Martial Arts, Dancing and Sports dataset: a Challenging Stereo and Multi-View Dataset for 3D Human Pose Estimation

Weichen Zhang*, Zhiguang Liu, Liuyang Zhou, Howard Leung, Antoni B. Chan

Department of Computer Science, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong SAR

Abstract

Human pose estimation is one of the most popular research topics in the past two decades, especially with the introduction of human pose datasets for benchmark evaluation. These datasets usually capture simple daily life actions. Here, we introduce a new dataset, the Martial Arts, Dancing and Sports (MADS), which consists of challenging martial arts actions (Tai-chi and Karate), dancing actions (hip-hop and jazz), and sports actions (basketball, volleyball, football, rugby, tennis and badminton). Two martial art masters, two dancers and an athlete performed these actions while being recorded with either multiple cameras or a stereo depth camera. In the multi-view or single-view setting, we provide three color views for 2D image-based human pose estimation algorithms. For depth-based human pose estimation, we provide stereo-based depth images from a single view. All videos have corresponding synchronized and calibrated ground-truth poses, which were captured using a Motion Capture system. We provide initial baseline results on our dataset using a variety of tracking frameworks, including a generative tracker based on the annealing particle filter and robust likelihood function, a discriminative tracker using twin Gaussian processes [1], and hybrid trackers, such as Personalized Depth Tracker [2]. The results of our evaluation suggest that discriminative approaches perform better than generative approaches when there are enough representative training samples, and that the generative methods are more robust to diversity of poses, but can fail to track when the motion is too quick for the effective search range of the particle filter. The data and the accompanying code will be made available to the research community.

Keywords: Human pose estimation, robust tracking, evaluation, martial arts, dancing and sports

1. Introduction

3D human pose estimation and tracking has been an active topic of research for over the past 20 years. The ability to recover the 3D articulated human pose from an image, or human motion from a video, has broad applications to human-computer interaction, surveillance, entertainment, and video understanding. Recently, with the development of more accurate depth sensors, e.g. Kinect, ToF camera and stereo cameras, recovering the human pose from depth features has also attracted much attention. Human pose estimation is challenging and suffers from three confounding problems: 1) the pose space is high-dimensional and hard to optimize over; 2) self occlusions make it difficult to localize invisible body parts; 3) the search space, even when initialized from the previous frame's result, is large due to the complexity of human motion. Existing methods for human pose estimation can be classified into three methodologies: generative approaches, discriminative approaches, and hybrid approaches. Generative methods track by matching the observation (images or depth map) to a 3d body model. Such methods do not require training data, and hence are suitable to track any kind of pose. However, since generative methods use the tracking result from the previous frame to reduce the search space of the pose in the current frame, they may have difficulties when the motion is large between frames. Discriminative methods directly learn a mapping from observations to the pose parameters. Since the pose is obtained directly from the observations, these methods can deal with quick motions well, and they are robust to the observation noise when both the training and testing data have the same noise distribution. However, discriminative methods require a large amount of training data, and they obtain poor results when the testing data contains different poses from the training data. To combine the advantages of both generative and discriminative approaches, uses a discriminative approach to predict pose candidates from the previous history or current observation, and then uses a generative approaches to refine the candidates to best fit the observation with the body model. Hybrid methods thus can achieve the best performance, working for quick motions and not severely losing track even when the current pose is not in the training data. However, all of these methods will fail when the observations are noisy and suitable features cannot be extracted.

Prior to the release of the HumanEva dataset [3], there was no widely accepted dataset for evaluating 3D human pose estimation methods, since obtaining the ground truth poses is difficult and requires specialized motion capture equipment. Although it only contains 5 simple actions with ground truth poses, the introduction of the HumanEva dataset [3] provided an im-

*Corresponding author
Phone: +852 5425 3773
Email address: wczhang4-c@my.cityu.edu.hk (Weichen Zhang)

 https://doi.org/10.1016/j.imavis.2017.02.002

portant benchmark dataset for the community to evaluate the various methodologies. At the time, they did not record depth data since the depth sensor was not widely used then. In subsequent years, [4] introduced the Multimodal Human Action Database (MHAD) dataset with more subjects, more actions and more data types. However, each sequence of MHAD is too short to evaluate the robustness of tracking algorithms. Data-driven methods such as random forests and deep neural networks require large amounts of training data. As a result, [5] introduced the Human3.6M dataset, which contains 3.6 million images and corresponding 3D human pose.

