human mesh model is fitted to the estimated 3D skeleton, providing a coarse approximation of the human body shape for later refinement.

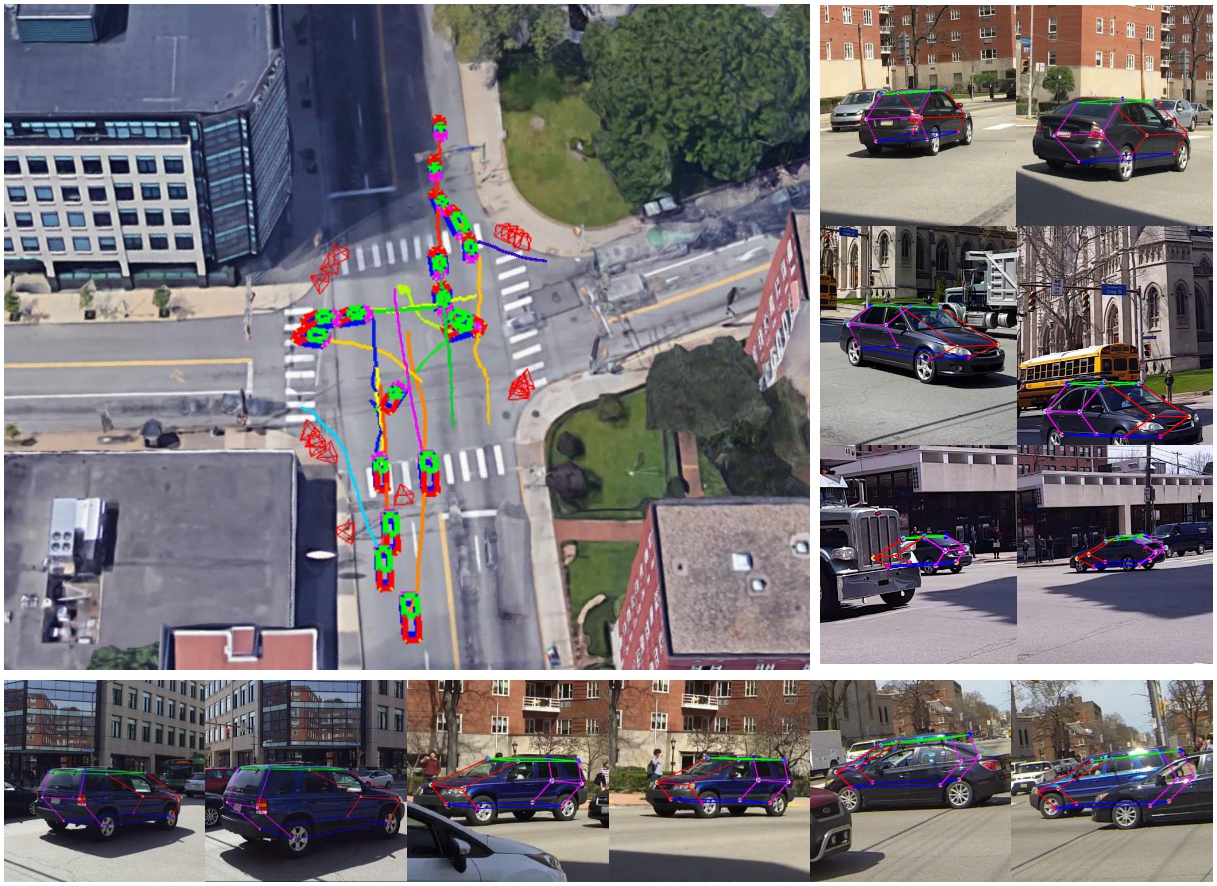


Figure 1.5: Reconstruction of vehicles crossing a busy intersection, making turns, going straight and changing lanes. A subset of vehicle skeletons (3D detector locations) and their 3D trajectories are augmented within the Google Earth view of the intersection. The reconstructions are reprojected into multiple views of two cars (a sedan and an SUV) demonstrating good performance under partial occlusions.

in actual chaotic live group events (see Figure 1.4 for one of those sequences [2]). The existence of spatiotemporally coherent 3D skeleton eases the human morphable mesh model fitting step providing a good initialization for further body shape refinement.

1.2.3 Fusion of Motion and Semantic Priors via Object Triangulation

We formulate a novel fusion algorithm to combine the imprecise but accurate multiview structured semantic keypoint with the sparse but precise single-view unstructured tracks for an object undergoing rigid motion. **No spatial constraints are needed for single-view tracking points. No temporal constraints are needed for multi-view matching points.** No scale ambiguity occurs in merging the motion estimation across cameras.

We apply the method to reconstruction moving cars at a busy intersection captured 21 hand-

[2] See link and `https://www.youtube.com/watch?v=ZDuaJzcLcdE` for more results.

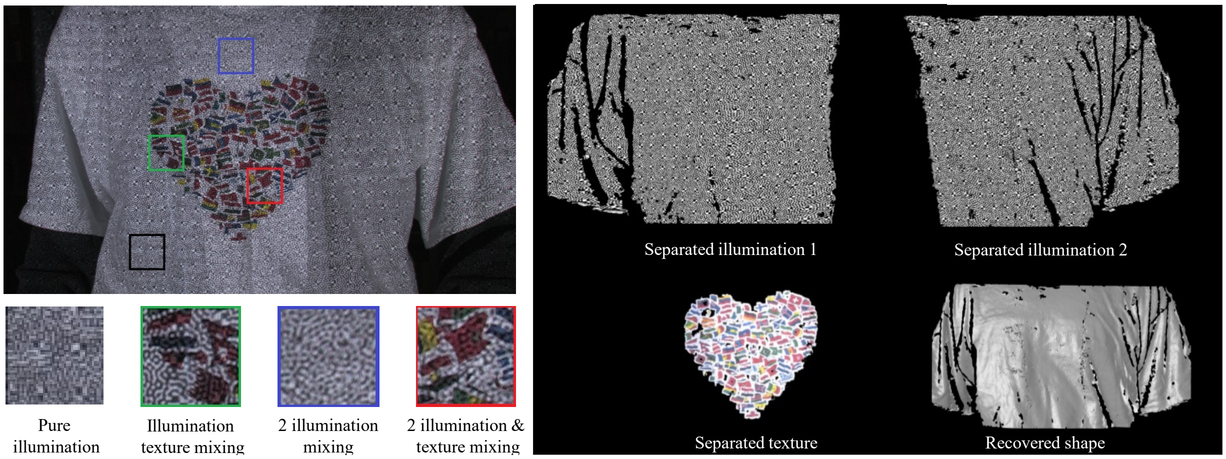


Figure 1.6: When a highly textured shirt illuminated by 2 projectors, the mixture of surface color and the illuminations makes it difficult to find reliable and dense camera-projector correspondences. By exploiting known illumination patterns and a partial observation of the surface color, we can decompose the mixed appearances into its original illuminations and the surface texture. The decomposed illumination facilitated dense camera-projector correspondences for detailed shape reconstruction and the decomposed surface texture allows dense surface motion tracking.

held cameras (see Figure 1.5) [3]. We demonstrate that incomplete and imprecise semantic keypoint detection across multiple views can be fused with precise but sparse single-view tracks of Harris feature to reconstruct moving vehicles even in severe occluded scenarios. The shapes (even sparse) and motions of the vehicles recovered using our approach can be invaluable to traffic analysis, including vehicle type, speed, density, 636 trajectory, and frequency of events such as near-accidents or for (semi-)autonomous vehicles approaching the intersection.

1.2.4 Illumination Decomposition for Active Sensing of Dynamic Shape

Active illumination sensing has a trade-off between acquisition time and resolution of the estimated 3D shapes. Multi-shot approaches can generate dense reconstructions but require stationary scenes. Single-shot methods are applicable to dynamic objects but can only estimate sparse reconstructions and are sensitive to surface texture. We develop a single-shot approach to produce dense shape reconstructions of highly textured objects illuminated by one or more projectors. Our key is an image decomposition scheme that can recover the illumination image of different projectors and the texture images of the scene from their mixed appearances. Our main assumption is that **the object shape deformation is locally smooth such that local deformation propagation is possible**. We focus on three cases of mixed appearances: the illumination from one projector onto textured surface, illumination from multiple projectors onto a textureless surface, or their combined effect. Our method can accurately compute per-pixel warps from the illumination patterns and the texture template to the observed image. The texture template is obtained by interleaving the projection sequence with an all-white pattern. The estimated warps are reliable even with infrequent interleaved projection and strong object deformation. Thus, we

[3] See link and https://youtu.be/RUtK7xRtyns for more results

obtain detailed shape reconstruction and dense motion tracking of the textured surfaces. The proposed method, implemented using a one camera and two projectors system, is validated on synthetic and real data containing subtle non-rigid surface deformations (see Figure 1.6 for one example result) [4].

1.3 Potential Impacts

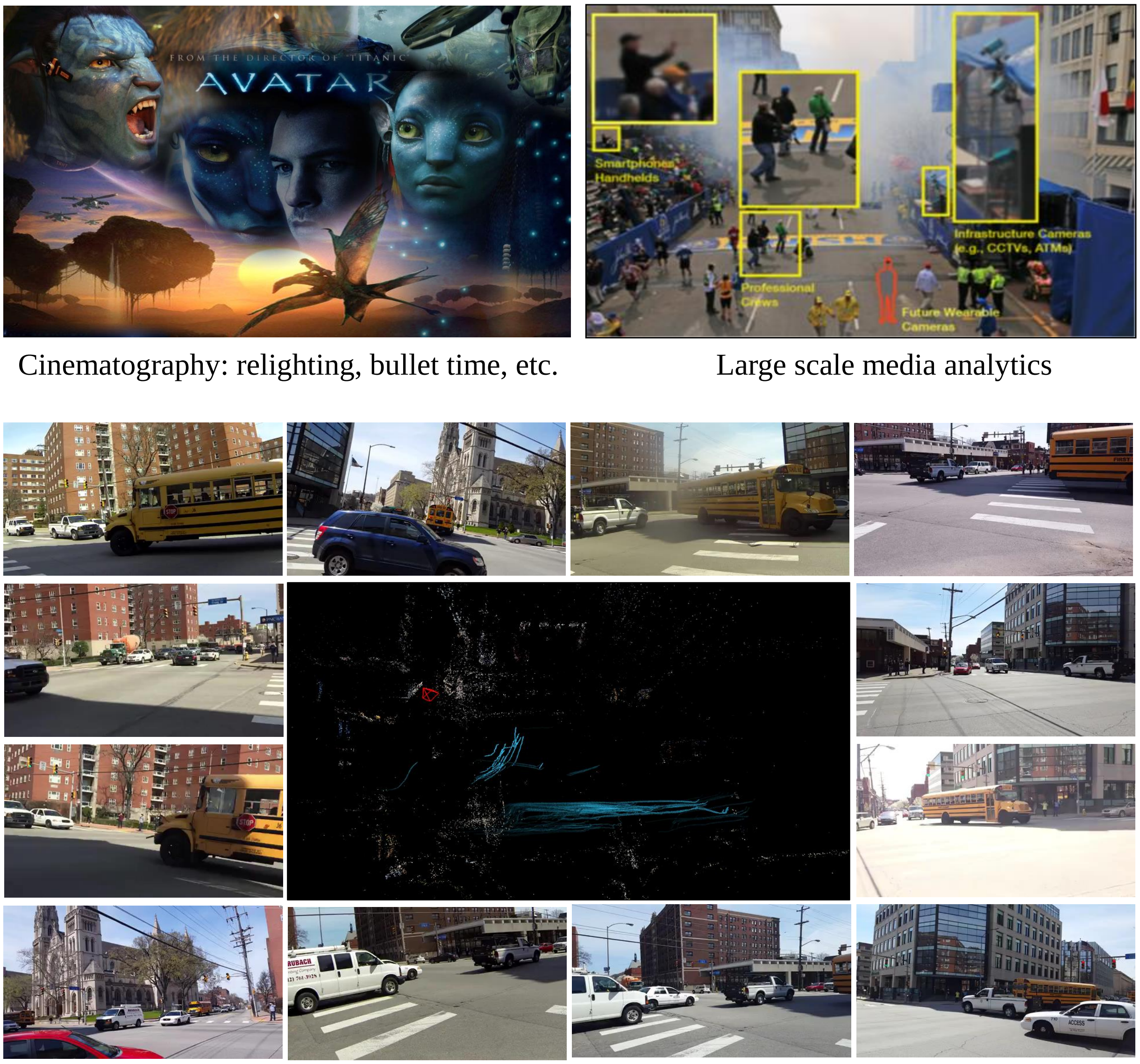
This thesis could have strong impacts on many fields: filming industry, surveillance, media analytics, and crowd psychology understanding. Figure 1.7 illustrates some of these applications.

Cinematography: Many compelling video processing effects can be achieved if accurate per-pixel depth information and the camera spatial and temporal calibrations are known. Handling camera motion, occlusions, or relighting entirely in image space can be extremely labor-intensive while dealing with these issues in scene-space is simple [62, 120, 124, 203, 249]. Our system uses handheld video cameras to reconstruct temporally consistent mesh models of the scene. This enables casual cinematography where everybody can create stunning visual effects only available in blockbuster movies.

Security surveillance: Consider a historical tragedy such as The Boston Bombing in 2013. This event was captured by infrastructure cameras and civilian personal cameras from many different angles. If the information was automatically fused from the vast amount of visual data, both the suspects and the explosive devices could be quickly identified. More importantly, the civilian could be evacuated along the save escape routes that were planned according to the reconstructed dynamic scene model.

Crowd psychology: Understanding human behavior is crucial for human-robot interactions or autonomous driving. For example, when a fully autonomous car travels to a new geodemographic region (India or China vs. USA), it has to understand the social behavior of both the other drivers and the pedestrians in other to operate safely. Our system measures the movements and activities of the people, the cars, etc. in natural settings, which is later used to understand the social norms of that geodemography.

[4] See link for more results


FROM THE DIRECTOR OF TITANIC
AVATAR
Smartphones Handhelds
Professional Crews
Infrastructure Cameras (e.g., CCTVs, ATMs)
Future Wearable Cameras

Cinematography: relighting, bullet time, etc.

Large scale media analytics

Crowd psychology:
how people drive at a busy intersection can be studied by monitoring the car motion

Figure 1.7: The ability to estimate dense and accurate temporally coherent 3D mesh models, the scene illumination and the object ”true” color from handheld cameras in unconstrained settings opens up many exciting applications in the movie industry, AR/VR, media analytics, and human behavior understanding.

1.4 Publication List

The relevant publications to this thesis are as follows:

- Chapter 2: Spatiotemporal Bundle Adjustment
  (1) Spatiotemporal Bundle Adjustment for Dynamic 3D Reconstruction in the Wild, in CVPR 2016
  (2) Spatiotemporal Bundle Adjustment for Dynamic 3D Human Reconstruction in the Wild, in TPAMI (under review)
  Project webpage: link

- Chapter 3: Self-supervised Learning ofSpatiotemporal People Association
  (1) Self-supervised Multiview Person Association and Its Applications, in TPAMI (under review)
  Project webpage: link

- Chapter 4: CarFusion: Fusion of Motion andSemantic Priors via Object Triangulation
  (1) CarFusion: Incorporating Point Tracking and Detection for Multi-view Dynamic Reconstruction, in CVPR 2018
  (2) Occlusion-Net: 2D/3D Occluded Keypoint Localization Using Graph Networks, in CVPR 2019
  Project webpage: link

- Chapter 5: Illumination Decomposition for DenseDynamic Capture in Active Sensing
  (1) Texture Illumination Separation for Single-shot Structured Light Reconstruction, in TPAMI 2016
  Project webpage: link

Chapter 2

Spatiotemporal Bundle Adjustment

When a moving point is observed from multiple cameras with simultaneously triggered shutters, the dynamic 3D reconstruction problem reduces exactly to the case of static 3D reconstruction. The classic point triangulation constraint [140], and the algorithmic edifice of bundle adjustment [211] built upon it, applies directly. Currently, there exists no consumer mechanism to ensure that multiple personal cameras, i.e., smartphones, consumer camcorders, or egocentric cameras, are simultaneously triggered [127]. Thus, in the vast majority of dynamic scenes captured by multiple independent video cameras, no two cameras see the 3D point at the same time instant. This fact trivially invalidates the triangulation constraint.

To optimally solve the dynamic 3D reconstruction problem, we must first recognize all the constituent sub-problems that exist. The classic problems of point triangulation and camera resectioning in the static case are subsumed. In addition, two new problems arise: reconstructing 3D trajectories of moving points and estimating the temporal location of each camera. Second, we must recognize that the sub-problems are tightly coupled. As an example, consider the problem of estimating 3D camera pose. While segmenting out stationary points and using them to estimate camera pose is a strategy that has been used in prior work [168], it ignores evidence from moving points that are often closer to the cameras and therefore provide tighter constraints for precise camera calibration. Imprecise camera calibration and quantization errors in estimating discrete temporal offsets result in significant errors in the reconstruction of moving points[1] [78, 174, 218].

In this chapter, we develop the novel concept of spatiotemporal bundle adjustment that jointly optimizes for all for sub-problems simultaneously. Just as with static 3D reconstruction, where the most accurate results are obtained by jointly optimizing for camera parameters and triangulating static points, the most accurate results for dynamic 3D reconstruction are obtained when jointly optimizing for the spatiotemporal camera parameters and triangulating both static and dynamic 3D points. Unlike traditional bundle adjustment, we recognize the need for a motion

[1] Consider this example: when a person jogging at 10m/s is captured by two cameras at 30Hz, one static and one handheld jittering at 3mm per frame, with the camera baseline of 1m, recording from 4m away. A simple calculation suggests that a näive attempt to triangulate points of the static camera with their correspondences of the best-aligned frame in the other camera results in up to 40 cm reconstruction error.

prior in addition to the standard reprojection cost that jointly estimates the 3D trajectories corresponding to the sub-frame camera temporal alignment. We evaluate several physics-based 3D motion priors (least kinetic energy, least force, and least action) on the CMU motion capture repository [1]. Such joint estimation is most helpful for dynamic scenes with large background/foreground separation where the spatial calibration parameters estimated using background static points are unavoidably less accurate for foreground points.

Direct optimization of the spatiotemporal objective is hard and susceptible to local minima. We address this optimization problem using an incremental reconstruction and temporal alignment algorithm. This optimization framework ensures the proposed 3D motion prior constraint is satisfied. Our algorithm naturally handles the case of missing data (e.g., when a point is occluded in a particular time instant) and scales to many cameras. Thus, we can produce accurate 3D trajectory estimation at much high temporal resolution than the frame rates of the input videos.

While the incremental reconstruction and alignment approach is effective and accurately optimizes the spatiotemporal bundle adjustment problem, it is not efficient. The computational complexity grows quadratically with the number of cameras. We solve this issue by dividing the optimization problem into overlapping groups of cameras with overlapping field of view, each of which is optimized independently using the incremental reconstruction and alignment scheme. These sub-problems are merged and globally optimized in the final pass. Empirically, this approach is at least 20 times faster and has marginal accuracy loss on our datasets.

We apply spatiotemporal bundle adjustment to reconstruct human dynamic scenes. While this algorithm can accurately reconstruct the 3D trajectory of the dynamic points, those points are usually sparse and hence, can be hard to visually interpret. We fit a statistical 3D human body model [141] to the unsynchronized and low frame rate videos to augment the reconstruction. This is in a similar spirit of Multiview stereo to sparse bundle adjustment [79, 188]. Due to the lack of triangulation constraints used in previous work [112], we employ the same physics-based motion prior and the sparse dynamic points to further constrain the fitting. Additionally, since we fit the mesh model to multiple unsynchronized videos simultaneously, unless the frame sequencing is properly estimated, the fitted mesh motion will contain significant jitters and loops. This is the key difference between unsynchronized multiple cameras model fitting and monocular [101, 114, 213] or image-based model fitting [31, 170]. While ideally we should re-optimize camera calibration parameters and 3D points jointly with the body shape and pose coefficients, the extracted semantic cues are often imprecise and hurt the spatiotemporal bundle adjustment. Thus, we fix the estimated spatiotemporal parameters during the shape fitting.

As a demonstration, we build an end-to-end pipeline that takes multiple uncalibrated and unsynchronized video streams and produces a dynamic reconstruction of the event without any constraints. Our framework is applicable to rolling shutter camera via a novel a self-calibration method to estimate the rolling shutter readout speed and the spatial pose of a moving rolling shutter camera. Because the videos are aligned with sub-frame precision, we reconstruct 3D trajectories and human body mesh of unconstrained outdoor activities at much higher temporal resolution than the input videos.

Contributions: (1) We present a spatiotemporal bundle adjustment framework that jointly optimizes four coupled sub-problems: estimating camera intrinsics and extrinsics, triangulating 3D static points, as well as subframe temporal alignment between cameras and estimating 3D trajectories of dynamic points. Key to our joint optimization is the careful integration of physics-based motion priors within the reconstruction pipeline, validated on a large motion capture corpus of human subjects. (2) We devise two efficient computational algorithms to strictly enforce the motion prior using the optimization: the incremental reconstruction and alignment and the divide and conquer algorithm which significantly speeds up the first solver with marginal loss in accuracy. (3) We build an end-to-end framework for human shape reconstruction via model fitting to unsynchronized low framerate video cameras.

2.1 Related Work

In order to reconstruct the dynamic scene, all the cameras must first be temporally aligned. Most approaches compute corresponding trajectories of the moving points and optimize for the time offsets such that the interpolated trajectories satisfy the mutiview constraints such the fundamental matrix [38, 63, 166] or rigidity rank-3 criteria [210]. Other interesting approaches, that exploit the object motion dynamics [224], the relative motions of the rigid objects [80], or the co-occurrence statistic of the spatiotemporal feature points [221, 239], are also explored. However, these methods are either fundamentally limited to the static world assumption, unable to estimate sub-frame alignment, or restricted by the strict assumption of the motion model for the entire observation.

Prior work in dynamic 3D reconstruction has mostly assumed known camera pose and temporal alignment. Under this assumption, Avidan and Shashua posed the problem of trajectory triangulation [17], where multiple noncoincidental projections of a point are reconstructed. Trajectory triangulation is an ill-posed problem and current algorithms appeal to motion priors to constrain reconstruction: linear and conical motion [17]; smooth motion [168, 215]; sparsity priors [261]; low rank spatiotemporal priors [194].

Recent concurrent work has considered the aggregate problem of temporal alignment and dynamic reconstruction jointly [108, 255]. However, the relation between static and dynamic reconstruction was made. We are the first to formally model the four sub-problems jointly to fully take advantage of their inter-connections.

2.2 Motion Prior for Dynamic 3D Capture

Consider the scenario of C video cameras observing N 3D points over time. The relation between the 3D point $X^n(t)$ and its 2D projection $x_c^n(f)$ on camera c at frame f is given by:

$$\begin{bmatrix} x_c^n(f) \\ 1 \end{bmatrix} \equiv K_c(f) \begin{bmatrix} R_c(f) & T_c(f) \end{bmatrix} \begin{bmatrix} X^n(t) \\ 1 \end{bmatrix}, \tag{2.1}$$

where $K_c(f)$ is the intrinsic camera matrix, $R_c(f)$ and $T_c(f)$ are the relative camera rotation and translation, respectively. For simplicity, we denote this transformation as $x_c^n(f) = \pi_c(f, X^n(t))$. The time corresponding to row r_c at frame f is related to the continuous global time t linearly: $f = \alpha_c t + \beta_c + \gamma_c * r_c$, where α_c and β_c are the camera frame rate and time offset, γ_c is the rolling shutter pixel readout speed. For global shutter camera, γ_c is zero.

Image reprojection cost: At any time instance, the reconstruction of a 3D point must satisfy Eq. 2.1. This gives the standard reprojection error S_I, which we accumulate over all 2D points observed by all C cameras for all frames F_c:

$$S_I = \sum_{c=1}^{C} \sum_{n=1}^{N} \sum_{f=1}^{F_c} I_c^n(f) \frac{\|\pi_c(f, X^n(t)) - x_c^n(f)\|^2}{\sigma_c^n(f)}, \tag{2.2}$$

where, $I_c^n(f)$ is a binary indicator of the point-camera visibility, and $\sigma_c^n(f)$ is a scalar, capturing the uncertainty in localizing $x_c^n(f)$ to S_I. Since the localization uncertainty of an image point $x_c^n(f)$ is proportional to its scale [248], we use the inverse of the feature scale as the weighting term for each residual term in S_I.

However, Eq. 2.2 is purely spatially defined and does not encode any temporal information about the dynamic scene. Any trajectory of a moving 3D point must pass through all the rays corresponding to the projection of that point in all views. Clearly, there are infinitely many such trajectories and each of these paths corresponds to a different temporal sequencing of the rays. Yet, the true trajectory must also correctly align all the cameras. This motivates us to investigate a motion prior that ideally estimates a trajectory that corresponds to the correct temporal alignment. The cost of violating such a prior S_M can be then added to the image reprojection cost to obtain a spatiotemporal cost function that jointly estimates both the spatiotemporal camera calibration parameters and the 3D trajectories:

$$S = \arg \min_{\mathbf{X}(t), \{\mathbf{K}, \mathbf{R}, \mathbf{t}\}, \boldsymbol{\alpha}, \boldsymbol{\beta}} S_I + S_M. \tag{2.3}$$

Given multiple corresponding 2D trajectories of both the static and the dynamic 3D points $\{\mathbf{x}_c(t)\}$ for C cameras, we describe how to jointly optimize Eq. 2.3 for the 3D locations $\mathbf{X}(t)$, the spatial camera parameters at each time instant $\{K_c(f), R_c(f), T_c(f)\}$ and the temporal alignment between cameras $\boldsymbol{\beta}$. We assume the frame rate $\boldsymbol{\alpha}$ is known.

2.2.1 Physics-based Motion Priors

In this section, we investigate several forms of motion prior needed to compute S_M in Eq. 2.3. We validate each of these priors on the entire CMU Motion Capture Database [1] for their effectiveness on modeling human motion.

When an action is performed, its trajectories must follow the paths that minimize a physical cost function. This inspires the investigation of the following three types of priors: least kinetic energy, least force[2], and least action [71]. See Figure 2.1 for the formal definition of these priors. In each of these priors, m denotes the mass of the 3D point, g is the gravitational acceleration force acting on the point at height $h(t)$, and $v(t)$ and $a(t)$ are the instantaneous velocity and acceleration at time t, respectively.

Mathematically, the least kinetic energy prior encourages constant velocity motion, the least force prior promotes constant acceleration motion, and the least action prior favors projectile motion. While none of these priors hold for an active system where forces are arbitrarily applied during its course of action, we conjecture that the cumulative forces applied by both mechanical and biological systems are sparse and over a small duration of time, the true trajectory can be approximated by the path that minimizes the costs defined by our motion priors. Any local errors in the 3D trajectory, either by inaccurate estimation of points along the trajectory or wrong temporal sequencing between points observed across different cameras, produce higher motion cost.

Least kinetic motion prior cost: We accumulate the cost over all N 3D trajectories for all time instances T^n:

$$S_M = \sum_{n=1}^{N} \sum_{i=1}^{T^n-1} w_n(t) \frac{m_n}{2} v_n(t^i)^2 (t^{i+1} - t^i), \tag{2.4}$$

where $\gamma_n(t)$ is the weighting scalar and m_n is the point mass, assumed be to identical for all 3D points and set to be 1. We approximate the instantaneous speed $v(t^i)$ at time t^i along the sequence $X^n(t)$ by a forward difference scheme, $v_n(t^i) \approx \|\frac{X^n(t^{i+1}) - X^n(t^i)}{t^{i+1} - t^i}\|$. We add a small constant ϵ to the denominator to avoid instability caused by 3D points observed at approximately same time. Eq. 2.4 is rewritten as:

$$S_M = \sum_{n=1}^{N} \sum_{i=0}^{T^n-1} \frac{w_n(t)}{2} \left\| \frac{X^n(t^{i+1}) - X^n(t^i)}{t^{i+1} - t^i + \epsilon} \right\|^2 (t^{i+1} - t^i), \tag{2.5}$$

Using the uncertainty $\sigma_c^n(f)$ of the 2D projection of 3D point $X_n(t)$, the weighting $w_n(t)$ can be approximated by a scaling factor that depends on the point depth λ and the scale μ, relating the focal length to the physical pixel size, as $w_n = \frac{\mu\lambda}{\sigma_c^n}$. The least force and least action prior costs can be computed similarly.

[2] We actually use the square of the resulting forces.

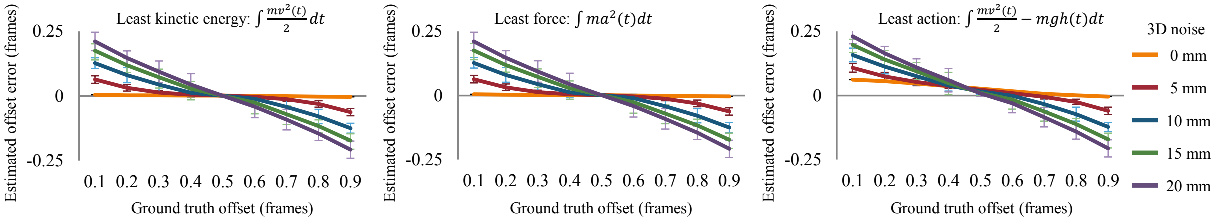


Figure 2.1: Evaluation of the motion priors on 3D motion capture data. The least kinetic energy prior and least force prior performs similarly and both estimate the time offset between two noisy sequences obtained by uniformly sampling a 3D trajectory from different starting times. The least action prior gives biased results even for the no-noise case.

2.2.2 Evaluation on Motion Capture Data

Consider a continuous trajectory of a moving point in 3D. Sampling this continuous trajectory starting at two different times produces two discrete sequences in 3D. We first evaluate how the motion prior helps in estimating the temporal offset between the two discrete sequences. We extend this to 2D trajectories recorded by cameras later. The evaluation is conducted on the entire CMU marker-based Motion Capture Database [1] containing over 2500 sequences of common human activities such as playing, sitting, dancing, running and jumping, captured at 120 fps.

Each trajectory is subsampled starting at two different random times to produce the discrete sequences. 3D zero-mean Gaussian noise is added to every point along the discrete trajectories. The ground truth time offsets are then estimated by a linear search and we record the solution with the smallest motion prior cost. For our test, the captured 3D trajectories are sampled at 12 fps and the offsets are varied from 0.1 to 0.9 frame interval in 0.1 increments.

As shown in Figure 2.1, the least kinetic energy prior and least force prior perform similarly in this setting and both estimate the time offset between the two trajectories well for low noise levels. When more noise is added to the trajectory sequences, the sequencing is noisier. Yet, our motion cost favors correct camera sequencing over closer time offset. This is a desirable property because wrong sequencing results in a trajectory with loops (see Figure 2.5). In contrast, the least action prior gives biased results even when no noise is added to the 3D data.

We further compare the sequencing expressiveness of the least kinetic energy prior to the least force prior for different cumulative framerate. The cumulative framerate is defined as the framerate of the virtual camera consists of all the frames from each camera. As shown in Figure 2.2, the least force prior is more expressive than the least kinetic prior for lower framerate. This is expected since the least force prior captures more local information of the trajectory. However, for modern video cameras, the cumulative framerate easily exceeds 120fps, where the alignment results using either priors are similar. Thus, we only use the least kinetic prior for the remainder of the work. Extension to the least force is straight forward.

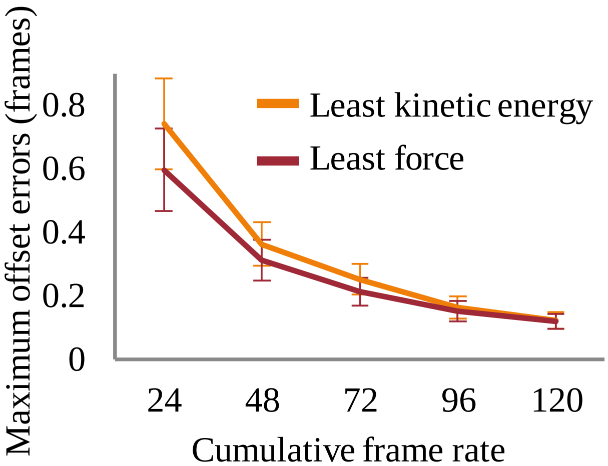


Figure 2.2: Comparison of the sequencing expressiveness between the least kinetic energy prior and the least force prior for different cumulative framerates.

2.3 Spatiotemporal Bundle Adjustment

Unlike traditional bundle adjustment [211], the spatiotemporal bundle adjustment must jointly optimize for four coupled problems: camera intrinsics and extrinsics, 3D locations of static points, temporal alignment of cameras and 3D trajectories of dynamic points. However, direct optimization of Eq. 2.3 is hard because: (a) it requires a solution to a combinatorial problem of correctly sequencing all the cameras and (b) motion prior cost is strongly discontinuous as small changes in time offsets can switch the temporal ordering of cameras. Thus, it is not possible to ensure the satisfaction of the motion prior constraint.

We solve this problem using an incremental reconstruction and alignment approach where the camera is sequentially added to the optimization problem. This algorithm is further speedup by a divide-and-conquer scheme where the groups of cameras are solved independently first and then merged and refined globally using continuous second-order optimization. We initialize temporal alignment and the 3D trajectory of the dynamic points using a geometry (or triangulation constraint) based method [63, 196]. Even though the triangulation constraint is not strictly satisfied, empirically, the estimations provide a good starting point for the incremental reconstruction and alignment.

2.3.1 Incremental Reconstruction and Alignment

Our incremental reconstruction and alignment (IRA) approach adds camera one at a time. For every new camera, a linear search for the best sequencing of this camera with respect to the previous cameras based on the motion prior cost is conducted. Once the sequencing order is determined, we use continuous optimization to jointly estimate all the spatiotemporal camera parameters, and static points and dynamic trajectories. Thank to the linear search step, we can

```
Input: $\{\mathrm{x}_c(t)\}, \{\mathbf{K}', \mathbf{R}', \mathbf{T}'\}, \boldsymbol{\beta}'$
Output: $\{\mathbf{X}(t)_p\}, \{\mathbf{K}, \mathbf{R}, \mathbf{T}\}, \boldsymbol{\beta}$
1. (Sec. 3.1.1) Refine the alignment pairwise
2. (Sec. 3.1.2) Generate prioritized camera list
3. (Sec. 3.1.3) while All cameras haved NOT been processed do
    for All cameras slots do
        Solve Eq. 2.3 for $\{\mathbf{X}_p(t)\}$ and $\boldsymbol{\beta}$
        if No sequencing flipped then
            Record the STBA cost and its solution.
        else
            Discard the solution;
        end
    end
    Accept the solution with the smallest cost
end
4. (Sec. 3.2) Solve Eq. 2.3 for $\{\mathbf{X}(t)_p\}, \{\mathbf{K}, \mathbf{R}, \mathbf{T}\}, \boldsymbol{\beta}$
```

Algorithm 1: Incremental reconstruction and alignment.

enforce the motion prior constraint strictly without any discontinuities due to incorrect time ordering of cameras. We summarize this method in Algorithm 1.

Temporal alignment of two cameras: We refine the initial guess by optimizing Eq. 2.3. However, just as in point triangulation, the 3D estimation from a stereo pair is unreliable. Thus, we simply do a linear search on a discretized set of temporal offsets and only solve Eq.2.3 for the 3D trajectories. The offset with the smallest cost is taken as the sub-frame alignment result. We apply this refinement to all pair of cameras.

Which camera to add next? As in incremental SfM [75, 196], we need to determine the next camera to include in the calibration and reconstruction process. For this, we create a graph with each camera as a node and define the weighted edge cost between any two cameras i^{th} and j^{th} as

$$E_{ij} = \sum_{k=1, k\neq i,j}^{C} S_{ij} \frac{|t_{ij} + t_{jk} - t_{ik}|}{N_{ij} B_{ij}}, \tag{2.6}$$

where t_{ij}, N_{ij}, B_{ij}, and S_{ij} are the pairwise offset, the number of visible corresponding 3D points, the average camera baseline, and the spatiotemporal cost evaluated for those cameras, respectively. Intuitively, $|t_{ij} + t_{jk} - t_{ik}|$ encodes the constraint between the time offsets among a camera triplet, and $N_{ij} B_{ij}$ is a weighting factor favoring the camera pair with more common points and larger baseline.

Similar to [63, 221], a minimum spanning tree (MST) of the graph is used to find the alignment of all cameras. We use the Kruskal MST, which adds nodes with an increasing cost at each step. The camera processing order is determined once from the connection step of the MST procedure.

Estimating the time offset of the next camera: We temporally order the current processed cameras and insert the new camera into possible time slots between them, followed by a non-linear optimization to jointly estimate all the offsets and 3D trajectories. Any trials where the relative ordering between cameras change after the optimization are discarded, ensuring that the motion prior is satisfied. The trial with the smallest cost is taken as the temporal alignment and 3D trajectories of the new set of cameras.

2.3.2 Divide and Conquer

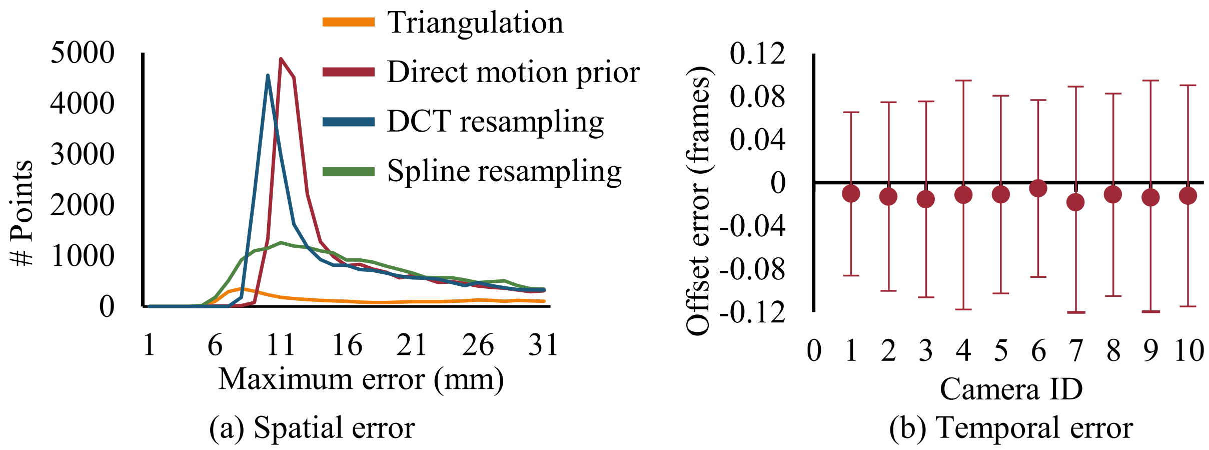
While the incremental reconstruction and alignment approach offers a tractable solution for optimizing Eq .2.3, its complexity increases quadratically with the number of cameras. For every new camera, we must optimize Eq. 2.3 $C-1$ times, where C is the number of the camera being processed, to determine the sequencing order with the least motion cost. To address the computational efficiency issue, we propose a divide and conquer approach to speed up to solver with minimal loss in reconstruction accuracy. This algorithm is based on the observation that the temporal alignment becomes stable after a small number of cameras is processed (4 cameras in our all of our experiments).

This algorithm proceeds in three stages. First, we form the camera groups with large co-visibility with them by creating a skeleton graph [197] of camera graph built-in Sec. 2.3.1. Here, each group is taken as two camera nodes in the skeleton graph and their connected cameras of the camera graph. We purposely let the overlapping groups share two cameras to better detect and discard temporal inconsistency when merging all the groups together. Second, we process each camera group independently processed using the incremental reconstruction and alignment approach. For every pair of inconsistency groups detected, these groups are merged and re-processed using the first approach. Third, we aggregate to temporal alignment parameters from all groups into a common timeline and optimize all cameras jointly for the spatiotemporal calibration parameters and the 3D position of the static and dynamic points.

2.3.3 Motion Resampling via Discrete Cosine Transform

Note that Eq. 2.5 approximates the speed of the 3D point using finite-difference. While this approximation allows better handling of missing data, the resulting 3D trajectories are often noisy. Thus, we further fit the weighted complete DCT basis function to the estimated trajectories. Our use of DCT for resampling is mathematically equivalent to our discrete motion prior [200] and is not an extra smoothing prior. For the uniform DCT resampling, the least kinetic energy prior cost can be rewritten as:

$$S'_M = \sum_{n=1}^{N} E^{n\top} W^n E^n \Delta t, \tag{2.7}$$



(a) Spatial error (b) Temporal error

Figure 2.3: Evaluation of the motion priors on the Motion Capture database for simultaneous 3D reconstruction and sub-frame temporal alignment. (a) Spatially, the trajectories estimated using the motion prior achieves higher accuracy than generic B-spline trajectories basis. Frame level alignment geometric triangulation spreads the error to all cameras and estimates less accurate 3D trajectories. (b) Temporally, our motion prior based method estimates the time offset between cameras with sub-frame accuracy.

where E^n is the DCT coefficient of the 3D trajectory n, W^n is a predefined diagonal matrix, weighting the contribution of the bases, and Δt is the resampling period. The 3D trajectory $X^n(t)$ is related to E^n by $X^n(t) = B^{n\top} E^n$, where B^n is a predefined DCT basis matrix. The dimension of B^n and W^n depend on the trajectory length. We replace the trajectory $X^n(t)$ by $B^{n\top} E^n$ and rewrite Eq. 2.3 as:

$$S = \underset{E}{\operatorname{argmin}} \, \lambda_1 S'_I + \lambda_2 S'_M, \tag{2.8}$$

where λ_1 and λ_2 are the weighting scalars and S'_I is the reprojection error computed using the resampled trajectories. While applying resampling to the incremental reconstruction loop can improve the 3D trajectories and the temporal alignment, it requires inverting a large and dense matrix of the DCT coefficients, which is computationally demanding. Thus, we only use this scheme as a post-processing step.

2.3.4 Evaluation on Motion Capture Data

We validate the proposed spatiotemporal bundle adjustment on synthetic data generated from the CMU Motion Capture database [1]. We sequentially distribute the ground truth trajectory, captured at 120 fps, to 10 global shutter perspective cameras with a resolution of 1920x1080 and 12fps. All cameras are uniformly arranged in a circle and capturing the scene from 3 m away. We randomly add 3000 background points arranged in a cylinder of radius 15 m centered at dynamic points. The relative offsets, discretized at 0.1 frames, are randomly varying for every sequence. None of the offsets generates cameras observing the 3D points synchronously. We assume that the initial offsets are within 2-3 frames accurate, which is the case for most geometry-based

	Incremental				Divide and conquer	
	Geometry	Spline	MP	R	MP	R
Static	3.45	2.93	2.54	2.41	2.56	2.41
Dynamic	17.8	1.68	0.85	0.74	0.89	0.75

Table 2.1: The reprojection error for the entire CMU Mocap dataset. MP is the results for all cameras without resampling (R).

alignment methods. We also add zero mean Gaussian noise of 2 pixels standard deviation to the observed 2D trajectories. For the divide and conquer approach, we split the camera into 3 groups of 4 nearby cameras each.

The reconstruction and alignment errors are summarized in Figure2.3 and Table ??. Spatially, the point triangulation of the frame-accurate alignment propagates the error to all cameras and gives the worst result. Trajectories reconstructed using 3D cubic B-spline basis gives much smaller error than the point triangulation. However, it also arbitrarily smooths out the trajectories and is inferior to our method. While both the direct motion prior and DCT resampling have similar mean error (direct: 6.6 cm, DCT: 6.5 cm), the former has a larger maximum error due to the noise in approximating the velocity. Temporally, our method can estimate ground truth offset at sub-frame accuracy with low uncertainty. The divide and conquer approach is quantitatively equally accurate with the incremental reconstruction and alignment method while being approximately 30 times faster. We also observe that this approach produces no temporal inconsistency between camera groups for all trials.

2.4 Dynamic Reconstruction of Human Body

While the spatiotemporal bundle adjustment can accurately estimate the 3D trajectory of dynamic points, the number of such trajectories is sparse and is insufficient to fully visualize the scene content. In this section, we leverage recent advances in semantic understanding to estimate the 3D human body shape and pose by fitting a statistical body model to the observed semantic cues. More specifically, the optimization is constrained by the 2D human body part segmentation from each image, the body anatomical keypoints, the sparse 3D trajectories recovered using sparse spatiotemporal bundle adjustment, the least kinetic energy motion prior, and the spatial pose and shape priors. Due to the imprecision in localizing the semantic cues, we fix camera parameters and only optimize for the 3D human body.

2.4.1 Statistical Body Model

We use the SMPL model [141], a linear blend shape model of body shape that can be deformed via linear blend skinning, to present the human body. This model $V(\Omega, \Phi, \Gamma)$ is a triangle mesh, composed of 6890 vertices, and is parameterized by gender, 10 identify shape coefficients Ω, 24 joints Φ, presented using angle-axis to model the relative rotation between body parts, and a body root translation Γ. The 3D location of the body joints X corresponding to a particular pose configuration are given by the joint regressor R, a matrix presenting a sparse linear combination of surface vertices around the joint, $X = RV(\Omega, \Phi, \Gamma)$. In our case, R is slightly different from [141] as our joints are defined according to the OpenPose keypoints format [36].

2.4.2 Body Alignment Objective

We fit the SMPL model to the observed semantic body part and keypoint detectors and the sparse 3D trajectory by optimizing the following cost:

$$S = \arg \min_{\Omega, \mathbf{\Phi}(t), \mathbf{\Gamma}(t)} \lambda_{S_J} S_J + \lambda_{S_T} S_T + \lambda_{S_S} S_S + \lambda_{S_M} S_M + \lambda_{S_B} S_B, \tag{2.9}$$

where $\{S_J, S_S\}$ are the image evidence cost, capturing the body joint, and body silhouette, respectively, S_T is the cost induced by the sparse 3D trajectory on the body, S_M is the least kinetic motion prior loss imposed on the 3D body joints, and S_B is the body pose and shape prior cost. We normalize each of these costs by the number of their contributing residuals before applying the weights $\lambda_{S_K}, \lambda_{S_T}, \lambda_{S_S}, \lambda_{S_M}, \lambda_{S_B}\}$ to each cost. The definition of these cost functions are described as follow.