Datasets for depth-based human pose estimation include the SMMC dataset [6], the EVAL dataset [7] and the PDT dataset [2]. The SMMC dataset [6] was captured using a time-of-flight (ToF) sensor, and the recorded actions are simple. Both the EVAL dataset [7] and PDT dataset were recorded with the Kinect sensor [8]. Correspondingly, most existing methods for depth-based human pose estimation use depth images from ToF cameras [6, 9] or the Kinect sensor [10, 11, 7, 2]. However, both Kinect and ToF are active sensors, and hence are limited to studio environments without infrared pollution and only work within a limited range, with the subjects constrained to a small area. In contrast, stereo-based depth maps have less limitations of working environment (e.g., they can work outdoors), but the depth maps are more noisy.[1]

In this paper, to further the research of 3D human pose estimation using image-based sensors and depth-based sensors, we introduce the Martial Arts, Dancing and Sports (MADS), consisting of both multi-view RGB videos and depth videos. Our dataset consists of Tai-chi, Karate, Hip-hop dance, Jazz dance and sports combo (basketball, volleyball, football, rugby, tennis and badminton) actions that are complex and challenging. These actions are unique, and almost all of them have not appeared in any other existing datasets. The multi-view color videos are captured with three color cameras, and the depth videos are captured with a stereo camera. The dataset also contains the calibrated and synchronized ground-truth poses, which were obtained using a motion capture system.

The contributions of our paper are summarized as follows. Firstly, the poses we captured are very different than those in existing datasets – they contains more self-occlusions, articulated motion of the limbs, motions with different speed and more body spinning. Because of these factors, our dataset provides a significant challenge to existing algorithms, and can help further the development of robust pose tracking algorithms. Secondly, we are the first to collect these challenging actions on both multi-view cameras and a single stereo-based depth camera. By introducing a new dataset using stereo-based depth, we hope to further the research on human pose tracking from noisy stereo-based depth maps, which will help to improve the overall robustness of depth-based pose estimation algorithms with potential outdoor applications. Thirdly, we conduct baseline experiments on our MADS dataset using several generative and discriminative algorithms. Our experiments show that current tracking methods still have room to improve on our challenging dataset (errors ranging from 100-200mm), as compared to other well-studied datasets (e.g., $\sim$ 50 mm for HumanEva [3]; $\sim$ 30 mm for SMMC-10 [6]).

The remainder of this paper is organized as follows. In Section 2, we present a literature review on existing human pose estimation methods and related datasets. In Section 3, we describe our Martial Arts, Dancing and Sports Dataset. In Section 4, we present the algorithms we used for baseline comparisons. Finally, in Section 5, we present experiment using the baseline algorithms on our dataset.

2. Related work

In this section we review methods for human pose estimation and human pose datasets.

2.1. Human pose estimation using multiple views

Human pose estimation has been researched by the community for over 20 years, and an overview can be found in [12, 13]. Early generative trackers include CONDENSATION [14], which proposed a Markov chain Monte Carlo (MCMC) strategy to match the body template to the contour, and covariance scaled sampling for motion prediction from the previous frame [15]. For multi-view human pose tracking, common image features include foreground silhouettes [16, 17, 18], and color cues to better localize body parts [18, 19]. The Chamfer distance transformation is widely used on both edge and silhouette descriptors[16, 17, 18, 20, 21], which makes the matching between a body model and the image features more robust. [22] proposed a 3D pose recovery strategy that used multi-view silhouettes to construct 3D visual hulls, which are then used to estimate the human skeleton. [23] used mean-shift to extract key poses as model references during tracking to avoid the influence of observation noise, such as missing body parts in the silhouettes. To extend the Gaussian diffusion model used by most generative methods, [19] trained on previous tracking results to predict future samples. Finally, [24] estimated the 3D human body shape from multi-view silhouettes. [25] proposed a robust likelihood function for tracking human poses using a Bayesian framework. The robust likelihood function combined color, silhouette and edge features with Chamfer transformation to make generative tracking more robust and accurate.