Body keypoint alignment cost: This cost function aims to reduce the difference between the projected SMPL keypoints and the detected keypoints, and is written as

$$S_J = \sum_{c=1}^{C} \sum_{f=1}^{F_c} \sum_{j=1}^{J} I_c^j(f) \sigma_c^j(f) \rho\Bigg(\frac{||\pi_c(f, X^j) - x_c^j(f)||}{\sigma_J} \Bigg), \tag{2.10}$$

where σ_J is a scalar approximating the uncertainty in detecting the body keypoints, X_j is a joint among the set $X(t) = RV(\Omega, \Phi(t), \Gamma(t))$, $I_c^j(f)$ is a binary indicator of the point visibility, and $\sigma_c^j(f)$ is the confidence of the keypoints detected by OpenPose [36].
Sparse 3D trajectory constraints: This cost function penalizes variance of the distances $L(.,.)$ between the point X^n along the 3D trajectories to the two nearest joints within the same body part, $X^j \in 2NN(X^n)$, and is expressed as

$$S_T = \sum_{c=1}^{C} \sum_{f=1}^{F_c} \sum_{n=1}^{N} \sum_{X^j} I_c^n(f) \rho\Bigg(\frac{||L(X^j, X^n) - \overline{L}(X^j, X^n)||^2}{\sigma_T} \Bigg), \tag{2.11}$$

where σ_T is a scalar capturing the uncertainty in estimating the location of point along the sparse 3D trajectory, $I_c^n(f)$ is a binary showing the availability of $X^n(t)$ on to camera c at frame f, and l is the Euclidean distance between two points. Here, we add the extra variable $\overline{L(.,.)}$ as the average distance between points over the entire course of motion, to the optimization. Despite the non-rigid body and cloth deformation, we expect the deviation of the instantaneous point distance to be close to its average over time. Empirically, we observe that this loss function improves tracking robustness especially in case of semantic detector failure when body parts corresponding different person are grouped together.

Silhouette alignment cost: This cost function discourages any projected body vertices not contain inside the detected body segmentation and is expressed as

$$S_S = \sum_{c=1}^{C} \sum_{f=1}^{F_c} \sum_{v \in V(\Omega, \Phi(t), \Gamma(f))} \rho\Bigg(\frac{DT_c(f)(\pi_c(f, v))}{\sigma_P} \Bigg), \tag{2.12}$$

where $DT_c(f)$ is the distance transform of the body part segmentation in camera c at frame f. $DT_c(f)$ is zeros for points inside projected mesh and is equal to the distance between the projected mesh vertex and its nearest point on the segmentation boundary contour otherwise. This loss is particularly useful for occluded body parts. We use DensePose [11] to compute body segmentation.

Motion prior cost: Due to the lack of triangulation contraints in unsynchronized camera setup, the motion prior is key to for accurate dynamic human body estimation, especially for occluded body parts under fast body motion and observed by low frame rate cameras. Similar to sparse 3D trajectory estimation, we use the least kinetic motion prior with forward differentiation approximation of the velocity imposed on the 3D body joints X^j to constraint to body motion over its entire observation duration T

$$S_M = \sum_{t \in T} \sum_{X_j} \Bigg\| \frac{X^j(t^(i+1)) - X^j(t^i)}{t^{i+1} - t^i + \epsilon} \Bigg\|^2 \frac{(t^{i+1} - t^i)}{\sigma_M}, \tag{2.13}$$

where σ_M is the expected variation in human instantaneous velocity. We set σ_M differently for different joints.

Body shape prior cost: The SMPL model is designed to fit semi-naked people, whereas we are interested in measuring people who are wearing arbitrary clothes and accessories. Moreover, the semantic detector that the model is being fit to could produce erroneous estimation, especially for occluded body parts. Such limitation of the body model and the incorrect correspondences can significantly affect both the body shape and pose parameters. We mitigate such defects by placing a zero-mean standard normal distribution over the pose $\Phi(f)$ (favors mean pose) and shape Ω parameters as

$$S_B = \lambda_\Omega \mathcal{N}(\Omega) + \lambda_\Phi \sum_{c=1}^{C} \sum_{f=1}^{F_c} \mathcal{N}(\Phi(f)), \tag{2.14}$$

where $\{\lambda_\Omega, \lambda_\Phi\}$ are weighting scalar between the residuals.

2.4.3 Optimization Strategy

Due to the complexity of the human body pose, a direct optimization of Eq. 2.9 converges slowly and often fails to produce accurate body fitting. We solve the problem in three stages. (1) full sequence spatiotemporally coherent 3D human skeleton estimation (2), per-time-instance human model fitting to the skeleton, and (3) window-based accurate and temporally coherent body pose and shape fitting. These stages are described below.

Stage 1: Coherent 3D human skeleton estimation For each person in the scene, we wish to estimate a temporally and physically consistent human skeleton model for the entire sequence. This is achieved by minimizing an energy function that combines of the reprojection cost on the detected semantic keypoint of Eq. 2.10, the least kinetic motion prior cost of Eq. 2.13, and the prior on human skeleton written as

$$S_b = \sum_{t \in T} \sum_{q \in Q} \left(\frac{L(q,t) - \overline{L}(q)}{\sigma_L} \right)^2, \tag{2.15}$$

$$S_{lr} = \sum_{t \in T} \sum_{(l,r) \in S} \left(\frac{L(l,t) - L(r,t)}{\sigma_S} \right)^2, \tag{2.16}$$

where Q is the set of keypoint connectivity within all rigid body parts, S denotes the set of joints of the corresponding left and right limb, $\{\sigma_L, \sigma_S\}$ captures the precision of the symmetry and left-right constancy constraints. These priors enforce the left-right symmetry of the body bone length and penalize large changes between the bone length estimated each time instance and average bone length $\overline{L}$ over the entire sequence. As in Sec. 3, the initial 3D skeleton is obtained by geometric triangulation. We weight the costs equally and optimize them for the entire sequence at once. Lastly, we resample the 3D joint along the temporal axis using DCT to fill in the missing skeleton due to occlusion.

Stage 2: Per-instance human model fitting Given the coherent 3D skeleton at all time instances, we fit the SMPL model to the skeleton independently at each time instances in order to gain resilience to fitting failure. This is done by optimizing a cost function composes of the 3D-3D SMPL joint to our skeleton cost, and body shape and pose prior, as in Eq. 2.14. The 3D-3D cost is written as

$$S_{3D-3D} = \sum_{t \in T} \sum_{j=1}^{J} \left(\frac{||X^j - \tilde{X}j||}{\sigma_{3D}} \right)^2, \tag{2.17}$$

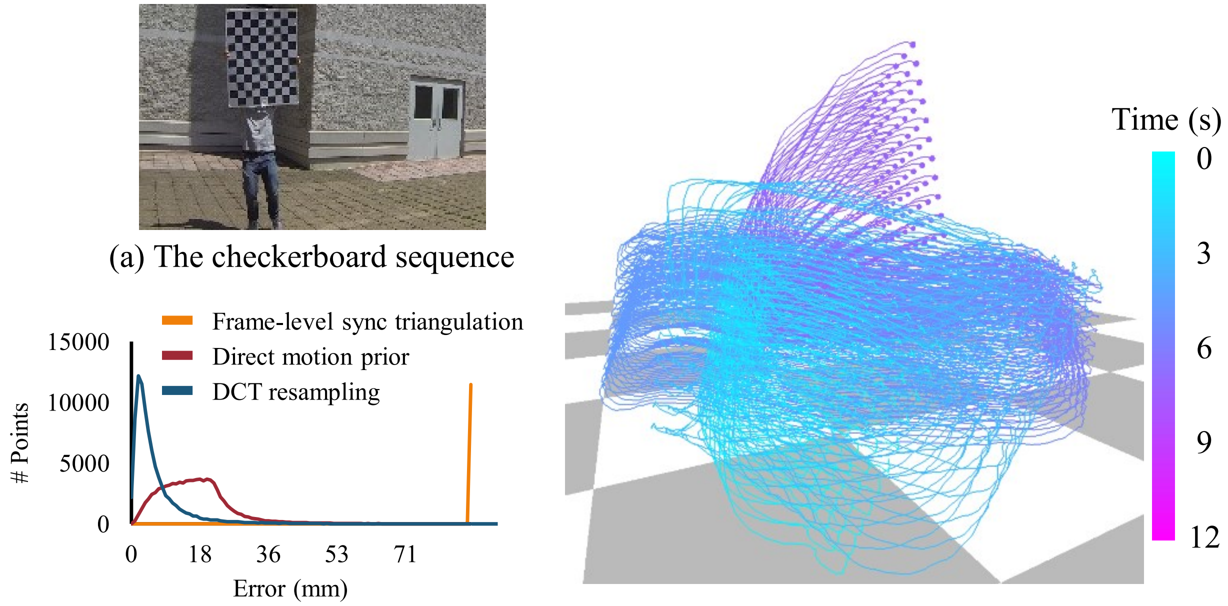
where $\tilde{X}$ is the estimated skeleton at stage 1 and σ_{3D} is the expected noise in 3D estimation. Empirically, we found this approach is fast and gives good approximation for the last stage.

Stage 3: Window-based human model fitting We optimize Eq. 2.9 for the SMPL body shape and pose in overlapping windows. For the overlapping region, we fix the optimized parameters to those of the previous windows to ensure consistent body shape estimation.

2.5 Analysis on Real Handheld Camera Data

2.5.1 Data Preprocessing

We create an end-to-end system that takes video streams for multiple temporally unaligned and spatially uncalibrated cameras and produces the spatiotemporal calibration parameters as well as the 3D static points and dynamic trajectories. We show the results for 3 scenes: checkerboard, jump, and dance, captured by either smartphone or GoPro Hero 3 cameras, all of which are rolling shutter camera. We quantify the error in 3D trajectory estimation and effect of sub-frame



(a) The checkerboard sequence

(c) Error histogram of the reconstructed checkerboard

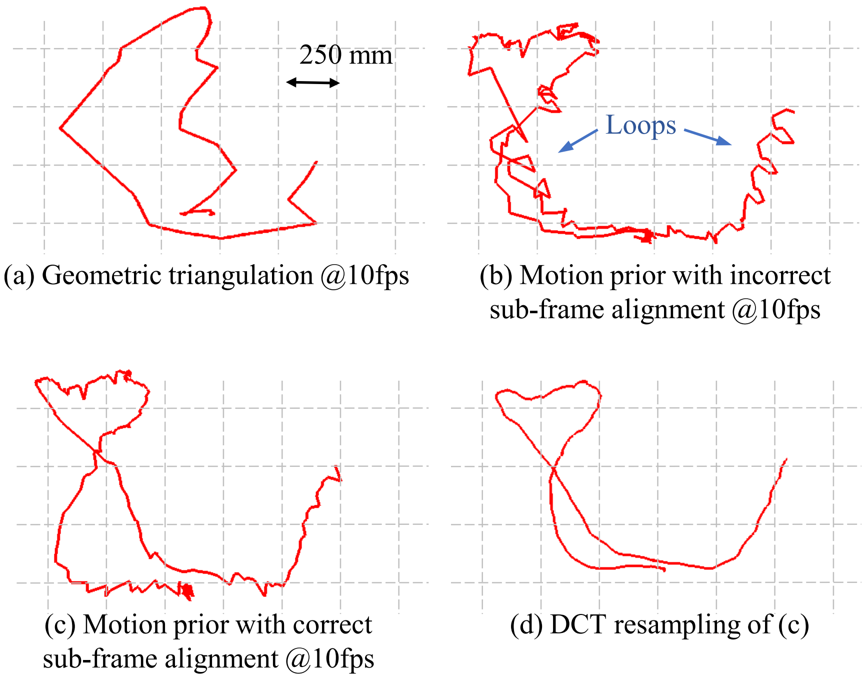
(b) The 3D trajectory of the checkerboard corners

Figure 2.4: Accuracy evaluation of the checkerboard corner 3D trajectories. While the reconstruction is conducted independently at every corner, collectively, the estimated 3D trajectories assemble themselves in the grid-like configuration. Our methods produce trajectories with significantly smaller error than naive geometric triangulation.

alignment using the Checkerboard sequence, captured by 7 Gopro at 1920×1080 and 60 fps. The Jump sequence, captured by 8 Gopro at 1280×720 at 120 fps, is served to demonstrate our ability to handle fast motion using low framerate cameras. The Dance scene, captured by five iPhone 6 and six Samsung Galaxy 6 at 1920×1080 and 60 fps, showcases the situation where the static background and dynamic foreground are separated by a large distance. For all scenes, we downsample the framerate to simulate faster motion, which invalidates the geometry constraints for unsynchronized cameras and stresses the essence of the motion prior: 10 fps, 30 fps, and 15 fps for Checkerboard, Jump, and Dance, respectively.

3D corpus and initial camera pose estimation: We track SIFT features using affine optical flow [18] and sample keyframes, defined as frames where the number of tracked features drop 40% from the last keyframe, from all videos. These keyframes are passed to a SfM pipeline [187, 229] to build the 3D corpus of the scene. We register other frames to this corpus using the r6P algorithm and refine their parameters using the Cayley transform model [9]. No temporal regularization is performed during the registration to preserve the abrupt a camera motion frequently observed due to the camera holder's footstep.

Rolling shutter scanning speed estimation: Consider a moving camera observing static features. Geometrically, this camera can also be viewed as being static and observing moving features. We estimate the camera rolling shutter readout speed by assuming the 3D location of



(a) Geometric triangulation @10fps

(b) Motion prior with incorrect sub-frame alignment @10fps

(c) Motion prior with correct sub-frame alignment @10fps

(d) DCT resampling of (c)

Figure 2.5: Effect of accurate sub-frame alignment for the 3D trajectory estimation. (a) Point triangulation of frame accurate alignment gives large reconstruction error and creates different 3D shape with respect to other methods. (b) Incorrect sub-frame alignment generates 3D trajectory with many loops. (c) Trajectory estimated from correct sub-frame alignment is free from the loops. (d) Using DCT resampling for (c) gives smooth and shape preserving 3D trajectory.

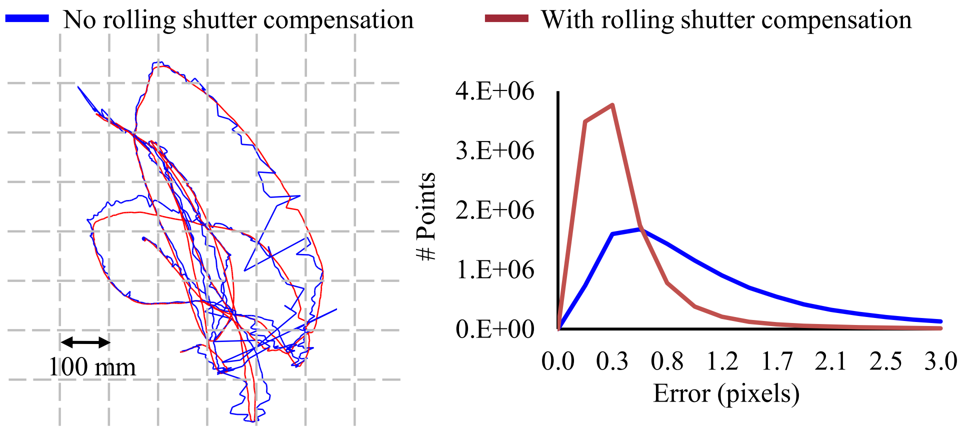
these moving features also obeys the least kinetic motion prior for the duration of 1 frame. Denote $X_v(f), X_v(f+1)$ as the virtual location of the static feature captured exactly at the first row of of frame $f, f+1$, respectively, and $X_(t)$ is the true location of the same feature observed in the image, $t \in [t(f), t(f+1)]$. Under the least kinetic assumption (e.g., constant velocity) and vertical rolling shutter readout, we can present the 3D location of the observed feature as

$$X(t) = X_v(f) + \gamma \frac{r}{h}(X_v(f+1) - X_v(f)), \tag{2.18}$$

where r is the image row of the observed feature and h is the image height. Using this representation of X to optimize Eq. 2.2, we can estimate the rolling shutter readout speed γ for each camera.

Corresponding 2D trajectory generation: We detect and match SIFT features across cameras at evenly distributed time instances. We discard matches with low gradient score and track the remaining points both forward and backward in time using affine template matching. The backward-forward consistency check is used to discard erroneous optical flow during tracking. Finally, we check for the appearance consistency between patches of the first and the last frame using Normalized Cross Correlation and remove the entire trajectory if the score is below 0.8.

Trajectory classification: We exploit the fact that triangulation based methods work for static points but produce large errors for dynamic points in order to identify 2D trajectories of dynamic



(a) Ground projection of the camera trajectory

(b) Reprojection error

Figure 2.6: Analysis of the spatial camera calibration for the Checkerboard sequence with different camera models.

	No γ		With γ	
	Static-Dynamic		Static-Dynamic	
Checkerboard	0.70	1.52	0.67	1.21
Jump	0.61	1.55	0.59	1.34
Dance	0.85	2.38	0.82	2.13

Table 2.2: Effect of modeling the rolling shutter readout on the reconstruction accuracy. While the temporal sequencing between cameras is still correct (due to the artificial down-sampling of the framerate), modeling γ results in smaller the reprojection error. Inaccurately reconstructed dynamic points also (slightly) negatively affect the reconstruction of static points.

points. This is done using these two heuristics: (1) the reprojection error of a static point should be small regardless of which camera frame it is triangulated from. We randomly sample frames along the 2D trajectory to triangulate and consider the 2D trajectory as belonging to a static point if the reprojection threshold is smaller than 3 pixels for more than 80% of the time. (2) the reprojection error of a dynamic point forms a steep valley as the time offset passes by its true value. We reject any set of trajectories as belonging to a dynamic point if the minimum of the cost valley is not smaller than 80% of the average cost.

2.5.2 Sparse spatiotemporal bundle adjustment

We first evaluate the effect of properly modeling the rolling shutter pose and its readout speed to the reconstruction. As shown in Figure 2.6, spatial modeling the rolling shutter produces significantly more stable camera trajectory and lower reprojection error. As showed in Table 2.2, although for artificially down sample frame rate videos, the rolling shutter readout becomes less significant given the artificially lengthen frame duration, the reconstructions with γ modeled are consistently more accurate. Theoretically, while the reconstruction of static points is not affected

	Geometry			
	#Trajectory	Avg samples per trajectory	RMSE (pixels) Static – Dynamic	
Checkberboard	88	179.8	0.67	6.59
Jump	717	36.4	0.59	1.91
Dance	577	22.3	0.82	5.23

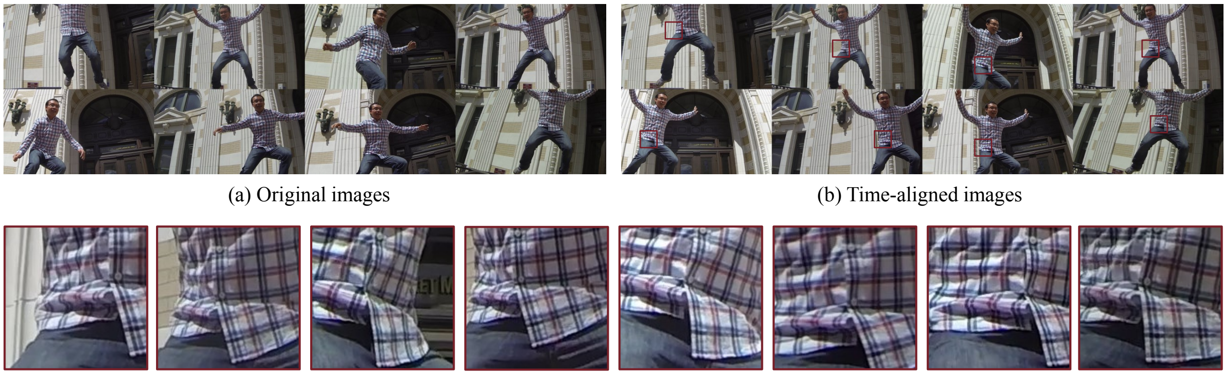
	Motion prior									
	#Trajectory	Avg samples per trajectory	RSME^1 (pixels) Static – Dynamic		RSME^{1*} (pixels) Static – Dynamic		RSME^2 (pixels) Static – Dynamic		RSME^{2*} (pixels) Static – Dynamic	
Checkberboard	88	1023.0	0.67	1.21	0.65	1.15	0.68	1.2	0.67	1.16
Jump	3231	127.8	0.59	1.34	0.5	1.26	0.59	1.36	0.51	1.6
Dance	4105	216.4	0.82	2.12	0.85	1.71	0.83	2.14	0.87	1.72

Table 2.3: Reconstruction accuracy comparison between geometric triangulation and our proposed method. RMSE^1 and RMSE2 are the results obtained by the incremental reconstruciton and alignment and divide and conquer approaches, repetitively. The $*$ denotes the results after resampling. Both approaches are equally accurate and the divide and conquer scheme is at least 10 times faster. Please see the text for more details.

by γ, inaccurate reconstruction of dynamic points negatively affect the camera calibration parameters, which in turn affects the static points. For all results represented below, both the rolling shutter camera pose and readout speed are employed.

Table 2.3 gives the complete quantitative evaluation on three video sequences in terms of (a) re-projection error in pixels for both stationary and dynamic points, (b) number and average length (time) of the 3D trajectories created using points from multiple views. Points with reprojection error exceeding the threshold are discarded. Noticeably, our proposed method produces several folds more trajectories, longer average trajectory length, and less reprojection error than geometry approach. For the checkerboard sequence, since the correspondences are known, its 3D points are intentionally not discarded. The resampling scheme consistently and noticeably further reduces re-projection error for all scenes. Similar to the analysis on synthetic data, the divide and conquer scheme is as accurate as the incremental reconstruction and alignment approach but is at least 10 times faster (40 times for Dance sequence).

Checkerboard scene: Since the ground truth location of the board is unknown, we quantify the reconstruction accuracy by measuring the deviation from the planar configuration for all the checkerboard corners. We reconstruct each corner independently. As depicted in Figure 2.4, the reconstruction using geometric triangulation is at least 80 mm inaccurate. Conversely, most 3D corners estimated from our method have much smaller error (direct motion prior: 33 mm, DCT: 15 mm). Visually, the estimated trajectories using the method assemble themselves in the grid-like configuration of the physical board.



(a) Original images

(b) Time-aligned images

(c) Insets of the time-aligned images

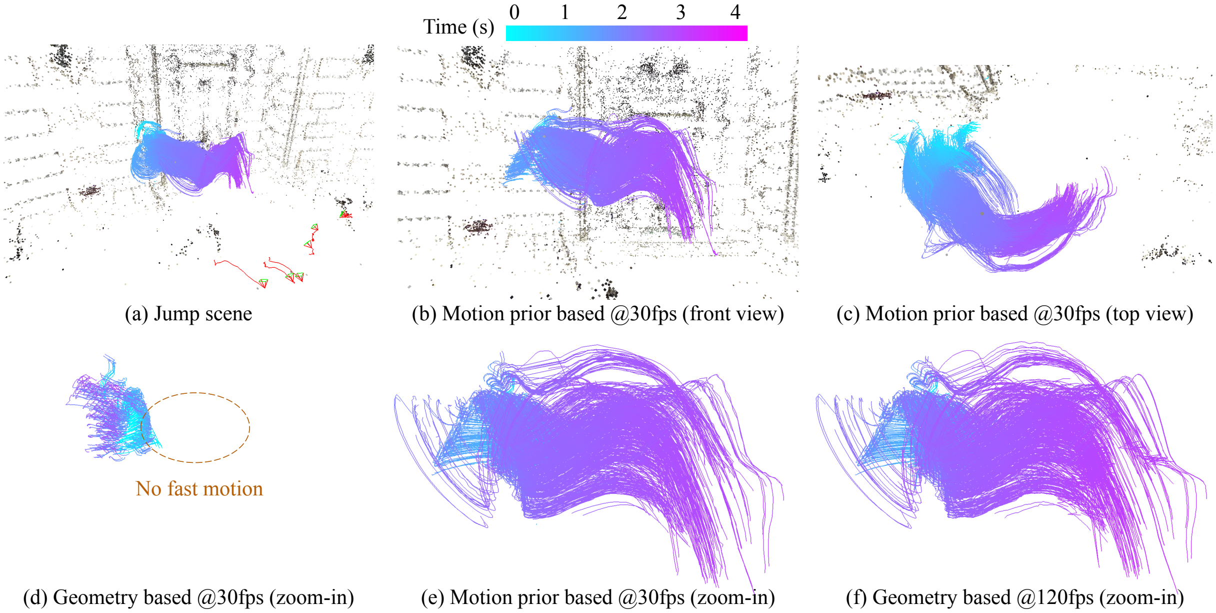
Figure 2.7: Temporal alignment. (a) Original unaligned images. (b) Our aligned images, estimated from temporally down-sampled video at 30 fps, are shown for the original video captured at 120 fps. (c) Inset of aligned images. The shadow cast by the folding cloth is well temporally aligned across images. This figure is best seen electronically.

Figure 2.5 shows the effect of accurate sub-frame alignment on the trajectory reconstruction. Due to the fast motion, geometry-based method produces trajectory with much different shape than the motion prior based method. We artificially alter the sub-frame of the offsets to create wrong frame sequencing between different cameras and optimize Eq. 2.3 for the trajectory. This results in trajectories with many small loops, a strong cue of incorrect alignment. Conversely, our reconstruction with correct time alignment is free from the loops. Our final result, obtained by DCT resampling, gives smooth and shape-preserving trajectories.

Jump scene: To visually evaluate the alignment, we scale the estimated offsets to show the alignment on the original footage at 120fps (see Figure 2.7). Notice that the shadow cast by the folding cloth are well aligned across images. Figure 2.8 shows our estimated trajectories for all methods. The point triangulation of frame-accurate alignment fails to reconstruct the fast action happening at the end of the action. Conversely, our method produces plausible metric reconstruction for the entire action even with relatively low frame-rate cameras. Due to the lack of ground truth data, we compare our reconstruction with the point triangulation using 120 fps videos, where few differences between the two reconstructions are observed.

Dance scene: We estimate per-frame camera intrinsic to account for the auto-focus function of smartphone cameras. Figure 2.9 shows our trajectory reconstruction results. Our method reconstructs fast motion trajectories (jumping), longer and higher temporal resolution trajectories than point triangulation results at 15 fps. Since we discard many short 2D trajectories (thresholded at 10 samples), we reconstruct fewer 3D trajectories than geometric triangulation at 60 fps. However, the overall shape of the trajectories is similar.

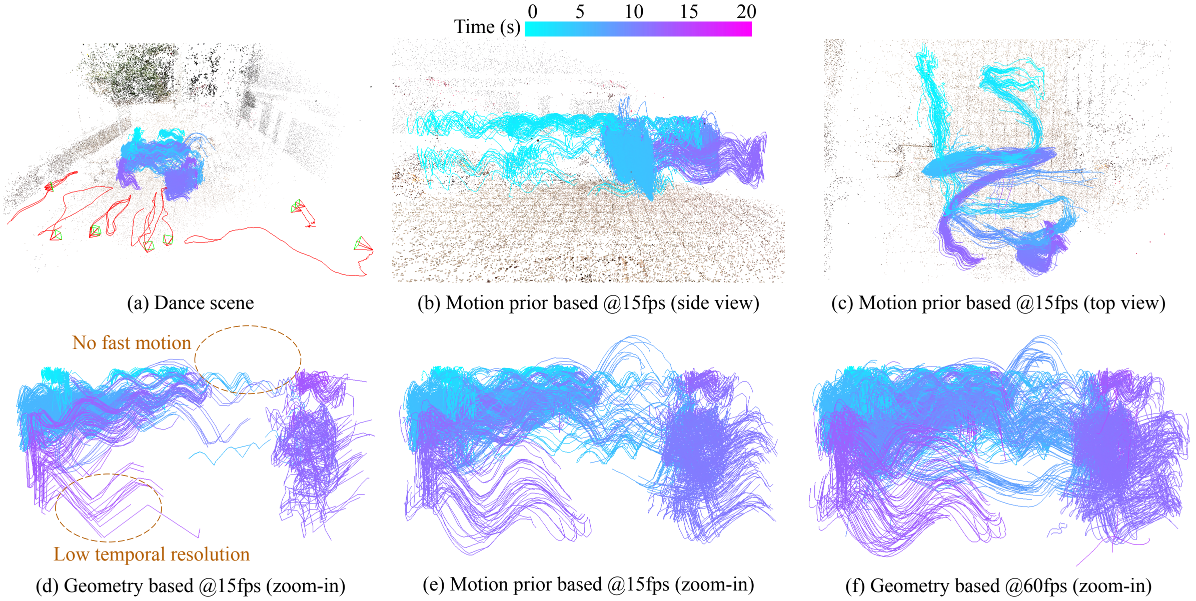
Interestingly, this scene has a large number of static background points. This adversely reduces the spatial calibration accuracy for the foreground points (see Figure 2.10). Our algorithm improves the spatial calibration for cameras with enough number of visible dynamic points.



(a) Jump scene (b) Motion prior based @30fps (front view) (c) Motion prior based @30fps (top view)

(d) Geometry based @30fps (zoom-in) (e) Motion prior based @30fps (zoom-in) (f) Geometry based @120fps (zoom-in)

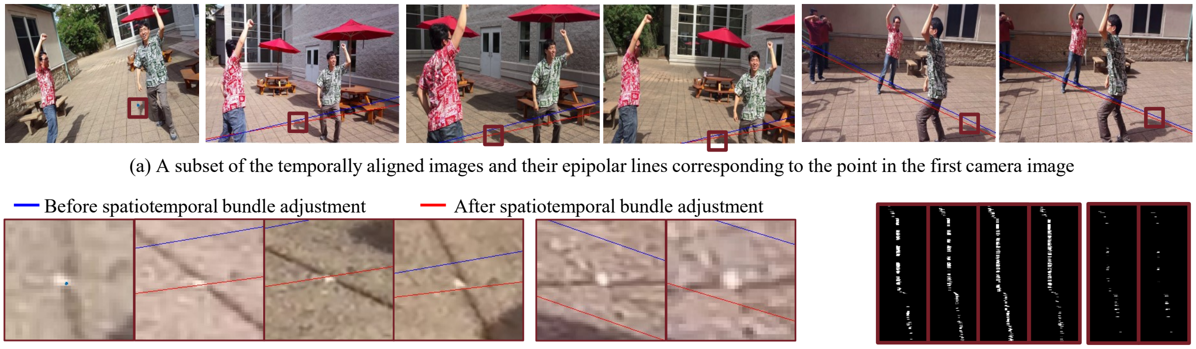
Figure 2.8: Jump scene. Point triangulation of frame-accurate alignment fails to reconstruct the fast action happened at the end of the sequence. Conversely, our motion prior based approach produces plausible reconstruction for the entire course of the action even with relatively low frame-rate cameras. Trajectories estimated from our approach highly resemble those generated by the frame-accurate alignment and triangulation at 120fps.



(a) Dance scene (b) Motion prior based @15fps (side view) (c) Motion prior based @15fps (top view)

(d) Geometry based @15fps (zoom-in) (e) Motion prior based @15fps (zoom-in) (f) Geometry based @60fps (zoom-in)

Figure 2.9: Dance scene. The 3D trajectories are estimated using 10 15 fps cameras. Noticeably, the trajectories generated from frame accurate alignment and triangulation are fewer, shorter, and have lower temporal resolution than those reconstructed from motion prior based approaches.


Before spatiotemporal bundle adjustment
After spatiotemporal bundle adjustment

(a) A subset of the temporally aligned images and their epipolar lines corresponding to the point in the first camera image

(b) Insets of the (a): Success (left) and failure (right) epipolar estimation.

(c) Visibility matrix of the dynamic points

Figure 2.10: Evaluation of the spatiotemporal calibration. The blue and red lines are the estimated epipolar lines before and after spatiotemporal bundle adjustment, respectively. The epipolar lines estimated after spatiotemporal bundle adjustment have noticeable improvement at the foreground for cameras with a large number of visible dynamic points. This figure is best seen electronically.

2.5.3 Human body fitting

Due to the lack of ground truth data, we qualitatively compare the different shape fitting algorithms by its alignment to the observed person silhouette. As shown in Figure 2.11, due to fast human activity and abrupt camera motion, shape fitting using geometric constraint and frame-level alignment either fails to estimate the body shape (first and second rows in the second column) or does not align well with the observed silhouette (third row, second column). Motion prior based shape fitting to the detected body keypoints better aligns the mesh to the images but fails when the detected keypoints are occluded (first, second, and fourth rows in the third column). Additionally due to the small misalignment when the SMPL model internal joint location and the detected joints, especially at the hip joints, using the only detected joints produces suboptimal body shape (third row, third column). Our full cost function considering the motion prior, the body joints, and detected silhouette produces the best fitting results. We notice that contribution of the sparse dynamic 3D points is negligible for both scenes. This is because most of the points are concentrated in the body torso, which is already the most stable tracked body part. We show snippets of the reconstructed body mesh for both scenes in Figure 2.12.

2.6 Discussion

One of the our biggest obstacles is the requirement of corresponding 2D trajectories across cameras. Just as SIFT descriptor for point matching has revolutionized static scene reconstruction, trajectory descriptor is needed for dynamic scene reconstruction. To disambiguate the matching, such a descriptor must accumulate information spatiotemporally. Another promising solution is to use semantic detection as coarse correspondences which can later guide the pixel-level correspondences in a tracking-by-detection framework. We explore this option further in Chapter 4.

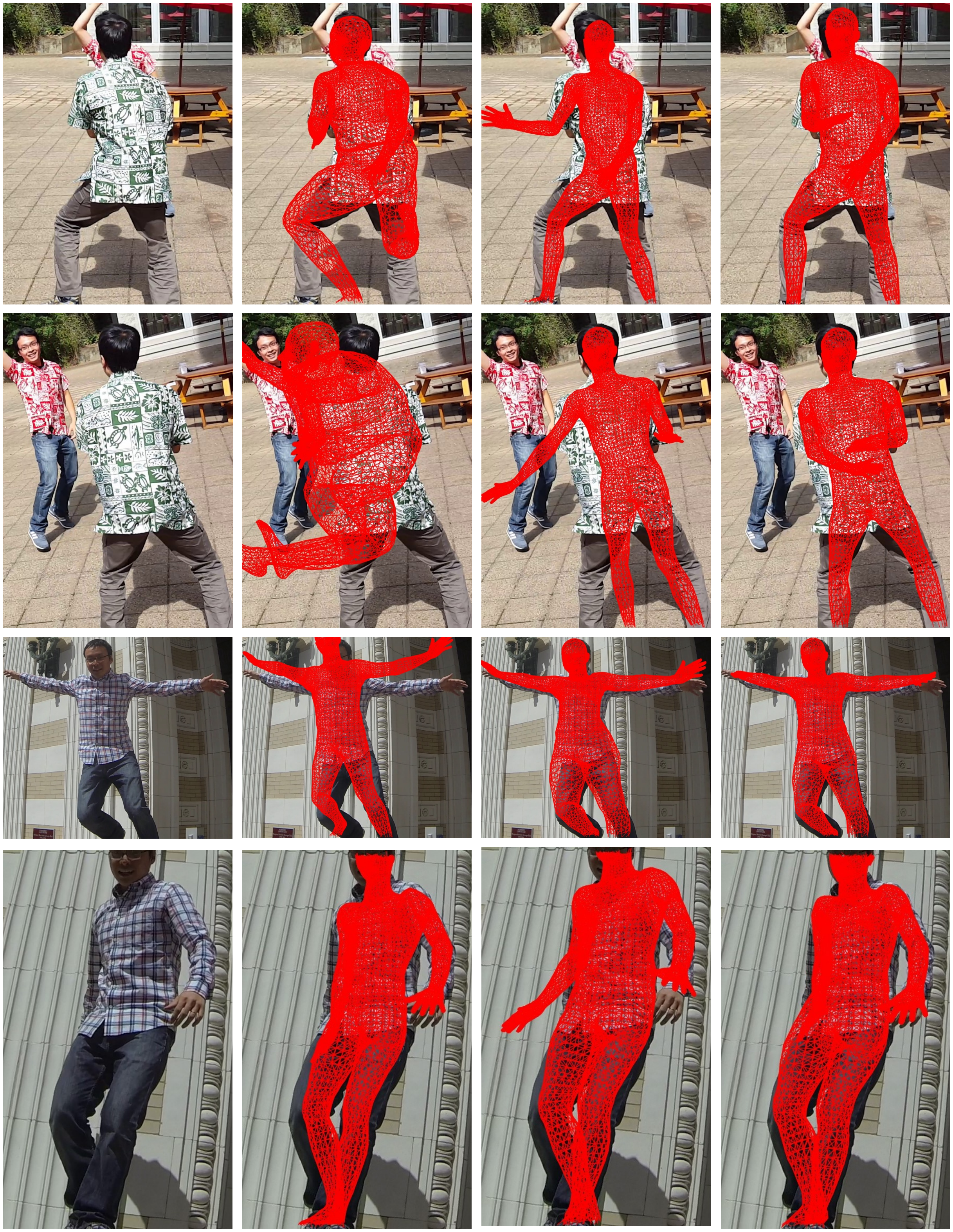


Figure 2.11: The effect of subframe alignment for human shape fitting for the Dance and Jump scenes. Given the input frames (first column), due to fast motion, shape fitting assuming frame-level alignment and geometric constraint (second column) fails to match the silhouette of the person. Our motion prior based shape fitting to the 2D body joints (third column) better aligns the observed silhouette but fails when the joints are not visible. Adding the silhouette constraint to the motion prior based fitting produces the best results (last column).

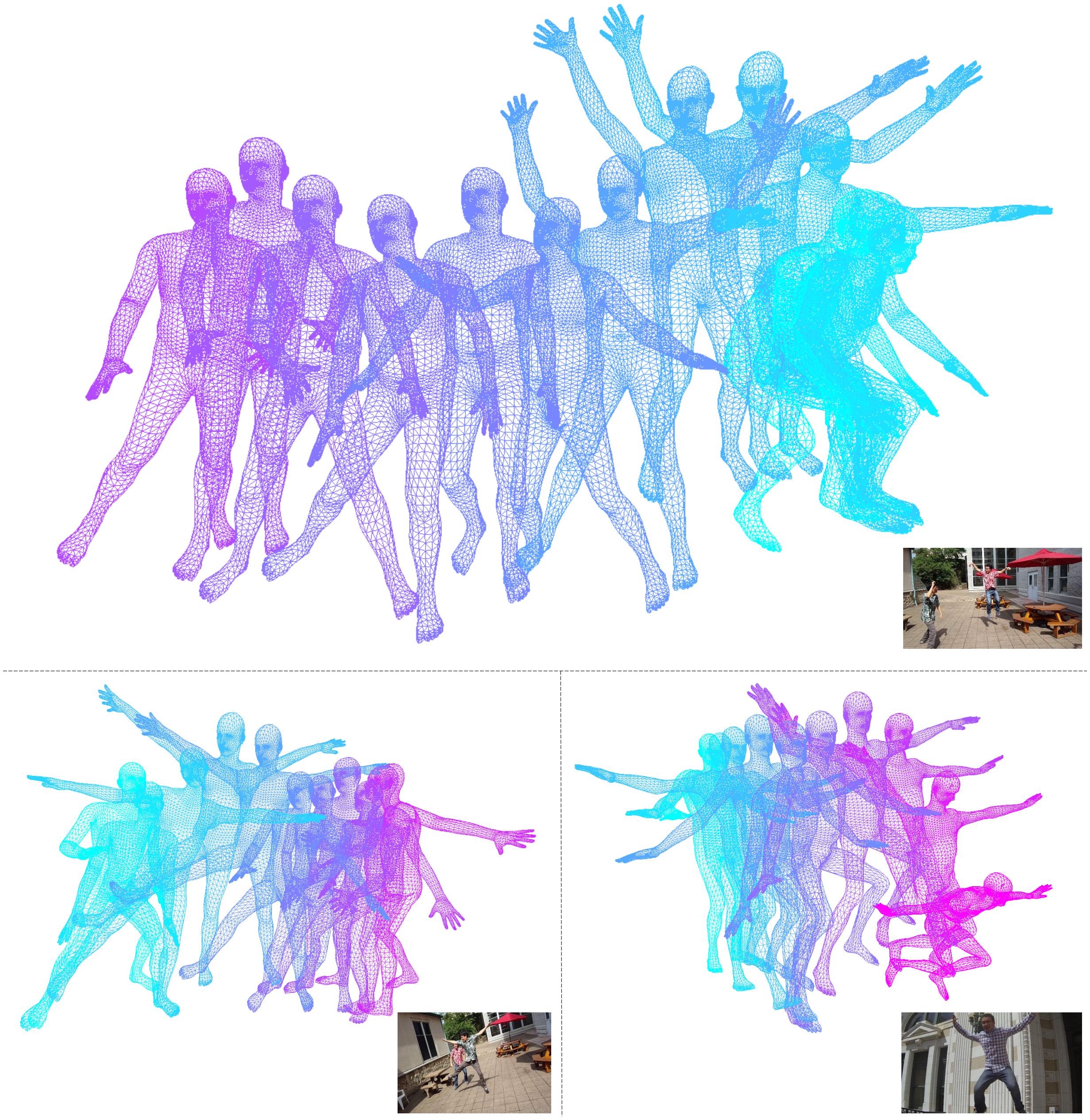


Figure 2.12: Human mesh fitting for different person in the Dance sequence (top and bottom left) and Jump sequence (bottom right). The color encodes the relative action time.

2.7 Summary

We present a framework for dynamic human reconstruction from unsynchronized video streams where geometric triangulation constraint is inapplicable. The key to our approach is the use of physics-based motion prior to joint spatiotemporally calibrate the cameras and reconstruct the observed feature points, as well as the rolling shutter scanning speed. We devise an incremental reconstruction and alignment that strictly enforces the motion prior during the optimization and a divide and conquer algorithm that speeds up the first algorithm many folds without loss of accuracy. For better visual interpretation of the scene, we fit a statistical human model of the observed unsynchronized video streams. This fitting is constrained by the same motion prior in addition to detected semantic cues in the images. We showcase the reconstruction on videos captured by uncalibrated and unsynchronized and moving rolling shutter cameras in the wild.

Chapter 3

Self-supervised Learning of Spatiotemporal People Association

With the rapid proliferation of consumer cameras, events are increasingly being recorded from multiple views, such as surprise parties, group games, and sports events. The challenges in tracking and reconstructing such events include: (a) large scale variation (close-up and distant shots), (b) people going in and out of the fields of view many times, (c) strong viewpoint variation, frequent occlusions and complex actions, (d) clothing with virtually no features or clothing that all look alike (school uniforms or sports gear), and (e) lack of calibration and synchronization between cameras. As a result, tracking methods (both single [50, 192, 251] and multi-view [27, 136, 181]) that rely on motion continuity produces short tracklets. And tracking-by-association methods relying on pretrained descriptors [15, 246] fail to bridge the domain gap between training data captured in (semi-)controlled environments and event videos captured in open settings.

In this chapter, we present a novel self-supervised person association framework that integrates tracking-by-continuity and tracking-by-association to overcome both their limitations. We show that even a state-of-art pretrained person appearance descriptor trained on large-scale labeled human re-identification dataset [133, 178, 256] is not sufficient to discriminate different people over a long duration and across multiple views. We bridge the domain gap by refining the pretrained descriptor to the event videos of interest with *no* manual interventions (such as manual labeling) and discriminatively learn a robust single-frame person association descriptor. Our insight to self-supervision is to exploit three *basic* sources of information in the target domain: (a) short tracklets from tracking-by-continuity methods, (b) multi-view geometry constraints, and (c) mutual exclusion constraints (one person cannot be at two locations at the same time). These constraints allow us to define contrastive and triplet losses [51, 189] on triplets of people images – two of the same person and one of a different one. Even using the most conservative definition of the constraint satisfaction (tiny tracklets, strict thresholds on the distance to epipolar lines) allows us to *automatically* generate millions of training triplets for domain adaptation.

While the above domain adaptation stage improves the descriptor discriminability of people with similar appearance, it could also lead to strong semantic bias for people rarely seen in

Scene			
# cameras	6 head-mounted + 12 hand-held	16 hand-held	9 hand-held
Video stats.	1920×1080, 60fps, 30s	1920×1080, 60fps, 60s	3840×2160, 30fps, 120s
# people	14	14	60
Tracklet noise	2%	11%	3%

Table 3.1: Three new group activity tracking datasets scenes: Chasing [C] (left) , Tagging [T] (middle), and Halloween [H] (right). [C] has 6 of people with camouflage and 3 others with dark clothing, and cannot be distinguished without strong attention to detail. Most people in [T] wear feature-less clothing making feature tracking hard. [H] is from an actual surprise birthday during the Halloween party and suffers from significant motion blur. The scene and camera behavior for any of these sequences are not staged. Tracklet noise is the percentage of tracklets with at least two people grouped into a single track.

the videos. We address this problem by jointly optimize the descriptor discrimination on the large labeled corpus of multiple publicly available human re-identification (ReID) datasets and the unlabeled domain videos using a multitask learning objective. Empirically, the proposed descriptor learning enables easier multi-view association of individual or tracklet of detections via clusterings. We show that even classical clustering algorithm such as k-means is sufficient given a known number of people. In practice, since the number of people is unknown, we adopt the continuous clustering framework [191] and enforce soft spatiotemporal constraints from our mined triplets during the construction of the clustering connectivity graph. Since the association is solved globally, there is no tracking drift.

We validate our framework on the recent WILDTRACK dataset [41] and three challenging new datasets of complex and highly dynamic group activity: Chasing [C], Tagging [T], and Halloween [H], captured by up to 18[1] mobile cameras (see Tab. 1). We show significant people association accuracy improvement over the state-of-art pretrained human ReID model (18% for [C], 9% for [T], and 9% for [H]).

To further demonstrate the impact of the improved descriptor, we use our association to drive a complete pipeline for 3D human tracking to estimate spatially stable and temporally coherent 3D skeleton for each tracked person. Compared to the baseline, our method shows significant improvement (5-10X) in 3D skeleton reconstruction, stability, minimizing tracking noise. We believe, for the first time, stable and long duration 3D human tracking is demonstrated in actual chaotic live group events.

We leverage the tracked 3D human skeletons and merge multiple video streams into a multi-

[1] For a pair of cameras are held by 1 person, we only use the left camera for learning human descriptor and skeleton reconstruction due to their small camera baseline.

angle video by selecting video chunks where the visible person is most frontal to the viewing camera. Such multi-angle cut is needed because it is unlikely that any person in the scene is well visible by one video stream for any complex group activity events. Our cut algorithm detects inter-human occlusion to determine the camera switching moment while still maintains the flow of the action well. This system provides an easy visual interface for people tracking from multiple cameras of crowded activities.

Contributions: (1) We present a simple but powerful self-supervised domain adaptation of person appearance descriptor framework using monocular motion tracking, mutual exclusive constraints, and multi-view geometry *without* manual annotations. (2) We demonstrate that discriminative appearance descriptor allows a reliable association via simple clustering. This advantage enables a first-of-a-kind accurate and consistent markerless motion tracking of multiple people participating in a complex group activity from mobile cameras *"in the wild"* with further application to multi-angle video for intuitive tracking visualization. These contributions are *orthogonal* to advances in single-view tracking (tracklet) or clustering algorithms.