Discriminative methods learn a mapping from image descriptors to the human poses. For example, [26] used relevance vector machines to map from silhouettes to human poses, while [27] used a linear kernel dependency estimation model. [1] trained a twin Gaussian process (TGP) model to map from HOG [28] features to pose vectors, and achieved state-of-the-art performance on the HumanEva-I dataset [3]. [29] used binary vectors of "posebits", which describe the geometrical structure of the human pose, as the image observations to predict human poses. Recently, random decision forests were used to label body parts in RGB images to help estimate the human pose from single image [30]. Deep neural networks have also been applied to estimate 2D human pose [31, 32] and 3D human pose [33] from single images.

[1] Nonetheless, our dataset must be recorded in a laboratory setting, since we must capture the ground-truth pose with a MOCAP system.

Finally, hybrid approaches typically train a motion model to predict pose candidates for the generative tracker. For example, GPLVM [34], GPDM [35] and CRBM [36] used latent motion models for prediction. Besides the commonly used kinematic body model, [37] learned a graphical model using MOCAP data to model the constraints of each joint. [38] extended the 2D pictorial model to 3D human pose estimation from multi-view images by avoiding limbs intersections. Following [38], [39] proposed a multi-human pose estimation method using 3D pictorial structures. [40] measures interactions among multiple people using 480 camera views, with the people's skeletons estimated by merging human body part detection methods on each view. [37, 38, 39, 40] used a discriminative body part detector to find human body part candidates.

2.2. Human pose estimation from depth images

Human pose estimation from depth images can also be categorized into generative, discriminative, and hybrid approaches. For generative approaches, the observed depth data is matched to a human body model, typically using iterative closest points (ICP). [41] estimated the human pose by using 2D image features to segment depth points, and then applies 3D ICP. [42, 43] proposed the Articulated ICP and its variant to estimate the human pose from a range sensor, but they didn't consider challenging poses with self-occlusions and only estimate the upper body pose. [7] used an extended ICP approach to optimize the joint positions of the pose, and constrains the model from entering the space between the camera and the observed depth surface. [44] proposed a GMM for modeling the relationship between pose parameters and the mesh body model, and used the EM algorithm to optimize both the pose and shape parameters of the body model. Finally, [45] proposed a combination of particle swarm optimization (PSO) and ICP for hand pose estimation.

Discriminative methods typically train a classifier to detect body parts from depth maps, e.g., [46]. [10] trained random decision forests to segment each depth point into different limbs, and the joint position of each limb is determined using the mean-shift algorithm. [47] also trained a random forest, but estimated the correspondences between the depth image and the model independently, and the parameters of the model were optimized in one-shot. [48] learned a key-point detector and optimized the skeleton by minimizing the error between the reference key-point vector and a predicted key-point vector. [49] proposed a discriminative method for estimating 3D human pose from stereo-based depth videos. It introduced a grid-based shape descriptor and trained random forests to classify the observation points into each body part.

One type of hybrid approach is to use discriminative part detectors in combination with model-based local search, e.g., [6] used a body-part detector from [46], while [50] first detected the head and hands, and then optimizes the whole pose with ICP. Another type of hybrid approach is to build databases containing a mapping from the point cloud features (e.g. geodesic extrema) to poses [11, 9]. Given a new depth image, its point cloud is used to retrieve a candidate pose from the database, and then the pose is refined with a generative approach, such as ICP. [51] used random forests to initialize the optimization of an objective function, which is based on projected 2D plane, instead of 3D point clouds. [2] constructed a shape-invariant personalized tracker based on the database of a single actor from [9]. Finally, [52] used an inertial sensor to provide cues of the body direction when optimizing the pose within a hybrid framework.

2.3. Human pose datasets

The current datasets for human pose estimation from multi-view or depth data are summarized in Table 1. Multi-view human pose estimation is typically evaluated on the HumanEva [3], MHAD [4] and Human3.6M [5] datasets. Before the HumanEva dataset [3], there was no widely accepted dataset for 3D human pose estimation (see [3, Table 1] for a list of datasets before HumanEva). HumanEva [3] used a motion capture (MOCAP) system in conjunction with a video capture system to obtain multi-view videos with synchronized MOCAP ground truth. The HumanEva dataset [3] contains 40,000 frames spanning 4 subjects and 6 actions. The dataset also provides an easy-to-use MATLAB [53] interface with a baseline algorithm based on the annealing particle filtering (APF) [16]. While HumanEva [3] has been successful in furthering research on human pose estimation, the actions performed are fairly simple and consist of walking, jogging, gesturing, boxing, or a combination thereof. Compared to the sports actions of HumanEva (jogging and boxing), our dataset provides more challenging and complex actions, with more joints moving simultaneuosly, and more poses consisting overlapping limbs.

Following the success of HumanEva [3], the Multimodal Human Action Database (MHAD) dataset was introduced in [4] and contains several types of data: multi-view color video, Kinect depth data, stereo image data, audio data, accelerometer data, and MOCAP ground-truth data. The dataset contains 12 subjects and 11 actions, but no sequence is longer than 15 seconds ($\sim$330 frames), which makes it difficult to evaluate the robustness of a method for long-term tracking. Finally, the Human3.6M dataset was recently introduced in [5], and contains 3.6 million images and corresponding 3D human poses. The dataset contains 11 subjects and 15 actions recorded from 4 views. The dataset is of high quality with high resolution color videos, accurate ground truth, and a diversity of actions. Although depth data is captured by a ToF sensor, the depth map is polluted by the infrared light of the MOCAP system reflecting off the MOCAP body markers. In addition, no calibration data for the ToF sensor is provided.

Besides the above datasets for human pose estimation and tracking, there are several other datasets that captured similar data but for human body modeling. These datasets are mainly for synthesizing 3D body models from a large number of camera views, but lack ground-truth joint locations and hence have limited use for human pose estimation. Using 22 stereo cameras and one high-resolution RGB camera, [54] captured 300 scans from 10 subjects in 30 different poses for 3D mesh model registration. Its ground-truth is not the joint locations, but the correspondences for non-rigid point registration. [55] captured videos of 8 subjects for 12 motions and two person interactions with a convergent eight camera setup. They represented

Table 1: The comparison between recent datasets for 3D human pose estimation.

	HumanEva[3]	MHAD[4]	Human3.6M[5]	SMMC-10[6]	EVAL[7]	PDT[2]	MADS
# of subjects	4	12	11	1	3	4	5
# of actions	6	11	15	28	8	4	30
# of video frames	40,000	-	900,000	>10,000	>10,000	>20,000	>53,000
# of camera views	3 or 4	4	4	1	1	1	3 or 1
# of images	122,500	-	3,600,000	>10,000	>10,000	>20,000	>100,000
resolution	640× 480	640× 480	1,000× 1,000	144× 176	640× 480	640× 480	1024× 768
average # frames per sequence	1500	200	2500	100 or 400	500	1500	800
average # seconds per sequence	25	10	50	4 or 16	16.5	50	60 or 80
frame rate per second	60	22	50	25	30	30	15 or 10 or 20
depth sensor	No	Kinect & Stereo	ToF	ToF	Kinect	Kinect	Stereo
calibration available	Yes	Yes	Yes, No for depth	Yes	Yes	Yes	Yes
body model	Cylinder	-	-	-	-	Scanned mesh	Cylinder
motion complexity	Simple	Simple	Simple to complex	Simple	Simple to complex	Moderate to complex	Moderate to complex
ground truth data	Marker	Marker	Marker	Marker	Joint	Marker and joint	Marker

the human pose with a mesh model generation from 8 views, but there is no motion capture data as the ground truth. [56] proposed a topology dictionary to describe 3D video content, and experiments used a Yoga pose dataset consisting of multiview videos. However, no motion capture ground-truth data is provided. [57] proposed an articulated mesh modeling algorithm from synchronized silhouettes. They captured 8-view high-resolution videos, and scanned the subject to generate the mesh template. The algorithm outputs 3D skeleton poses together with animated 3D mesh models, but the dataset also lacks ground-truth data for the 3D joint locations.