3.1 Related Work

Our work is related to the themes of people Re-Identification (ReID) and multi-view motion tracking. People ReID focuses on learning descriptors that match people across views and time. Recent advances can be attributed to large and high-quality datasets [133, 178, 256], and strong end-to-end descriptor learning, which eclipses handcrafted features [135, 143, 149, 254, 257]. Common modern ReID approaches include verification models [7, 46, 98, 133] and classification models [231, 235]. In general, verification models only use weak re-ID labels and do not take all the annotated information into consideration, these models are outperformed by many classification-only models by a margin [216, 230, 231]. One drawback of the identification model is that the training objective is different from the testing procedure, i.e., it does not account for the similarity measurement between image pairs, which can be problematic during the people retrieval process. [151, 201, 202, 228] combines approaches for both image-based and video-based ReID and show improvement in accuracy. Some recent works also consider body part information [131, 253] for fine-grained descriptor learning. We adopt similar architecture [201, 235] but show how a previous generic person descriptor trained on *labeled* data is insufficient for a reliable human association on the multi-view videos captured in the wild. Our key is to exploit basic constraints available in testing scene to adapt the person descriptor with *no* manual annotations. Thus, our model is event (scene) specific rather than generic human ReID models.

People tracking approaches formulate person association as a global graph optimization problem by exploiting the continuity of object motion; examples include [59, 155, 251] for single view tracking, and [27, 193, 223] for multi-view tracking from surveillance cameras. These approaches use relatively simple appearance cues such as the histogram of color, optical flow, or just the overlapping bounding box area [12, 34, 50, 53, 164, 192] for monocular settings or 3D occupancy map from multi-view systems [73, 136, 232]. Although data-association based tracking methods are theoretically promising, the performance is highly reliant on the quality of the object detector. Since not all objects can be detected in each frame, false detections may be present, and some objects may occlude others, solving the tracking-by-detection problem via graph optimization is very challenging. Thus, these methods usually generate reliable short-term tracklets as the targets permanently disappear after a short time. We tackle people tracking in recurrent scenes and our algorithm takes those single-view tracklets as inputs to produce their associations for the entire scene. Additionally, whereas existing multi-view tracking algorithms are limited on fixed, calibrated and synchronized cameras [23, 73, 136, 232], our framework is applicable to uncalibrated moving cameras and can temporally align multiple videos automatically.

Recently, [60, 156, 205, 246] combine global graph optimization and discriminative appearance descriptors and show clear improvements over isolated approaches. Zhang et al. [252] are probably the first to apply online discriminative learning in tracking multiple objects. Dehghan et al. [60] formulate an online discriminative learning, which solves the detection and global data association jointly by integrating global data association into the inference of a structured learning tracker. The closest to our work is Yu at al. [246]. However, their method assumes *known* number of people captured in controlled settings and solve a challenging L_0 optimization using a sophisticated solution path approach. We tackle a similar problem but in unconstrained settings with *unknown* number of people and moving cameras using simpler clustering algorithm. This is possible because of our discriminative but automatic scene-aware person descriptor.

We use our association to drive a complete pipeline for 3D human tracking. While markerless motion tracking has been widely demonstrated in laboratory setups [64, 110, 139, 199] and more recently in general settings [65, 157, 172, 177], thank to advances in CNN-based body pose detectors [35, 163], these methods showcase the results on activity involving 1 or 2 people staying in fixed volume (never have to re-associate people) with minimal interactions (inter-occlusion is not considered). In contrast, we target 3D motion tracking of complex group activities of up to 14 people in unconstrained settings captured by up to 11 cameras with people frequently move in and out of the field of view.

Our application to multi-angle video cut is most related to Arev at. el [14]. However, our goal is obtained the best front-facing camera view to track a selected person whereas they aim to capture the recorders joint attention. Moreover, the key constraints in choosing the best view are fundamentally different as we have underlying 3D skeleton models whereas they rely on co-visibility of the camera frustums.

3.2 Person Appearance Descriptor

Our goal is to learn a robust appearance descriptor extractor $u_x = f(x)$ of a person image x that is similar for images of the same person and dissimilar for different people regardless of the viewing direction, pose deformation, and other factors (like illumination) for our domain videos. We start with an extractor $f(x)$, initially trained on a large labeled corpus of multiple publicly

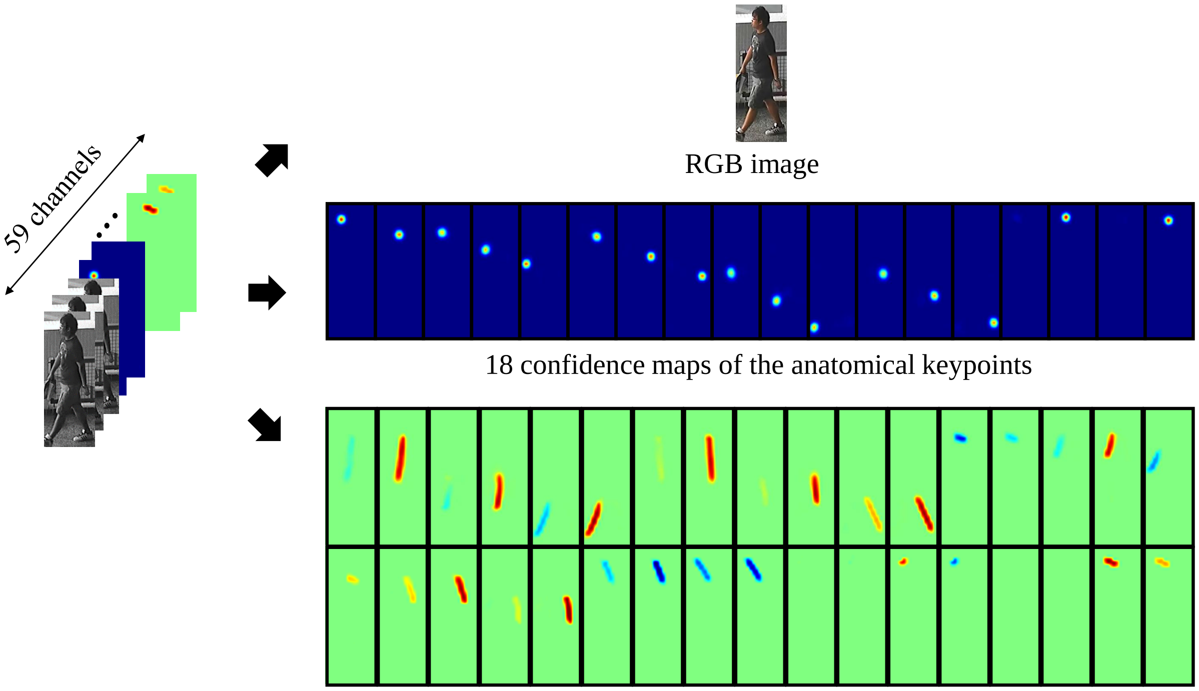

59 channels
RGB image
18 confidence maps of the anatomical keypoints
38 confidence maps of the affinity field between keypoints

Figure 3.1: The input to our CNN is a 59-channel feature maps, consisting of the color image, the feature maps of the 18 anatomical keypoints and their affinity fields computed by CPM.

available people ReID datasets, and finetune it using the Siamese triplet loss on triplets of images automatically mined from the domain videos with no human intervention. While this finetuning stage improves the descriptor discriminability of people with similar appearance, it could also lead to strong semantic bias for people rarely seen in the videos. We address this problem using a multitask learning objective and jointly optimize the descriptor discriminability on the labeled corpus labeled human ReID datasets and the unlabeled domain videos. We iteratively mine the triplets and retrained the descriptor for several (triplet mining) iterations.

3.2.1 Person Appearance Descriptor

This section describes our pose insensitive person descriptor extractor $f(x)$. For current ReID datasets, the ground truth detections and labels are mostly not generated manually. Pedestrian misalignment, which mainly arises from person detector errors and pose variations, is a critical problem for a robust person reID system. With bad alignment, the background noise will significantly compromise the feature learning and matching process. One straight forward way to achieve invariance to such misalignment and various pose configuration is to rectify the input image into a canonical frame [253]. However, the rectification is problematic due to 2D warping artifact and wrong pose detection. Instead, we augment the RGB image with the heatmaps of keypoints and their part affinity fields provided by CPM model [35] (see Figure 3.1). This representation avoids the viewing direction quantization in rectifying the body parts [46, 133] and

name	patch size/ stride	output side	#1×1	#3X3 reduce	#3x3	double #3X3 reduce	double #3X3	pool+proj
input		59x288x112						
conv 0–5	3x3/2	32x72x28						
inc 1a		256x72x28	64	64	64	64	64	avg+64
inc 1b	stride 2	384x36x14	64	64	64	64	64	max+identity
inc 2a		512x36x14	128	128	128	128	128	avg+128
inc 2b	stride 2	768x18x7	128	128	128	128	128	max+identity
inc 3a		1024x18x7	256	256	256	256	256	avg+256
inc 3b	stride 2	1536x9x4	256	256	256	256	256	max+identity
fc7		256						
fc8		256						
fc9		M						

Table 3.2: The structure of our CNN model for person ReID. This model is inspired by the Inception architecture, known for its efficiency and expressiveness.

takes the detection confidence into account to down weight possible pose detection failures.

Table 3.2 shows our detailed CNN model for the extractor $f(x)$. Inspired by the ability to learn multi-scale features and model compactness of the Inception architecture [204], our customized CNN model passes an input of size 288x112x59 through six convolution layers, six Inception modules, and three fully connected layers. The person descriptor is extracted at $fc8$ layer. We use Batch Norm [104] followed by ReLU activation at every layer. We regularize training by randomly switching off 50% of the neurons in the $fc7$ layer during training.

3.2.2 Spatiotemporal Descriptor Adaptation

Due to possible discrepancies between the appearances of the training sets and our domain application videos, we finetune the $f(x)$ on each of our test video sequences using the contrastive and triplet loss [51, 189]. The input to our process are triplets of 2 images of the same person and 1 image of a different person. We optimize the CNN such that the distance between query and anchor is small and the distance between query and the negative example is large. Our loss function is defined as:

$$\begin{aligned}
L_S(u_i, u_i^+, u_i^-) &= \|u_i - u_i^+\|_2^2 \\
&\quad + \max\left(0, \|u_i - u_i^-\|_2^2 - m\right) \\
&\quad + \max\left(0, \|u_i^+ - u_i^-\|_2^2 - m\right), \\
L_T(u_i, u_i^+, u_i^-) &= \max\left(0, \|u_i - u_i^+\|_2^2 - \|u_i - u_i^-\|_2^2 + m\right), \\
L_{ST}(u_i, u_i^+, u_i^-) &= L_S(u_i, u_i^+, u_i^-) + L_T(u_i, u_i^+, u_i^-),
\end{aligned}$$

where (u_i, u_i^+, u_i^-) is triplet of two positive and a negative unit norm descriptor, respectively, and m (set to 2 for all experiments) is the margin parameter between two distances. Our total loss function for finetuning is defined as:

$$E_{ST} = \min_{f} \sum_{i=1}^{N_d} L_{ST}(u_i, u_i^+, u_i^-),$$

where, N_d is the number of triplets in the domain videos. We optimize the model using Stochastic Gradient Descent. Empirically, we found hard-negative mining hurts the learning due to possibly erroneous triplets in the first triplet mining stage and thus, only use this trick in later iterations.

Automatic Triplet Generation

Single-view triplets: For every video, we first apply CPM to detect all the people and their corresponding anatomical keypoints. Given these detections, we can easily generate negative pairs by exploiting mutual exclusive constraints, i.e. the same person cannot appear twice in the same image. In addition, we can create positive pairs by using short-term motion tracking. We create motion tracklets by combining three trackers: bidirectional Lucas-Kanade tracking of the keypoints, bidirectional Lucas-Kanade tracking of the Difference of Gaussian features found within the detected person bounding box, and person descriptor matching between consecutive frames. The tracklet is split whenever any of the trackers disagree. We also monitor the smoothness of the keypoint 2D trajectories and split the tracklet whenever the instantaneous 2D velocity is 3 times greater than its current average value. More sophisticated approaches such as [59, 155] can also be used for better tracklet generation. Images corresponding to the same motion tracklet constitute positive pairs for our finetuning.

Multi-view triplets: We enrich the training triplets with positive pairs across views by using multi-view geometry – pairs of detections corresponding to a single person in 3D space must satisfy epipolar constraints. Since our videos are captured in the wild, they are unlikely to be synchronized. Thus, we must first estimate the temporal alignment between cameras to use multi-view geometry constraints. Assuming known camera framerate and start time from the video metadata, which aligns the videos up to a few seconds, we linearly search for the temporal offset with the highest number of inliers satisfying the fundamental matrix. A bi-product of this alignment process is the corresponding tracklets across views, which form our positive pairs.

More specifically, let $\mathbf{k}_i^n(t) = \{k_i^{n_{t,1}}, ..., k_i^{n_{t,18}}\}$ be the set of anatomical keypoints of the people detection n at frame t of camera i, and $\mathbf{T}_i^l = \{n_0, .., n_F\}$ be a tracklet l containing the images of the same person for F frames. Let $M_c = (\mathbf{T}_i^l, \mathbf{T}_j^k)$ be the candidate tracklet pair c of the same person, computed by examining the median of the cosine similarity score of between all pairs of descriptors [2] within the tracklets, for camera pair (i, j) and $\mathbf{M}_{i,j}$ be all putative matched tracklets for camera pair (i, j). We set the similarity threshold to 0.5 and add those candidate matches to the hypothesis pool until their ratio-test threshold drops below 0.7. We use RANSAC with the point-to-line (epipolar line) distance as the scoring criteria to try all possible time offsets within the window of $[-2W, 2W]$ frames to detect the hypothesis with the highest number of

[2] At this stage, the descriptors are extracted using a pretrained CNN.

geometrically consistent matched tracklets:

$$I \leftarrow \underset{M_c \in \mathbf{M}_{i,j}}{\textbf{RANSAC}} \sum_{w=-W}^{W} \sum_{t=1}^{F} \sum_{\substack{n=1\\ n\in\mathbf{T}_i^l(t)\\ m=\mathbf{T}_j^k(t+w)\\ (\mathbf{T}_i^l,\mathbf{T}_j^k)\in M_c}}^{N_i(t)} \sum_{p=1}^{18} d(k_i^{n,p}, k_j^{m,p}, \mathbf{F}_{i,j}(t)),$$

where, $N_i(t)$ is the number of people detected in camera i at frame t, I is the number of inliers, and $d(x_1, x_2, \mathbf{F}_{i,j}(t))$ is the bidirectional point-to-line distance characterized by the fundamental matrix $\mathbf{F}_{i,j}(t)$ between the camera pair. $\mathbf{F}_{i,j}(t)$ can either be estimated by calibrating the cameras with respect to the scene background or explicitly searched for using the body keypoints during the time alignment process. We prune erroneous matches by enforcing cycle-consistency within any triplet of cameras with overlapping field of view. We set W to twice the camera framerate and use the video start time to limit the search.

3.2.3 Multitask Person Descriptor Learning

While finetuning the person appearance descriptor exclusively on the test domain could potentially improve discrimination of similar looking people, using it alone may result in semantic drift. The learned descriptor has a strong bias toward frequently observed people. The descriptor of different people who are rarely observed together from a single camera cannot be forced to be different due to the lack of mutual exclusive constraints.

We jointly learn the person descriptor from both the large scale labeled human identity training data and the scene specific videos. Since the model must predict the identity of the person from the labeled dataset, it is expected to output discriminative descriptors for rarely seen people in the domain videos. On the other hand, since we finetune the model on the domain videos, it should also discriminate people in those sequences better than training solely on other datasets. Mathematically, our multitask loss function is defined as:

$$E_D = \min_f (1-\alpha) E_{SM} + \alpha E_{ST},$$

where α is the scalar balancing the contribution of two learning tasks. E_{SM} is the standard classification loss:

$$E_{SM} = \underset{f}{\text{argmin}} \sum_{i}^{N_s} L_{SM}(g(f(x_i)), y_i)),$$

where, N_s is the number of training examples in the labeled corpus datasets, g is a linear function mapping the person appearance descriptor, $f(\cdot)$, to a vector of the dimension of the number of people in the training corpus, and L_{SM} is the softmax loss penalizing wrong prediction of the people ID label. We set α equal to 0.5 for all experiments.

3.3 Analysis of Human Descriptor Learning

We validate our method on three challenging new sequences: [C], [T], and [H] (see Table 3.1). In [T], the camera holders are mostly static and appear in low resolution which does not provide

	#identity	#camera	#images	Resolution	Annotation
VIPeR [84]	632	2	1.2k	48x128	Manual
3DPes	192	8	1k	37x88 – 236x278	Manual
ETH[190]	149	1	8.5k	44x85 – 175x449	Manual
iLIDS[68]	250	8	0.5k	32x76 – 115x294	Manual
CAVIARA[47]	72	2	1.2k	16x43 – 72x144	Manual
PRID[99]	200	2	4.4k	128x48	Manual
V47[222]	47	2	0.4k	51x154 – 222x446	Manual
WARD[148]	70	3	0.5k	48x128	Computer
CUHK02[132]	1816	10	7k	60x160	Manual
CUHK03[133]	1467	10	28k	31x92– 201x308	Computer
RAiD[57]	43	4	7k	64x128	Manual
Shinpuhkan[117]	24	16	22k	48x128	Manual
CUHK04[236]	8432	-	32k	19x53 – 141x375	Manual
MARS[256]	1261	6	59k	128x256	Computer
CMU[109]	33	31	13k	222x367 – 631x958	Computer
DUKE_MTMC [178]	1843	8	36k	64x156 –114x362	Computer
SAIVT[28]	150	8	7k	73x137 – 200x400	Manual

Table 3.3: Statistic of the people re-ID dataset. There are large variations in the image resolution, number of people, number of views for these datasets. For most large scale datasets (MARS, DUKE_MTMC), the ground truth people bounding boxes were generated by computer algorithms, which heavily suffers from people misalignment.

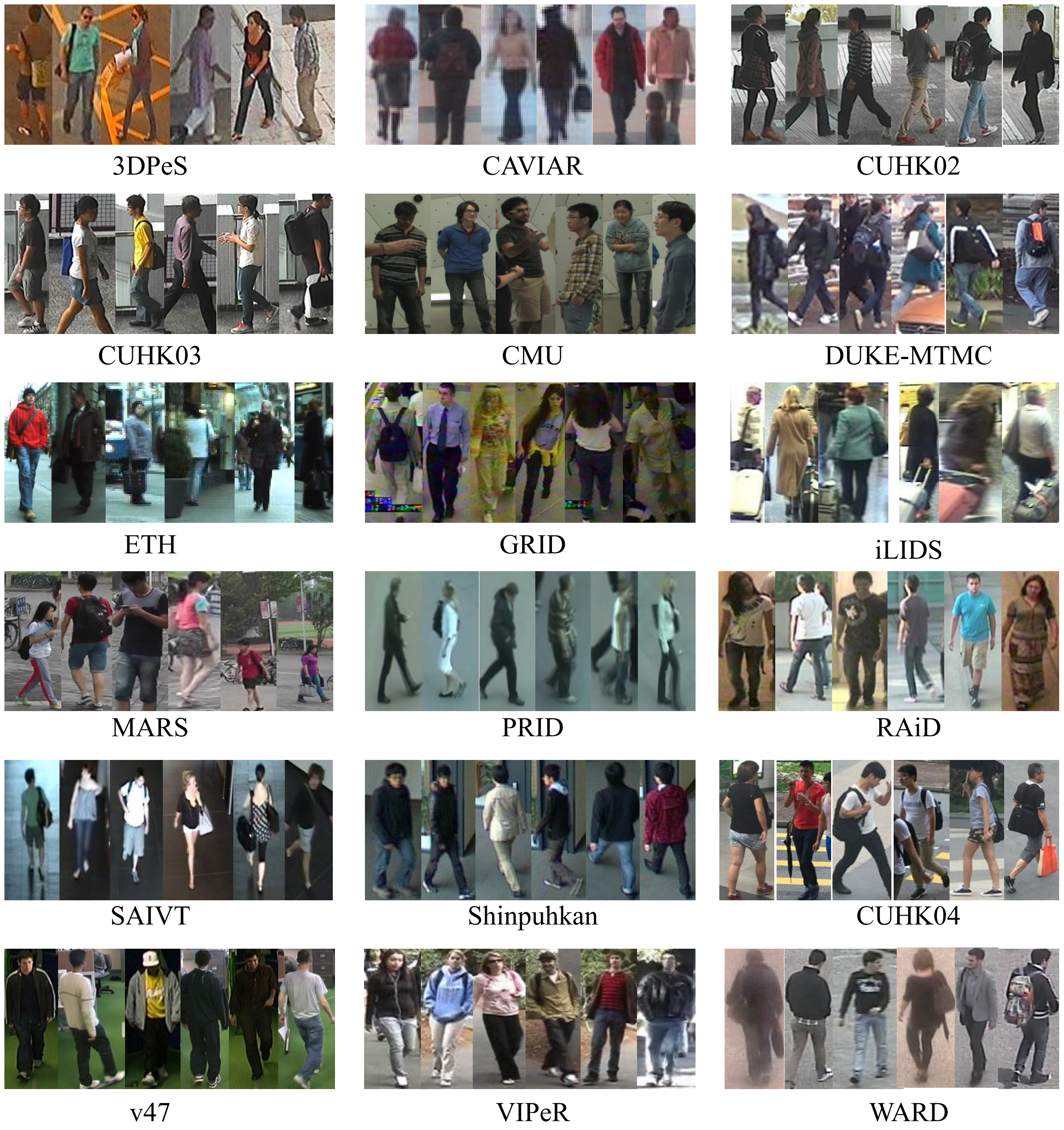


Figure 3.2: We learn the discriminative appearance feature embedding using CNN on a training corpus consists of 18 different datasets. Each dataset was collected with different setups, e.g., multi-camera surveillance system or single camera multi-target tracking (ETH), at different places, e.g., campus (CUHK02, CUHK03, MARS, DUKE˙MTMC), shopping mall (Shinpuhkan), airport (iLIDS), or streets (ETH, WARD, PRID), and of different image resolution. Note that there are strong people misalignment in MARS.

enough appearance variation for strong descriptor learning. It also has many noisy single-view tracklets with different people grouped together due to the lack of texture on the clothing and frequent inter-occlusion. There were no constraints on the camera motion or the scene behavior for any sequences. All the cameras in [C] and [T] are spatially calibrated using the ColMap library [187]. Calibration fails for [H] due to human motion which frequently occludes the background and strong motion blur. We manually annotate the people ID in our datasets for quantitative evaluations.

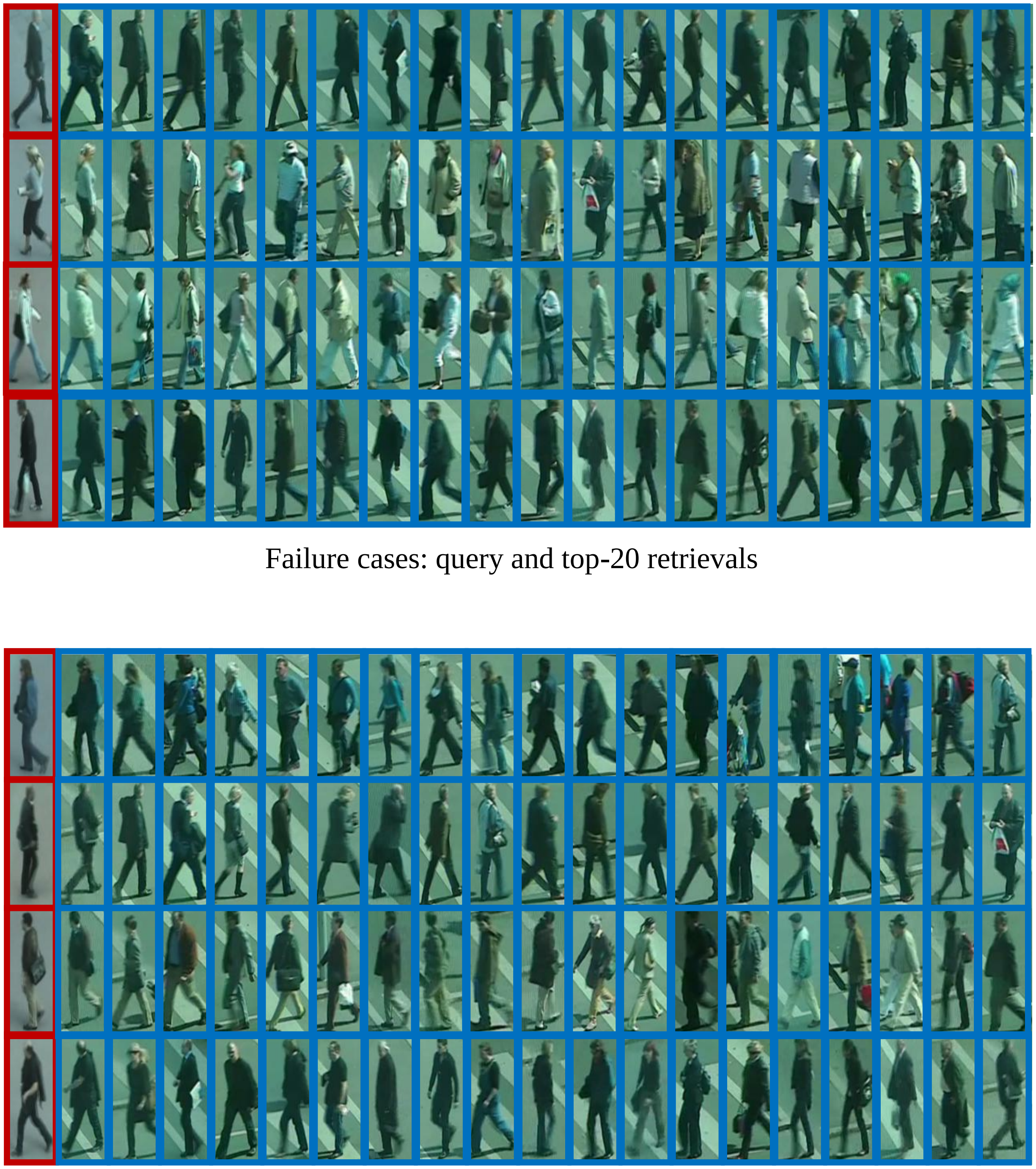
We first train the generic person descriptor extractor $f(x)$ on a combination of 16 different publicly available ReID datasets: VIPeR [84], 3DPes [20], ETH [190], iLIDS [68], CAVIARA[47], PRID[99], V47[222], WARD[148], CUHK02 [132], CUHK03 [133], CUHK04[236], RAiD[57], Shinpuhkan[117], MARS[256], CMU Panoptics studio [111], and SAIVT[28]. Each dataset was collected with very different locations with different demographics, various camera setups, and image resolution. CUHK02 CUHK03, DUKE-MTMC, MARS were captured on campus, where many students wear backpacks. PRID contains pedestrians in street views, where crosswalks appear frequently in the dataset. VIPeR images has significant illumination variation across different camera views. iLIDS was captured at the airport with many people dragging the luggage. The CMU dataset, captured in the CMU Panoptics studio, contains strong viewpoint and pose variations. ETH was captured from a single moving camera in street views for multi target tracking purpose. Notably, CUHK04, contained both images captured around an urban city and movie snapshots, has rich variations of viewpoints, lighting, background conditions. This dataset has the largest number of people identity but with very few views for each of them (on average 3 images). For most large scale datasets (MARS, DUKE_MTMC), the ground truth people bounding boxes were generated by computer algorithms, which heavily suffers from people misalignment. The statistics for each of these datasets is summarized in Table ??. The resulting labeled training corpus has approximately 200k images of nearly 16k different people. Please see Figure 3.2 for an illustration of the datasets.

We employ the standard softmax loss and train the extractor model $f(x)$ from scratch using Stochastic Gradient Descent with mini-batch of size 240 for 100 epoch. To train the MTL descriptor, we split samples evenly between unlabeled domain data and the labeled ReID datasets for each training batch and train until 1 epoch of the domain data is reached.

3.3.1 Analysis of Pose-Insensitive Person Descriptor

We analyze the accuracy of our descriptor on six commonly used datasets: CUHK03, MARS, PRID, iLIDS, ViPER, and 3dPES. We follow the same testing protocols for each dataset and use the Cumulated Matching Characteristics (CMC) to evaluate the performance of different ReID methods. Effectively, the CMC curve measures how the accuracy changes as a function of the number of nearest neighbor used in the gallery set.

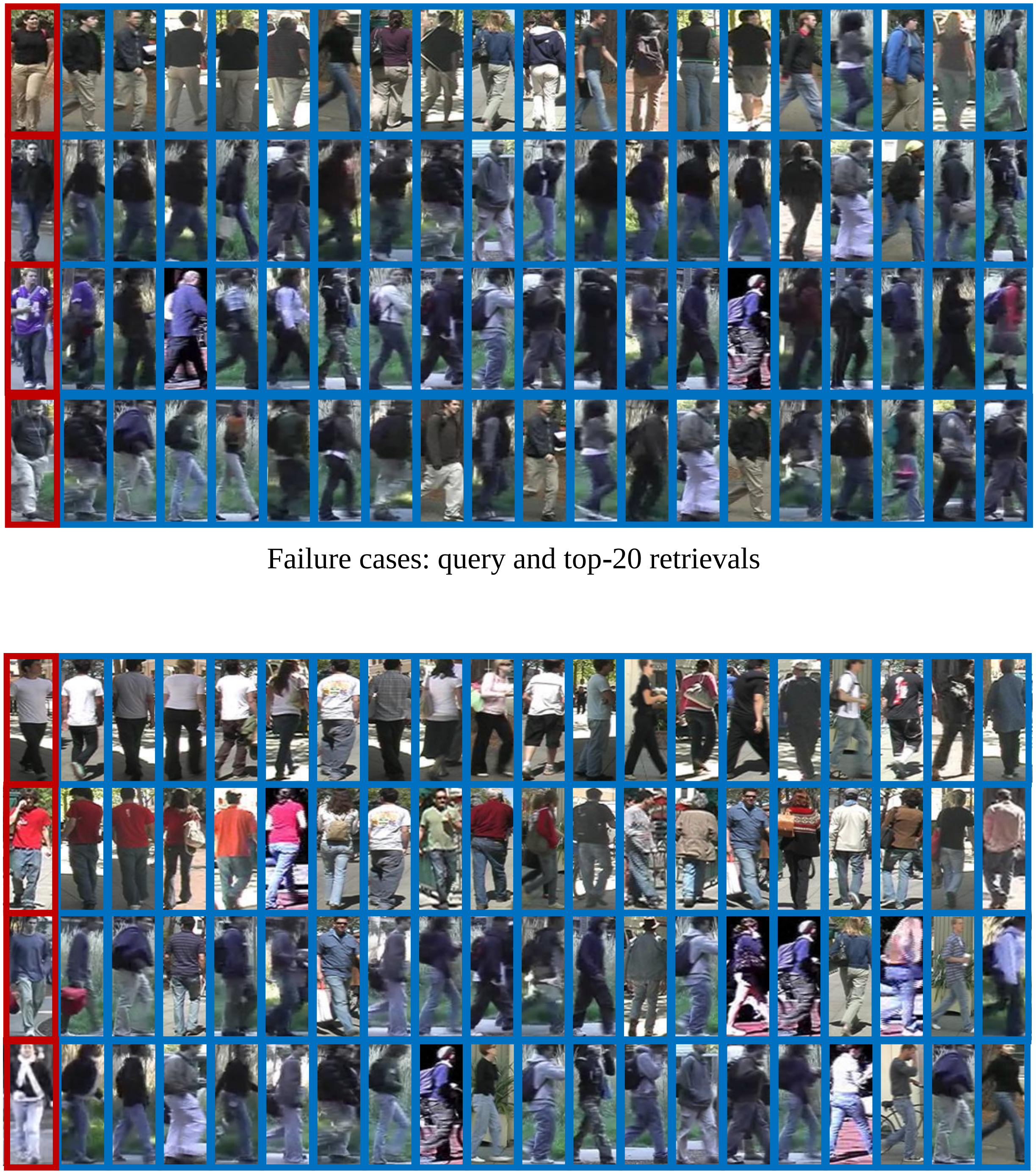
Qualitative: Figures 3.3, 3.4, 3.6, 3.5, 3.7, and 3.8 show the qualitative results of the success and failure cases for PRID, ViPER, 3dPES, iLIDS, CUHK03, and MARS, respectively. While the top retrievals for PRID, ViPER, iLIDS, or 3dPES bare some global appearance similarities



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

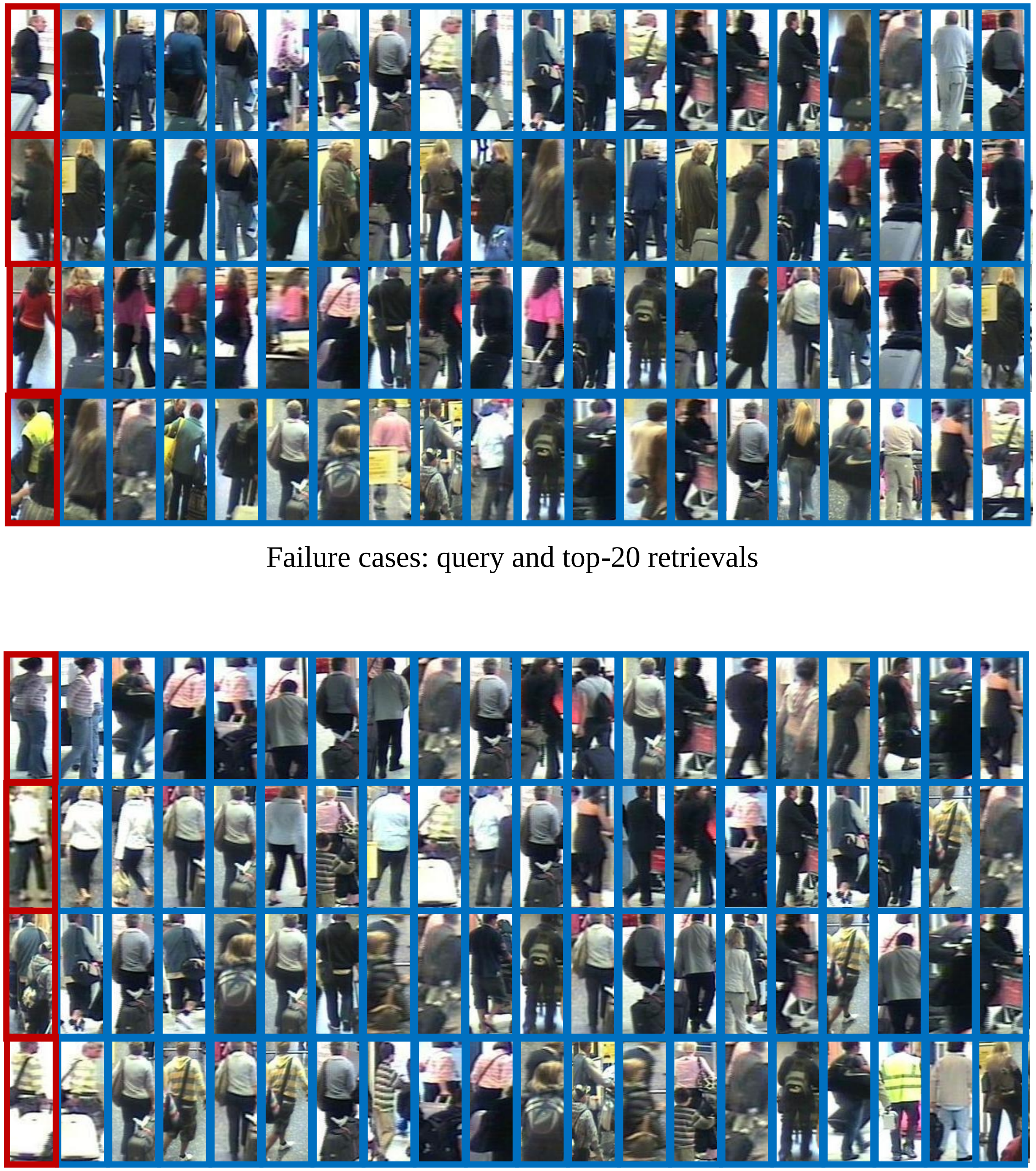
Figure 3.3: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for PRID dataset. For every row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query. While the top retrievals bare some global appearance similarities with the query, the retrievals are quite scattered in general. Note that this dataset is small and contains low-quality images.



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

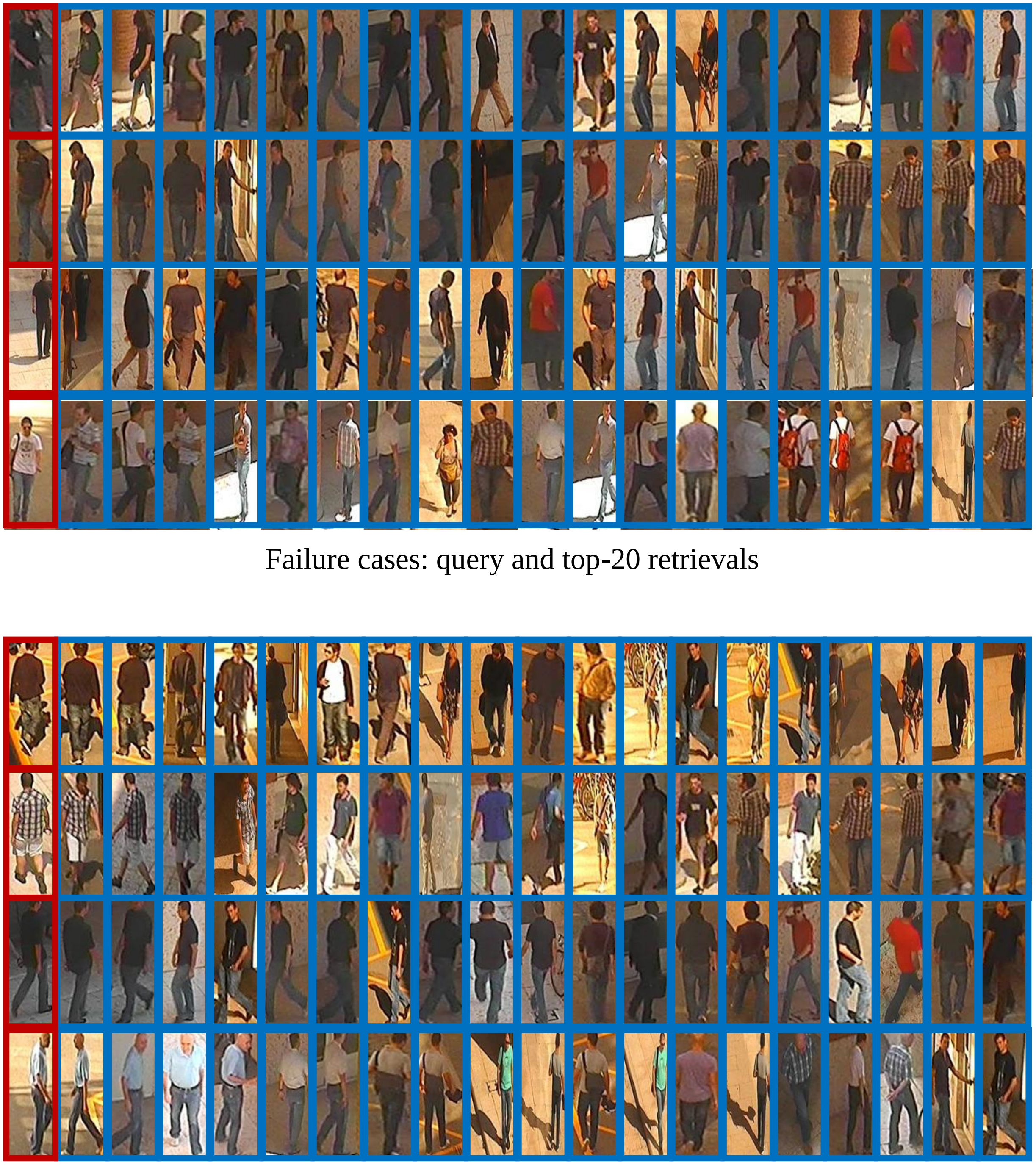
Figure 3.4: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for ViPER dataset. For every row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query. While the top retrievals bare some global appearance similarities with the query, the retrievals are quite scattered in general. Note that this dataset is small and contains low-resolution images with large viewpoint and illumination variations.



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

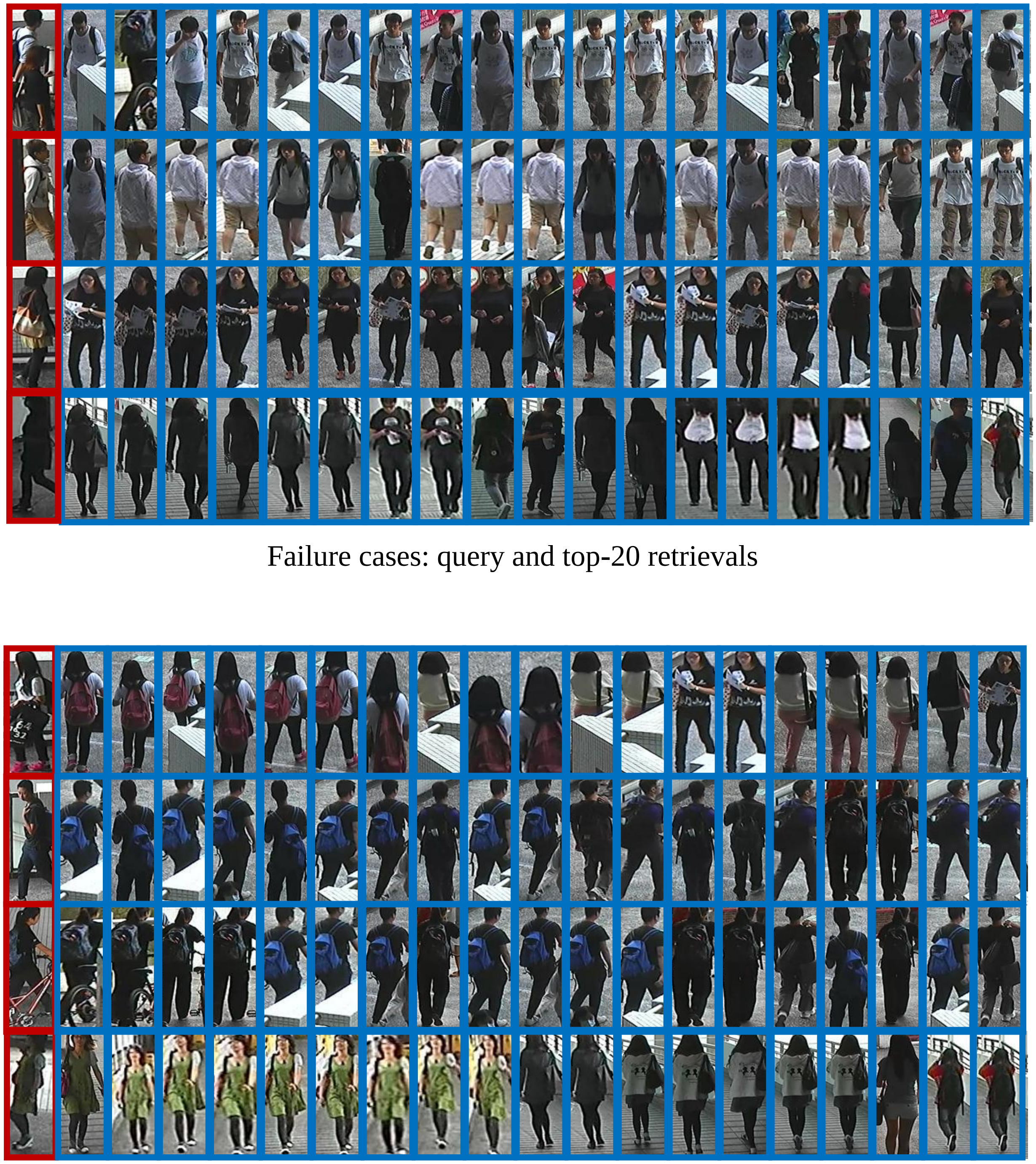
Figure 3.5: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for iLIDS dataset. For every row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query. While the top retrievals bare some global appearance similarities with the query, the retrievals are quite scattered in general. Note that this dataset is small and contains quality resolution images with strong occlusion.



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

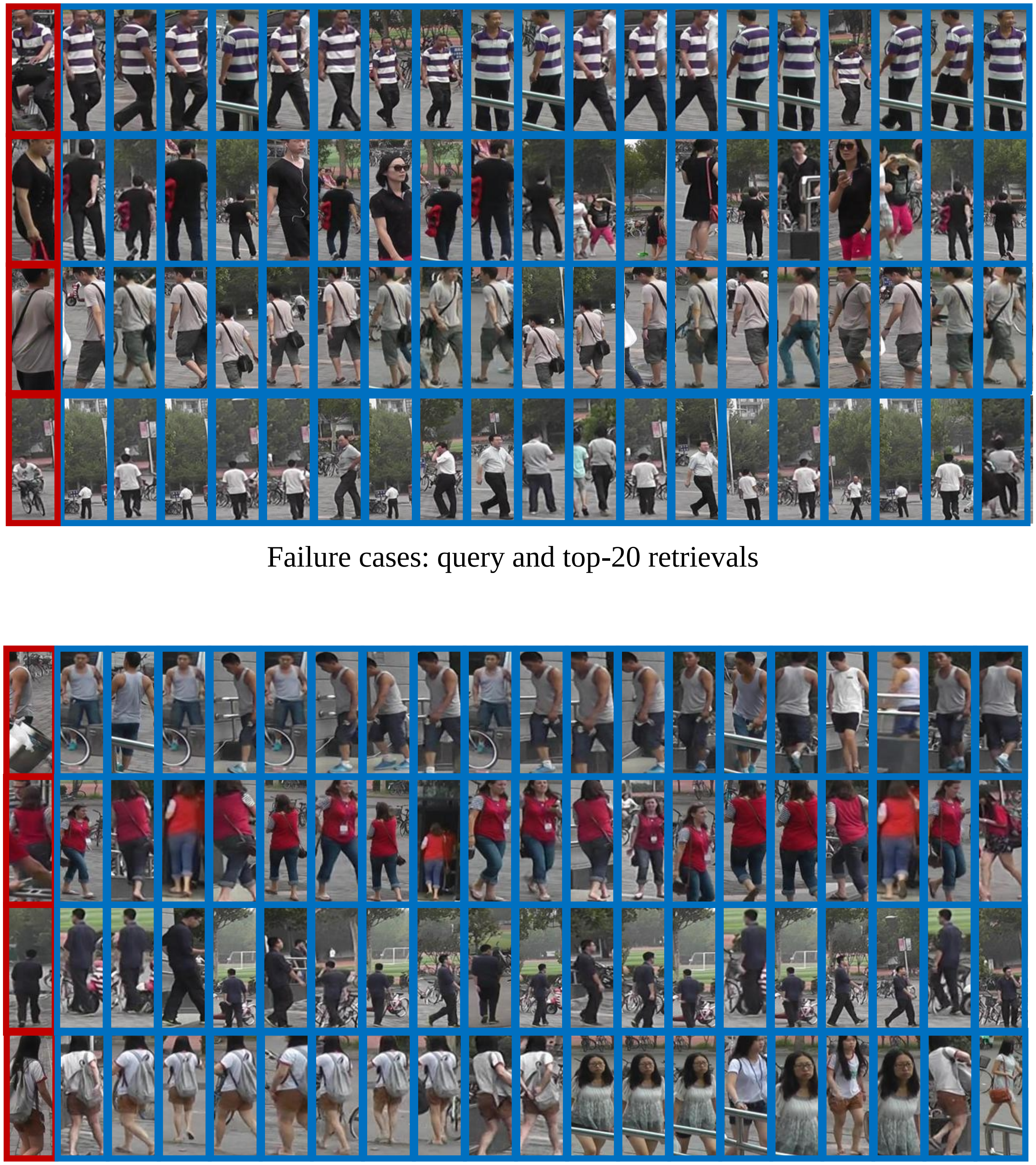
Figure 3.6: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for 3dPES dataset. For every row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query. While the top retrievals bare some global appearance similarities with the query, the retrievals are quite scattered in general. Note that this dataset is small and has large viewpoint and illumination variations.



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

Figure 3.7: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for CUHK03 dataset. For every row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query.



Failure cases: query and top-20 retrievals

Success cases: query and top-20 retrievals

Figure 3.8: Some typical the query images (red) where our model fails to find the correct matches after 20 nearest neighbors (blue) matchings for MARS dataset. For very row, the first image is the query and other are the nearest neighbors, sorted in decreasing distance from the query. Note the subtle differences between the query and the retrievals in case of failure. In some cases, the provided people identity is wrong (top row).

Input	CUHK03	MARS	PRID	iLDS	ViPER	3dPES
Baselines	85.4[81]	77.4 [98]	43.6 [167]	64.6 [235]	**56.3** [81]	56.0 [235]
Big RGB	91.1	76.9	55.0	84.5	42.4	70.0
Big RGB+KP	92.8	**79.8**	60.0	84.4	51.9	78.0
Big RGB+PAF	93.1	79.4	59.0	84.5	49.7	79.4
Big RGB+KP+PAF	**93.7**	**79.8**	**62.0**	**85.2**	**52.5**	**78.9**

Table 3.4: Ablative analysis of the pose heatmaps for the top-1-NN accuracy. Using all the heatmaps generated by CPM yields the best accuracy, albeit modest improvements over the key points (KP) or the part affinity fields (PAF) alone. Except for ViPER, which is a small dataset with strong illumination and viewpoint variations, our method consistently outperforms the baselines by a margin.

with the query, the retrievals are quite scattered in general. In contrast, the failure cases for CUHK03 and MARS are more subtle with many of them coming from the same tracklet (the third row in MARS), some even due to wrong ground truth labels (the first row in MARS). Note that CUHK03 and MARS are many times bigger than other datasets.

We show the pose and viewpoint insensitive properties on the data collected from the CMU Panoptic Studio using t-SNE visualization [145] in Figure 3.9. his dataset consist of 30k images for 80 different people. None of them are the same people as in the CMU dataset used in the training stage. Despite the similar appearance of multiple people, the images of the same person are clustered together. This shows a strong evidence of the pose and viewpoint insensitivity of our descriptor.

Quantitative: Table 3.4 shows the comparison between our approach and the recent methods for the top-1 matches on six commonly used datasets and our ablative analysis of how different heatmap categories affects the matching accuracy. For video dataset such as MARS, most methods compute the averaging distance of the learned feature descriptor over all pairs of time instances of the tracklets to match between trajectories. We perform per-frame matching, which is more challenging. Our approach outperforms most other methods by a margin except for ViPER, which is a small dataset with strong variations in viewpoint, image quality, and lighting condition. Since the total number of images from PRID, iLIDS, ViPER, and 3dPES comprises less than 5% of the training images, their appearance statistics is likely to be dominated by larger datasets. This potentially explains for their lower accuracies compared to CUHK03 and MARS. Augmenting the color images with the CPM heatmaps improves the accuracy, among which ViPER is boosted by 10.1%. Using both the keypoints and part affinity field heatmaps gives the best accuracy, albeit a modest improvement over keypoints or affinity field heatmaps alone.

3.3.2 Analysis of Descriptor Adaptation

Figure 3.10 shows 10-NN cross-view matching of images of several people with similar appearance or motion blur for all sequences and their cosine similarity score using the pretrained model (P) and our multitask descriptor learning (MTL). The pretrained model retrieves multiple incorrect matches. Our method is notably more accurate. Our similarity score often has a clear transition between correct and incorrect retrievals. Figure 3.11 and Figure 3.12 show a comparison of the 2D t-SNE embedding [145] between the descriptors using P and our MTL approach. While the descriptors extracted from the pretrained model are scattered, our descriptor groups images of the same person from all views and time instances into cleanly separated clusters. Note that while [H] is very complex scene with up to 60 people, our discriminative MTL still better clusters images of the same person into a single group.

Figure 3.13 shows the Cumulative Matching Characteristic (CMC) for all sequences: Chasing [C], Tagging [T], and Halloween [H]. There are clear improvements over the pretrained model as more sophisticated stages of your algorithm is applied. We further visualize the association accuracy in Figure 3.14. For the all sequences, the adapted descriptor improves the discrimination of frequently visible actors: 94% vs. 68% 1-NN classification accuracy for [C] and 90% vs. 75% for [T]. However, the discrimination of descriptor for the camera holders decreases: 56% vs. 85% for [C] and 35% vs. 42% for [T]. Our MTL, combining the strength of the classification and metric learning loss, performs best (92%/95% for actors/holders on [C] and 89%/61% for [T]) and has an overall baseline improvement of 17% [C], 9% for [T][3], and 9% for [H]. False matches due to the similar descriptor extracted from the generic CNN model are largely suppressed.

Figure 3.15 shows our analysis of the number of cameras, the tracklet noise, the training videos length on 1-NN matching accuracy, and the triplet mining iterations. Multi-view constraints are more helpful than temporal constraints as there are small improvements compared to the pretrained model P when a single camera is used. The algorithm shows noticeable improvement even with as few as 4 cameras. Yet, for the current scenes, such improvement saturates when more than 6 cameras are used. Regarding tracklet noise, our algorithm can improve the baseline if the noise percentage is less than 4%. High noise leads to fewer, and potentially incorrect, multi-view tracklets from pairwise matches and leads to slightly inferior accuracy compared to P. Even finetuning on $1/6th$ of the sequences leads to a notable improvement over P and performance converges after $2/6th$ of the sequence is used; this indicates that our method could be used on a smaller training set (e.g., first 15 minutes of a game) and applied to the rest. Lastly, we observe marginal accuracy improvement after the 1-st triplet mining iteration. This suggests that hard-negative mining is probably not needed and should be avoided in our framework.

[3] The results for [T] was obtained with cleaned tracklets.

3.4 Applications

3.4.1 Multi-View People Tracking via Clustering

Using the person descriptor, we cluster detections of the same person across all space-time instances. Since each video contains tens of thousands of detections, jointly clustering all detections for all videos is computationally costly. We adaptively sample the people detector according to their 2D proximity with other detectors and the speed of the detector within each tracklet. All close-by detectors are sampled. Detectors that can be linearly interpolated by others within the same tracklet are ignored. Unreliable detectors with less than 9 keypoints (partially occluded people) detected are also ignored.

We use the robust continuous clustering framework of Shah and Koltun [191] but explicitly enforce soft constraints from motion tracklets, mutual exclusive constraints, and geometry matching to link detections. Depending on the discrimination of u, the correct number of cluster can be automatically determined during the optimization process. This clustering is formulated as the optimization problem:

$$C = \min_{\mathbf{m}} \sum_{i=1}^{N} \|u_i - m_i\|_2^2 + \lambda \sum_{(p,q)\in Q} w_{p,q}\rho(\|m_p - m_q\|_2),$$

where, N is the number of people detectors, Q is graph connecting data points u_i, $\mathbf{m} = \{m_1, .., m_N\}$ are the representative of the input descriptors $\mathbf{u}$, λ is scalar balancing the maximum curvature between the data and the regularization during the optimization [191], and ρ is the German-McClure estimator. $w_{p,q} = \frac{\sum_i^N N_i}{N\sqrt{N_p N_q}}$, where N_i is the number of edges connecting x_i in Q, balances the strength of the connection (p, q).

In our settings, the graph Q is mutual k-NN graph [32]. To form Q, we first determine the similarity between tracklets by taking the median of the similarity score between all possible person descriptor pairs within the two tracklets. The number of nearest neighbors for each tracklet is chosen such that the distance between different tracklets is 2 times larger than the median of the tracklet self-similarity score. All detectors belonging to the same tracklet are connected with detectors of their k mutually nearest tracklets. We then prune connections that violate the multi-view triplets mined in Section 3.2.2.

Analysis of the Descriptor Benefits: Tab. 3.5 quantifies the performance of different descriptor learning algorithms by the number of clusters automatically determined by the algorithm, the Adjusted Rand Index (ARI)[4], and cluster accuracy for all detected people in both sequences. Using [191] on MTL descriptor performs best. However, for a known number of people, performing the classical K-means clustering on the MTL descriptor also yields comparable accuracy (the precise number of people is only available in [C]). This confirms the effectiveness of our descriptor learning, not the clustering algorithm.

[4] The ARI is a measure of the similarity between two clusters with different labeling systems and is widely used in statistics [103].

	[C]			[T]			[H]		
	C	ARI	Acc.	C	ARI	Acc.	C	ARI	Acc.
[191]+ P	21	.88	90.1	66	.85	86.8	86	.77	79.5
[191] + MTL	16	.97	98.3	45	**.94**	**95.1**	71	.85	88.1
Kmeans+MTL	16	**.98**	**98.7**	14	.87	88.2	60	**.87**	**88.7**

Table 3.5: Analysis of the clustering algorithms by the number of clusters C, ARI measure and clustering accuracy. Although all methods detected clusters than needed, they correspond to the small cluster of pedestrians who do not participate in the activity (often seen in [T]) or not fully visible bodies due to occlusion. Using [191] on our (MTL) descriptors performs best, achieving near perfect clustering accuracy (98.3% for [C] and 95.6% for [T]))).

3.4.2 Markerless Human Motion Capture

We build a pipeline for markerless motion tracking of complex group activity from handheld cameras. We first cluster the descriptors from all camera to obtain person tracking information. For each person (cluster), we wish to estimate a temporally and physically consistent human skeleton model for the entire sequence. This is achieved by minimizing an energy function that combines an image observation cost, motion coherence, and a prior on human shape:

$$E(\mathbf{K},\overline{\mathbf{L}}) = E_I(\mathbf{K}) + E_L(\mathbf{K},\overline{\mathbf{L}}) + E_S(\mathbf{K}) + E_M(\mathbf{K}),$$

where, $\mathbf{K}$ is the 3D location of the anatomical keypoints over the entire sequence, $\overline{\mathbf{L}}$ is the set of mean limb length for each person. The image evidence cost E_I encourages the image reprojection of the set of keypoints 3D position to be close to the detected 2D keypoints. The human constant limb length cost E_L minimizes the variations of the human limb length over the entire sequence. The left-right symmetric cost E_S penalizes large bone length differences between the left and right side of the person. The motion coherency cost E_M prefers trajectory of constant velocity [219]. The formulation for each of these terms are given in Tab. 3.6. We weight these costs equally.

We initialize $\mathbf{K}, \overline{\mathbf{L}}$ by per-frame RANSAC triangulation of the corresponding person obtained from the person from descriptor clustering and minimize $E(\mathbf{K},\overline{\mathbf{L}})$ using Levenberg-Marquardt optimizer [6]. Lastly, we fit the SMPL mesh model [141] to the skeleton to improve the visualization quality (See Section 2.4 for the fitting details).

Analysis of the Descriptor Benefits: As a baseline, we use the ground truth people association to perform a per-frame multi-view triangulation along with limb length symmetry constraints link this reconstruction temporally using ground truth person tracking for visualization. As shown in Figure 3.16, our method succeeds despite the strong occlusion and complex motion pattern. Quantitatively, we show 5 to 10X improvement over the baseline (see Tab. 3.7). We visualize the reprojection of 3D keypoints to all views for [C] in Figure 3.17. The reprojected points are close to the anatomical keypoints. These results validate our algorithm ability to perform accurate markerless motion capture completely in the wild.

$E_I(\mathbf{K})$	$\sum_{c=1}^{C} \sum_{t=1}^{F} \sum_{n=1}^{N} \sum_{p=1}^{18} \rho\left(V_c^{np}(t) \frac{\pi_c(K^{np},t)-k_c^{np}(t)}{\sigma_I} \right)$
$E_L(\mathbf{K}, \overline{\mathbf{L}})$	$\sum_{t=1}^{F} \sum_{n=1}^{N} \sum_{q \in Q} \left(\frac{\overline{L}^{nq} - L_c^{nq}(t)}{\sigma_L} \right)^2$
$E_S(\mathbf{K})$	$\sum_{t=1}^{F} \sum_{n=1}^{N} \sum_{(l,r) \in S} \left(\frac{L_c^{nl}(t) - L_c^{nr}(t)}{\sigma_S} \right)^2$
$E_M(\mathbf{K})$	$\sum_{n=1}^{N} \sum_{p=1}^{18} \sum_{i=1}^{F-1} \left(\frac{K^{np}(i+1) - K^{np}(i)}{\sigma_M^p \Delta(i+1,i)} \right)^2$

C: number of cameras
F: number of frames
N: number of tracked people
$\pi_c(K^p, t)$: projection matrix
$V_c^{np}(t)$: visibility indicator
$L^{nq}(t)$: 3D distance between two points
Q: keypoint connectivity set
S: corresponding left and right limb set
$\Delta(.,.)$: absolute time differences
σ_I: variation in 2D detection
σ_L: variation in bone length
σ_M^p: variation in 3D speed

Table 3.6: 3D human-aware tracking cost functions.

	[C]]		[T]	
	Baseline	Ours	Baseline	Ours
Length Dev. (cm)	8.0	1.5	13.6	1.5
Symmetry Dev. (cm)	9.1	1.2	10.2	1.4

Table 3.7: Comparison between per-frame 3D skeleton reconstruction using ground truth association and human aware tracking. Temporal integration and the physical body constrains improve the 3D skeleton stability by 5-10X.

	[C]		[T]	
	ARI	Acc.	ARI	Acc.
[246]+P+known #cluster	.82	85.3%	.76	78.5%
[246]+MTL+known #cluster	.89	91.7%	.81	82.4%
[246]+P+GT 3D location+known #cluster	.88	90.3%	.84	84.1%
[246]+MTL+GT 3D location+known #cluster	.96	98.2%	.87	89.8%
Ours:MTL+unknown # clusters	.97	98.3%	.94	95.6%

Table 3.8: Due to the inaccurate sparse association between detections (tracklet noise in [T]), wrong number of people (in [T]), and early commitment to perframe 3D human position estimation (sensitive to errors due to wrong/missing inliers in RANSAC), [246] is not as competitive as ours. Discriminative descriptor learned by MTL **outperforms** pretrained P regardless of the association algorithms. GT 3D is the estimated location with ground truth association.

r	[23]	Ours: No tracklets	Ours: Full
0.3	67.0%	71.6%	71.9%
0.5	74.1%	75.8%	76.2%

Table 3.9: MOT-Accuracy comparison for different threshold radius r on WILDTRACK [41]. Due to large number of negative samples, our method outperforms [23] even without using single-view tracklet for triplet generation. We observe modest gain in our full method because frequent occlusions and frame sub-sampling prohibit long single-view tracklets.

We also compare our tracking via clustering approach to current arts in multi-view people tracking of [246] on [C] and [T] (results on [H] is not possible due to calibration failure) in Tab. 3.8 and with [23], the current best published 3D tracker on the WILDTRACK[41] dataset, the current most challenging multi-view tracking dataset in Tab. 3.9[2]. In both cases, we observe clear improvements using our MTL descriptors. Note that while previous methods require *fixed* cameras (in [23]) and *known* number of people (in [246]) in addition to being synchronized and calibrated, our algorithm can perform long term tracking of group activities in the wild without any of aforementioned requirements. We show our 3D tracking and the projected skeleton to observed images for the WILDTRACK dataset in Figure 3.18. We observe that the 2D projection of the keypoints to all views corresponds well to the expected person anatomical keypoints and tracks people even through occlusions.

3.4.3 Semantic Cut for Multi-angle Video

For any complex group activity events, it is unlikely that any person in the scene is well recorded by one video stream. We leverage track3d 3D human skeleton and merge multiple video streams into a multi-angle video by selecting video chunks where the selected person is visible and most

[2] Due to strong inter-person occlusion, many correctly associated detection are seen by less than 3 cameras, which are discarded by our 3D reconstruction algorithm. Thus, we only show the MOT-Accuracy for reconstructed 3D skeletons.

frontal to the camera. Similar to Arve et al. [14], we model the selection of the camera on a trellis graph and seek for the smallest cost path transversing the graph. The nodes in this graph are frames of camera c and edges are the connection of all consecutive frames from all cameras.

Node cost: This cost is determined by the normal vector of the person torso $n(K)$ and the camera c viewing direction d_c and the distance of the projection of the skeleton K to image center and is written as:

$$E_n(c, f) = V_c(f)\Big(\lambda_1 n(K).d_c + \lambda_2 g(|\pi_c(K, f) - c_c|^2, \tau)\Big),$$

where $V_c(f)$ is a binary visibility indicator, $\pi_c(f)$ is the camera projection matrix, c_c is the 2D image center location, $g(., \tau)$ is a thresholding function, only penalizes the cost if it exceeds τ, and (λ_1, λ_2) are the weights between two error terms.

Edge cost: This cost function is weighted combination of constant cost γ penalizing rapid camera switching and transitability cost penalizing camera switching during action. We determine the transitability by the instantaneous velocity of the skeleton. Our edge cost is written as:

$$\begin{aligned} E_e(c_i(f), c_j(f+1)) = \lambda_3 \left| \frac{K(f+1) - K(f)}{\Delta(f+1, f)} \right|^2 \\ + \lambda_4 \gamma [c_i(f) \neq c_j(f+1)], \end{aligned} \tag{3.1}$$

where $[.]$ is the Iverson bracket, and (λ_3, λ_4) are the weights between two error terms.

We compute the smallest path using Dijkstra's shortest path algorithm. Figure 3.19 shows a camera switching case where it correctly switches the camera upon inter-human occlusion. Since this algorithm does not plan several steps ahead, we observe that abrupt camera switching still occurs. We smooth the path using an average filter in the post-processing stage.

3.5 Discussion

During the triplet mining for descriptor adaptation, the temporal offsets between cameras are estimated. The accuracy of the estimated offset depends highly on the relative speed of the moving objects and the camera framerate. In the extreme case of static scenes, although these offsets can be arbitrarily inaccurate, the multiview constraints is still satisfied. Thus, interestingly, we could mine multiple cross views tracklets of the same person despite inaccurate offset estimation, as long as those tracklets satisfy the multiview constraints.

We note the importance of accurate tracklet generation for descriptor bootstrapping. Noisy tracklets can severely degrade the descriptor discrimination. While more sophisticated algorithms could be used to improve the tracklet generation quality [59, 60], the problem may still remain for scenes with people wearing similar and textureless clothing. One prominent solution is the use of robust estimator for the distance metric loss under the graduated non-convexity framework [191, 258].

The ultimate assumption of our framework is accurate semantic keypoints detection. Unless these keypoints are well localized, our multiview synchronization and triplet mining will break down. While this assumption is acceptable given recent dramatic advances in human keypoint detection [36, 67, 96, 163, 237], the detection are much less accurate for other objects, such as vehicles. We develop an algorithm to combine local unstructured single-view tracking and multiview structured semantic keypoints for cars to improve both 3D motion tracking and semantic detection in Chapter 4.

3.6 Summary

We have presented a simple but powerful framework for scene-adaptive person descriptor. This is demonstrated on challenging scenes captured by mobile cameras. The learned descriptor reliably associates the same person over distant space and time instances. Our descriptor outperforms the baseline by 18% and our 3D skeleton reconstruction is 5-10X more stable than naive reconstruction even with ground truth people correspondences on events captured in the wild. Our algorithm works even with few cameras. This enables has potential applications to broadcast sport (e.g, basketball or football) with domain adaptation using prior *unlabeled* footage as fine-tuning on a small subset of the test sequence suffices for generalization.

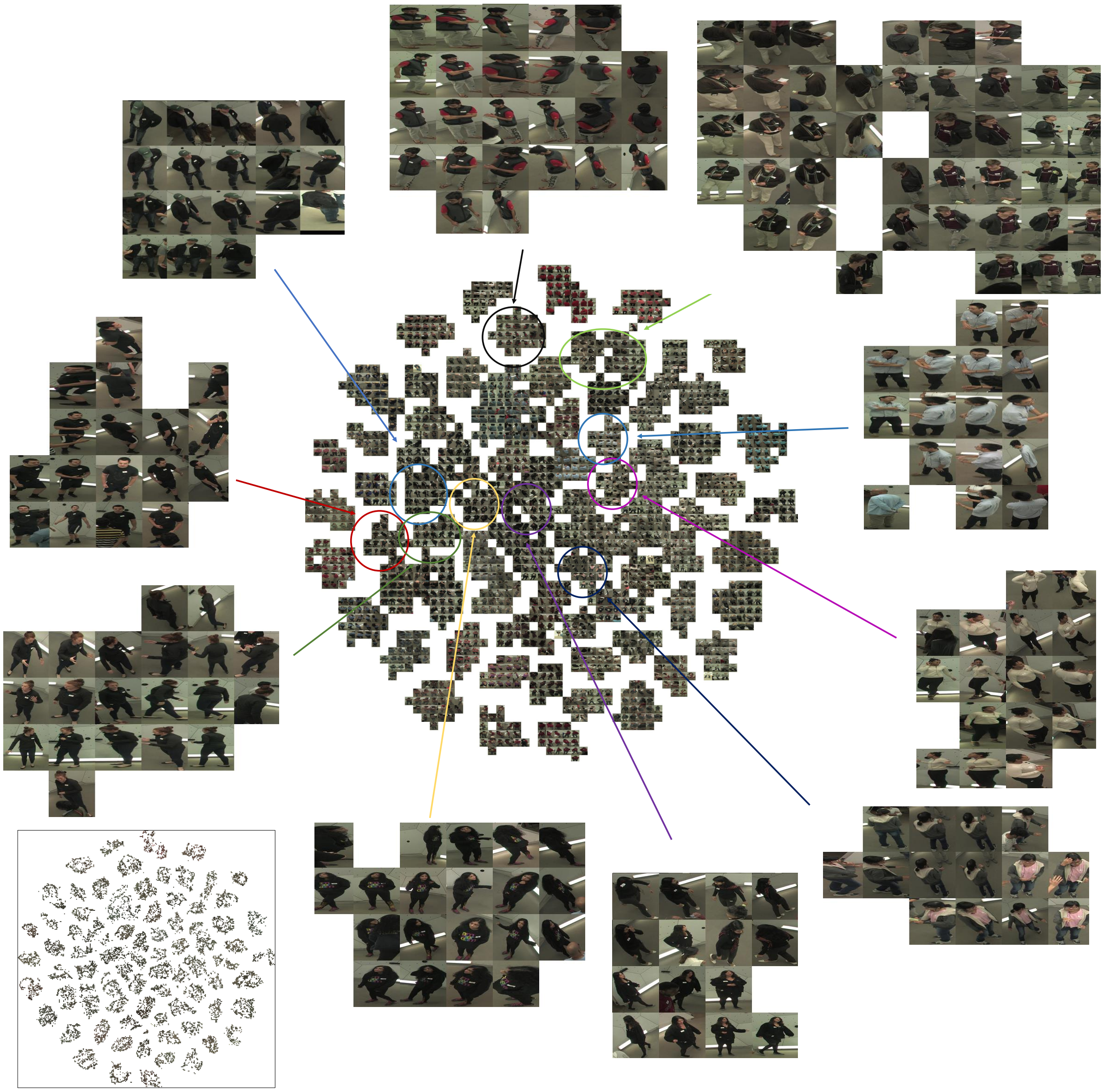


Figure 3.9: The Barnes-Hut t-SNE visualization of our embedded descriptor for 30k images of 80 people collected by the CMU Panoptic studio. Despite similar appearances, the images of the same person are clustered together. We magnified the image resolution of the original t-SNE clusters for better visualization. This figure is best viewed on a monitor when zoomed in.

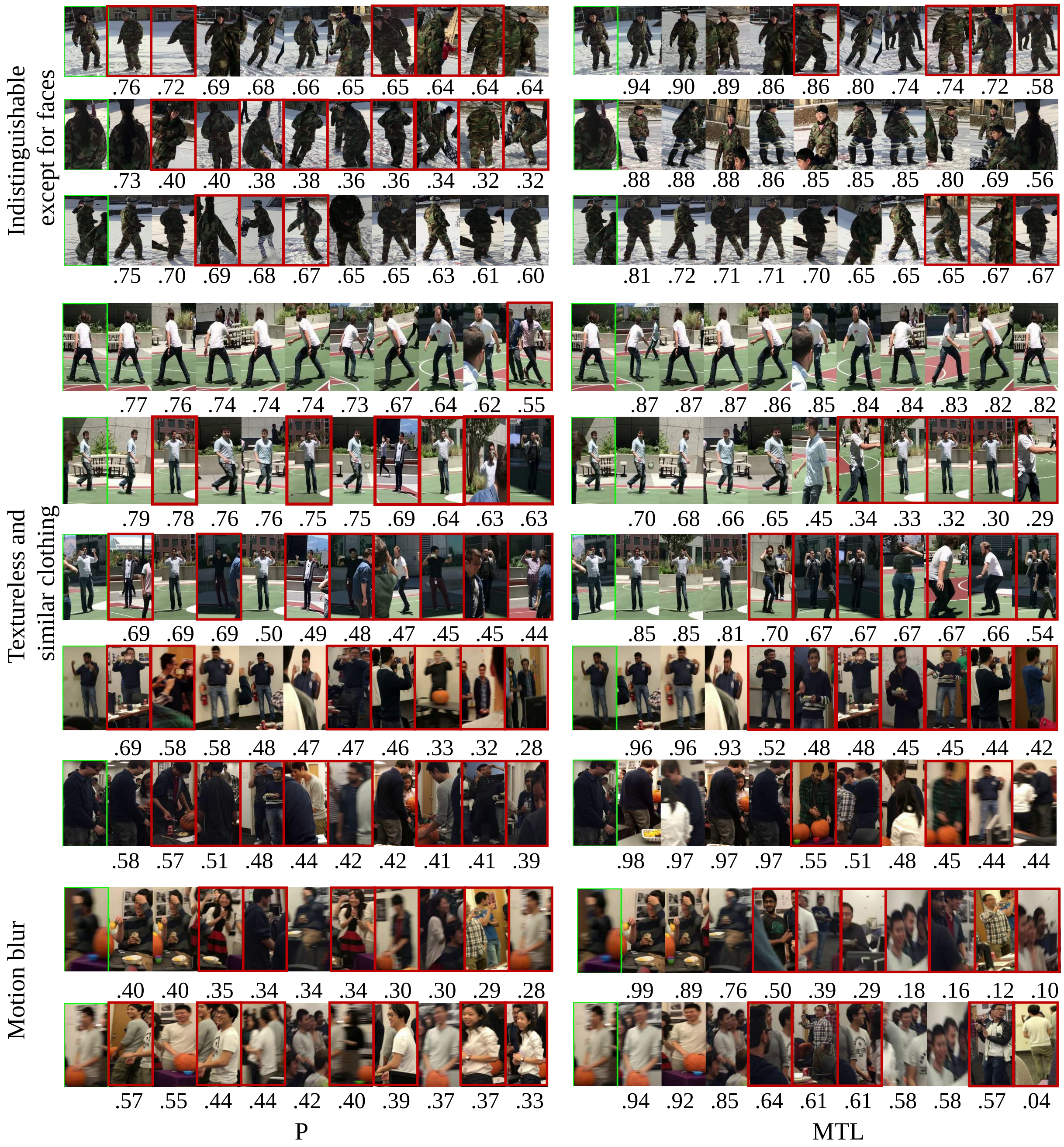


Figure 3.10: 10-NN cross-view matching of the several people with confusing appearance and their cosine similarity score using the pretrained model and our multitask descriptor learning (MTL). Green denotes the query and red denotes incorrect matches. We label the query in green and wrong association in red. Our method retrieves more positive matches and provides easy-to-separate similarity score. All top three neighbors are of the same person.

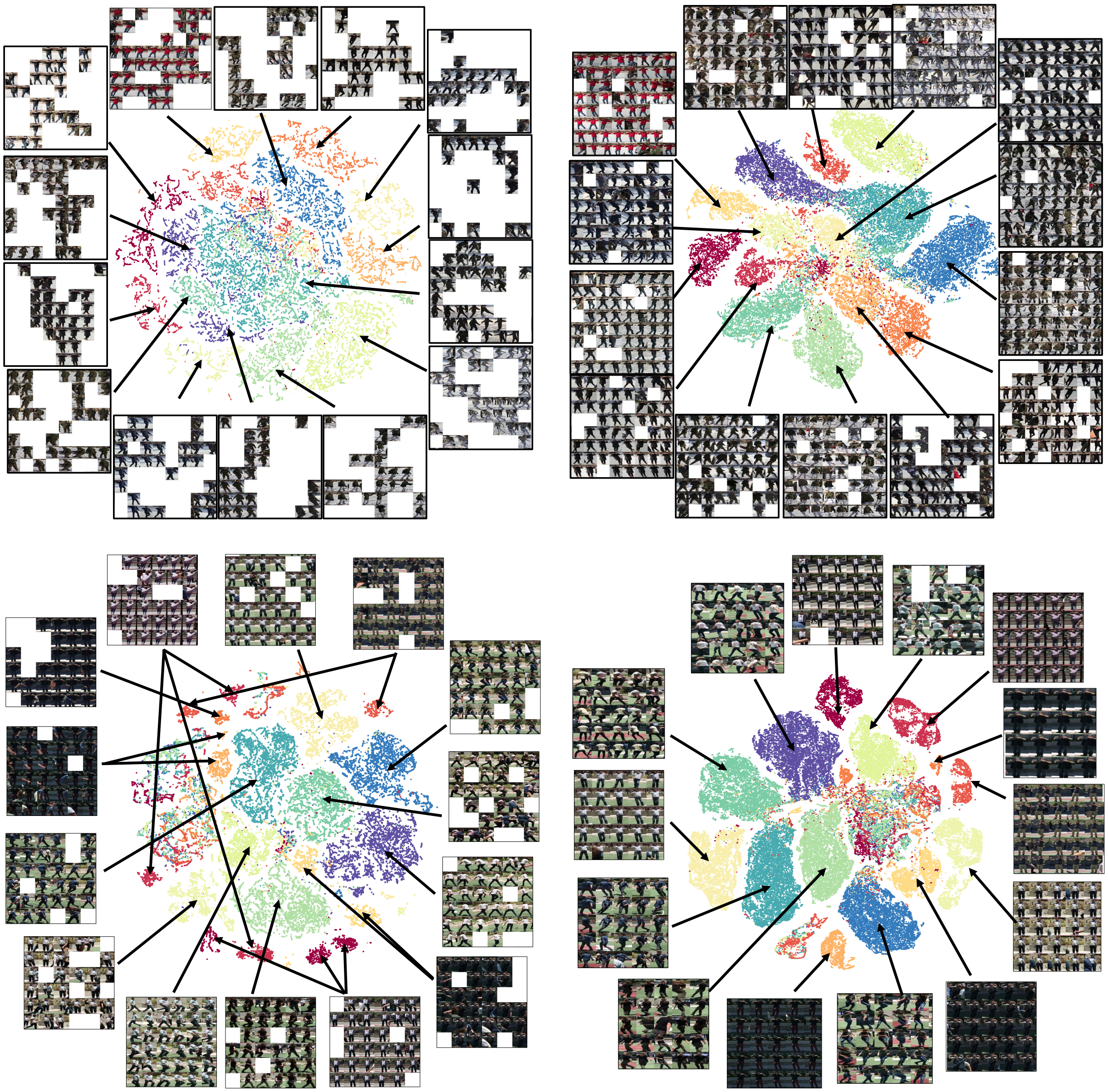


Figure 3.11: t-SNE visualization of the person descriptor extracted using a pretrained model and our multitask learning (MTL) for sequence [C]. Except for images of the same tracklet within a single view, the pretrained descriptors are scatter. Our descriptor groups images of the same person from all views and time instances into cleanly separated clusters. See Figure 3.14 for extra quantitative evidences.

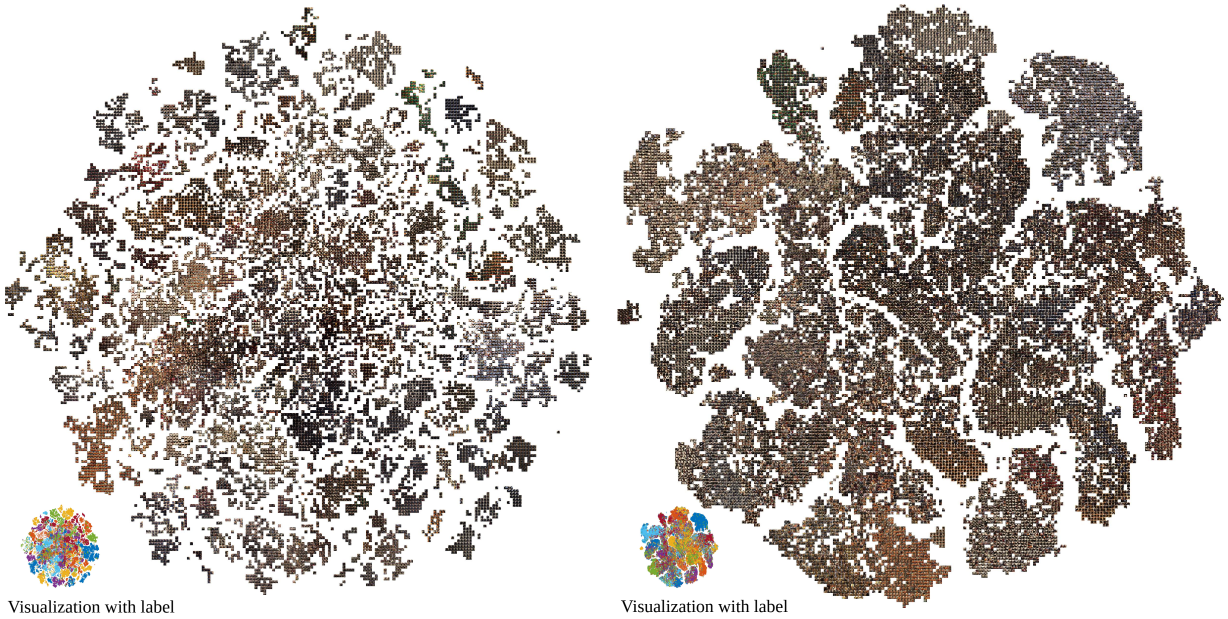


Figure 3.12: t-SNE visualization of the person descriptor extracted using a pretrained model (left) and our multitask learning model (right) for sequence [H]. Despite being a very complex scene with high number of people, our proposed algorithm shows better discrimination and the same person is better grouped into a single cluster.

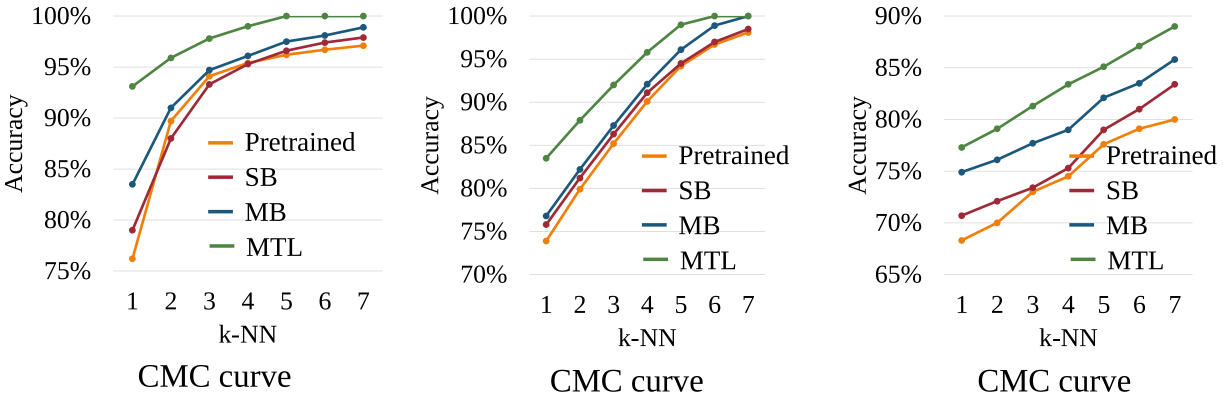
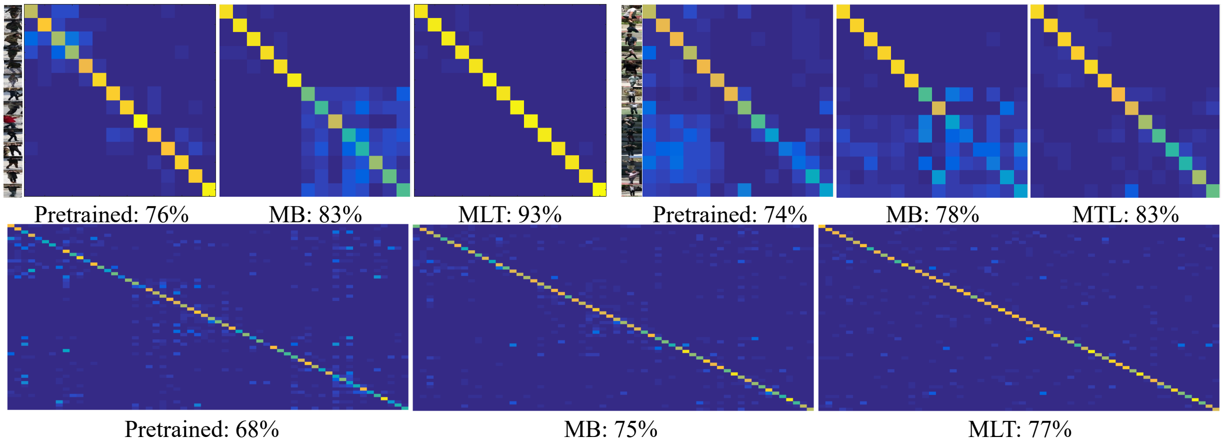


Figure 3.13: The CMC for the Chasing (left), Tagging (middle), and Halloween (right) scene at different stage of our algorithm. Our method outperforms the pretrained model at every stages.


Pretrained: 76% MB: 83% MLT: 93% Pretrained: 74% MB: 78% MTL: 83%

Pretrained: 68% MB: 75% MLT: 77%

Figure 3.14: The confusion matrix of the top-1 matches for the all sequences ([C] top left, [T] top right, [H] bottom) at different stages: pretrained model, multi-view bootstrapping (MB), and multitask learning (MTL). There are consistent improvements in accuracy as more sophisticated stage is executed.

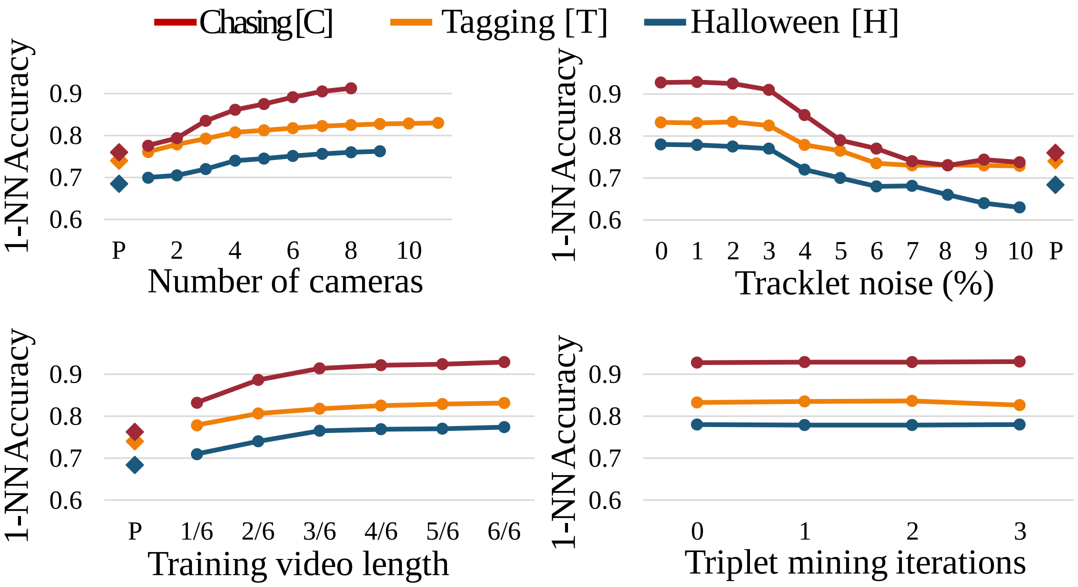
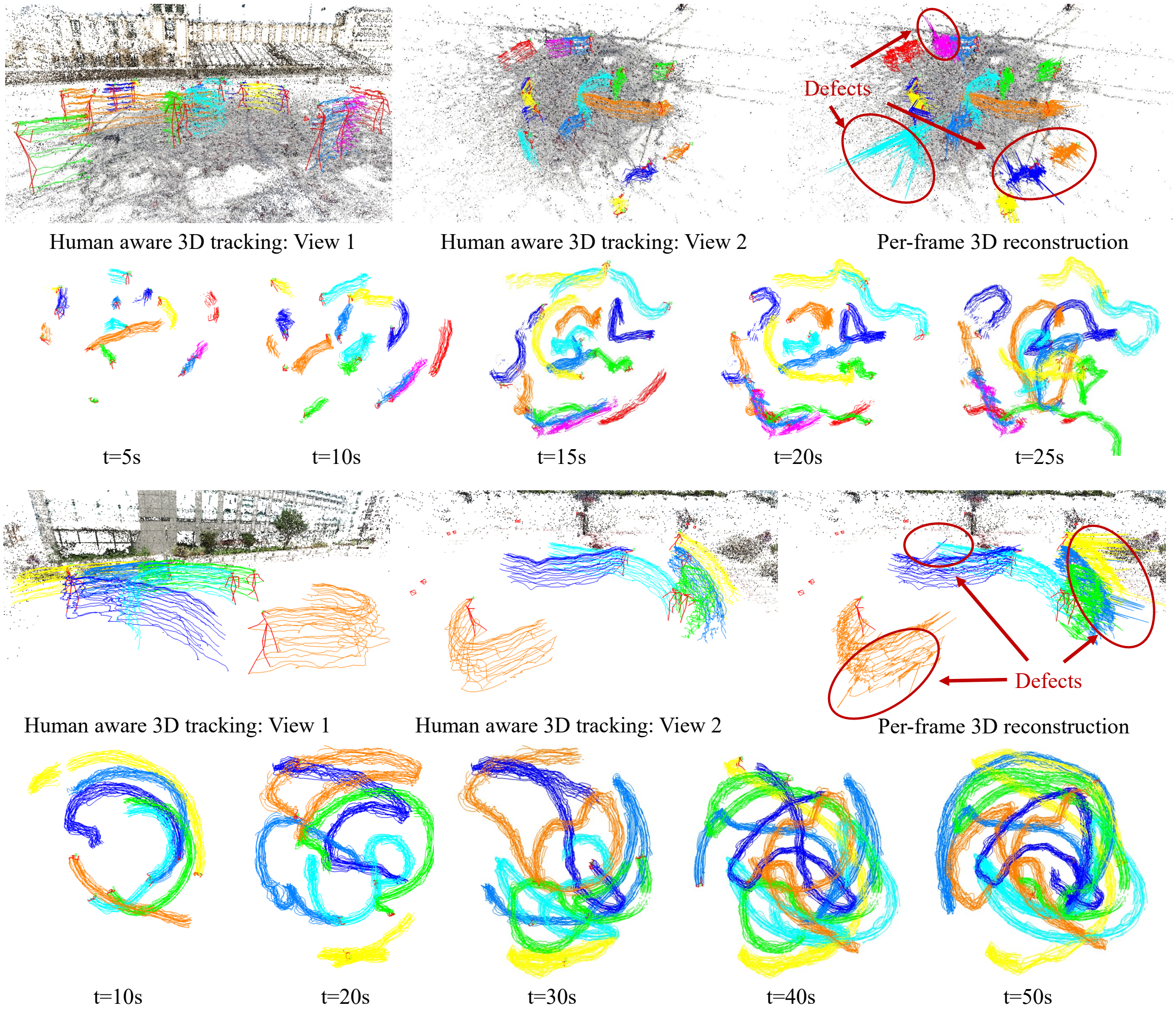


Figure 3.15: 1-NN matching accuracy analysis of the proposed method for a different number of cameras, percentage of tracklet noise (two or more people grouped in 1 tracklet), fraction of domain data required for generalization, and triplet mining iterations. P denotes the pretrained model. Please refer to the text for the details.



Human aware 3D tracking: View 1

Human aware 3D tracking: View 2

Per-frame 3D reconstruction

t=5s

t=10s

t=15s

t=20s

t=25s

Human aware 3D tracking: View 1

Human aware 3D tracking: View 2

Per-frame 3D reconstruction

t=10s

t=20s

t=30s

t=40s

t=50s

Figure 3.16: 3D tracking for [C] (top) and [T] (bottom) for the entire event. Owing to accurate association, our method gives smooth and clean trajectories despite strong occlusion, similar people appearance, and complex motion pattern.

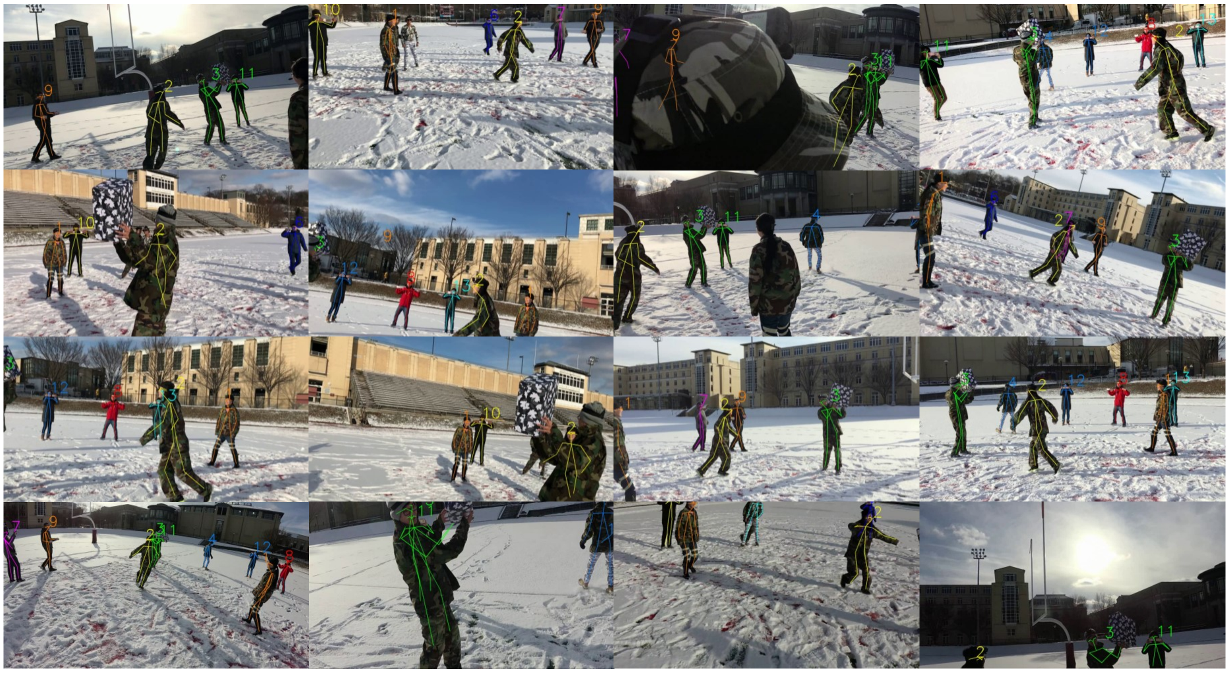


Figure 3.17: The 2D projection of the keypoints to all views corresponds well to the expected person anatomical keypoints and tracks people even through occlusions.

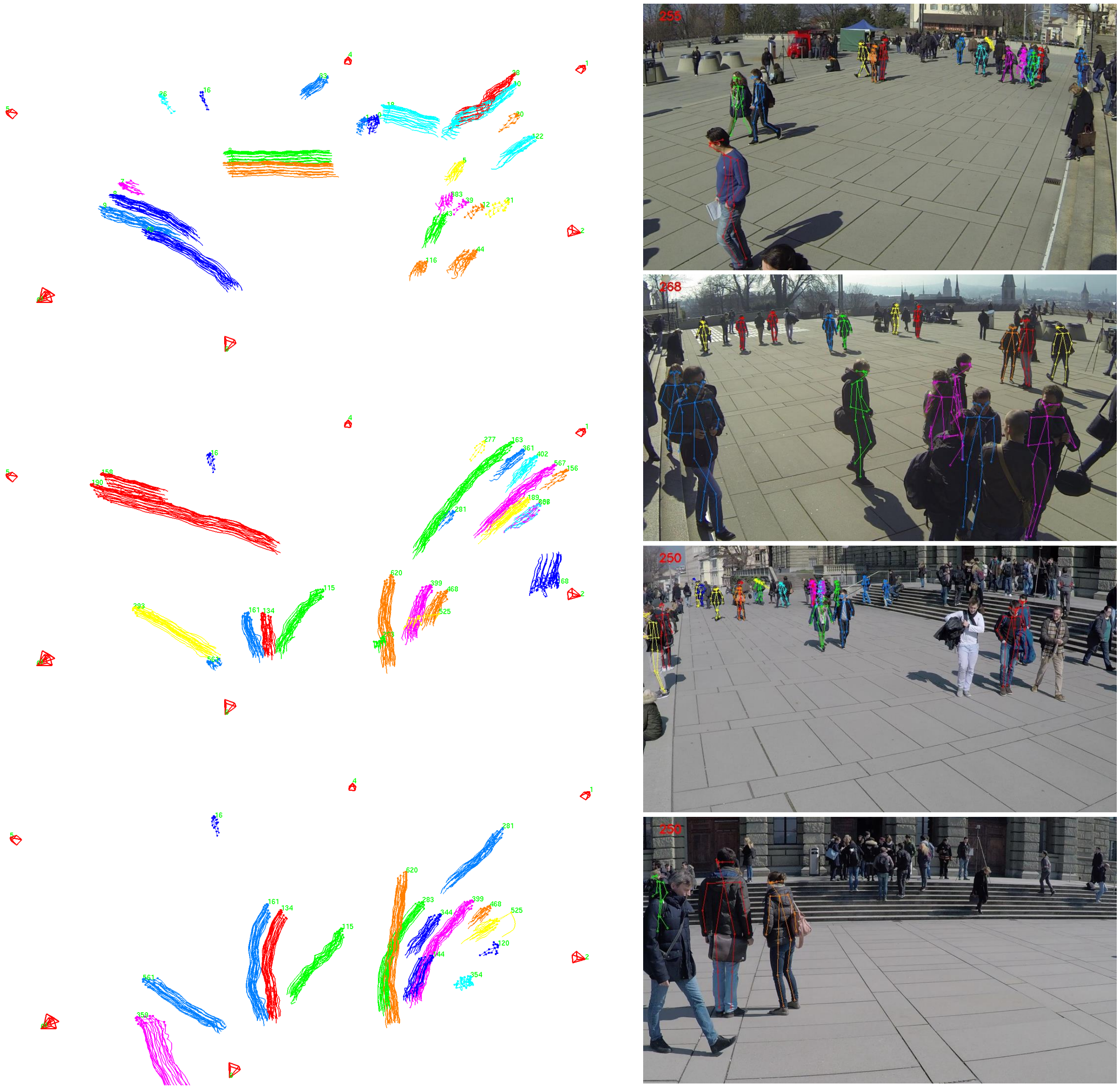


Figure 3.18: 3D tracking from 7 cameras of the WILDTRACK dataset. Owing to the accurate association, our method gives smooth and clean trajectories despite strong occlusion, similar people appearance, and complex motion pattern. The 2D projection of the keypoints to all views corresponds well to the expected person anatomical keypoints and tracks people even through occlusions.

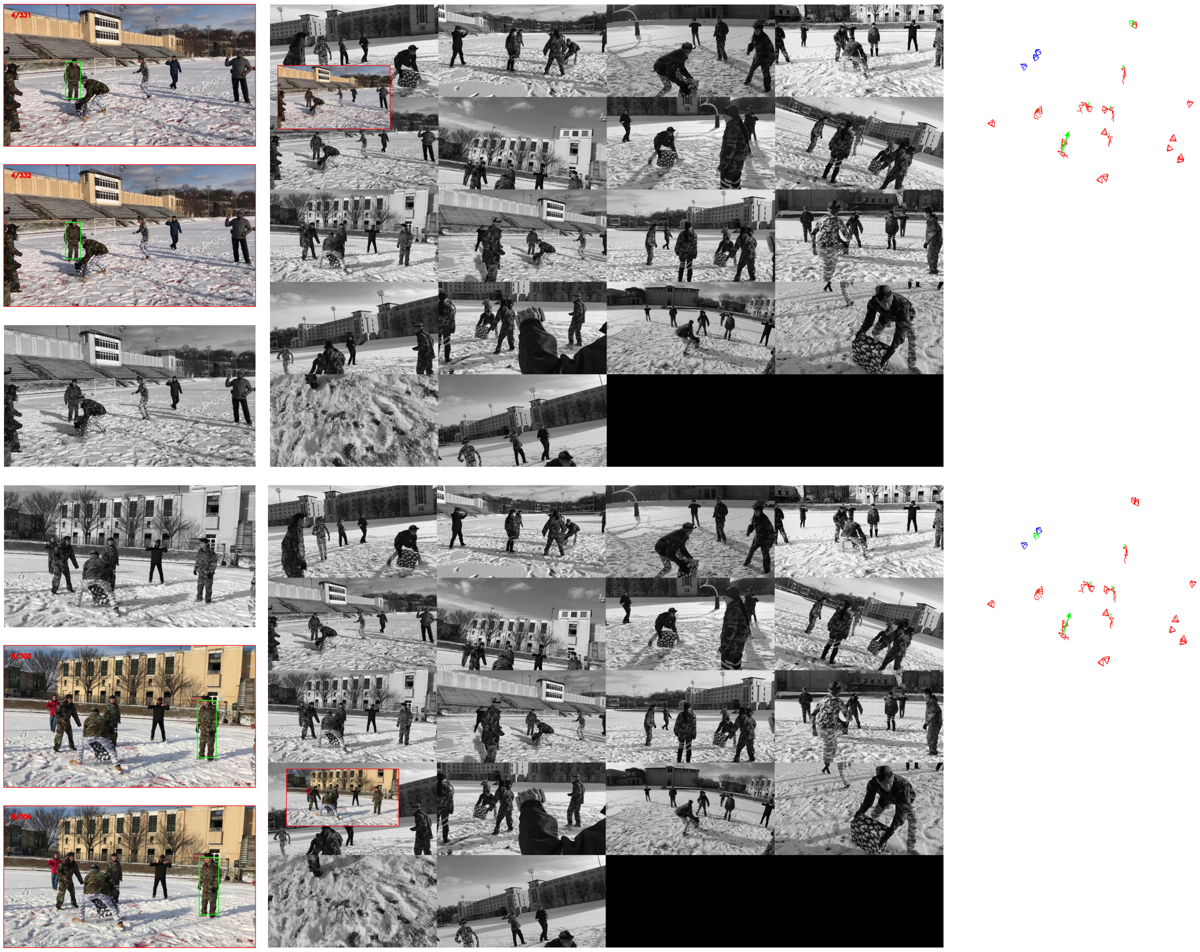


Figure 3.19: A visualization of two shots cut created by our algorithm for the Chase sequence. The red square indicates the selected camera (frame). The top row shows the input images from all camera at a particular time instance and the corresponding 3D view. The tracked person is highlighted in the green bounding box. The green arrow shows his front-facing direction. All visible cameras are shown in blue and the selected camera is shown in green. The bottom row shows consecutive frames of the final video. The not selected frames are shown in grayscale. Our algorithm detects and switches the camera upon inter-human occlusion.

Chapter 4

CarFusion: Fusion of Motion and Semantic Priors via Object Triangulation

The increasing availability of multiview camera systems at urban traffic intersections provides us a strong opportunity to reconstruct moving vehicles crossing those intersections. Even coarse shapes and motions of the vehicles can be invaluable to traffic analysis, including vehicle type, speed, density, trajectory and frequency of events such as near-accidents, that can be fed to (semi-) autonomous vehicles approaching the intersection. That said, reconstructing moving vehicles in a busy intersection is hard because of severe occlusions. Moreover, the cameras are often unsynchronized, have little overlap in fields of view, and need to be calibrated each frame as they are often not rigidly attached and sway because of wind or vibrations.

Given recent advances in semantic keypoints detection of vehicles (wheels, headlights, doors, etc.) [130, 163, 225], one naive approach to reconstruct the moving vehicles is to triangulate the detected 2D keypoints. Unfortunately, current car keypoint detector significantly lags the human keypoint counterpart due to the lack of large-scale dataset. Thus, the detected keypoint locations are often incomplete and not precise enough to directly apply triangulation-based 3D reconstruction methods in the presence of occlusions. For the same reason, tracking via per-frame detection is not stable enough to be useful for structure-from-motion approaches. On the other hand, in spite of significant improvement on feature point tracking [18, 209] in structure-from-motion/SLAM [58, 119, 161, 162], corresponding these features across wide-baseline views is near impossible given that each camera sees only parts of a vehicle (front, one side, or back) at any given time instant. Currently, there are no methods for consistent 3D reconstruction of moving vehicle under heavy occlusion.

In this chapter, we present a comprehensive framework for 3D reconstruction of moving vehicles even in severe occluded scenarios by fusing single-view *unstructured* motion tracking and semantic *structured* keypoints detection. Our insight is that the imprecise but accurate structured keypoint is *rigidly* tight to the sparse but precise single-view unstructured tracks over the entire course of motion. No spatial correspondences are needed for the unstructured points. No temporal correspondences are needed for the structured points. We optimize for both the 3D points and the car motion over all viewing cameras jointly using bundle adjustment. We call this frame-

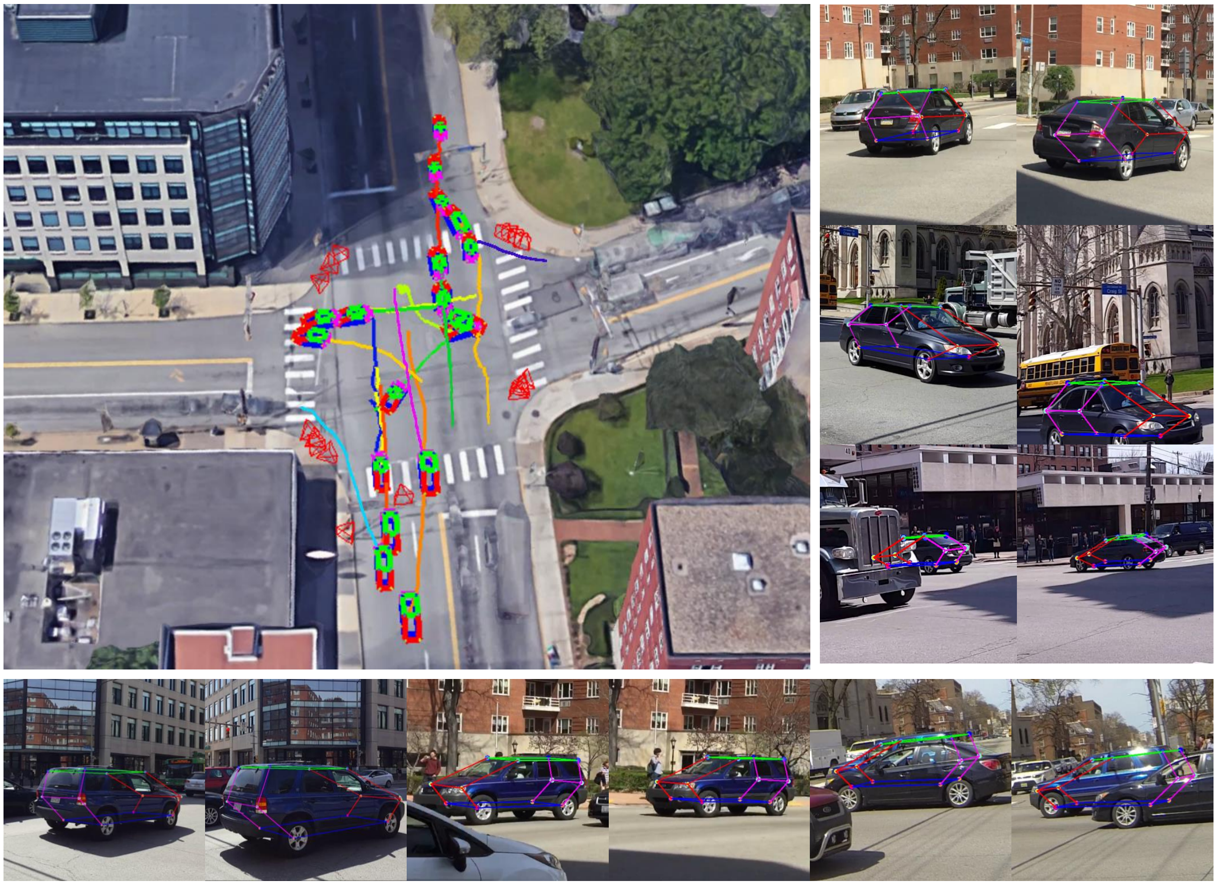


Figure 4.1: Reconstruction of vehicles crossing a busy intersection, making turns, going straight and changing lanes. A subset of vehicle skeletons (3D detector locations) and their 3D trajectories are augmented within the Google Earth view of the intersection. The reconstructions are reprojected into multiple views of two cars (a sedan and an SUV) demonstrating good performance under partial occlusions.

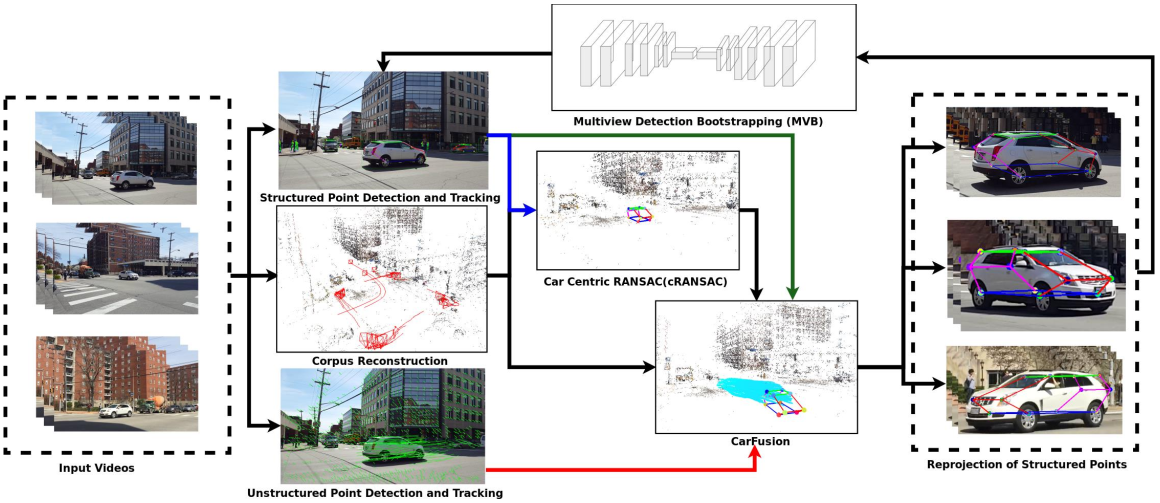

Multiview Detection Bootstrapping (MVB)
Structured Point Detection and Tracking
Car Centric RANSAC(cRANSAC)
Corpus Reconstruction
CarFusion
Input Videos
Unstructured Point Detection and Tracking
Reprojection of Structured Points

Figure 4.2: CarFusion: Our overall pipeline for dynamic 3D reconstruction of multiple cars from uncalibrated and unsynchronized video cameras. We fuse the structured points (detected vehicle parts) and the tracks of the unstructured feature points to obtain precise reconstruction of the moving vehicle. The reconstructions are reprojected into all the views and are used to bootstrap and improve the detectors.

work as "CarFusion" and it consists of three main stages: (1) a novel object-centric (as opposed to feature-centric) RANSAC approach to provide a good initialization of the 3D geometry of the structured points of the vehicle, (2) a novel algorithm that fully exploits the complementary strength of the structured and unstructured points via rigidity constraints, and (3) closing-the-loop by reprojecting the reconstructed structured points to all views to retrain the part detectors. A schematic overview of our system is illustrated in Figure 4.2.

We demonstrate the reconstruction of vehicles at a busy intersection shown in Figure 4.1. About 62 vehicles were detected, tracked and reconstructed within a 3-minute duration captured from 21 handheld cameras that are uncalibrated and unsynchronized and were panning to cover wider fields of view. A subset of vehicle structured point trajectories is augmented within the Google Earth image of the intersection. They include cars of different types (sedans, SUVs, hatch-backs, jeeps, etc.) making left and right turns, going straight-ahead as well as changing lanes. Several views of two specific cars in various occluded scenarios are shown with the reprojections of the structured points.

We also validate the performance of each stage of our framework and compare our approach to alternative methods that rely only on tracking-by-detection or feature-based structure-from-motion. By treating structured and unstructured points in a unified framework, we are able to show significant improvements in vehicle detection rates, vehicle trajectory lengths (or tracks) and reconstruction accuracies. Our approaches are designed to handle partial occlusions but fail when a vehicle is mostly occluded at all times. The estimated 3D vehicle tracks are accurate but slightly wobbly and will benefit from additional domain-specific priors.

Contributions: We develop a framework to fuse both the single-view feature tracks and multi-view semantic keypoints to significantly improve the detection, localization, and reconstruction of rigid dynamic objects even in the presence of strong occlusions.

4.1 Related Work

The literature on 3D reconstruction of vehicles can be largely classified into two categories: coarse, object-centric reconstruction using a single image or monocular video and dense reconstruction using multiview stereo. Unlike works that employ different sensor modalities [45, 129], our work is purely based on RGB cameras and thus, we only review methods using RGB sensors.

Single-View Reconstruction in the Wild: Reconstructing 3D information from a single view has been the subject of study for multiple decades. The earlier approaches assumed an isolated object for analysis similar to a projection of a CAD model on a plain background [29, 30, 180]. With the onset of better recognition algorithms [125, 128, 233], recent works utilize state-of-the-art object detectors [82] and instance segmentation [96, 244] algorithms to isolate an object, and follow various recipes to extract 3D information [21, 159, 212]. Multi-stage pipelines involve detecting and segmenting objects in the scene, estimating 3D poses, fitting shape models to the segment masks, enabling coarse to fine improvement [115, 158, 171]. Notably, Xiang et al. [234] estimated 3D voxels of the object directly from detection and segmentation results instead of estimating viewpoints and keypoints. Approaches that regress depth from a monocular video have also been explored [83, 259]. In general, these approaches produce coarse and category specific reconstruction (e.g., car, chair). The reconstruction is also potentially geometrically inconsistent if re-projected to multiple views.

Active-Shape Model Reconstruction: Many works have been motivated by using active shape models [54] for vehicle reconstruction [262, 263]. These algorithms exploit strong part detectors learned from CNNs [40, 130, 137, 163, 225] and deform the shape model to fit the observations. Recent works have also combined SLAM with active shape priors for the reconstruction of objects [49]. In general, these approaches produce more detailed 3D shape than those with category-specific reconstruction.

Multi-view reconstruction: Multiview stereo systems are widely used in the context of vehicle reconstruction for both dense shape and velocity estimation [26, 87, 152, 153]. These approaches exploit cues from 2D bounding box detection, image instance segmentation or object category shape to regularize the stereo disparity for large displacement and textureless/glossy regions. Our work also employs multiple cameras but reconstructs both the car skeleton and sparse trajectories of the car body using 3D priors on symmetry, link length, and rigidity constraints. Our multiview detector bootstrapping is similar in spirit to Simon et al.[195] for hand keypoint detection. However, their work is conducted in a laboratory studio equipped with a massive number of cameras and the method can produce a good hand skeleton using multiview triangulation alone. Our work is "in the wild" where stable vehicle reconstruction is hard even with ground truth correspondences.

4.2 Multiview Reconstruction of Moving Cars

Consider C video cameras observing M rigidly moving cars over F video frames. At any time instance f, the car m has a fixed number of structured points $S_m(f)$ and an arbitrary number of unstructured points $U_m(f)$. The structured points are semantically meaningful 3D locations of parts on the car. They can be reliably but imprecisely detected and can be matched to different images at any time instance. The unstructured points are the 3D locations of the local features (say, Harris corners) in the observed image. They can precisely be detected and reliably matched only within the same video. Their 2D locations are $s^c_m(f)$ and $u^c_m(f)$, respectively. The motion of an individual car is characterized by a rigid transformation $[R_m(f), T_m(f)]$ at frame f. Denote $x^c(f) = \pi^c(X(f))$ as the image projection of an arbitrary 3D point X to camera c at time f by the camera projection matrix $\pi_c(f)$. The visibility of X in camera c is given by $V^c(X(f))$. We assume all the cameras are globally calibrated with respect to the static background and temporally synchronized. The 3D locations of the unstructured points can be computed using SfM algorithms [187], which are later transformed to the global coordinate system of the static background. We note that each set of unstructured points computed from different cameras is up to an unknown scale, which is optimized jointly during the global optimization process. Our goal is to precisely estimate and track the 3D configurations of the structured points.

4.2.1 cRANSAC: Car-Centric RANSAC

To reconstruct the vehicle from multiple views, we must find correspondences across views first. We propose a car-centric RANSAC procedure for finding such correspondences. Compared to common point-based RANSAC [94], we consider the entire car as a hypothesis, which allows explicit physical constraints on the car link length and its left-right symmetry to be enforced. Due to the uncertainty in detecting the 2D location of the structured points from different views, these additional constraints are needed for reliable multiview correspondence estimation.

Concretely, consider a set of 2D car proposals $\boldsymbol{h}(f) = \{h^1(f), ..., h^c(f)\}$ available from all the cameras at frame f. Each proposal consists of a set of structured points s^c_m. We want to find a set $\boldsymbol{g_m} = \{g^1_m, ..g^c_m\}$, where $g^i_m \in h^i$, for every car m visible in the cameras. At every RANSAC iteration, we sample proposals within a triplet of cameras with sufficiently large baselines and triangulate the hypothesis to obtain $S_m(f)$. These points are back-projected to all cameras to find a better hypothesis g_m. We optimize a car-centric nonlinear cost function E_C and prune proposals with large error within g_m. This procedure is applied for a fixed number of iterations. The hypothesis with the largest number of elements is taken as the inlier proposal for that car. These proposals are removed from the proposal pool h and the process is restarted until no good hypothesis is left. The car-centric cost function is defined as:

$$E_C = \alpha_I E_I + \alpha_S E_S + \alpha_L E_L \tag{4.1}$$

where, $\{E_I, E_S, E_L\}$ are the image evidence cost, car link length symmetry cost, and car link length consistency cost, respectively, and $\{\alpha_I, \alpha_S, \alpha_L\}$ are the weights balancing the contributions of each cost. The cost functions are described below.

Image evidence cost: This cost function penalizes the deviation between the 3D projection of a point and its detected 2D location:

$$E_I(f) = \sum_{c=1}^{C} \sum_{p=1}^{S_m} V_p^c(f) \rho \bigg(\frac{\pi^c(S_p(f)) - s_p^c(f)}{\sigma_I} \bigg),$$

where, ρ is the Tukey Biweight estimator and σ_I is the deviation in 2D localization of the structured point x_p^c.

Link length consistency cost: This cost incorporates prior information about the expected length of two structured points and penalizes the deviation of the estimated length with respect to its mean:

$$E_L(f) = \sum_{\{p,q\} \in \mathbf{L}} \rho \bigg(\frac{L_{p,q}(f) - \overline{L}_{p,q}}{\sigma_L} \bigg)^2,$$

where, $L_{\{p,q\}}$ is the Euclidean distance between two structured points $\{p, q\}$, in the connectivity graph $\mathbf{L}$, and its expected length $\overline{L}_{\{p,q\}}$, defined based on the vehicle type, e.g. sedan or truck, and σ_L is the expected variation in length.

Left-right symmetry cost: We penalize large differences between the left and right link length of the car. This constraint is useful in fusing detectors visible from the opposite side in other views. This cost function is given as:

$$E_S(f) = \sum_{\{l,r\} \in S} \bigg(\frac{L_l(f) - L_r(f)}{\sigma_s} \bigg)^2,$$

where, S is the set of corresponding left and right links, and σ_S is the expected variation in the left and right link lengths.

We rescale the SfM reconstruction into metric units and set $\{\sigma_I, \sigma_L, \sigma_S\}$ to $\{10, 1.5, 0.1\}$, $\{\alpha_S, \alpha_S, \alpha_L)\}$ to $\{1, 1, 0.5\}$, respectively for our experiments.

4.2.2 Fusion of Structured and Unstructured Points

By exploiting the physical constraints on link length and left-right symmetry, we can estimate plausible 3D configurations of S_m from multiple wide baseline cameras at any time instances. Yet, these estimations remain spatiotemporally unstable due to large uncertainty in detected locations of structured points. On the other hand, the unstructured points can be detected and tracked precisely for every camera. However, it is difficult to reliably establish correspondence between

```
Input: $\{s^c_m(f), u^c_m(f)\}$, $\pi^c(f)$, $\boldsymbol{h}(f)$
Output: $\{S_m(f), U_m(f)\}$,$\{R_m(f), T_m(f)\}$
repeat
    while No more cars available do
        while Inliers $<$ Min Inliers do
            repeat
                $g_m \leftarrow$ Sample $h$ from three cameras;
                $S_m \leftarrow$ DLT($g_m$);
                $g_m \leftarrow$ Project $S_m$ to all cameras;
                $g_m \leftarrow$ Optimize Eq. 4.1 and prune $g_m$;
                if $g_m > g_{mbest}$ then
                    $g_{mbest}$ = $g_m$;
                end
                iter++;
            until iter $<$ Max Iteration;
        end Section 4.2.1
        Remove $g_m$ from $h$;
        Reconstruction $U_m(f)$ objects ;
        Optimize Eq.4.2 for $S_m(f)$, $\{R_m(f), T_m(f)\}$ (Section 4.2.2);
    end
    Project $S_m$ and retrain the detector (Section 4.2.3);
until iter $<$ Max Iteration;
```

Algorithm 2: CarFusion algorithm

unstructured points across cameras due to large viewpoint changes.

Our fusion cost combines the complementary strengths of the structured and unstructured points using rigidity constraints. It enables precise and spatiotemporally stable estimation of the 3D configuration of the structured points. This cost function is formulated as:

$$\begin{aligned} e(f) &= \left(\frac{\|R_m(f)S_i^c(f_s)+T_m(f)-\lambda^c U_j^c(f))\|_2-\|(S_i^c(f_s)-\lambda^c U_j^c(f_s))\|_2}{\sigma_R} \right)^2, \\ E_R &= \textstyle\sum_c^C \sum_f^F \sum_j^{U_m^c} \sum_i^{S_m^c} e(f), \end{aligned}$$

where λ_c is the global unknown scale factor for estimating the unstructured points from camera c, σ_R, set to 0.1, is the expected deviation from rigid deformation of the car 3D configurations over time, and f_s is the frame where the car is first reconstructed (with sufficient inliers) using our RANSAC algorithm. We initialize λ_c using the average scale of the structured points. Our formulation links structured and unstructured points between all the visible cameras seamlessly over space and time. The cost function promotes fixed distances between the structured and unstructured points (definition of rigid motion) during the course of motion. No spatial correspondences are needed for the unstructured points. No temporal correspondences are needed for the structured points.

Since the car motion is a rigid transformation, we explicitly enforce this constraint into the image evidence cost and integrate it over all time instances:

$$\begin{aligned} e(f) &= \rho\left(\frac{\pi^c(R_m(f)(f)S_m(0)+T_m(f))-s_p^c(f)}{\sigma_I} \right) \\ E_{I2} &= \textstyle\sum_{c=1}^C \sum_f^F \sum_{p=1}^{S_m} V_p^c(f)e(f), \end{aligned}$$

We then optimize the following total cost for precise 3D reconstruction of each car:

$$E = \min_{\mathbf{S}_{m(t_0)},\overline{\mathbf{L}}_m,\{R_m(f),T_m(f)\}} E_{I2} + E_S + E_L + E_R, \tag{4.2}$$

where, $\overline{\mathbf{L}}_m$ is set of mean link lengths and is initialized using mean of the 3D configurations S_m estimated in Sec. 4.2.1.

For efficiency, we start the reconstruction of each vehicle progressively, starting from the first time when our RANSAC detects the 3D object, and optimize Equation 4.2 for its structured point trajectories. The reconstructed cars are removed from the hypotheses pool. We iterate this process until no more cars can be reconstructed. Please refer to Algorithm 2 for the entire process.

4.2.3 Multiview Detection Bootstrapping

Precise and temporally stable 3D reconstruction of the car from multiple views can bootstrap the 2D detection of the structured points (loop-back shown in Figure 4.2). In turn, better 2D localization of the structured points enables more precise 3D estimation of the car. Given the 3D locations of structured points and their visibilities, we project the 3D points onto all the views. We use the reprojected points as automatically computed labels for fine-tuning the car detector. We recompute the reconstruction using the improved detectors for better fitting of the structured points and further minimization of the reprojection error. The emphasis is to improve detections using reconstruction and vice-versa from cameras captured in the wild.

4.3 Analysis

We evaluate our framework on a traffic scene captured with six Samsung Galaxy 6, ten iPhone 6, and six Gopro Hero 3 cameras at 60 fps in a busy intersection for 3 minutes. In total the algorithm is run on nearly 210000 frames. These videos were captured by 13 people, some of whom carried two cameras. The sequence is challenging as there are no constraints on the camera motion or the vehicle motion in the scene. We manually annotate the 2D locations of the structured points for every visible cameras for 2793 frames from different viewing angles in the Intersection dataset.

We evaluate our reconstruction pipeline at its progressive stages: car-centric RANSAC (cRANSAC), temporal integration using only the structured points (T-cRANSAC), and the fusion of both structured points and unstructured track (CarFusion). The T-cRANSAC is the result of optimizing the cost function 4.2 but without the fusion term E_R. This method can be considered as reconstruction using tracking-by-detection. We also compare the evolution in accuracy of the 2D structured point detector before and after the multiview bootstrapping. We use the Stacked Hourglass architecture [163] referred to as ”pretrained”, to detect the structured points. The same architecture is used for finetuning with the point labels obtained using our multiview reconstruction. The finetuned detector is referred to as MVB (multiview bootstrapping).

4.3.1 Data Pre-processing

Structured points detection and tracking: We used the FCIS model [244] to obtain the car proposal hypotheses. For each hypothesis, The structured points are 14 car keypoints, obtained by training the Stacked hourglass CNN architecture [163] the KITTI dataset [106, 130]. We generate tracklet of each proposal by examining the overlapping area of the bounding boxes in consecutive frames. We split the tracklet if there are other bounding boxes with 70% overlapping area in one frame.

Camera calibration and 3D background estimation: We estimate the camera intrinsics and extrinsics at keyframes and reconstruct the stationary background points using ColMap [187]. The camera poses are propagated from the keyframes to all other frames using the affine Lucas-Kanade tracking and PnP pose estimation. The time offsets between cameras are estimated using

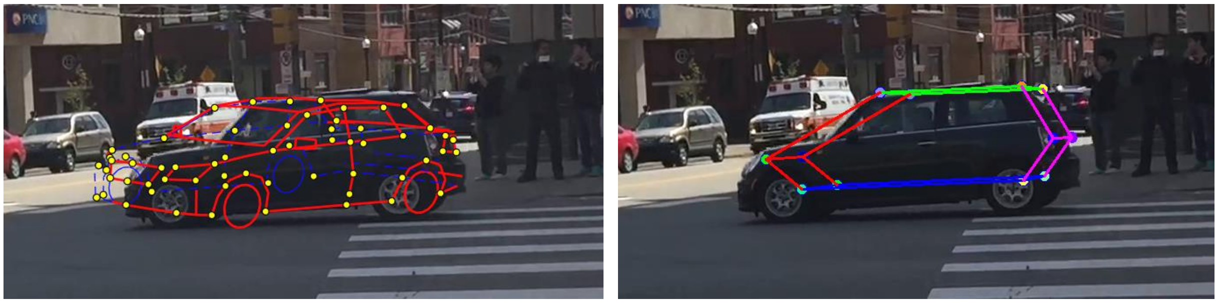


Figure 4.3: [Left] CAD based reconstruction [137] shows 22^o angular error with respect to the motion of the vehicle. [Right] In comparison, our result shows 1^o angular error. The CAD basis based methods fail due to inaccurate keypoint predictions, especially when only one side of the vehicle is visible.

the approach in [217].

3D reconstructing of the unstructured points: For every car proposal, we detect the Harris features and track them using the affine Lucas-Kanade algorithm. We initialize the detection every 30 frames and the track them for 120 frames in both backward and forward directions. We estimate their 3D locations using single view SFM.

4.3.2 Comparison with Baseline Approaches

Comparison with CAD/Active Shape Model Fitting: Figure 4.3 shows a comparison between the CAD fitting algorithm of Lin et al. [137] applied to a single image and our approach. CAD fitting approaches are sensitive to errors in 2D keypoint localization, especially in the presence of occlusions. In unconstrained settings as ours, CAD fitting orientation error is approximately 22 degrees (while our method shows only 1 degree error). Further, since current methods were trained on a small range of CAD models, they cannot generalize to arbitrary vehicles in the wild.

Comparison with single video methods: Figure 4.4 shows a comparison between our method and a traditional single-video based SFM reconstruction of structured points on the intersection dataset. While single video-based reconstruction has a reprojecion error of 35 pixels and fails to properly match the structured points of a different viewpoint, our method produces more accurate reprojection (2.5 pixels reprojection error).

4.3.3 Ablation Analysis

Quantitative: We analyze the improvement in the accuracy of reconstructed structured points with respect to the ground truth annotations according to the tracking length and the number of unstructured points in Figure 4.5. As expected, the increase in visibility (track length) of structured points better stabilize the structured points which leads to higher quality reconstruction. We also find that the larger number of unstructured points improve the quality of the structured

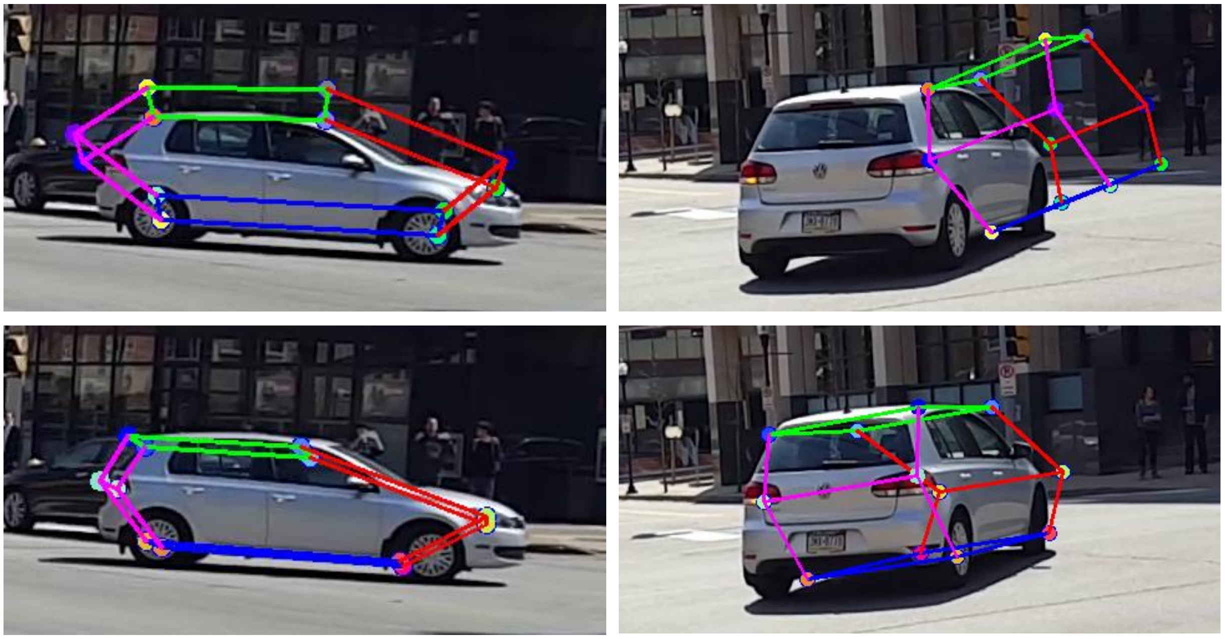


Figure 4.4: [Top-Left] Single-video based SFM(SVB-SFM) reconstruction of structured points. [Top-Right] The reprojection of the reconstruction onto a different view. [Bottom] Projection of reconstructed car using CarFusion to the above corresponding views.

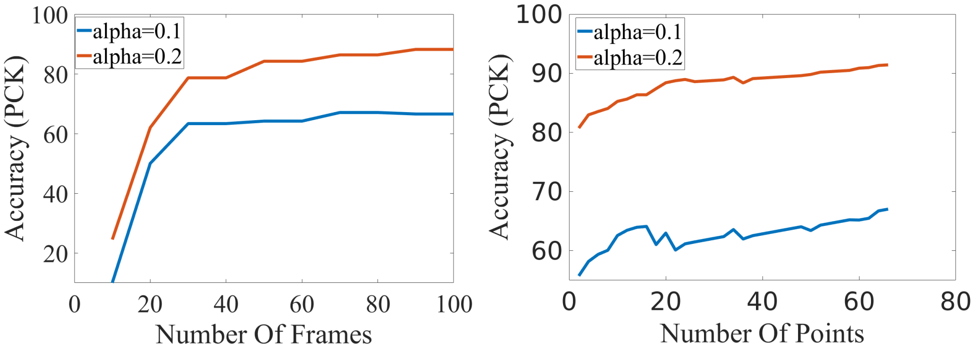


Figure 4.5: Analysis of accuracy with respect to the increase in number of frames (left) and increase in number of unstructured points (right) used in the CarFusion algorithm.

	cRANSAC			T-cRANSAC			CarFusion		
	# Traj	RMSE		#Traj	RMSE		#Traj	RMSE	
		Pretrain-MVB			Pretrain-MVB			Pretrain-MVB	
Straight	14	12.24	8.52	14	17.8	7.1	112	16.8	2.5
Turning	14	8.94	6.95	14	12.5	5.83	101	15.5	3.1
Multi	42	7.45	5.3	42	14.3	4.47	414	17.4	2.2

Table 4.1: Reprojection error of the reconstructed tracks at different stages of the pipeline. The rows refer to cases where one car is moving straight, turning left or right and multiple cars in the intersection. The number of trajectories using cRANSAC and T-cRANSAC is fixed to the number of parts, while with point fusion we have a combination of structured and unstructured tracks. The full pipeline (CarFusion + MVB) performs best, reducing the error of cRANSAC and T-cRANSAC by 4 and 2 times, respectively.

	$\alpha = 0.1$	$\alpha = 0.2$
Pretrained	87.1	91.8
MVB	91.4	94.5

Table 4.2: Comparing the structured point detectors using the Percentage of Correct Keypoint (PCK) metric. Our multiview bootstrapping (MVB) shows clear improvement over the state-of-art baseline detector [163].

points due stronger rigidity constraints and the improvement is more evident for stricter threshold $(\alpha = 0.1)$.

We adopt the widely used PCK metric [240] to evaluate the accuracy of 2D structured point detection. Under this metric, a 2D prediction is deemed correct when it lies within specified radius $\alpha * B$ of the ground-truth label, where B is the larger dimension of the car bounding box. We provide fine-grain analysis of the methods in Table 4.1, using three sub-sequences: car moving straight in a single lane, car turning, and a three cars scene. The first sub-sequence is observed for 234 frames, the second sub-sequence is observed for 172 frames, and last sub-sequence is observed for 202 frames. We report the root mean square error (RMSE) of the difference between the re-projected points and the detected points. We observe that the RMSE of the cRANSAC algorithm is large because of many detections with high variation in part localization. This error is reduced by finetuning (MVB) and can be attributed to the fact that better detection produces a more consistent 3D model. Interestingly, without multiview bootstrapping the error increases for T-cRANSAC. This could be because the detections are not temporally consistent. As expected, this error drops after detector finetuning. Using the unstructured tracks reduces the overall reprojection error of the 3D tracks by at least 5 times (12.24 to 2.5 or 7.45 to 2.2). However, the finetuned network gives modest improvement over the reconstruction of the structured tracks. This could be due to the limitation of the CNN architecture where the training image is downsampled substantially. The length of the trajectory of the car is the max length of the bounding box tracks over all the inlier videos.

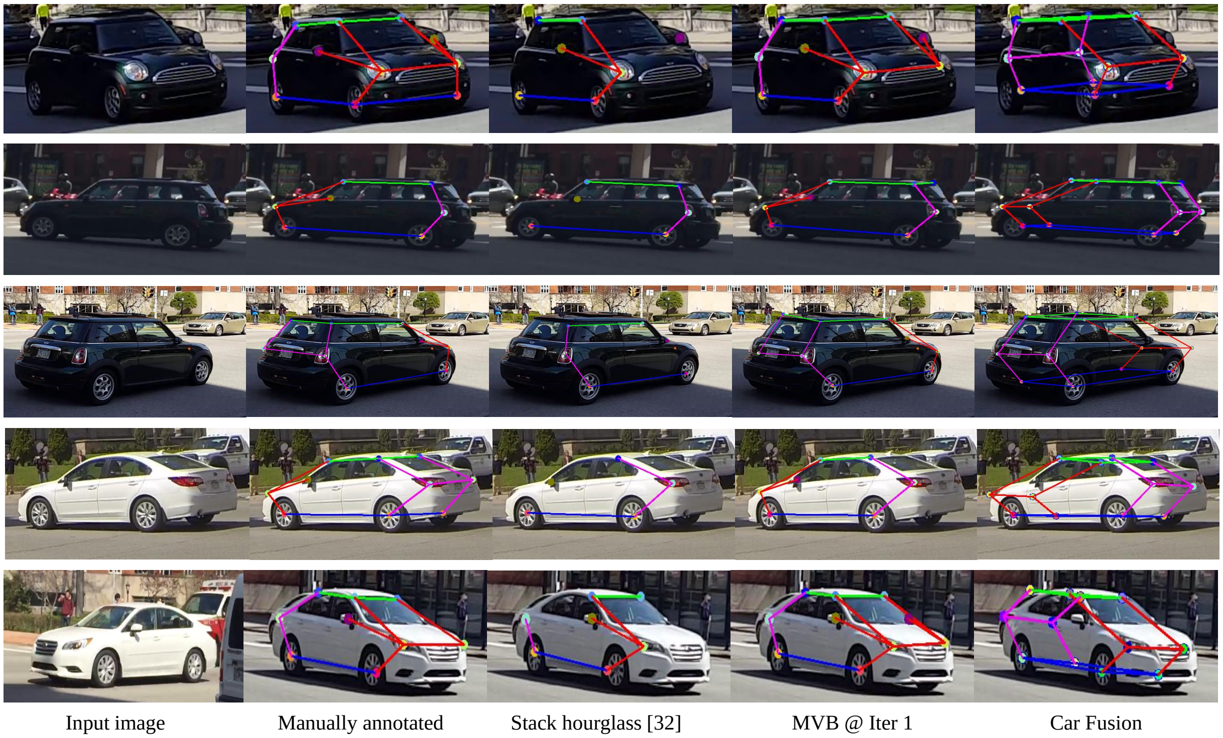


Figure 4.6: Qualitative analysis of the structured point detector before and after multiview bootstrapping (MVB), shown for two cars in three different views. Initial detectors were trained using Newell et al. [163]. The CarFusion approach was used to reconstruct the cars. Then the resulting 3D structured points were re-projected to all the views and used to retrain/bootstrap the detectors. The MVB approach shows clear visual improvement over the baseline, even in the presence of occlusions.

As showed in Table 4.2, our finetuned detector improves the accuracy of the baseline method by 4.3% with $\alpha = 0.1$ and 2.7% with $\alpha = 0.2$ just by finetuning the detector from the 2D re-projection of the reconstructed structured points. This result clearly demonstrates the benefit of CarFusion for accurate 3D structured points reconstruction and multiview bootstrapping for more accurate structured point detection.

Qualitative: Figure 4.6 compares detection of the structured points before and after multiview bootstrapping with respect to the ground truth labels for two cars observed in three different views. We visualize only detections with more than 50% confidence. Our multiview bootstrapping shows clear improvements over the baseline method as more confident points are accurately detected. Using CarFusion, the reprojected points accurately localize the structured points and provide a plausible prediction for occluded locations, as shown for twelve snapshots of another car in Figure 4.7. We attribute this property to the use of symmetry, link length, and rigidity constraints in the reconstruction stage. Although some of the structured points are not visible from any of the views, for example, the left front wheel of the car in Figure 4.7, we are still able to accurately reconstruct the point in 3D due to our left-right symmetry and link length con-

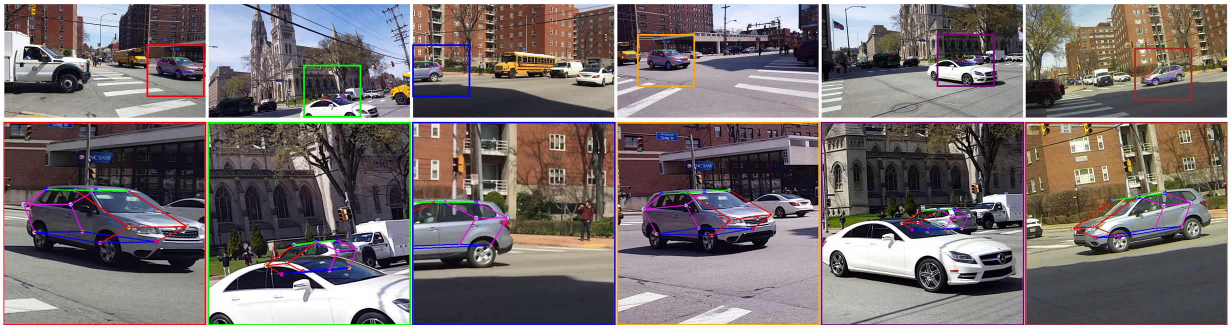


Figure 4.7: The 2D re-projections of the steel gray colored car in many occluded configurations. The CarFusion method can accurately reconstruct the 3D configuration of car despite strong occlusion. The top row shows the full field of views and the bottom row shows zoomed in insets.

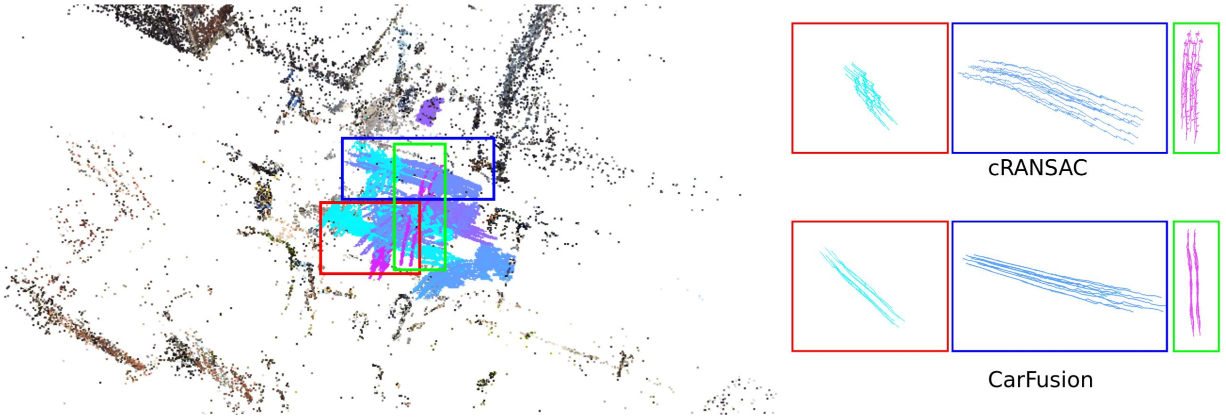


Figure 4.8: Visualization of the reconstructed trajectories of multiple cars using cRANSAC. The insets on the right show detailed comparisons of the trajectories stability between cRANSAC and CarFusion. CarFusion produces clearly more stable trajectories. Visually, they correspond well to the motion of a moving car.

straints. Without these constraints, the reconstruction of the structured points, even fully visible from multiple views, often explodes due to erroneous detection hypothesis.

Figure 4.8 shows a comparison between the quality of the reconstructed trajectories of the structured points using cRANSAC and the complete CarFusion pipeline. The trajectories are smoother by incorporating the Fusion of points compared to SFM on structured points. Our smooth trajectories are verified to be more accurate (see Table 4.2). Regardless of the finetuning step, cRANSAC performs poorly. This is due to erroneous detection that leads to frequent failure of cRANSAC . We observe a significant boost in the accuracy by temporal smoothing of the cRANSAC results over time. Our full CarFusion algorithm with multiview bootstrapping performs best, with 79.4% inliers detected.

In Figure 4.9 we illustrate the complete 3D reconstruction of trajectories of structured points on moving cars using CarFusion and the 2D projection to inlier views for several cars. As can be

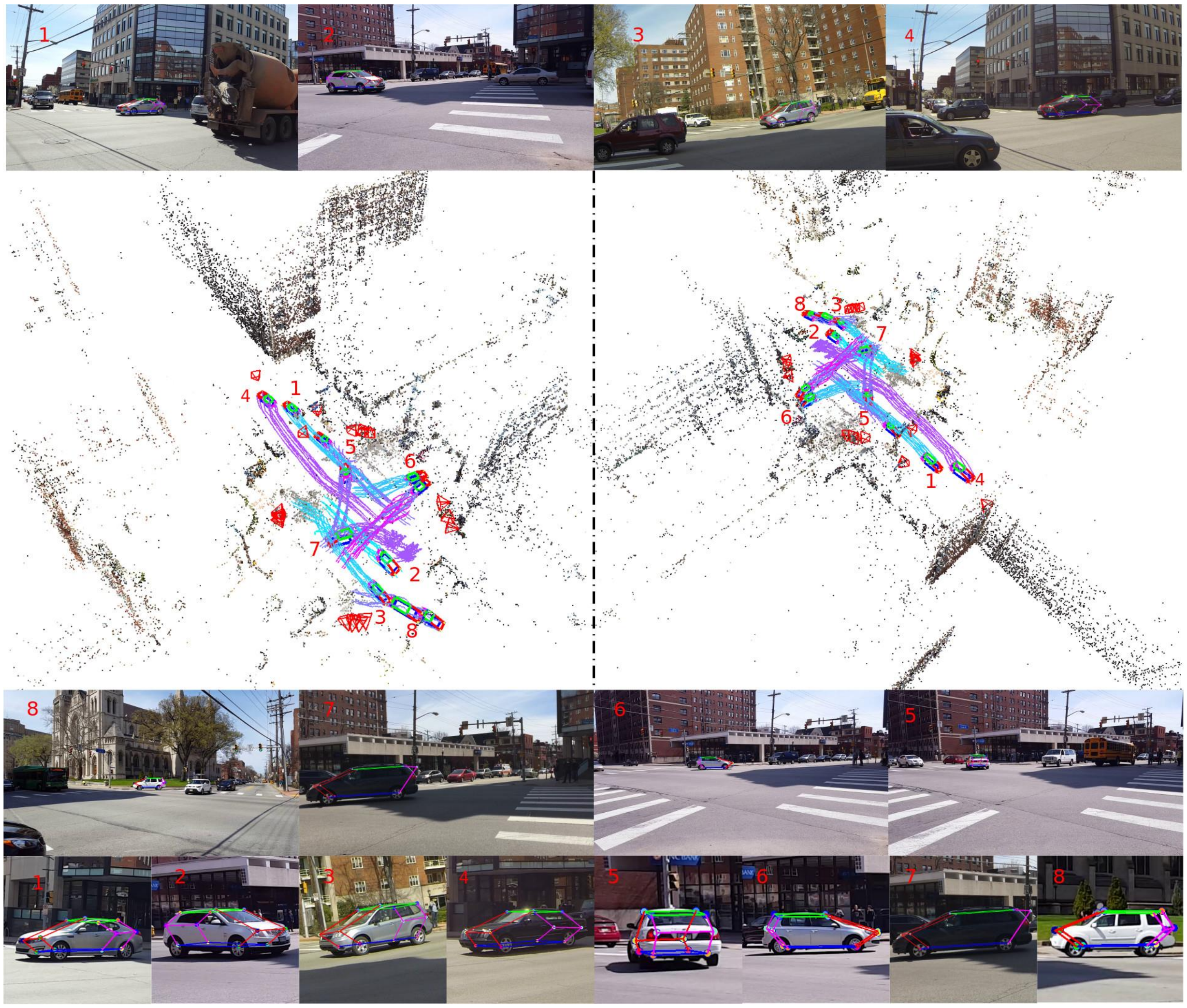


Figure 4.9: Visualization of the 8/43 reconstructed cars using CarFusion. We show the 2D re-projection of the reconstructions onto sample frame containing those cars. All the re-projected points fit the cars well.

seen from the results we are able to accurately reconstruct the trajectories of the cars over time captured from unsynchronized videos.

4.4 Discussion

While our multiview bootstrapping can significantly improve the keypoint detection accuracy, we notice that it could be overly confident and incorrectly predict occluded points. We expect an explicit modeling of the relation between occluding and occluded keypoints using graph neural net in order to refine the occluded points can help.

As shown in Chapter 2, To reconstruct a dynamic scene from multiple cameras, the most important aspect is the temporal alignment of the C cameras. Currently, the alignment is estimated offline by solving the spatiotemporal bundle adjustment on a very sparse set of unstructured point correspondences across cameras. However, similar to the point trajectory smoothness assumption in spatiotemporal bundle adjustment, we could potentially solve the alignment by considering the smoothness of the rigid body motion when aggregating the estimated motion from individual cameras.

4.5 Summary

We have presented a method to fuse imprecise and incomplete part detections of vehicles across multiple views and the more precise feature tracks within a single view to obtain better detection, localization, tracking and reconstruction of vehicles. This approach works well even in the presence of strong occlusions. We have quantified improvements due to the different stages of the end-to-end pipeline that only uses videos from multiple uncalibrated and unsynchronized cameras as input. We believe this approach can be useful for stronger traffic analytics at urban intersections.

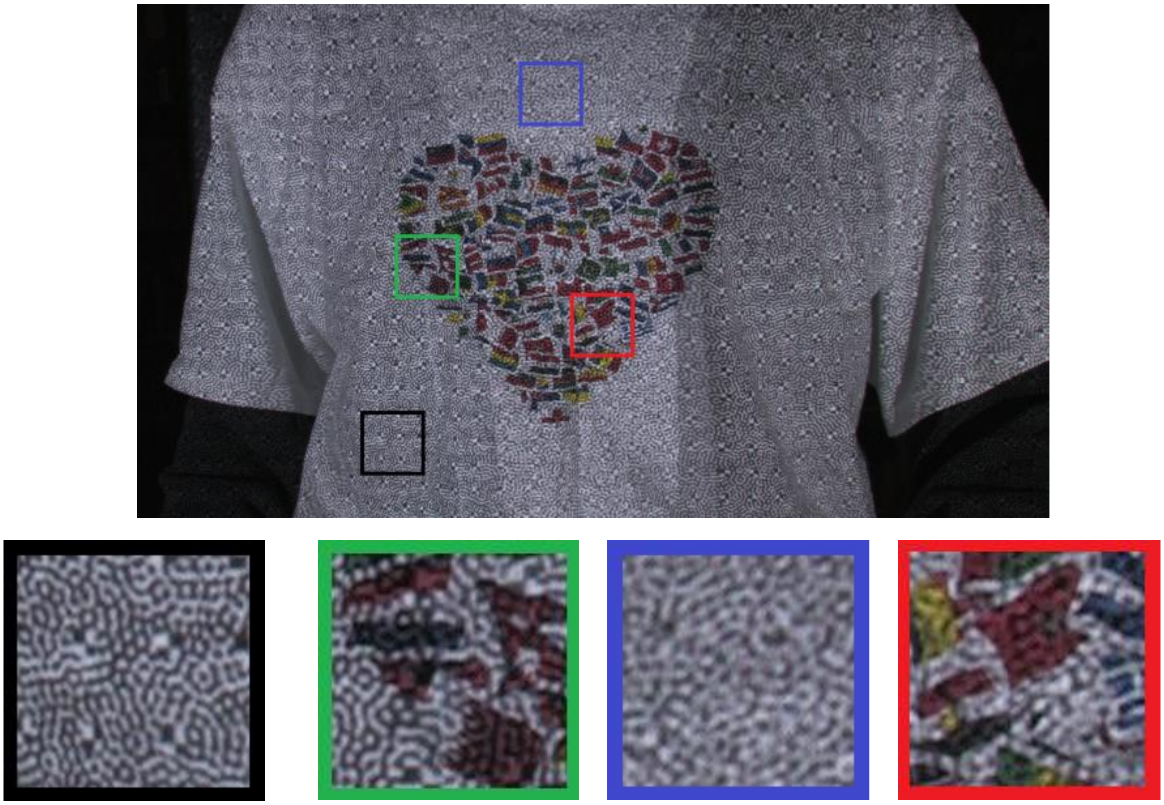
Chapter 5

Illumination Decomposition for Dense Dynamic Capture in Active Sensing

Structured light systems have been the method of choice to obtain dense and accurate surface models of complex objects for many industrial applications such as inspection of manufactured parts, biometrics, etc. Current illumination coding strategies can be mainly classified into two categories: multi-shot [55, 89, 218] and single-shot [118, 214, 238]. Multi-shot methods can estimate per-pixel depth map for various types of objects using temporal coding of the illumination patterns but require the scene to be stationary during the image acquisition. By decoding the spatial structure embedded in the illumination pattern, single-shot methods are applicable to dynamic objects but generate low spatial resolution reconstructions. Moreover, single-shot systems are susceptible to high frequency textured objects which distorts the appearance of the light pattern (see Figure 5.1 for an example). Such different requirements of multi-shot and single-shot methods lead to a trade-off between the spatial and the temporal resolution of the 3D shape and is hampering wider success of structured light systems.

Another well-known limitation of the current structured light approaches is that the multiple projectors cannot be used concurrently because of the illumination mixing phenomena. With the emergence of the virtual reality and tele-presence engines [25, 147], there is an increasing need for surround structured light systems where the use of multiple projectors is a must. Unfortunately, instead of producing high spatial resolution reconstructions in the overlapping regions, large holes are obtained. Currently, the problem is mitigated using additional hardware [33]. However, this approach alters the calibrated extrinsic between individual units and may require online recalibration of the whole system to integrate individual 3D shapes into one piece.

In this chapter, we present a structured light system that can handle the two aforementioned issues. First, despite being a single-shot method, it can use high frequency illumination patterns to estimate high spatial resolution shape information of highly textured objects. Second, it allows concurrent use of multiple projectors without additional hardware. The proposed method is single-shot because it does not use the illuminated images in the previous frames to temporally decode the illumination pattern. Figure 5.1 shows a typical example of surface texture that our method can handle.



Pure illumination

Texture illumination mixing

Illuminations interference

Texture & Illuminations interference

Figure 5.1: Conventional single-shot structured light systems often fail for highly textured objects. For multiple projectors system, problematic effects such as the mixture of albedo variations and the interference between high frequency illumination patterns arises, which makes it difficult to establish reliable and dense camera-projector correspondences.

Due to the use of multiple projectors, the observed image not only contains the mixed appearance between the surface texture and the illumination but also the mixture of the illumination from different projectors and their combinations. Our solution decomposes possible combinations of the surface texture and the projected illumination patterns used to recreate the observed mixed appearance image. More concretely, we develop an optimization framework that reliably estimates local warps of the illumination patterns visible in the observed image regions and a reference texture template to compose the observed image. The texture template is obtained by periodically interleaving the projection sequence with an all-white pattern. The warping functions are computed starting from a sparse set of correspondences between the camera and the projectors. These initial matches are greedily propagated into textured areas where spatial correspondences cannot be directly estimated. Our separation scheme allows the use of multiple projectors to accurately capture the 3D shape of large objects.

As a result of the separation, the dense correspondences between projectors and camera even in the textured, and (or) mixed illumination regions are established. Subsequently, these correspondences are triangulated to reconstruct the high resolution 3D shape of the object. In addition, we obtain dense tracking inside the texture region in the presence of the illumination pattern.

Since the method computes warps to a template, it does not exhibit drift over time. Moreover, unlike conventional single-shot structured light systems whose performances degrade as surface texture frequency increases, our method achieves better decomposition accuracy with higher frequency texture. Finally, despite being presented for single-shot approaches, this method can also be used in conjunction with multi-shot systems by spatially modulating the illumination patterns with a random pattern [241]. Structured light systems using our decomposition method can take advantage of the consumer trends in resolution increases for both camera and projector and thus, avoid the need for expensive high quality time-of-fight sensors.

The proposed method is implemented on a structured light system consisting of one camera and two projectors, where four possible types of appearance could be observed: pure illumination from individual projector, illumination from two projectors, surface texture and illumination from individual projector, and illumination from two projectors and surface texture. We demonstrate dense and accurate decomposition and reconstruction results on both synthetic and real data with non-rigidly deforming objects.

5.1 Related Work

Active illumination for shape estimation mainly focuses on designing coded patterns that are robust to occlusion, depth discontinuity [37, 42, 186, 218, 250], defocus [4], or global illumination [43, 44, 55, 85, 90, 91]. A thorough review of the current state of the art techniques is given in [185]. Generally, these systems have to select appropriate illumination patterns depending on the temporal or spatial resolution requirements. Since multi-shot methods can robustly generate high spatial resolution but require stationary scenes, motion compensation schemes have been developed to handle slowly moving objects [226]. Another common approach is to interleave the patterns for structure estimation with patterns optimized for computing motion [86].

In contrast with multi-shot approaches, single-shot methods sacrifice spatial resolution for high temporal resolution reconstruction and usually cannot deal with textured objects. In the context of single-shot structured light reconstruction, the problem of textured objects is almost unexplored with the exception of the work from Koninckx et al. [122, 123]. Their methods rely on a feedback loop that changes patterns according to the error in the decoding process and heuristic rules to set the color codes depending on different textured surfaces. In principle, this method treats the surface texture as a nuisance and designs illumination patterns robust to the texture. Conversely, we consider the texture as an additional source of information that needs to be recovered along with the 3D shapes.

While single camera-projector systems are most popular, there have been few attempts to deploy the multiple systems at once because of the distorted appearance of multiple illumination patterns superimposed on each other. Recently, Furukawa et al. [76] present a two projectors-one camera system that projects de Bruijn stripe pattern of different colors from different projectors to reduce the interference between those patterns. Sagawa et al. [184] extend the idea further and obtain higher resolution shape by interpolating the de Bruijn sequence. Inherit from these works, a surround structured light system is introduced [116]. Due to the use of sparse color grid, all these methods produce sparse shape reconstruction and are not robust to color objects. Another notable surround structured light reconstruction is presented by Lanman et al. [126] where the planar mirrors are elegantly arranged to simulate the projector patterns coming from different views. There are also efforts to create the surround 3D reconstruction systems for immersive visualization and tele-presence using the Kinects I depth sensor [25, 34, 147]. Generally, Kinect-based methods rely on hole filling or smoothing algorithms [147] and external mechanical vibrator to make their light patterns blurred when viewed by the other Kinects [25].

Although the theme of image separation has not been applied to structured light system, its history dates back to the early 90s when researchers extract motion between different depth layers from image sequences [105, 107, 220, 227]. Their key observation is that the background and the reflection layers undergo different motions due to their different distances to the transparent layer, and hence, can be separated. These methods require multiple images with motions of various directions and they only focus on motion separation, not image layer restoration.

A similar type of image separation is performed while estimating intrinsic images [72, 206, 243]. Prevailing intrinsic image decomposition algorithms assume that natural images contain piecewise-smoothness reflectance and smoothly varying shading. Recently, by exploiting the difference in the distribution of the two intrinsic components, Yu and Brown [245] present a fast separation algorithm that produces state of the art results with a single image. In contrast with these methods, our work decomposes the high frequency illumination patterns and texture from the observed mixed appearance image. Such high frequency mixture breaks the smoothness assumption of both transparent/semi-reflection removal and intrinsic images.

5.2 Decomposition Framework

5.2.1 Mathematical Formulation

Consider an object being illuminated by one or more projectors. The brightness $I(x, y)$ at location (x, y) in the observed image is modeled as a multiplication of a texture image $I_T(x, y)$ and the illumination image $I_L(x, y)$ at that location:

$$I(x, y) = I_T(x, y)I_L(x, y). \tag{5.1}$$

The texture image I_T is the image observed if the projector illuminates an all-white pattern on the object. The illumination image I_L is the incident lighting pattern. This equation can explain different types of mixed appearance: illumination light from one projector onto textured surface, illumination from multiple projectors onto a textureless surface, or illumination lights from multiple projectors onto a textured surface. When the object is being illuminated by multiple light sources, the incident light is the combination of all lighting coming from individual projectors.

Equation 5.1 is ill-posed because it has more unknowns, e.g. $I_T(x, y)$ and $I_L(x, y)$, than the equations. Thus, we must rely on additional knowledge of the illumination and texture source. Since the illumination image is a projection of the known projector patterns, these patterns serve as our referenced templates. The reference texture template can be obtained by interleaving the projection sequence with an all-white pattern.

As the object deforms, the appearance of both the texture image I_T and the illumination image I_L change accordingly. This suggests that we must be able to warp the texture template and projecting patterns to the current their current hidden appearances in order for Equation 5.1 still to hold true. Because image deformation is high-dimensional and non-linear, analytic forms that describe consistent deformation behavior over the entire image do not exist. Thus, we model the warping function locally within a small patch to approximate for these distortions. More specifically, we employ an affine warping function f, a constant gain a_T and an offset b_T to map patches in the texture template T to the texture image I_T:

$$I_T(x, y) = a_T T(f(x, y)) + b_T, \tag{5.2}$$

where the constant terms, a_T and b_T, are defined patch-wise and help compensating for the intensity changes between the two images due to changes in surface normal, light directions and ambient illumination. The warping function f is written as:

$$f(x_k, y_l) = \begin{bmatrix} x_0 + p_0 + p_2 k + p_3 l \\ y_0 + p_1 + p_4 k + p_5 l \end{bmatrix}, \tag{5.3}$$

where (k, l) are the row and column of the texture template patch centered at point (x_0, y_0), and $p_{0..5}$ are the warping parameters.

Unlike the texture template warping where the deformation between two time instances can be small, the appearance of a patch in the illumination pattern is quite different when observed in the camera image due to perspective transformation. Assuming photometrically calibrated projectors, we adopt a homography warping g, a set of constant gain a_{Li}, and offset b_{Li} to transform the illumination pattern L_i to the illumination image I_L:

$$I_L(x,y) = \sum_{I_l=1}^{N} a_{Li} L_i(g_i(x,y)) + b_{Li}, \tag{5.4}$$

where N is the number of projectors visible at pixel x, y in the camera image. The role of a_{Li} and b_{Li} are also to compensate the intensity changes between the illumination patterns and the illumination image. Empirically, we found the homography warping more robust than affine warping. This warping function h is expressed as:

$$g_i(x_k, y_l) = \begin{bmatrix} \frac{x_0+q_{i0}+q_{i2}k+q_{i3}l}{1+q_{i6}k+q_{i7}l} \\ \frac{y_0+q_{i1}+q_{i4}k+q_{i5}l}{1+q_{i6}k+q_{i7}l} \end{bmatrix}, \tag{5.5}$$

where (k,l) denotes the row and column of the texture template patch centered at point (x_0, y_0), and $q_{i0,..i7}$ are the warping parameters of the i^{th} projector.

We minimize the following cost function over a patch of size $(2M+1) \times (2M+1)$ centered at the decomposing point (x_0, y_0) for the warp parameters $p_{0..6}$, $q_{i0..i7}$ and their photometric compensation coefficients a_T, b_T, a_{Li}, b_{Li}:

$$\sum_{k=-M}^{M} \sum_{l=-M}^{M} \left[I_T(x_k, y_l) I_L(x_k, y_l) - I(x_k, y_l) \right]^2. \tag{5.6}$$

Once the optimization converges, we quantify the decomposition by the Normalized Cross Correlation (NCC) score of the patch in the observed image and the patch synthesized by Equation 5.1. A decomposition is deemed successful if the ZNCC score is greater than predefined threshold, set to 0.9 for all experiments.

Decomposition complexity: The complexity of Equation 5.6 grows linearly with the number of projectors. For a typical setup of two projectors and one camera, at every patch, the number of unknowns are 17 (8 parameters for I_T and 9 parameters for I_L) for texture and one illumination pattern mixing, 18 parameters for two projector illumination patterns mixing, and 26 (8 parameters for I_T and 18 parameters for I_L) for texture and two illumination patterns mixing. For our local decomposition scheme, the ratio of the number of constraints, i.e. the number of pixels in a patch, over the number of unknowns is small, which makes solving for these warp parameters challenging. Thus, proper exploitation of additional geometric constraints and a good initial warp estimation are crucial to the decomposition scheme.

5.2.2 Illumination Flow Constraint

For a dynamic object illuminated by a stationary structured light system, its observed image changes at every time instance. Besides the motion of the textured surface observed on this

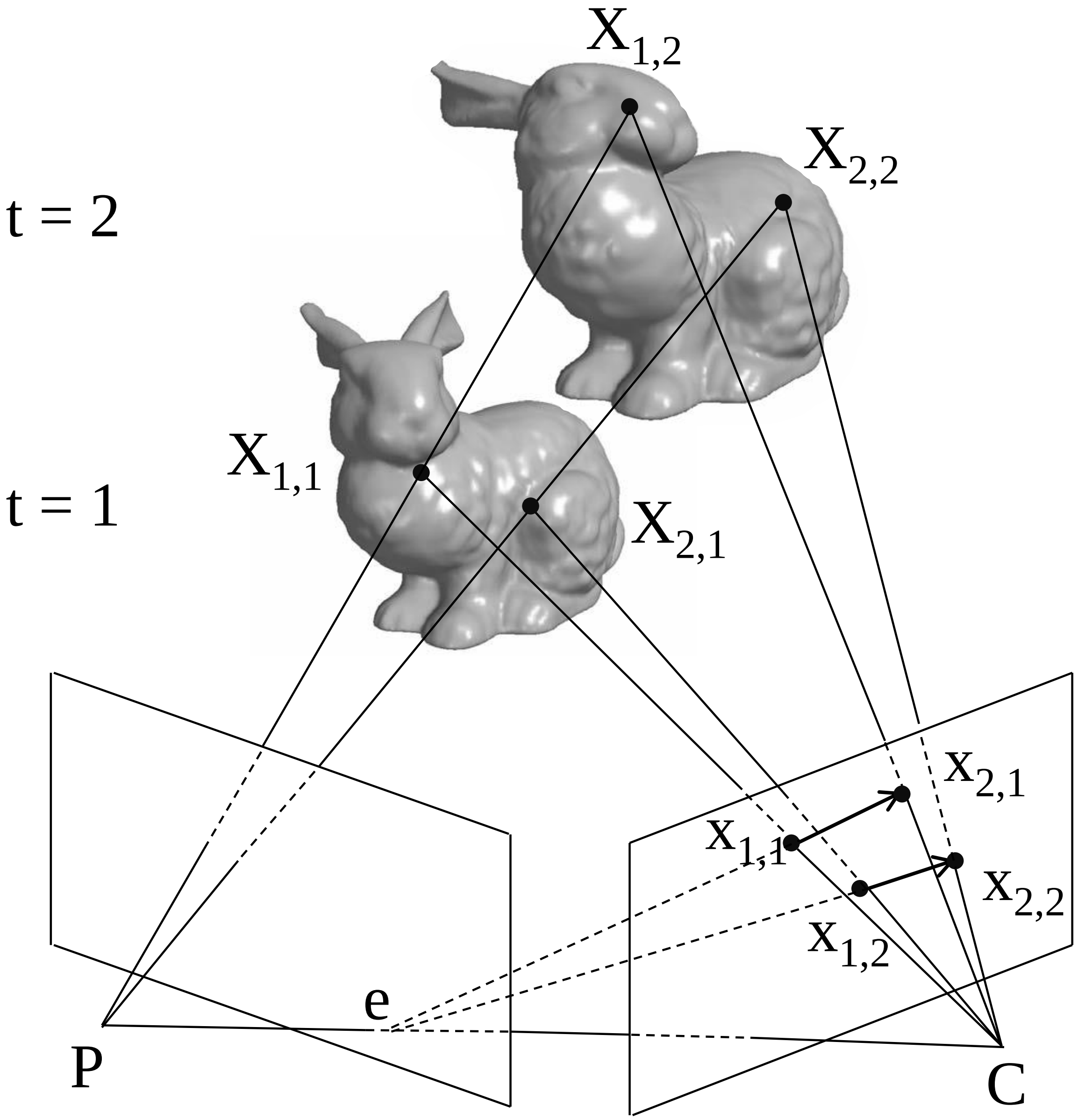


Figure 5.2: Illumination flow constraint. Two rays emanating from the projector P hit the object at $[X_{1,1}, X_{2,1}]$ and $[X_{1,2}, X_{2,2}]$ at time instance 1 and 2, respectively. Their corresponding projection to camera C are $[x_{1,1}, x_{2,1}]$ and $[x_{1,2}, x_{2,2}]$. The illumination flow vectors, $\vec{x}_{1,1} - \vec{x}_{1,2}$ and $\vec{x}_{2,1} - \vec{x}_{2,2}$, intersect at the epipole e of the projector. This geometry constraint reduces the dimensionality of the illumination flow by one.

image, there is also an apparent motion due to the illumination pattern. Interestingly, the flow direction of the illumination pattern and the texture are radically different from each other. Unlike the motion flow which can be in any directions according to the movement of points on the object surface, the direction of individual illumination flow is only along specific lines. Figure 5.2 illustrates this phenomenon. Because the illumination flow lies on the projection of the ray emanating from the projector, their directions must be along the epipolar lines of the camera-projector system. Furthermore, the illumination flow field encodes the changes in depth of rays emanating from the light source that hit the object. Because establishing spatial correspondences between camera and projector is much more difficult than estimating temporal correspondences, especially in the wide baseline scenario, any structured light systems can gain benefit from the temporal coherency of the illumination flow for shape reconstruction.

For our scenario, the illumination flow constraint allows us to simplify Equation 5.5 to:

$$g_i(x_k, y_l) = \begin{bmatrix} \frac{x_0+q_{i0}d_x+q_{i1}k+q_{i2}l}{1+q_{i5}k+q_{i6}l} \\ \frac{y_0+q_{i0}d_y+q_{i3}k+q_{i4}l}{1+q_{i5}k+q_{i6}l} \end{bmatrix}, \tag{5.7}$$

where the center of the patch (x_0, y_0) is forced to lie on the epipolar line parameterized by normalized direction vector (d_x, d_y) of that line in the projector image. Compared to Equation 5.5, for each projector, this equation reduces the dimensions of the optimization in Equation 5.6 by one and allows us to search for the decomposed illumination patch only along the epipolar line.

5.3 Greedy Correspondence Growing

Due to perspective distortion in the illumination image I_L and the large deviation from the texture template T for fast moving objects, good initial guesses for the warping parameters are required to optimize Equation 5.6. Even with random patterns [55], the local minimum of the cost function still exist due to the repetitive nature of the spatial encoding illumination pattern. We solve the initialization problem greedily. Starting from a few correspondences established in pure illumination or low textured regions, the results are gradually expanded to the mixed regions by optimizing for the warp parameters in Equation 5.6. Figure 5.3 gives an overview of this propagation process.

5.3.1 Types of Mixed Appearance

Solving for the warp parameters in Equation 5.6 requires knowing the mixture type of the decomposed point. Let the point being decomposed be the "child" and the point initializing the child's warping parameters be the "parent". The choice of possible appearance mixture of the child point depends on the type of mixture of its immediate parent in the propagation chain. We allow at most one change in the mixture's component with respect to its parent. For example, for the case of two projectors and one camera, if the parent composes of the light from one projector and the texture surface, its child can only be created from the pure illumination from that projector, the illumination and the texture, or mixture of two illumination patterns and the textured

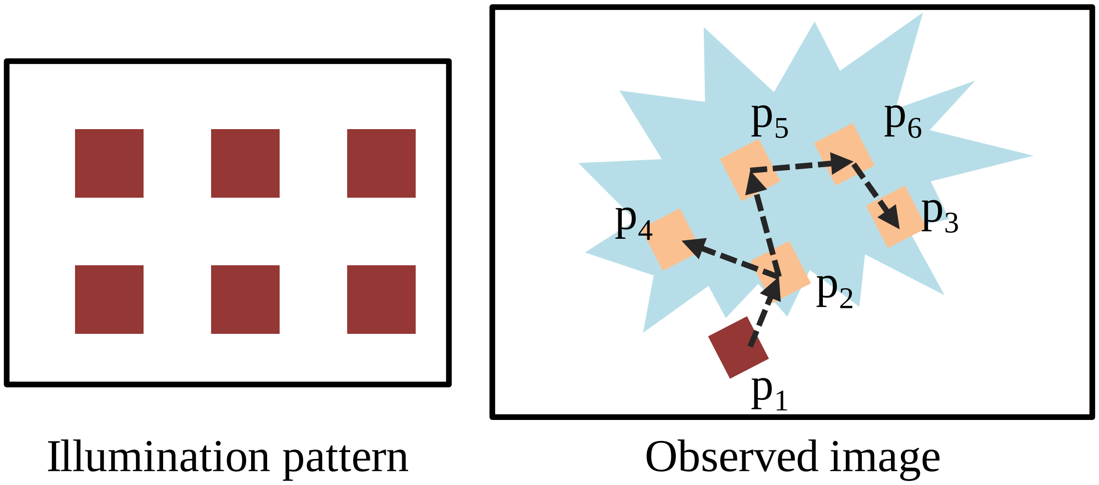


Figure 5.3: Correspondence propagation. The blue textured region is illuminated by the square pattern. Since the appearance of the squares in the textured region have changed, direct matching between the observed image and the projector pattern is not possible. The correspondences are propagated from the p_1, whose match in the projector pattern is easily found, to other squares inside the textured region. For the propagation path showed by dotted arrows, p_1 is the immediate parent of the child p_2, p_2 is the immediate parent of the child p_4, and etc.

surface. Although this assumption can be violated such as when the illumination and texture boundaries are at the same location, empirically they rarely happen and hence, are ignored. Because evaluating all the possible choices of the child mixture type is computationally expensive, we give more priority in the processing of the mixture type that is the same as the parent first. Table ?? shows all allowed mixture type for this structured light configuration.

5.3.2 Initialization of Warp Parameters

We initialize the warping parameters in two steps.

Table 5.1: Determination of mixture type

Parent type	**Possible child type**
L_1	L_1, TL_1, L_1L_2
L_2	L_2, TL_2, L_1L_2
L_1L_2	L_1, L_2, L_1L_2, TL_1L_2
TL_1	L_1, TL_1, TL_1L_2
TL_2	L_2, TL_2, TL_1L_2
TL_1L_2	TL_1, TL_2, L_1L_2, TL_1L_2

TL_1: texture and projector 1 mixture
TL_2: texture and projector 2 mixture
L_1L_2: projector 1 and 2 mixture
TL_1L_2: texture, projector 1 and 2 mixture

Step 1: Grow a sparse set of correspondences between the camera and the projectors using a greedy correspondence growing algorithm [39]. This greedy growing strategy and the use of a random illumination pattern allows us to establish dense correspondences everywhere except for the mixed appearance regions.

Step 2: For a pixel that is close to the mixed region boundary, we exhaustively search its local neighbors for patches on the illumination patterns and texture template that minimizes the cost defined in Equation 5.6. The choice of possible combination of patches are given from Table **??**. The mixed boundary is defined as locations where the direct correspondences between camera and projectors computed in Step 1 fail.

Depending on the motion of the objects, the range of the exhaustive search is set apriori. Since the deformation between the template patches and the ones in the pure texture and illumination images could be large, we pre-warp them using the warping parameters of the parent point before examining their contribution to the cost function for different hypothesized mixture.

5.3.3 Optimization

The initial warping parameters obtained in Section 4.2 are refined by minimizing Equation 5.6 with a standard Gauss-Newton method. Given the complexity of the decomposition, two heuristics are applied to ensure accurate and convergence of the optimization.

Coordinate descent: Even with good initial guess, optimizing Equation 5.6 could be challenging due to its high dimensionality, e.g. two illumination patterns and the textured surface mixing requires optimizing for 26 parameters per patch. We mitigate the problem by the coordinate descent scheme where we alternate between optimizing for the warping parameters of one source, say texture, and fixing the rest, say illumination pattern one and two.

Coarse to Fine Decomposition: Since both the texture and illumination images have high spatial frequencies, a coarse to fine scheme of multiple Gaussian pyramid results in severe aliasing artifacts. However, since the texture surface usually has lower frequency content than the illumination, the texture image must be decomposed at an appropriate scale to avoid drifting in low frequency textured regions. Thus, we try different sizes of the local patch, arranged in descending order, to handle the differences in scale of texture image. The process is terminated as soon as a certain patch size yields successful decomposition.

5.4 Results on Synthetic Dataset

The performance of our approach is validated on synthetic cloth sequences containing a range of texture frequencies. The cloth composed of 64,000 vertices is generated using the OpenCloth engine [142] and can deform in subtle and non-rigid ways. Its physical size is 750×400mm^2

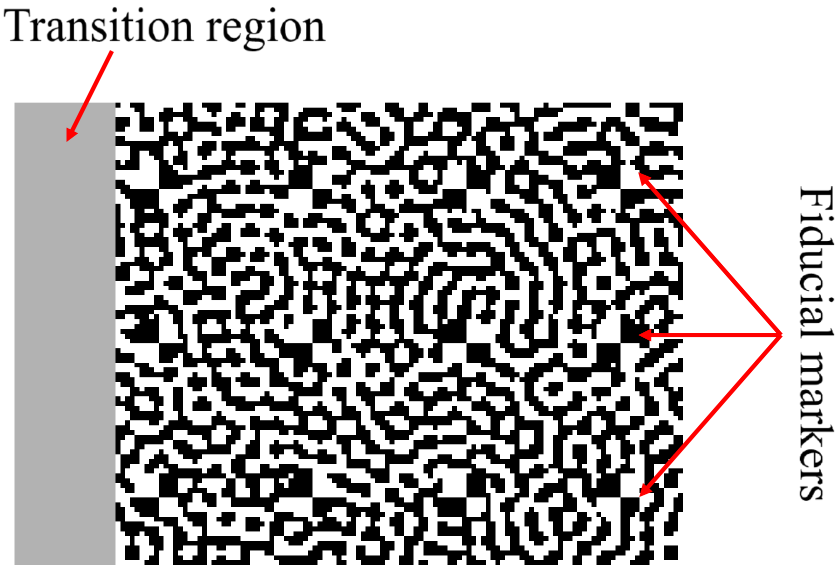


Figure 5.4: Part of the illumination pattern. The fiducial markers are embedded into the random pattern to provide a sparse set of correspondences between the camera and the projector.

and is placed approximately 1250 mm away from the camera, which resembles our real experimental setup. The non-rigidity of cloth makes dense decomposition and shape reconstruction challenging. The all-white pattern is projected once in 30 frames. For all of our experiments, this interleaving interval gives good trade-off between the temporal resolution of the results and the moving speeds of the objects.

We set the camera and the projector resolution to 1920×1080 and 1280×800, respectively. The decomposed patch size is varied from 21×21 to 15×15 with a step size of 2 to account for different scale of the surface texture. We split the region of interest into sub-regions and independently execute them to take advantage of the multi-core architecture of modern computer. Since the runtime of the algorithm depends on the size of the patch and the size of the image to be decomposed, we report the runtime in points per second unit. Decomposing texture-illumination, illumination-illumination, and illumination-illumination-texture take on average 2009, 2152, 1535 points every second on a Quad-core i7 CPU, respectively. The 3D shapes are estimated by triangulating the correspondences obtained after the decomposition. The results are presented without any post-processing.

For a succinct description of the result, we use the following notations: TL_i means texture and illumination pattern i^{th} mixing; L_1L_2 means two illumination patterns mixing ; TL_1L_2 means texture and two illumination patterns mixing.

5.4.1 Illumination pattern

Figure 5.4 shows one of our static bandpass random binary illumination pattern [55]. The size of the speckle in this pattern can be tuned to provide suitable contrast for illuminating objects of different size. Fiducial checkerboard markers are uniformly seeded at every 32 pixels inside this pattern to provide set of sparse spatial correspondences. These correspondences are computed using template matching along epipolar lines. Because the distance between these markers is usually magnified in the camera image, these markers do not cause ambiguities in the propagation process. For ease of implementation, a transition region is added to the side of the pattern to avoid the boundary issue when only a partial illumination patch lies on the illumination pattern, which commonly happens when multiple light patterns are mixed.

5.4.2 Qualitative Evaluation

We show our decomposition results on different texture frequency cloth: flower in Figure 5.5 and bear in Figure 5.6. The bear texture is very similar to the illumination pattern. This extreme case violates the assumptions of any methods powered by independent component analysis or smoothness prior. Thus, such methods are not applicable. The small but noticeable defects in the decomposition of frame 2 do not expand during the greedy propagation and are fixed in frame 29. Hence, there is little-to-no drift in the estimated warping functions. Visually, the method achieves better decomposition for higher frequency texture as there are less visible defects in the highly textured bear than the flower cloth.

5.4.3 Quantitative Evaluation

Figure 5.7 shows our depthmap estimation of the flower and bear cloth at frame 29^{th}, respectively. Without our separation, the correlation score between the projector patch and its corresponding patch in the observed image is very low in the mixed appearance regions. The depth estimated in these regions is eliminated, which results in large holes in the 3D shape. By separating the pure texture and illumination images, the proposed method accurately estimates the 3D shape with the exceptions of places where strong foreshortening occurs. For a particular point, its depth is retained only if its reprojection onto the projectors and camera images is less than 1 pixel error.

To better quantify the decomposition error, we compare the correspondences obtained after the decomposition with those estimated on the synthesized pure illumination and texture images. The pure illumination image is created by projecting the illumination pattern onto a textureless cloth for each individual projector while keeping the others off. Texture image is obtained by texturemapping the cloth with binarized Perlin-noise pattern generated by error-diffusion dithering and projecting white light on the cloth. This binarized random pattern is known to give accurate tracking results [16, 121]. We compute the spatial correspondences between camera and projector on the pure illumination image and the temporal correspondences on the pure texture image using patch-based matching methods [18]. These correspondences are the best possible

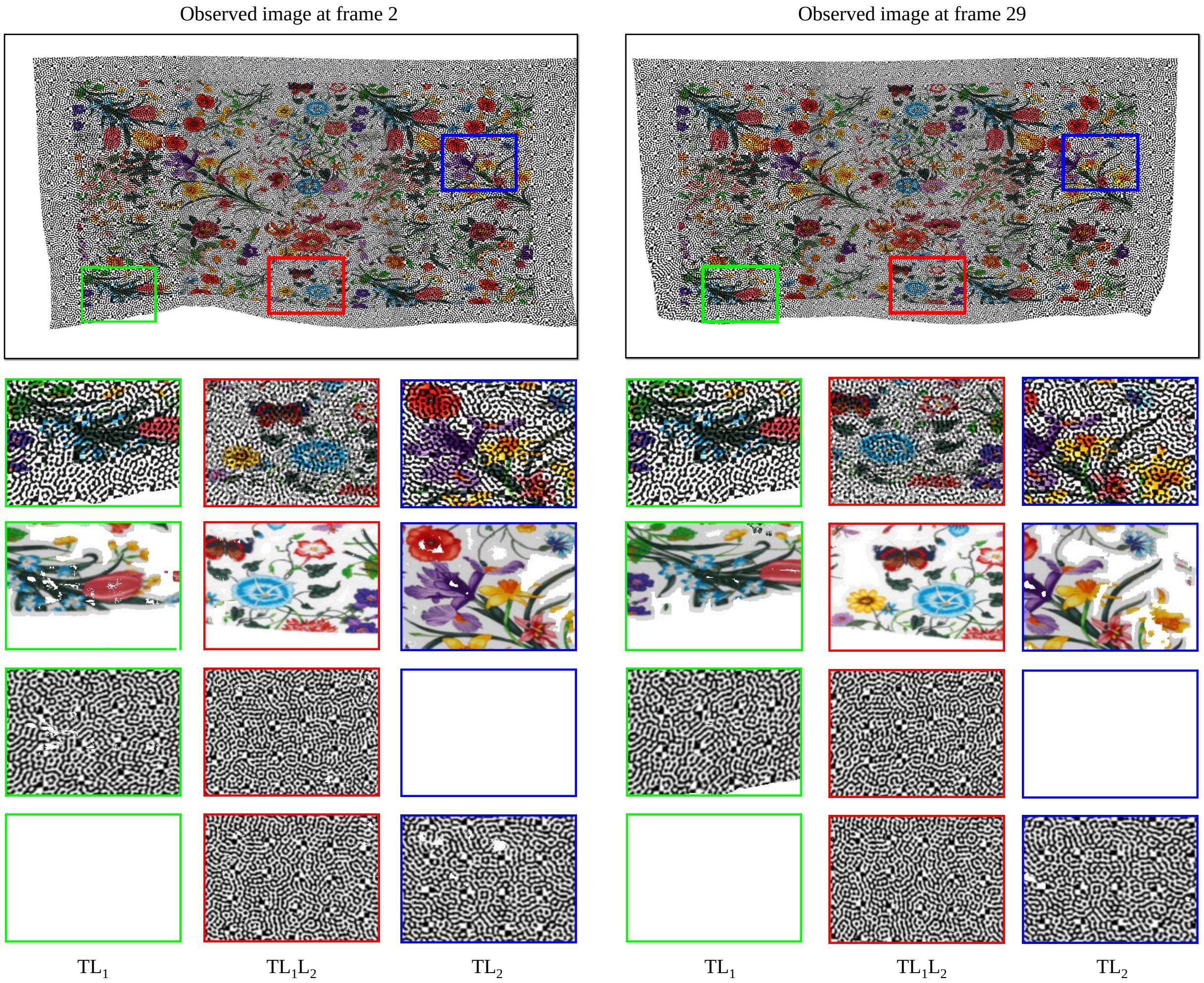


Figure 5.5: Decomposition of the texture and illumination images from a synthetic flower cloth sequence with interleaving pattern projected every 30 frames. We show insets of three types of mixture: texture-illumination mixing, illumination interference, and texture and illumination interference. The same regions of the observed images are magnified to show how its intricate appearance changes over time. From the first row to the fourth row are the magnified regions of the observed image, the estimated texture, the estimated illumination for projector 1, and the estimated illumination for projector 2. Note that the small white spots in decomposed illuminations, an indication of decomposition failure, at frame 2 are not expanding and have been fixed in frame 29.

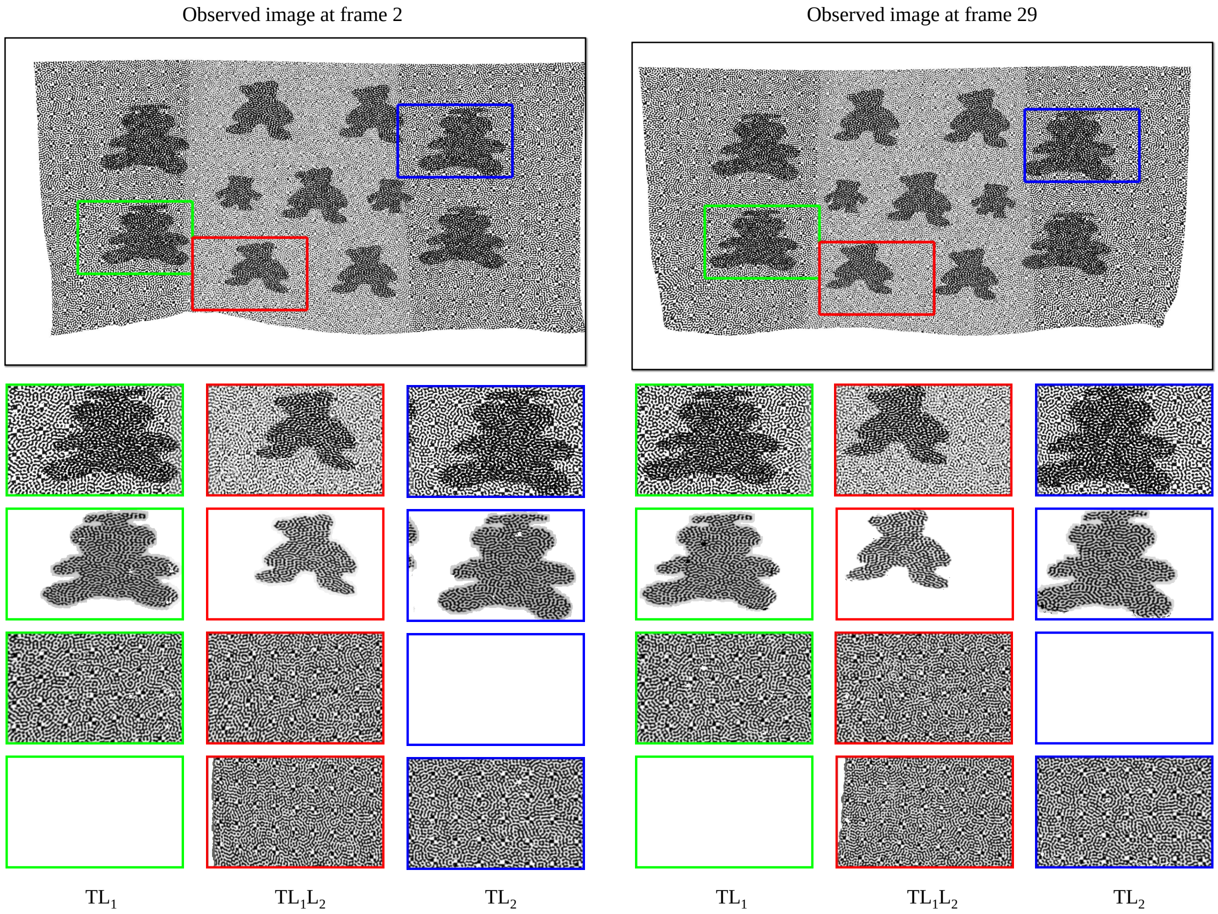
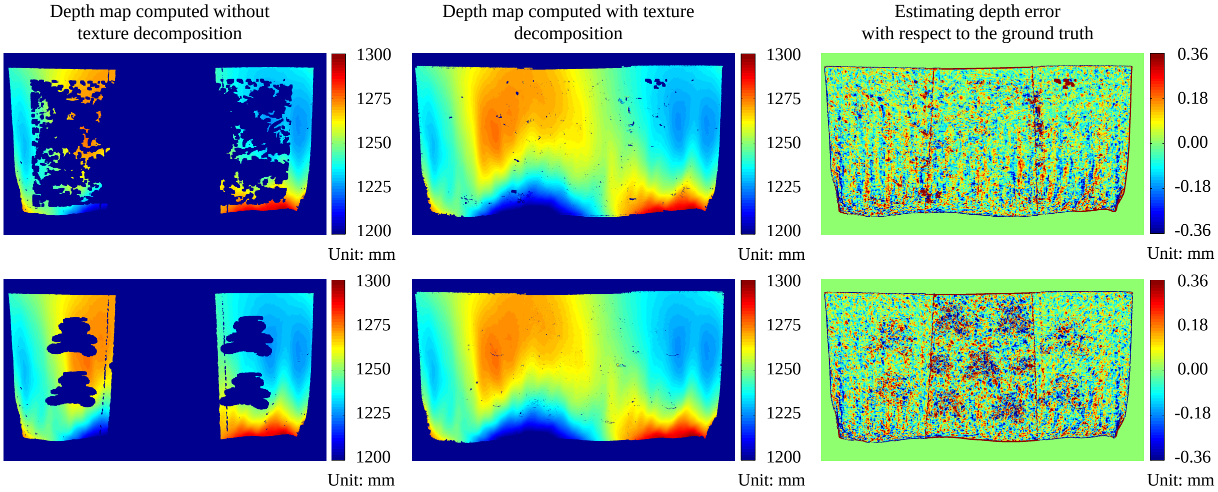


Figure 5.6: Decomposition of the texture and illumination images from a synthetic bear cloth sequence with interleaving pattern projected every 30 frames. We show insets of three types of mixture: texture-illumination mixing, illumination interference, and texture and illumination interference. The same regions of the observed images are magnified to show how its intricate appearance changes over time. From the first row to the fourth row are the magnified regions of the observed image, the estimated texture, the estimated illumination for projector 1, and the estimated illumination for projector 2. Our method achieves better decomposition for higher frequency texture as white spots showing missing data are less visible.


Depth map computed without texture decomposition

Depth map computed with texture decomposition

Estimating depth error with respect to the ground truth

Figure 5.7: Estimated depthmap of synthetic bear cloth sequence with interleaving pattern projected every 30 frames at the 29^{th} frame . The depth map estimated with our decomposition scheme not only shows its completeness but its high accuracy with respect to ground truth depth.

estimation at a given frame and hence, serve as ground truth.

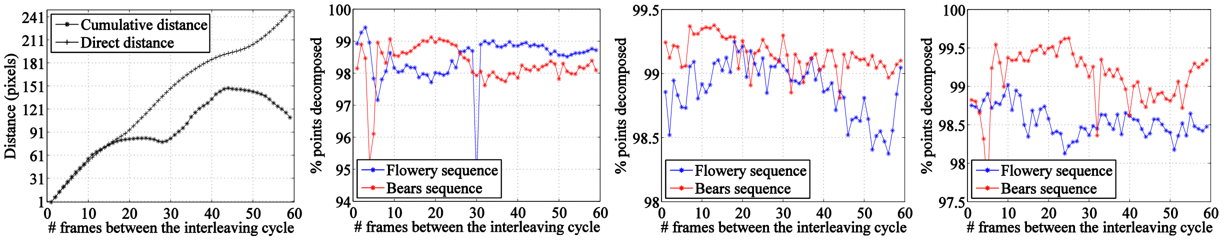
We define our error metric as the normalized error in decomposing texture and illumination images:

$$\frac{1}{N}\sqrt{\left(\frac{\widehat{x_i}-x_i}{W}\right)^2+\left(\frac{\widehat{y_i}-y_i}{H}\right)^2}, \tag{5.8}$$

where N is the number of points inside the mixed appearance regions, $(\widehat{x_i},\widehat{y_i})$, and (x_i,y_i) are the ground truth and the estimated correspondence locations, respectively. Depending on whether the error of the illumination or texture is being evaluated, (W,H) could be either the resolution of the projector or camera image.

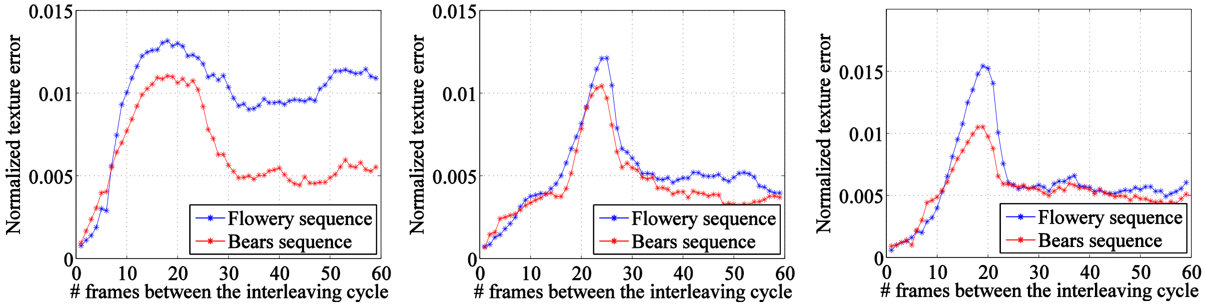
We compute two metrics for measuring the difference between the current image and the reference template: direct warp distance with respect to the template and the cumulative warp distance computed along the deformation path from the reference template to the current image. Since the motion of the cloth is quite repetitive, the second distance metric is needed to measure the difficulty in estimating the warping parameters.

Figure 5.8 shows the warp distance and percentage of points successfully decomposed for different type of appearance mixtures for the bear and flower cloth sequences along with the distance of the current frame to its reference template as a function of the interleaving period. Despite long interleaving period, the fraction of points decomposed does not experience significant drops. The texture decomposition error and the fraction of correspondences estimated for flower sequence are not as good as for the bear sequence. This indicates that the algorithm performs better with higher frequency texture. The explanation for this phenomenon is similar to the optical flow problem: tracking highly textured surfaces exhibits less drift.



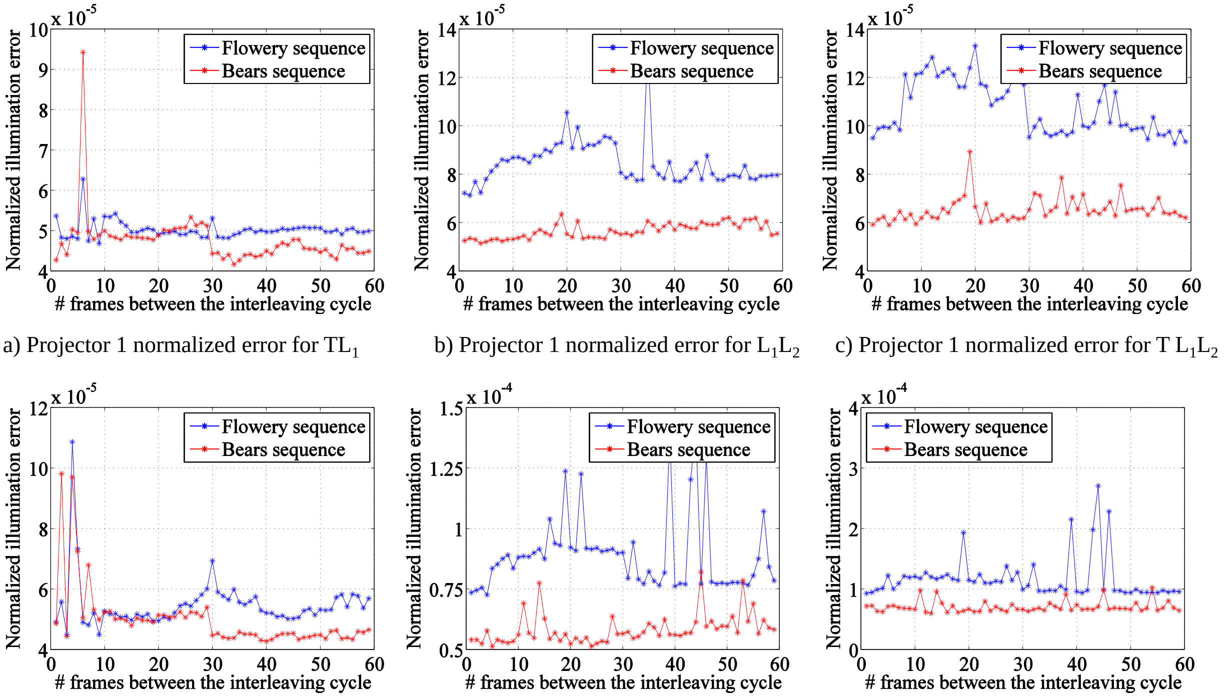
a) Warp distance b) Fraction of decomposed T L c) Fraction of decomposed L_1L_2 d) Fraction of decomposed T L_1L_2

Figure 5.8: Warping distance and the percentage of points successfully decomposed with respect to the interleaving period. The larger the distance means difficult decomposition. The decomposed points are categorized into three types: texture and illumination, two illumination patterns, and texture and two illumination patterns. The number of points successfully decomposed is stable over time. Because of its higher frequency texture, the bear cloth has higher percentage of points decomposed than the flower cloth.



a) Normalized error for TL_1 b) Normalized error for TL_2 c) Normalized error for T L_1L_2

Figure 5.9: The accuracy and robustness of the texture decomposition with respect to the interleaving period. The normalized texture errors are classified into different type of mixtures: texture-illumination pattern 1, texture-illumination pattern 2, and texture-illumination-illumination. Because of its higher frequency texture, decomposing the bear cloth sequence yields better accuracy than the flower cloth.



a) Projector 1 normalized error for TL_1

b) Projector 1 normalized error for L_1L_2

c) Projector 1 normalized error for $T\ L_1L_2$

d) Projector 2 normalized error for TL_2

e) Projector 2 normalized error for L_1L_2

f) Projector 2 normalized error for $T\ L_1L_2$

Figure 5.10: The accuracy and robustness of the illumination decomposition with respect to the interleaving period. The normalized illumination errors for individual projectors are classified into different type of mixtures: texture-illumination,illumination-illumination, and texture-illumination-illumination. Because of its higher frequency texture, decomposing the bear cloth sequence yields better accuracy than the flower cloth.

We show the texture decomposition accuracy of our method for the bear and flower cloth sequences on different type of mixed regions in Figure 5.9. Since the error does not increase dramatically over time, our method does not require frequent interleaving. The maximum normalized error for texture and multiple illumination patterns mixing is slightly higher than the mix of the texture and only one illumination pattern. This is expected because of its high warping complexity. While the bear cloth achieves better accuracy than the flower cloth when only one projector is visible to it, the difference appears to be smaller when both projectors are visible. This is because the multiple illuminations create more image gradient in the observed images, only very accurate estimation of texture patch can re-synthesize the observed image.

Figure 5.10 shows our accuracy and robustness in decomposing the illumination. Despite some sudden spikes, the effect of the interleaving period on the accuracy is minimal. Except for the mixing between projector 2 and the texture, estimating the warping parameters for the bear cloth is twice more accurate than the flower cloth. This behavior follows the trend of the previous graphs.

5.5 Results on Real Dataset

We conduct several experiments with real cloth sequences of different texture frequencies for three different scenarios: one projector and textured object, two projectors and textureless object, and two projectors and textured object. In the first case, we interleave the all-white pattern every 30 frames. In the second case, since the two illumination patterns have provided the warping templates, no interleaving is needed. In the third case, we project the interleaving pattern every 10 frames because of the complexity of the scene appearance. For all of our experiments, the scenes are illuminated using a 1280×800 DLP View Sonic projector and the images are acquired by the Canon XH-G1s HD 1920×1080 camera operating at 30fps. The camera and the projectors are calibrated using the method of Vo et al [218]. The results are presented without any post-processing.

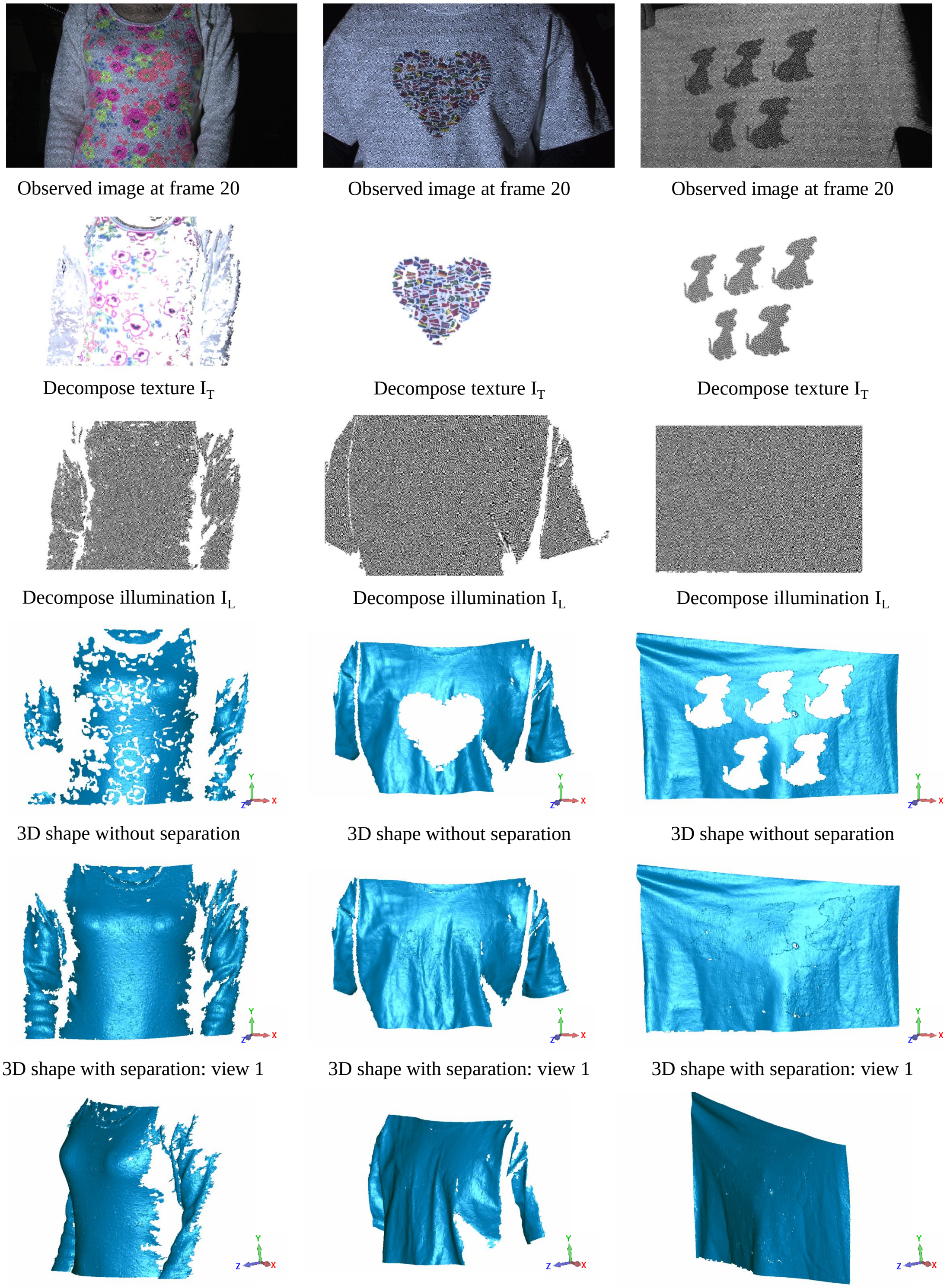
5.5.1 Texture and Illumination Mixing

We validate our algorithm for the texture and illumination separating task on three textured cloth sequences: flower, flag, and dog, arranged in increasing order of the texture frequency. Figure 5.11 shows the decomposition results. While the decomposed texture image can be imcompleted for low texture frequency object, such as the flower shirt, such texture does not cause confusion in the illumination decoding as the estimated illumination image contains few empty regions. For high frequency texture object, such as the dog cloth, both the texture and illumination image are well decomposed. Interestingly, even for seemingly textureless regions, as in the sleeves of the flower shirt, the composite of the rough but textureless cloth and the illumination explains the observed image well, which does not happen with pure illumination alone. This enables us to recover the surface shape in such regions.

Since the flower shirt has relatively low texture frequency, without texture and illumination separation, its 3D shape is best-estimated among the 3 sequences. However, the abrupt changes in color at the boundary of the flowers still alter the appearance of the illumination pattern and the estimated 3D shape cannot be completed. As the texture frequency increases, the chance of reconstructing the surface shape in the mixed region decreases as shown for the flag and dog cloths. With our separation algorithm, the 3D shape is accurately recovered even in the mixed appearance regions for all three sequences. We obtain subtle 3D deformation, as vividly shown in the flag shirt sequence. The shape information in such large holes cannot be accurately recovered with any hole filling methods. These results, shown at frame 20 of the 30 frames interleaving period, indicates that the decomposition algorithm suffers from little drifting over time. Thus, infrequently interleaving is possible so that high temporal resolution of the estimated 3D surface can be computed.

5.5.2 Two Illumination Mixing

We also validate the decomposition algorithm on the textureless shirt illuminated by two projectors, as shown in Figure 5.12, where the superposition of the illumination patterns is a well-



Observed image at frame 20

Observed image at frame 20

Observed image at frame 20

Decompose texture I_T

Decompose texture I_T

Decompose texture I_T

Decompose illumination I_L

Decompose illumination I_L

Decompose illumination I_L

3D shape without separation

3D shape without separation

3D shape without separation

3D shape with separation: view 1

3D shape with separation: view 1

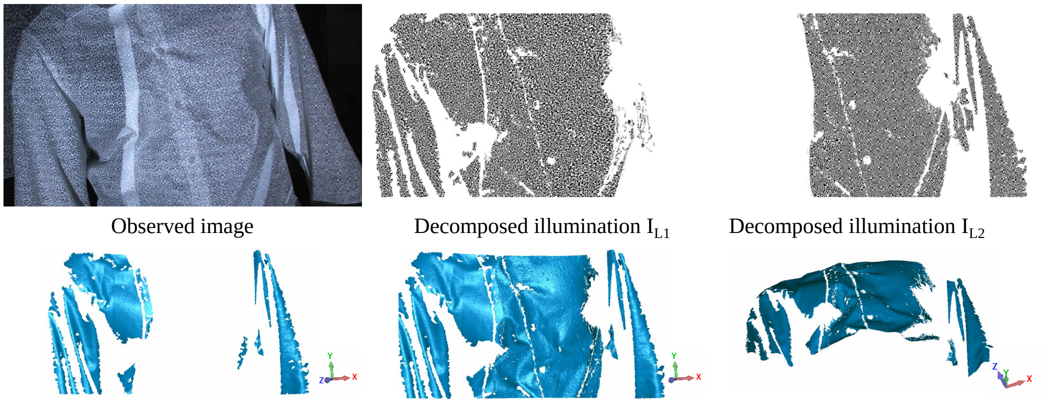
3D shape with separation: view 1

3D shape with separation: view 2

3D shape with separation: view 2

3D shape with separation: view 2

Figure 5.11: Texture-Illumination decomposition of the real cloth sequences with 30 frames interleaving. Due to the relatively low texture frequency of the flower shirt, its texture image cannot be complete recovered. Yet, all the sequences, the illumination images are well computed. Compared to the 3D shape without the separation scheme, detailed surface information is recovered after texture and illumination separation. Please refer to our project website for the videos of the results.



Observed image

Decomposed illumination I_{L1}

Decomposed illumination I_{L2}

3D shape without separation

3D shape with separation: view 1

3D shape with separation: view 2

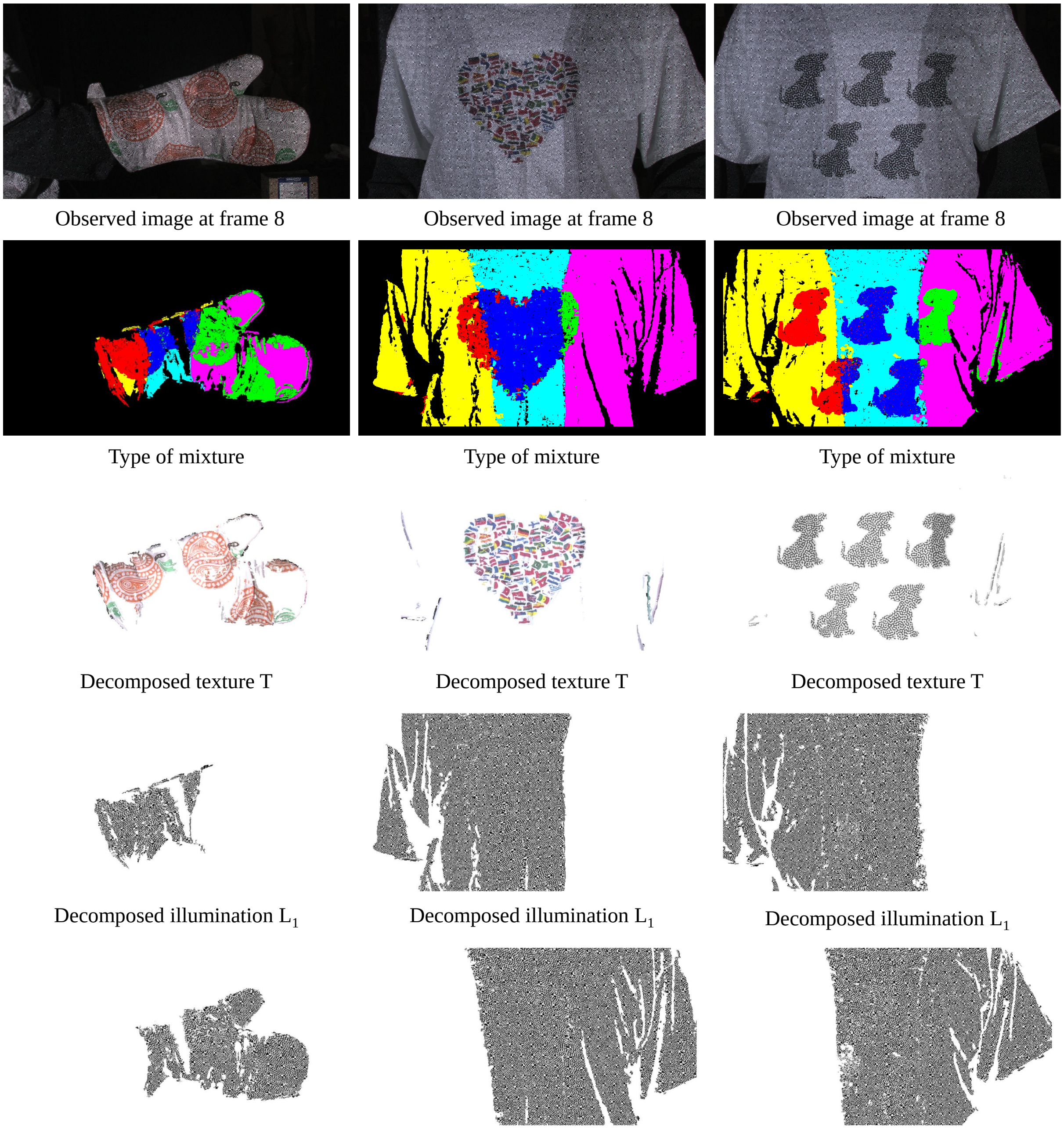
Figure 5.12: Illumination decomposition and 3D shape of the real cloth sequences. Estimating the 3D shape of the cloth in the mixed illumination regions is not possible without the decomposition scheme. After the two illumination patterns are separated, the details of the folding cloth are clearly revealed. The results are without any post-processing. Please refer to our project website for the videos of the results.

known problem. In this setup, while we experimented with two different high frequencies illumination pattern, the same patterns can be used as well, as shown in the next section. Since the illumination patterns have provided the reference templates for the warp function, no interleaving is needed. As can be seen, heavily distorted appearance of the illumination patterns in their overlapping regions makes shape recovery impossible. However, after our decomposition, dense and accurate shape information can be computed in a single-shot. The decomposition fails when occlusion or strong foreshortening occurs.

5.5.3 Texture and Two Illumination Mixing

Figure 5.13 shows our decomposition results for the glove, flag, and dog sequences. The dog sequence contains the highest texture frequency. As encoded in the mixture type image, the scenes contain six different type of appearances: pure illumination from projector 1 (yellow), pure illumination from projector 2 (pink), projector 1 and texture (red), projector 2 and texture (green), projector 1 and projector 2 (light blue), and projector 1, projector 2, and texture (blue). Such complication of the mixture types makes image decomposition challenging. Consequently, while there could be places where the mixture types are not visually correct, the majority of them happen in either the transition region of the illumination pattern (see Figure 5.4), the small but homogeneous textured regions, or the occluding boundary. As can be seen in the decomposed images, despite the simple imaging model, our approach can handle complex appearances of real world textured objects illuminated by the two projectors. We believe this is due to the local block decomposition strategy which is robust to global lighting variation.

Figure 5.14 shows the 3D shape of those cloth sequences. Without our separation scheme, the estimated 3D shapes are largely incomplete with obvious holes. In contrast, after our de-



Observed image at frame 8

Observed image at frame 8

Observed image at frame 8

Type of mixture

Type of mixture

Type of mixture

Decomposed texture T

Decomposed texture T

Decomposed texture T

Decomposed illumination L_1

Decomposed illumination L_1

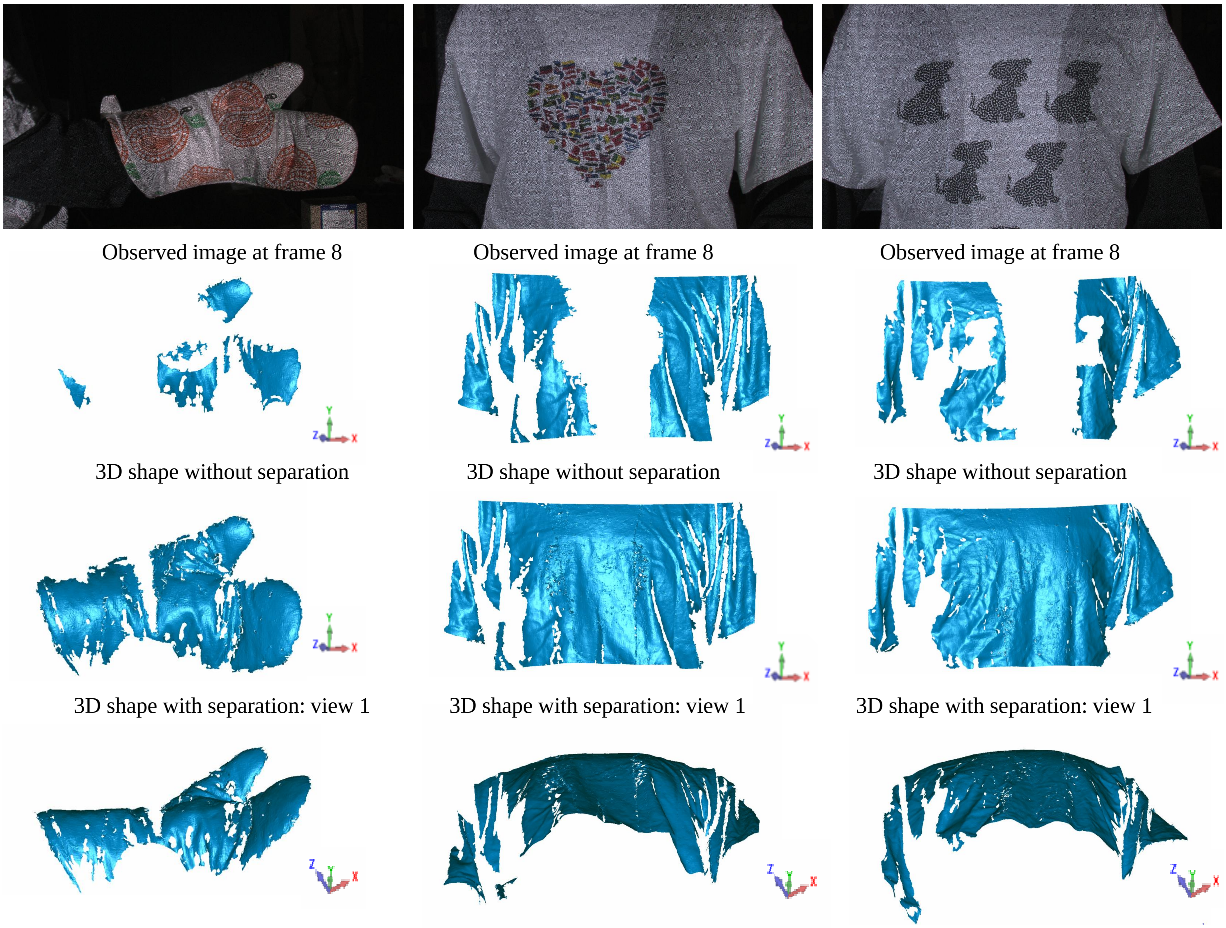
Decomposed illumination L_1

Decomposed illumination L_2

Decomposed illumination L_2

Decomposed illumination L_2

Figure 5.13: Decomposition of the real cloth sequences. The interleaving pattern is projected every 10 frames. The scenes contain six different type of appearances: pure illumination from projector 1 (yellow), pure illumination from projector 2 (pink), projector 1 and texture (red), projector 2 and texture (green), projector 1 and projector 2 (light blue), and projector 1, projector 2, and texture (blue).



Observed image at frame 8

Observed image at frame 8

Observed image at frame 8

3D shape without separation

3D shape without separation

3D shape without separation

3D shape with separation: view 1

3D shape with separation: view 1

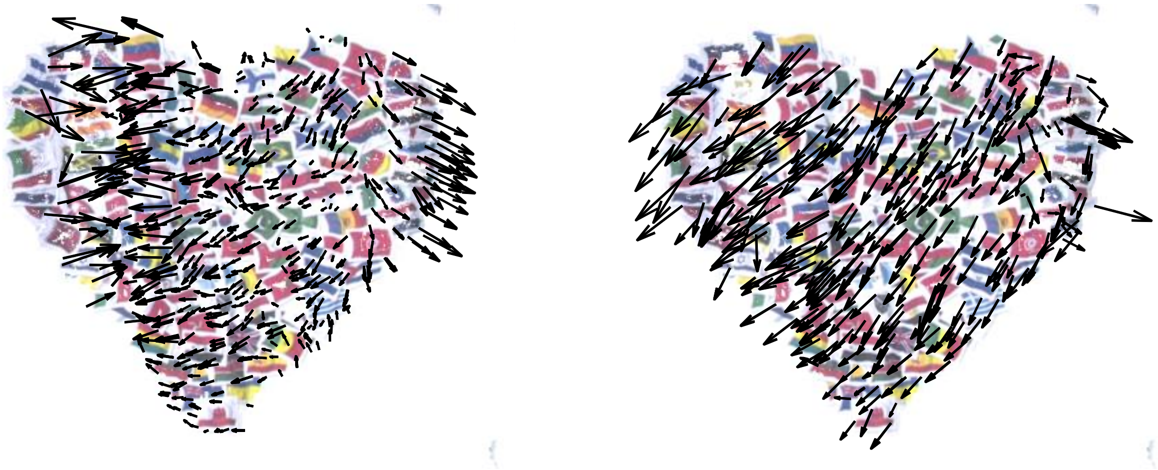
3D shape with separation: view 1

3D shape with separation: view 2

3D shape with separation: view 2

3D shape with separation: view 2

Figure 5.14: Estimated 3D shape for the cloth sequences. The interleaving pattern is projected every 10 frames. No post-processing is applied. Without the decomposition, estimated 3D shapes contain large holes that hole-filling methods cannot yield reasonable results. Conversely, our decomposition scheme allows accurate 3D surface estimation with clearly visible 3D shape of the folded cloth. Please refer to our project website for the videos of the results.



Direct tracking on the observed image

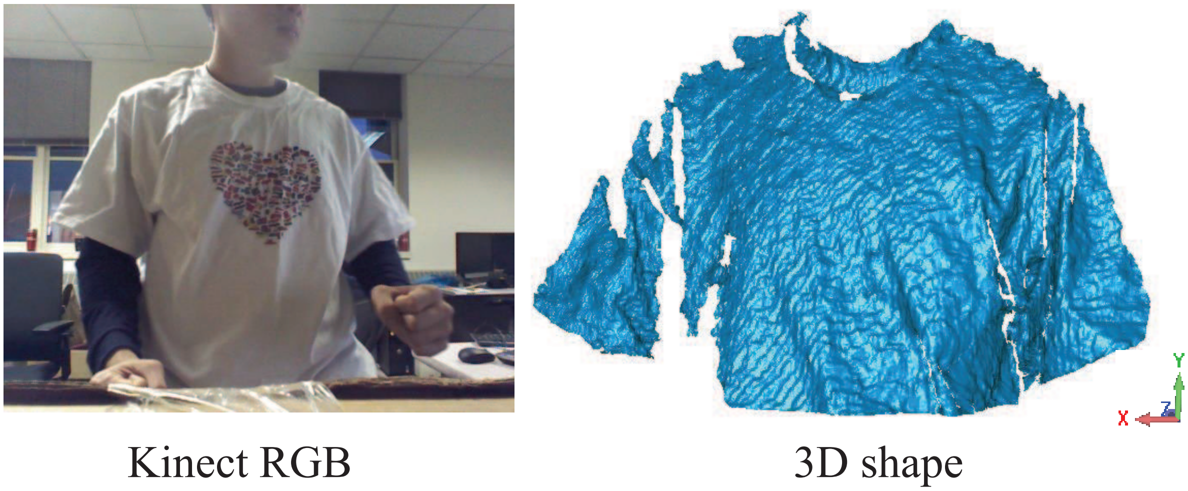
Tracking on the decomposed texture

Figure 5.15: Benefit of texture separation for tracking. Applying the tracking algorithm directly on the observed image results erroneous optical flow. Conversely, after texture separation, the same tracking algorithm can faithfully show how the cloth is moving.

composition, the shape of the cloth is much better represented. While we could not obtain good reconstruction at the texture and illumination mixing regions where there are abrupt intensity changes at transition region the illumination patterns, the estimated shapes in the texture and two illuminations mixing are visually plausible. Our estimated shape of folded cloth, which cannot be obtained by hole-filling methods, appears naturally.

Figure 5.15 shows the benefit of texture separation for surface tracking. Because of the mixed appearance, direct application of the patch based tracking algorithm [18] on the observed image does not yield plausible optical flow. The directions of the flows are noisy and they do not any dominant directions. Conversely, when the same tracking algorithm is applied on the decomposed texture image, the direction of the flow is less noisy and they faithfully present the true motion of the cloth. Such motion tracking could be helpful to register shape to create a complete 3D model of the object.

Figure 5.16 shows the 3D shape obtained from the Kinect I sensor. For fair comparison with the performance of the proposed method shown in the flag sequence (see Figure 5.13), the same subject stands at a similar distance to the sensor. Visually, the quality 3D shape from our method outperforms that of the Kinect. It is noteworthy that unless smoothing filter is applied to raw Kinect results, the mesh generation fails as the surface normal computed from the raw point cloud is too noisy.



Kinect RGB

3D shape

Figure 5.16: 3D shape from the Kinect. Note that smoothing filters has been applied to generate the mesh from raw point cloud data.

5.6 Discussion

Since we use the texture warping parameters estimated in the previous frame as initialization, empirically, we can robustly warp an observed patch to its reference template that is temporally far away. Thus, only infrequent projection of the interleaving white frame is needed and the obtained results have high temporal resolution. This strategy is analogous to the approach of Tian and Narasimhan [208] who use less distorted patches to estimate globally optimal sets of warping parameters. Clearly, since the interleaving sequence is dependent on the motion of the object, the interleaving period has to be adapted to different applications. Nevertheless, in the era 60-fps consumer-grade cameras, there is no need to interleave every other frame.

In spite of the greedy growing strategy, erroneous matches cannot propagate long as examined and thresholded by the cost function in Equation 5.6. Hence, our method avoids both the global ambiguity of the illumination pattern and being stuck in regions where occlusion, surface discontinuity, or severe foreshortening occurs. Moreover, since only a few seed points are needed initially, the good correspondences in the current frame can quickly propagate to non-decomposable regions in the earlier frames. Such automatic recovery from previous failures is another important property of our method.

For objects with high frequency texture everywhere, the proposed algorithm will fail because it cannot be properly initialized. Yet, such cases are rare and most objects have a mixture of low and high frequency texture regions (see Figure 5.11). While our separation method may also fail for objects with low frequency texture, the camera-projector correspondence can be established to reconstruct 3D shape for those cases.

While we only show the results for a single deforming object, our algorithm is applicable to general scenes containing multiple objects. As long as the seed points, i.e. correspondences established in low frequency textured regions, are available on each object, these correspondences can propagate to the entire object. Nevertheless, because of the nature of the patch decomposition approach, our algorithm cannot handle well the textured regions at the occluding boundary.

Theoretically, the decomposition model of Equation 5.1 can be generalized to more projectors where the type of mixture grows exponentially. However, the increase in the number of projector faces a diminishing return when random illumination pattern from multiple projectors are averaged out. In such cases, the contrast of the observed mixed regions become too low and our decomposition fails. Nevertheless, in practice, even for a surround structured light system, as in [116], there are at most two projectors to illuminate any point on the object surface.

Lastly, due to the sequential nature of our decomposition, it does not run in realtime. Unless knowledge about the surface geometry is known, reasonable initial guess of the optimization is not possible. Using Deep Neural Nets (DNN) for coarse shape retrieval or illumination decoding [77, 183] could lead to the next performance improvement for structured light system.

5.7 Summary

We present a single-shot approach to produce dense shape reconstructions of highly textured objects illuminated by one or more projectors. Our key is an image decomposition scheme that can recover the illumination image of different projectors and the texture images of the scene from their mixed appearances. We focus on three cases of mixed appearances: the illumination from one projector onto textured surface, illumination from multiple projectors onto a textureless surface, or their combined effect. Our method can accurately compute per-pixel warps from the illumination patterns and the texture template to the observed image. The texture template is obtained by interleaving the projection sequence with an all-white pattern. The estimated warps are reliable even with infrequent interleaved projection and strong object deformation. Thus, we obtain detailed shape reconstruction and dense motion tracking of the textured surfaces.

Chapter 6

Conclusion

6.1 Summary

Throughout this thesis, we have worked towards the goal of creating a computational pipeline for automatic dynamic event reconstruction from multiple video cameras in unconstrained crowd capture settings. This goal is challenging due to the current limited understanding of dynamic motion reconstruction without geometric constraints and the ability to find reliable correspondences across multiple cameras in the wild. The key to our solution is to exploit physical motion dynamics (Chapter 2), shape deformation (Chapter 4 and 5), scene semantics (Chapter 3), and their couplings (Chapter 4). We have validated the proposed algorithms on several large scale datasets and achieved unprecedented results. We believe this thesis could have strong impacts on many fields: movie industry, surveillance, media analytics, and crowd psychology understanding.

6.2 Future Work

We see two exciting future directions that bring this thesis closer to daily consumers applications. Looking back our pipeline in Figure 1.2, the future extensions can be broken down as shown in Figure 6.1. These directions will be our guiding north star for the next 5 years.

6.2.1 Large Scale Dynamic Event Analysis

There are two big exciting trends in the current technological world: the online media sharing platforms and the emergence of always-on AR glasses (e.g, Snap spectacles or Google Glass). Each of these trends generates enormous visual data. While this thesis was aimed to tackle such massive data organization problem directly, the actual implementation of such pipeline requires a better tool for camera temporal alignment, faster and more reliable data association across multiple modalities, and the ability to reconstruct 3D object from a monocular camera. We believe each of these problems is interconnected and progress in each dimension will benefit the others. For example, a motion prior can improve temporal association, and better association from the

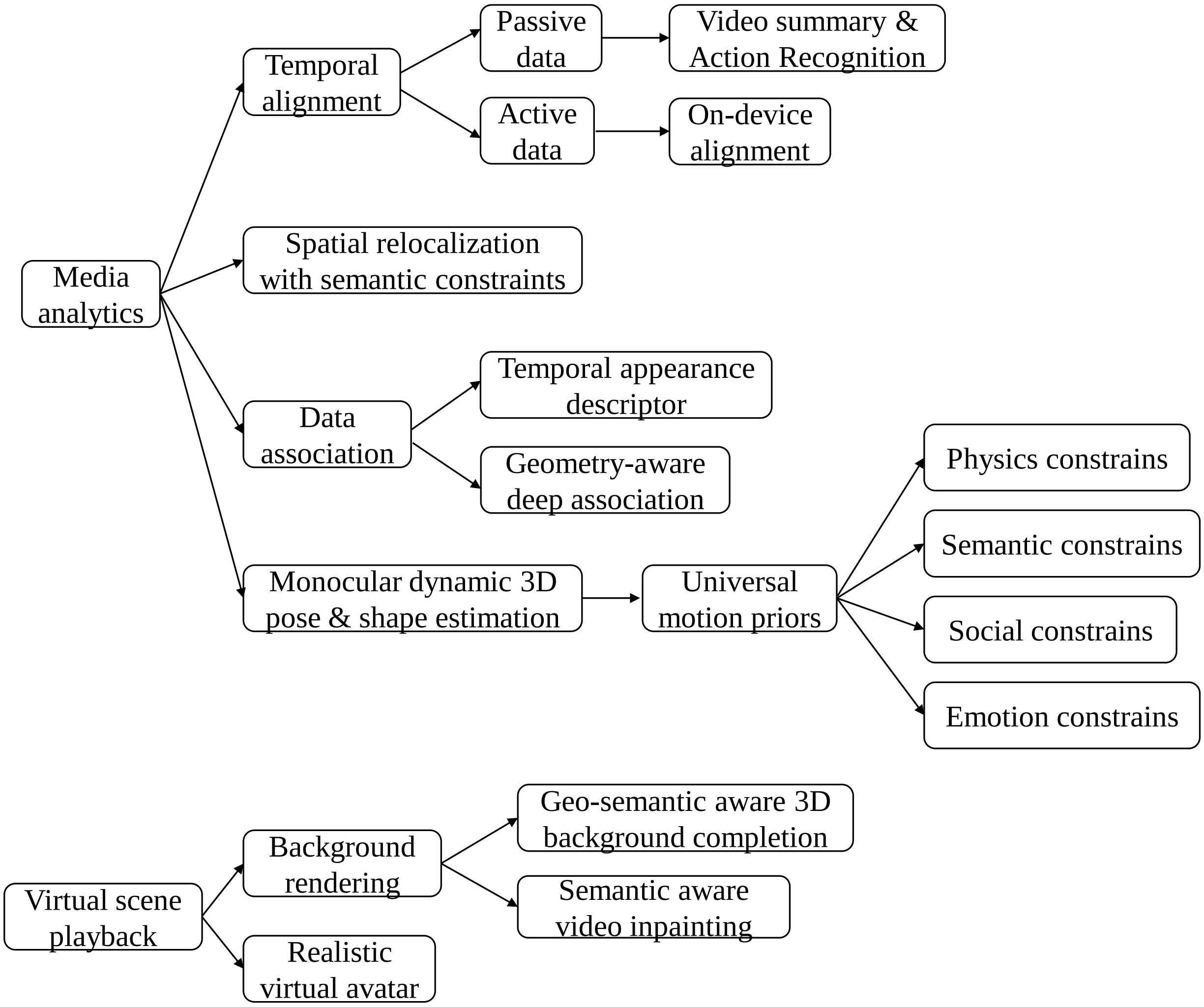

Media analytics
Temporal alignment
Passive data
Video summary & Action Recognition
Active data
On-device alignment
Spatial relocalization with semantic constraints
Data association
Temporal appearance descriptor
Geometry-aware deep association
Monocular dynamic 3D pose & shape estimation
Universal motion priors
Physics constrains
Semantic constrains
Social constrains
Emotion constrains
Virtual scene playback
Background rendering
Geo-semantic aware 3D background completion
Semantic aware video inpainting
Realistic virtual avatar

Figure 6.1: Future extensions to tackle large scale media analytics and virtual scene browsing.

multiview systems reconstruct better 3D motion which is a data source for learning motion prior.

Temporal alignment: For all algorithms developed in this thesis, unless all cameras to be temporally aligned to second-level accuracy, the algorithms cannot bootstrap itself (Chapter 3. However, the metadata of the videos available from media sharing platforms is severely biased in their recording time depending on the wireless connection settings. It is also unclear how if the original video framerate is preserved upon retrieval. These factors make temporal modeling difficult. For passive visual data that are mined from the Internet, one interesting direction is to exploit learning-based semantic action recognition [69, 70, 207] and scene summarization [92, 146, 242] for rough camera alignment. For active data generated from AR glasses or a group of users running customized camera applications, temporal alignment must be done on-device using camera API [13] and wireless network protocol [127]. This will give the highest alignment precision with minimal computational cost.

Robust object tracking and geometric-aware data association with deep learning: While adaptive per-frame appearance descriptor really unlocks 3D motion tracking at unprecedented scale and accurate tracking improve semantics detections. Unfortunately, these approaches are only suited for offline applications. Furthermore, regardless of the training data scope, there will always be testing data that falls outside the training data distribution. Such failure could be catastrophic for applications such as self-driving cars. To tackle the issues, the appearance descriptor should be temporally resilient and the descriptor distribution could be multi-modal or keyframe-based as inspired by SLAM [66, 160] and motion tracking systems [61]. Another interesting direction is to build the geometric knowledge directly into the association network [176] and exploit multiway graph matching for one-shot multiview association [100, 260].

Online camera pose relocalization of crowded scene: For any interesting dynamic event that involves multiple human actively interacting with each other, the static objects in the background such as buildings and walls are mostly occluded by humans. Since the number of static features observed from multiple views is significantly smaller than the number of dynamic features, camera localization is very challenging. There are few visual features of static objects available to build a reliable model of the scene. As the number of people and the complexity of the activity increases, both camera localization and human tracking become considerably more challenging due to frequent occlusions and strong viewpoint and appearance variations. From a high-level reasoning perspective, one potential approach to understand the geometry of the scene from the same set of visual data is to associate the same person across different cameras and fuse the keypoint association with the static feature to constrain the camera relocalization.

Monocular dynamic 3D reconstruction using universal motion priors: While we specifically assume the availability of multiview cameras, depending on the scale and the diversity of the dynamic event, it often occurs that only a single camera can observe the object. While monocular 3D tracking is inherently ill-posed, we believe it can be constrained by better understanding of the body motion and its physical environment (such as ground plane penetrations and object collisions [247]), semantic 3D environment (such as affordance surfaces [88, 134]), social interactions [113, 182], and even more subtle cues such as motion style changes due to the state of emotion [175]. In case of active data, the reconstruction could be further constrained by accounting for the 6-dof ego-motion of the capturing devices or simply the IMU information [102].

6.2.2 Realistic Virtual Scene Browsing

Virtual scene browsing, as a direct product of 4D scene reconstruction, is probably the most exciting application to the consumers. Unlike static scene browsing, the visual demand for dynamic scene browsing is much higher because the browsing experience is mostly personal. Human eyes are sensitive to those artifacts, especially for scenes familiar with themselves. Research in this direction should tackle both the background scene and the foreground rendering using modern deep learning approaches, which has strong potentials of exceeding classical texturemap rendering pipelines [74, 97, 154, 198].

Background rendering: Despite significant efforts in event reconstruction, artifacts in both

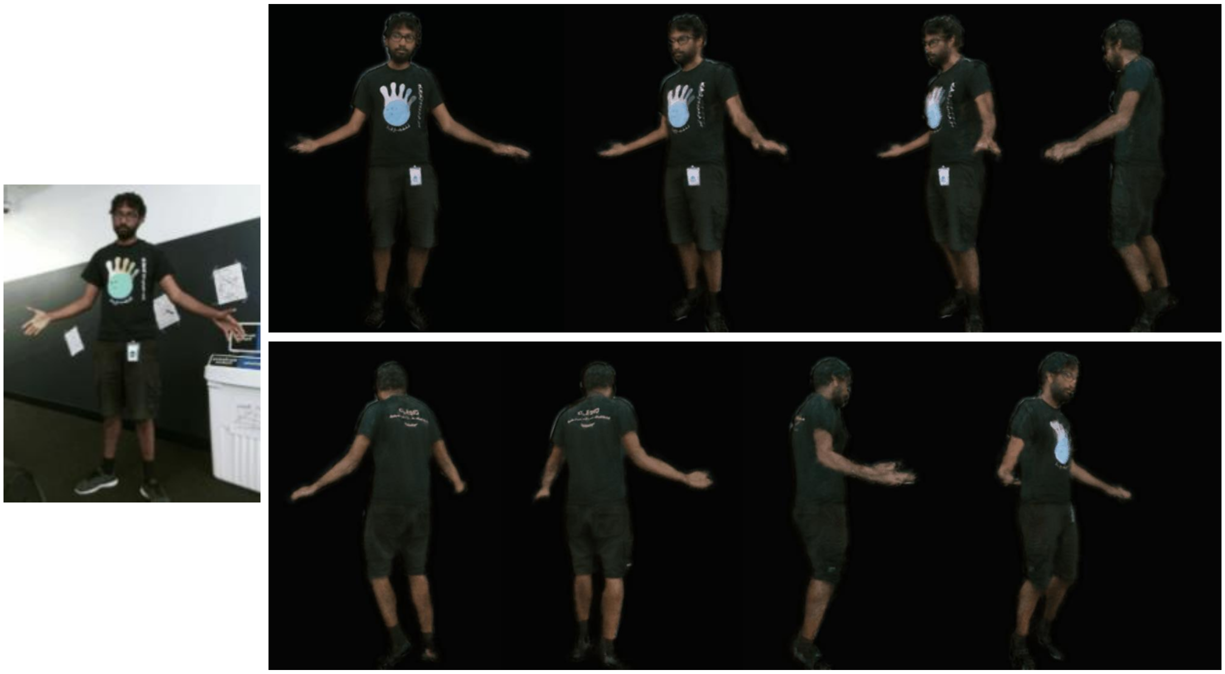


Figure 6.2: Our early attempt for realistic full-body animatable human avatar. We build the appearance learning on top of the statistical 3D human mesh representation and achieve coherent and realistic appearance rendering from only a single few-minutes-long video data.

shape and motion may still occur, sometimes simply due to the lack of observations (insufficient number of cameras). One very promising direction is to reconstruct the semantic 3D scene as a whole [48, 93]. This approach leverages the generic prior in real-world object configurations and produces reconstruction without holes. The reconstructions can be later realistified using by learning image priors from the event itself [22, 173].

Foreground rendering: Even with a state of art laboratory multiview capture system, the current state of art methods fail to create 3D human model that can fool the human perceptual system. Clearly, such a task is out of reach for in-the-wild data. However, while intrinsic body parts such as the face should remain faithful, other parts such as the hairs or clothing deformation can be modified slightly during the view transition. Building on top of a statistical 3D human body mesh fitting as a data presentation for learning the view dependence appearance, we have achieved promising results for human rendering from single few-minutes-long video clip (See Figure 6.2). We believe the learning could be further augmented with user-specific training data from social media platforms to further constraint the appearance generation.

Bibliography

[1] http://mocap.cs.cmu.edu.

[2] http://www.crowdflik.com/.

[3] http://www.intel.com/content/www/us/en/sports/360-replay-technology-overview.html.

[4] Supreeth Achar and Srinivasa G Narasimhan. Multi focus structured light for recovering scene shape and global illumination. In *ECCV*, 2014.

[5] Supreeth Achar, Joseph R. Bartels, William L. 'Red' Whittaker, Kiriakos N. Kutulakos, and Srinivasa G. Narasimhan. Epipolar time-of-flight imaging. *TOG*, 2017.

[6] Sameer Agarwal, Keir Mierle, and Others. Ceres solver. http://ceres-solver.org.

[7] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015.

[8] Cenek Albl, Zuzana Kukelova, and Tomas Pajdla. R6p - rolling shutter absolute camera pose. In *CVPR*, 2015.

[9] Cenek Albl, Zuzana Kukelova, and Tomas Pajdla. R6p-rolling shutter absolute pose problem. In *CVPR*, 2015.

[10] Cenek Albl, Akihiro Sugimoto, and Tomas Pajdla. Degeneracies in rolling shutter sfm. In *ECCV*, 2016.

[11] Rıza Alp Güler, Natalia Neverova, and Iasonas Kokkinos. Densepose: Dense human pose estimation in the wild. In *CVPR*, 2018.

[12] Nicolas Alt, Stefan Hinterstoisser, and Nassir Navab. Rapid selection of reliable templates for visual tracking. In *CVPR*, 2010.

[13] Sameer Ansari, Neal Wadhwa, Rahul Garg, and Jiawen Chen. Wireless software synchronization of multiple distributed cameras. In *ICCP*, 2019.

[14] Ido Arev, Hyun Soo Park, Yaser Sheikh, Jessica Hodgins, and Ariel Shamir. Automatic editing of footage from multiple social cameras. *TOG*, 2014.

[15] Shayan Modiri Assari, Haroon Idrees, and Mubarak Shah. Human re-identification in crowd videos using personal, social and environmental constraints. In *ECCV*, 2016.

[16] B Atcheson, W Heidrich, and I Ihrke. An evaluation of optical flow algorithms for background oriented schlieren imaging. *Exp in Fluids*, 2009.

[17] Shai Avidan and Amnon Shashua. Trajectory triangulation: 3d reconstruction of moving points from a monocular image sequence. *PAMI*, 2000.

[18] Simon Baker and Iain Matthews. Lucas-kanade 20 years on: A unifying framework. *IJCV*, 2004.

[19] Luca Ballan, Gabriel J Brostow, Jens Puwein, and Marc Pollefeys. Unstructured video-based rendering: Interactive exploration of casually captured videos. *TOG*, 2010.

[20] Davide Baltieri, Roberto Vezzani, and Rita Cucchiara. 3dpes: 3d people dataset for surveillance and forensics. In *Proc. of the 2011 joint ACM workshop on Human gesture and behavior understanding*, 2011.

[21] Aayush Bansal, Bryan Russell, and Abhinav Gupta. Marr revisited: 2d-3d alignment via surface normal prediction. In *CVPR*, 2016.

[22] Aayush Bansal, Yaser Sheikh, and Deva Ramanan. Shapes and context: In-the-wild image synthesis manipulation. In *CVPR*, 2019.

[23] Pierre Baqué, François Fleuret, and Pascal Fua. Deep occlusion reasoning for multi-camera multi-target detection. In *ICCV*, 2017.

[24] Jean-Charles Bazin and Alexander Sorkine-Hornung. Actionsnapping: Motion-based video synchronization. In *ECCV*, 2016.

[25] Stephan Beck, Andre Kunert, Alexander Kulik, and Bernd Froehlich. Immersive group-to-group telepresence. *TVCG*, 2013.

[26] Aseem Behl, Omid Hosseini Jafari, Siva Karthik Mustikovela, Hassan Abu Alhaija, Carsten Rother, and Andreas Geiger. Bounding boxes, segmentations and object coordinates: How important is recognition for 3d scene flow estimation in autonomous driving scenarios? In *ICCV*, 2017.

[27] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua. Multiple object tracking using k-shortest paths optimization. *TPAMI*, 2011.

[28] Alina Bialkowski, Simon Denman, Sridha Sridharan, Clinton Fookes, and Patrick Lucey. A database for person re-identification in multi-camera surveillance networks. In *International Conference on Digital Image Computing Techniques and Applications*, 2012.

[29] Irving Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 1987.

[30] Thomas O Binford. Visual perception by computer. In *IEEE Conf. on Systems and Control*, 1971.

[31] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it smpl: Automatic estimation of 3d human pose and shape from a single image. In *ECCV*. Springer, 2016.

[32] MR Brito, EL Chavez, AJ Quiroz, and JE Yukich. Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection. *Statistics & Probability Letters*, 1997.

[33] D Alex Butler, Shahram Izadi, Otmar Hilliges, David Molyneaux, Steve Hodges, and David Kim. Shake'n'sense: reducing interference for overlapping structured light depth

cameras. In *SIGCHI Human Factors in Computing Systems*, 2012.

[34] Asad A Butt and Robert T Collins. Multi-target tracking by lagrangian relaxation to min-cost network flow. In *CVPR*.

[35] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017.

[36] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018.

[37] Dalit Caspi, Nahum Kiryati, and Joseph Shamir. Range imaging with adaptive color structured light. *TPAMI*, 1998.

[38] Yaron Caspi, Denis Simakov, and Michal Irani. Feature-based sequence-to-sequence matching. *IJCV*, 2006.

[39] J Cech, J Sanchez-Riera, and R Horaud. Scene flow estimation by growing correspondence seeds.

[40] Florian Chabot, Mohamed Chaouch, Jaonary Rabarisoa, Céline Teulière, and Thierry Chateau. Deep manta: A coarse-to-fine many-task network for joint 2d and 3d vehicle analysis from monocular image. *arXiv preprint arXiv:1703.07570*, 2017.

[41] Tatjana Chavdarova, Pierre Baqu, Stphane Bouquet, Andrii Maksai, Cijo Jose, Timur Bagautdinov, Louis Lettry, Pascal Fua, Luc Van Gool, and Franois Fleuret. Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[42] HJ Chen, J Zhang, DJ Lv, and J Fang. 3-d shape measurement by composite pattern projection and hybrid processing. *Optics express*, 2007.

[43] Tongbo Chen, Hendrik PA Lensch, Christian Fuchs, and Hans-Peter Seidel. Polarization and phase-shifting for 3d scanning of translucent objects. In *CVPR*, 2007.

[44] Tongbo Chen, Hans-Peter Seidel, and Hendrik PA Lensch. Modulated phase-shifting for 3d scanning. In *CVPR*, 2008.

[45] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. 2017.

[46] De Cheng, Yihong Gong, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, 2016.

[47] Dong Seon Cheng, Marco Cristani, Michele Stoppa, Loris Bazzani, and Vittorio Murino. Custom pictorial structures for re-identification. In *BMVC*, 2011.

[48] Ian Cherabier, Johannes L. Schonberger, Martin R. Oswald, Marc Pollefeys, and Andreas Geiger. Learning priors for semantic 3d reconstruction. In *ECCV*, 2018.

[49] Falak Chhaya, Dinesh Reddy, Sarthak Upadhyay, Visesh Chari, M Zeeshan Zia, and K Madhava Krishna. Monocular reconstruction of vehicles: Combining slam with shape priors. In *ICRA*, 2016.

[50] Wongun Choi. Near-online multi-target tracking with aggregated local flow descriptor. In *ICCV*, 2015.

[51] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, 2005.

[52] Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *TOG*, 2015.

[53] Robert T Collins. Multitarget data association with higher-order motion models. In *CVPR*, 2012.

[54] Timothy F Cootes, Christopher J Taylor, David H Cooper, and Jim Graham. Active shape models-their training and application. *CVIU*, 1995.

[55] Vincent Couture, Nicolas Martin, and Sebastien Roy. Unstructured light scanning to overcome interreflections. In *ICCV*, 2011.

[56] Yuchao Dai, Hongdong Li, and Laurent Kneip. Rolling shutter camera relative pose: Generalized epipolar geometry. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[57] Abir Das, Anirban Chakraborty, and Amit K Roy-Chowdhury. Consistent re-identification in a camera network. In *ECCV*, 2014.

[58] Andrew J Davison, Ian D Reid, Nicholas D Molton, and Olivier Stasse. Monoslam: Real-time single camera slam. *TPAMI*, 2007.

[59] Afshin Dehghan, Shayan Modiri Assari, and Mubarak Shah. Gmmcp tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking. In *CVPR*, 2015.

[60] Afshin Dehghan, Yicong Tian, Philip HS Torr, and Mubarak Shah. Target identity-aware network flow for online multiple target tracking. In *CVPR*, 2015.

[61] Mingsong Dou, Sameh Khamis, Yury Degtyarev, Philip Davidson, Sean Ryan Fanello, Adarsh Kowdle, Sergio Orts Escolano, Christoph Rhemann, David Kim, Jonathan Taylor, et al. Fusion4d: Real-time performance capture of challenging scenes. *TOG*, 2016.

[62] Sylvain Duchêne, Clement Riant, Gaurav Chaurasia, Jorge Lopez-Moreno, Pierre-Yves Laffont, Stefan Popov, Adrien Bousseau, and George Drettakis. Multi-view intrinsic images of outdoors scenes with an application to relighting. *TOG*, 2015.

[63] A Elhayek, C Stoll, K Kim, H Seidel, and C Theobalt. Feature-based multi-video synchronization with subframe accuracy. *Pattern Recognition*, 2012.

[64] Ahmed Elhayek, Carsten Stoll, Nils Hasler, Kwang In Kim, H-P Seidel, and Christian Theobalt. Spatio-temporal motion tracking with unsynchronized cameras. In *CVPR*, 2012.

[65] Ahmed Elhayek, Edilson de Aguiar, Arjun Jain, J Thompson, Leonid Pishchulin, Mykhaylo Andriluka, Chris Bregler, Bernt Schiele, and Christian Theobalt. Marconi: Convnet-based markerless motion capture in outdoor and indoor scenes. *TPAMI*, 2017.

[66] Jakob Engel, Vladlen Koltun, and Daniel Cremers. Direct sparse odometry. *TPAMI*, 2017.

[67] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[68] Michela Farenzena, Loris Bazzani, Alessandro Perina, Vittorio Murino, and Marco Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, 2010.

[69] Yang Feng, Lin Ma, Wei Liu, Tong Zhang, and Jiebo Luo. Video re-localization. In *ECCV*, 2018.

[70] Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. Spatio-temporal video re-localization by warp lstm. In *CVPR*, 2019.

[71] Richard P Feynman, Rober B Leighton, and Matthew Sands. The Feynman lectures in physics, Mainly Electromagnetis and Matter, Vol. ii, 1963.

[72] G. D Finlayson, S. D Hordley, C Lu, and M. S Drew. On the removal of shadows from images. *TPAMI*, 2006.

[73] Francois Fleuret, Jerome Berclaz, Richard Lengagne, and Pascal Fua. Multicamera people tracking with a probabilistic occupancy map. *TPAMI*, 2008.

[74] John Flynn, Ivan Neulander, James Philbin, and Noah Snavely. Deepstereo: Learning to predict new views from the world's imagery. In *CVPR*, 2016.

[75] Jan-Michael Frahm, Pierre Fite-Georgel, David Gallup, Tim Johnson, Rahul Raguram, Changchang Wu, Yi-Hung Jen, Enrique Dunn, Brian Clipp, Svetlana Lazebnik, et al. Building rome on a cloudless day. In *ECCV*. 2010.

[76] Ryo Furukawa, Ryusuke Sagawa, Hiroshi Kawasaki, Kazuhiro Sakashita, Yasushi Yagi, and Naoki Asada. One-shot entire shape acquisition method using multiple projectors and cameras. In *2010 Fourth Pacific-Rim Symposium on Image and Video Technology*, 2010.

[77] Ryo Furukawa, Daisuke Miyazaki, Masashi Baba, Shinsaku Hiura, and Hiroshi Kawasaki. Robust structured light system against subsurface scattering effects achieved by cnn-based pattern detection and decoding algorithm. In *ECCV*, 2018.

[78] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *PAMI*, 2010.

[79] Yasutaka Furukawa, Brian Curless, Steven M Seitz, and Richard Szeliski. Towards internet-scale multi-view stereo. In *CVPR*, 2010.

[80] Tiago Gaspar, Paulo Oliveira, and Paolo Favaro. Synchronization of two independently moving cameras without feature correspondences. In *ECCV*, 2014.

[81] Mengyue Geng, Yaowei Wang, Tao Xiang, and Yonghong Tian. Deep transfer learning for person re-identification. *arXiv preprint arXiv:1611.05244*, 2016.

[82] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014.

[83] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017.

[84] Douglas Gray and Hai Tao. Viewpoint invariant pedestrian recognition with an ensemble

of localized features. *ECCV*, 2008.

[85] Jinwei Gu, Toshihiro Kobayashi, Mohit Gupta, and Shree K Nayar. Multiplexed illumination for scene recovery in the presence of global illumination. In *ICCV*, 2011.

[86] Stefan Gumhold and Sören König. Image-based motion compensation for structured light scanning of dynamic surfaces. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2008.

[87] Fatma Guney and Andreas Geiger. Displets: Resolving stereo ambiguities using object knowledge. In *CVPR*, 2015.

[88] Abhinav Gupta, Scott Satkin, Alexei A Efros, and Martial Hebert. From 3d scene geometry to human workspace. In *CVPR*, 2011.

[89] Mohit Gupta and Shree K Nayar. Micro phase shifting. In *CVPR*, 2012.

[90] Mohit Gupta, Yuandong Tian, Srinivasa G Narasimhan, and Li Zhang. A combined theory of defocused illumination and global light transport. *IJCV*, 2012.

[91] Mohit Gupta, Amit Agrawal, Ashok Veeraraghavan, and Srinivasa G Narasimhan. A practical approach to 3d scanning in the presence of interreflections, subsurface scattering and defocus. *IJCV*, 2013.

[92] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In *ECCV*, 2014.

[93] Christian Häne, Christopher Zach, Andrea Cohen, and Marc Pollefeys. Dense semantic 3d reconstruction. *TPAMI*, 2016.

[94] Richard Hartley and Andrew Zisserman. *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[95] Nils Hasler, Bodo Rosenhahn, Thorsten Thormahlen, Michael Wand, Jürgen Gall, and Hans-Peter Seidel. Markerless motion capture with unsynchronized moving cameras. In *CVPR*, 2009.

[96] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017.

[97] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. In *TOG*, 2018.

[98] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017.

[99] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. Person re-identification by descriptive and discriminative classification. In *Scandinavian conference on Image analysis*, 2011.

[100] Qi-Xing Huang and Leonidas Guibas. Consistent shape maps via semidefinite programming. In *CGF*, 2013.

[101] Yinghao Huang, Federica Bogo, Christoph Lassner, Angjoo Kanazawa, Peter V Gehler, Javier Romero, Ijaz Akhter, and Michael J Black. Towards accurate marker-less human

shape and pose estimation over time. In *3DV*, 2017.

[102] Yinghao Huang, Manuel Kaufmann, Emre Aksan, Michael J. Black, Otmar Hilliges, and Gerard Pons-Moll. Deep inertial poser learning to reconstruct human pose from sparseinertial measurements in real time. *TOG*, 2018.

[103] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of classification*, 1985.

[104] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[105] Michal Irani, Benny Rousso, and Shmuel Peleg. Computing occluding and transparent motions. *IJCV*, 1994.

[106] Sarthak Sharma J. Krishna Murthy and K. Madhava Krishna. Shape priors for real-time monocular object localization in dynamic environments. In *IROS*, 2017.

[107] Allan Jepson and Michael J Black. Mixture models for optical flow computation.

[108] Dinghuang Ji, Enrique Dunn, and Jan-Michael Frahm. Spatio-temporally consistent correspondence for dense dynamic scene modeling. In *ECCV*, 2016.

[109] Hanbyul Joo, Hyun Soo Park, and Yaser Sheikh. Map visibility estimation for large-scale dynamic 3d reconstruction. In *CVPR*, 2014.

[110] Hanbyul Joo, Hao Liu, Lei Tan, Lin Gui, Bart Nabbe, Iain Matthews, Takeo Kanade, Shohei Nobuhara, and Yaser Sheikh. Panoptic studio: A massively multiview system for social motion capture. In *ICCV*, 2015.

[111] Hanbyul Joo, Tomas Simon, Xulong Li, Hao Liu, Lei Tan, Lin Gui, Sean Banerjee, Timothy Godisart, Bart Nabbe, Iain Matthews, et al. Panoptic studio: A massively multiview system for social interaction capture. *TPAMI*, 2017.

[112] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total capture: A 3d deformation model for tracking faces, hands, and bodies. In *CVPR*, 2018.

[113] Hanbyul Joo, Tomas Simon, Mina Cikara, and Yaser Sheikh. Towards social artificial intelligence: Nonverbal social signal prediction in a triadic interaction. In *CVPR*, 2019.

[114] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018.

[115] Abhishek Kar, Shubham Tulsiani, Joao Carreira, and Jitendra Malik. Category-specific object reconstruction from a single image. In *CVPR*, 2015.

[116] Nozomu Kasuya, Ryusuke Sagawa, Ryo Furukawa, and Hiroshi Kawasaki. One-shot entire shape scanning by utilizing multiple projector-camera constraints of grid patterns. In *ICCVW*, 2013.

[117] Yasutomo Kawanishi, Yang Wu, Masayuki Mukunoki, and Michihiko Minoh. Shinpuhkan2014: A multi-camera pedestrian dataset for tracking people across multiple cameras. In *20th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, 2014.

[118] Hiroshi Kawasaki, Ryo Furukawa, Ryusuke Sagawa, and Yasushi Yagi. Dynamic scene shape reconstruction using a single structured light pattern. In *CVPR*, 2008.

[119] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In *ISMAR*.

[120] Felix Klose, Oliver Wang, Jean-Charles Bazin, Marcus Magnor, and Alexander Sorkine-Hornung. Sampling based scene-space video processing. *TOG*, 2015.

[121] S Konig and S Gumhold. Image-based motion compensation for structured light scanning of dynamic surfaces. *Int J. of Intell. Sys. Tech. and App*, 2008.

[122] Thomas P Koninckx and Luc Van Gool. Real-time range acquisition by adaptive structured light. *TPAMI*, 2006.

[123] Thomas P Koninckx, Andreas Griesser, and Luc Van Gool. Real-time range scanning of deformable surfaces by adaptively coded structured light. In *3DIM*, 2003.

[124] Johannes Kopf, Michael F Cohen, and Richard Szeliski. First-person hyper-lapse videos. *TOG*, 2014.

[125] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[126] Douglas Lanman, Daniel Crispell, and Gabriel Taubin. Surround structured lighting: 3-d scanning with orthographic illumination. *CVIU*, 2009.

[127] Richard Latimer, Jason Holloway, Ashok Veeraraghavan, and Ashutosh Sabharwal. Socialsync: Sub-frame synchronization in a smartphone camera network. In *ECCV*. Springer, 2014.

[128] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 2015.

[129] Bo Li, Tianlei Zhang, and Tian Xia. Vehicle detection from 3d lidar using fully convolutional network. In *Robotics: Science and Systems*, 2016.

[130] Chi Li, M Zeeshan Zia, Quoc-Huy Tran, Xiang Yu, Gregory D Hager, and Manmohan Chandraker. Deep supervision with shape concepts for occlusion-aware 3d object parsing. *arXiv preprint arXiv:1612.02699*, 2016.

[131] Dangwei Li, Xiaotang Chen, Zhang Zhang, and Kaiqi Huang. Learning deep context-aware features over body and latent parts for person re-identification. In *CVPR*, 2017.

[132] Wei Li and Xiaogang Wang. Locally aligned feature transforms across views. In *CVPR*, 2013.

[133] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014.

[134] Xueting Li, Sifei Liu, Kihwan Kim, Xiaolong Wang, Ming-Hsuan Yang, and Jan Kautz. Putting humans in a scene: Learning affordance in 3d indoor environments. In *CVPR*, 2019.

[135] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, 2015.

[136] Martijn C Liem and Dariu M Gavrila. Joint multi-person detection and tracking from overlapping cameras. *CVIU*, 2014.

[137] Yen-Liang Lin, Vlad I Morariu, Winston Hsu, and Larry S Davis. Jointly optimizing 3d

model fitting and fine-grained classification. In *ECCV*, 2014.

[138] Christian Lipski, Christian Linz, Kai Berger, Anita Sellent, and Marcus Magnor. Virtual video camera: Image-based viewpoint navigation through space and time. In *Computer Graphics Forum*, 2010.

[139] Yebin Liu, Juergen Gall, Carsten Stoll, Qionghai Dai, Hans-Peter Seidel, and Christian Theobalt. Markerless motion capture of multiple characters using multiview image segmentation. *TPAMI*, 2013.

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

[141] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *TOG*, 2015.

[142] Movania M. Opencloth. https://code.google.com/p/opencloth.

[143] Lianyang Ma, Xiaokang Yang, and Dacheng Tao. Person re-identification over camera networks using multi-task distance metric learning. *TIP*, 2014.

[144] Yi Ma, Stefano Soatto, Jana Koseck, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. Springer Publishing Company, Incorporated, 2010. ISBN 1441918469, 9781441918468.

[145] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *JMLR*, 2008.

[146] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In *CVPR*, 2017.

[147] Andrew Maimone and Henry Fuchs. Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In *10th IEEE International Symposium on Mixed and Augmented Reality*, 2011.

[148] Niki Martinel and Christian Micheloni. Re-identify people in wide area camera network. In *CVPRW*, 2012.

[149] Tetsu Matsukawa, Takahiro Okabe, Einoshin Suzuki, and Yoichi Sato. Hierarchical gaussian descriptor for person re-identification. In *CVPR*, 2016.

[150] Chris McGlone, Edward Mikhail, and Jim Bethel. Manual of photogrammetry. 1980.

[151] Niall McLaughlin, Jesus Martinez del Rincon, and Paul Miller. Recurrent convolutional network for video-based person re-identification. In *CVPR*, 2016.

[152] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *CVPR*, 2015.

[153] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis*, 2015.

[154] Moustafa Meshry, Dan B Goldman, Sameh Khamis, Hugues Hoppe, Rohit Pandey, Noah Snavely, and Ricardo Martin-Brualla. Neural rerendering in the wild. In *CVPR*, 2019.

[155] Anton Milan, Stefan Roth, and Konrad Schindler. Continuous energy minimization for multitarget tracking. *TPAMI*, 2014.

[156] Anton Milan, Laura Leal-Taixé, Konrad Schindler, and Ian Reid. Joint tracking and segmentation of multiple targets. In *CVPR*, 2015.

[157] Francesc Moreno-Noguer. 3d human pose estimation from a single image via distance matrix regression. In *CVPR*, 2017.

[158] Roozbeh Mottaghi, Yu Xiang, and Silvio Savarese. A coarse-to-fine model for 3d pose estimation and sub-category recognition. In *CVPR*, 2015.

[159] Arsalan Mousavian, Dragomir Anguelov, John Flynn, and Jana Kosecka. 3d bounding box estimation using deep learning and geometry. In *CVPR*, 2017.

[160] Raul Mur-Artal and Juan D Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *TR*, 2017.

[161] Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *T-RO*, 2015.

[162] Richard A Newcombe, Steven J Lovegrove, and Andrew J Davison. Dtam: Dense tracking and mapping in real-time. In *ICCV*, 2011.

[163] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016.

[164] Hieu Tat Nguyen and Arnold WM Smeulders. Fast occluded object tracking by a robust appearance filter. *TPAMI*, 2004.

[165] Matthew O'Toole, Supreeth Achar, Srinivasa G. Narasimhan, and Kiriakos N. Kutulakos. Homogeneous codes for energy-efficient illumination and imaging. *TOG*, 2015.

[166] Flávio LC Pádua, Rodrigo L Carceroni, Geraldo AMR Santos, and Kiriakos N Kutulakos. Linear sequence-to-sequence alignment. *PAMI*, 2010.

[167] Sakrapee Paisitkriangkrai, Chunhua Shen, and Anton van den Hengel. Learning to rank in person re-identification with metric ensembles. In *CVPR*, 2015.

[168] Hyun Soo Park, Takaaki Shiratori, Iain Matthews, and Yaser Sheikh. 3d trajectory reconstruction under perspective projection. *IJCV*, 2015.

[169] Alonso Patron-Perez, Steven Lovegrove, and Gabe Sibley. A spline-based trajectory representation for sensor fusion and rolling shutter cameras. *IJCV*, 2015.

[170] Georgios Pavlakos, Luyang Zhu, Xiaowei Zhou, and Kostas Daniilidis. Learning to estimate 3d human pose and shape from a single color image. In *CVPR, year=2018*.

[171] Georgios Pavlakos, Xiaowei Zhou, Aaron Chan, Konstantinos G Derpanis, and Kostas Daniilidis. 6-dof object pose from semantic keypoints. In *ICRA*, 2017.

[172] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *CVPR*, 2017.

[173] Francesco Pittaluga, Sanjeev J Koppal, Sing Bing Kang, and Sudipta N Sinha. Revealing scenes by inverting structure from motion reconstructions. In *CVPR*, 2019.

[174] K Raguse and C Heipke. Photogrammetric synchronization of image sequences. In *Proc. of the ISPRS Commission V Symp. on Image Eng. and Vision Metrology*, 2006.

[175] Tanmay Randhavane, Aniket Bera, Kyra Kapsaskis, Uttaran Bhattacharya, Kurt Gray, and Dinesh Manocha. Identifying emotions from walking using affective and deep features. *arXiv preprint arXiv:1906.11884*, 2019.

[176] René Ranftl and Vladlen Koltun. Deep fundamental matrix estimation. In *ECCV*, 2018.

[177] Helge Rhodin, Nadia Robertini, Christian Richardt, Hans-Peter Seidel, and Christian Theobalt. A versatile scene model with differentiable visibility applied to generative pose estimation. In *ICCV*, 2015.

[178] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCVW*, 2016.

[179] Nadia Robertini, Dan Casas, Helge Rhodin, Hans-Peter Seidel, and Christian Theobalt. Model-based outdoor performance capture. In *3DV*, 2016.

[180] Lawrence G Roberts. *Machine perception of three-dimensional solids*. PhD thesis, Massachusetts Institute of Technology, 1965.

[181] Artem Rozantsev, Sudipta N Sinha, Debadeepta Dey, and Pascal Fua. Flight dynamics-based recovery of a uav trajectory using ground cameras. In *CVPR*, 2017.

[182] Amir Sadeghian, Vineet Kosaraju, Ali Sadeghian, Noriaki Hirose, Hamid Rezatofighi, and Silvio Savarese. Sophie: An attentive gan for predicting paths compliant to social and physical constraints. In *CVPR*, 2019.

[183] R. Sagawa, Y. Shiba, T. Hirukawa, Kawasaki H. Ono, S., and R. Furukawa. Automatic feature extraction using cnn for robust active one-shot scanning. In *ICPR*, 2016.

[184] Ryusuke Sagawa, Ryo Furukawa, and Hiroshi Kawasaki. Dense 3d reconstruction from high frame-rate video using a static grid pattern. *TPAMI*, 2014.

[185] Joaquim Salvi, Sergio Fernandez, Tomislav Pribanic, and Xavier Llado. A state of the art in structured light patterns for surface profilometry. *Pattern recognition*, 2010.

[186] Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. In *CVPR*, 2003.

[187] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016.

[188] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016.

[189] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[190] William Robson Schwartz and Larry S Davis. Learning discriminative appearance-based models using partial least squares. In *XXII Brazilian Symposium on Computer Graphics and Image Processing*, 2009.

[191] Sohil Atul Shah and Vladlen Koltun. Robust continuous clustering. *Proceedings of the National Academy of Sciences*, 2017.

[192] Horesh Ben Shitrit, Jerome Berclaz, Francois Fleuret, and Pascal Fua. Tracking multiple people under global appearance constraints. In *CVPR*, 2011.

[193] Horesh Ben Shitrit, Jérôme Berclaz, François Fleuret, and Pascal Fua. Multi-commodity network flow for tracking multiple people. *TPAMI*, 2014.

[194] Tomas Simon, Jack Valmadre, Iain Matthews, and Yaser Sheikh. Separable spatiotemporal priors for convex reconstruction of time-varying 3d point clouds. In *ECCV*. 2014.

[195] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. 2017.

[196] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3D. In *SIGGRAPH*. ACM, 2006.

[197] Noah Snavely, Steven M Seitz, and Richard Szeliski. Skeletal graphs for efficient structure from motion. In *CVPR*, 2008.

[198] Pratul P. Srinivasan, Richard Tucker, Jonathan T. Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely. Pushing the boundaries of view extrapolation with multiplane images.

[199] Carsten Stoll, Nils Hasler, Juergen Gall, Hans-Peter Seidel, and Christian Theobalt. Fast articulated motion tracking using a sums of gaussians body model. In *ICCV*, 2011.

[200] Gilbert Strang. The discrete cosine transform. *SIAM review*, 1999.

[201] Yi Sun, Yuheng Chen, Xiaogang Wang, and Xiaoou Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014.

[202] Yi Sun, Xiaogang Wang, and Xiaoou Tang. Deeply learned face representations are sparse, selective, and robust. In *CVPR*, 2015.

[203] Kalyan Sunkavalli, Neel Joshi, Sing Bing Kang, Michael F Cohen, and Hanspeter Pfister. Video snapshots: Creating high-quality images from video clips. *TVCG*, 2012.

[204] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[205] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multi people tracking with lifted multicut and person re-identification. In *CVPR*, 2017.

[206] M. F Tappen, W. T Freeman, and E. H Adelson. Recovering intrinsic images from a single image. *TPAMI*, 2005.

[207] Antonio Tejero-de Pablos, Yuta Nakashima, Tomokazu Sato, and Naokazu Yokoya. Human action recognition-based video summarization for rgb-d personal sports video. In *ICME*, 2016.

[208] Y Tian and S.G Narasimhan. Globally optimal estimation of nonrigid image distortion. *IJCV*, 2012.

[209] Carlo Tomasi and Takeo Kanade. Detection and tracking of point features. 1991.

[210] Philip A Tresadern and Ian D Reid. Video synchronization from human motion using rank constraints. *CVIU*, 2009.

[211] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment A modern synthesis. In *Vision algorithms: theory and practice*. 2000.

[212] Shubham Tulsiani and Jitendra Malik. Viewpoints and keypoints. In *CVPR*, 2015.

[213] Hsiao-Yu Tung, Hsiao-Wei Tung, Ersin Yumer, and Katerina Fragkiadaki. Self-supervised learning of motion capture. In *NIPS*, 2017.

[214] Ali Osman Ulusoy, Fatih Calakli, and Gabriel Taubin. One-shot scanning using de bruijn spaced grids. In *ICCVW*, 2009.

[215] J Valmadre and Simon Lucey. General trajectory prior for non-rigid reconstruction. In *CVPR*, 2012.

[216] Rahul Rama Varior, Mrinal Haloi, and Gang Wang. Gated siamese convolutional neural network architecture for human re-identification. In *ECCV*, 2016.

[217] Minh Vo, Srinivasa G. Narasimhan, and Yaser Sheikh. Spatiotemporal bundle adjustment for dynamic 3d reconstruction. In *CVPR*.

[218] Minh Vo, Zhaoyang Wang, Bing Pan, and Tongyan Pan. Hyper-accurate flexible calibration technique for fringe-projection-based three-dimensional imaging. *Optics Express*, 2012.

[219] Minh Vo, Srinivasa G Narasimhan, and Yaser Sheikh. Spatiotemporal bundle adjustment for dynamic 3d reconstruction. In *CVPR*, 2016.

[220] John YA Wang and Edward H Adelson. Representing moving images with layers. *TIP*, 1994.

[221] Oliver Wang, Christopher Schroers, Henning Zimmer, Markus Gross, and Alexander Sorkine-Hornung. Videosnapping: interactive synchronization of multiple videos. *SIGGRAPH*, 2014.

[222] Simi Wang, Michal Lewandowski, James Annesley, and James Orwell. Re-identification of pedestrians with variable occlusion and scale. In *ICCVW*, 2011.

[223] Xinchao Wang, Engin Türetken, Francois Fleuret, and Pascal Fua. Tracking interacting objects using intertwined flows. *TPAMI*, 2016.

[224] Daniel Wedge, Du Huynh, and Peter Kovesi. Motion guided video sequence synchronization. In *ACCV*. 2006.

[225] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *CVPR*, 2016.

[226] Thibaut Weise, Bastian Leibe, and Luc Van Gool. Fast 3d scanning with automatic motion compensation. In *CVPR*, 2007.

[227] Yair Weiss and Edward H Adelson. A unified mixture framework for motion segmentation: Incorporating spatial coherence and estimating the number of models. In *CVPR*, 1996.

[228] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016.

[229] Changchang Wu. Visualsfm: A visual structure from motion system. `http://http://ccwu.me/vsfm`, 2011.

[230] Lin Wu, Chunhua Shen, and Anton van den Hengel. Deep linear discriminant analysis on

fisher networks: A hybrid architecture for person re-identification. *Pattern Recognition*, 2017.

[231] Shangxuan Wu, Ying-Cong Chen, Xiang Li, An-Cong Wu, Jin-Jie You, and Wei-Shi Zheng. An enhanced deep feature representation for person re-identification. In *WACV*, 2016.

[232] Zheng Wu, Ashwin Thangali, Stan Sclaroff, and Margrit Betke. Coupling detection and data association for multiple object tracking. In *CVPR*, 2012.

[233] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, 2015.

[234] Yu Xiang, Wongun Choi, Yuanqing Lin, and Silvio Savarese. Data-driven 3d voxel patterns for object category recognition. In *CVPR*, 2015.

[235] Tong Xiao, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *CVPR*, 2016.

[236] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search. In *CVPR*, 2017.

[237] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[238] Shuntaro Yamazaki, Akira Nukada, and Masaaki Mochimaru. Hamming color code for dense and robust one-shot 3d scanning. In *BMVC*, 2011.

[239] Jingyu Yan and Marc Pollefeys. Video synchronization via space-time interest point distribution. In *ACIVS*, 2004.

[240] Yi Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *CVPR*, 2011.

[241] Zhe Yang, Zhiwei Xiong, Yueyi Zhang, Jiao Wang, and Feng Wu. Depth acquisition from density modulated binary patterns. In *CVPR*, 2013.

[242] Ting Yao, Tao Mei, and Yong Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *CVPR*, 2016.

[243] G Ye, G Garces, Y Liu, Q Dai, and D Gutierrez. Intrinsic video and applications. *TOG*, 2014.

[244] Jifeng Dai Xiangyang Ji Yi Li, Haozhi Qi and Yichen Weil. Fully convolutional instance-aware semantic segmentation. In *CVPR*, 2017.

[245] L Yu and Michael Brown. Single image layer separation using relative smoothness.

[246] Shoou-I Yu, Deyu Meng, Wangmeng Zuo, and Alexander Hauptmann. The solution path algorithm for identity-aware multi-object tracking. In *CVPR*, 2016.

[247] Andrei Zanfir, Elisabeta Marinoiu, and Cristian Sminchisescu. Monocular 3d pose and shape estimation of multiple people in natural scenes-the importance of multiple scene constraints. In *CVPR*, 2018.

[248] Bernhard Zeisl, Pierre Fite Georgel, Florian Schweiger, Eckehard G Steinbach, and Nassir

Navab. Estimation of location uncertainty for scale invariant features points. In *BMVC*, 2009.

[249] Guofeng Zhang, Zilong Dong, Jiaya Jia, Liang Wan, Tien-Tsin Wong, and Hujun Bao. Refilming with depth-inferred videos. *TVCG*, 2009.

[250] Li Zhang, Brian Curless, and Steven M Seitz. Rapid shape acquisition using color structured light and multi-pass dynamic programming. In *3DV*, 2002.

[251] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008.

[252] Lu Zhang and Laurens van der Maaten. Structure preserving object tracking. In *CVPR*, 2013.

[253] Haiyu Zhao, Maoqing Tian, Shuyang Sun, Jing Shao, Junjie Yan, Shuai Yi, Xiaogang Wang, and Xiaoou Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017.

[254] Rui Zhao, Wanli Ouyang, and Xiaogang Wang. Unsupervised salience learning for person re-identification. In *CVPR*, 2013.

[255] Enliang Zheng, Dinghuang Ji, Enrique Dunn, and Jan-Michael Frahm. Self-expressive dictionary learning for dynamic 3d reconstruction. *IEEE TPAMI*, 2017.

[256] Liang Zheng, Zhi Bie, Yifan Sun, Jingdong Wang, Chi Su, Shengjin Wang, and Qi Tian. Mars: A video benchmark for large-scale person re-identification. In *ECCV*, 2016.

[257] Wei-Shi Zheng, Xiang Li, Tao Xiang, Shengcai Liao, Jianhuang Lai, and Shaogang Gong. Partial person re-identification. In *ICCV*, 2015.

[258] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *ECCV*, 2016.

[259] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, 2017.

[260] Xiaowei Zhou, Menglong Zhu, and Kostas Daniilidis. Multi-image matching via fast alternating minimization. In *ICCV*, 2015.

[261] Yingying Zhu and S. Lucey. Convolutional sparse coding for trajectory reconstruction. *PAMI*, 2015.

[262] M Zeeshan Zia, Michael Stark, Bernt Schiele, and Konrad Schindler. Detailed 3d representations for object recognition and modeling. *TPAMI*, 2013.

[263] M Zeeshan Zia, Michael Stark, and Konrad Schindler. Towards scene understanding with detailed 3d object representations. *IJCV*, 2015.