For depth-based human pose estimation, there are the SMMC-10 [6], EVAL [7], and PDT [2] datasets. The SMMC-10 dataset [6] contains only one subject with 27 actions, but most of the actions are very simple. The EVAL datset [7] contains 3 subjects and more complex movements (e.g., cart-wheels, hand standing, and sitting on the floor) than SMMC-10. Each of subjects contains 8 sequences and each sequence has about 500 frames. The PDT dataset [2] contains 4 subjects with 4 actions. The sequences in PDT contain challenging motions such as sitting on the floor, spinning around, and fast kicking. In PDT, the Kinect sensor failed to capture several frames in PDT when the subject is out of the working range of Kinect.

These three current datasets are based on Kinect or ToF sensors, which are active sensors that emit infrared rays and measure properties of the reflected rays to determine the depth of each pixel. Although accurate, such active sensors typically only operate within a limited range and over a limited area, and often are influenced by external environmental factors, such as background sunlight and interference from other active sensors. Besides these two types of depth sensors, another method for obtaining depth images is through stereo cameras. Typically, a disparity map is calculated between matched features in the left and right images [58]. Because the stereo matching procedure is influenced by the textures (or lack thereof) in the image, the depth maps from stereo cameras are typically more noisy and blurred than ToF and Kinect cameras. Nonetheless, stereo cameras are more robust in infrared-noisy and range varying environments (e.g., outdoors) and are not affected by infrared interference, and hence devising tracking algorithms to robustly work with stereo images is a promising research area.

3. Martial Arts, Dancing and Sports dataset

The goal of collecting the Martial Arts, Dancing and Sports dataset (MADS) is to provide challenging action sequences for human pose estimation from multiview or depth data. The MADS dataset contains 5 challenging actions types, which are Tai-chi, Karate, Jazz, Hip-hop and sports. All actions are performed by professional players, and we provide video data of real world challenging poses, which have not been been collected by existing datasets. For depth videos, we choose the stereo camera as the modality because there are no existing datasets that capture human actions with it – although noisier, stereo cameras do not suffer from infrared interference and can work outdoors.

3.1. Motion description

Our dataset contains actions from martial arts, dancings and sports, which are common, but non-daily life motions. Tai-chi is a traditional Chinese martial art, containing smooth circular actions. Karate is a Japanese martial art that contains many striking actions, such as punching, kicking, knee strikes, elbow strikes, and open hand techniques. The selected martial arts actions are very different that those used in daily-life, and have more requirements for power, speed and balance.

The jazz dance is developed from African American vernacular dance, and consists of many body spinning actions and arm actions with a large range of motions. The hip-hop dance refers to the street dance with hip-hop music, and consists of hip movements, as well as many shoulder and torso movements. For the dance actions, there are more body spinning than other types of motions, which makes kinematic-based tracking more difficult.

The sports combo sequences includes badminton, basketball, football, tennis and volleyball actions, which are common sports played around the world. For convenience of data collection, we let the actor perform actions without balls or rackets. Actions for basketball, football, and volleyball include shooting, passing, and defense. For badminton and tennis, the recorded actions are serving and hitting. For rugby, the actions include passing the ball and defence.

The actions in MADS are more complex and challenging than normal actions. Firstly, they have a larger range of motion, while some poses will not appear in normal actions. Secondly, there are more self-occlusions and more interactions between limbs. Thirdly, some actions are very quick, compared to the

MOCAP	Manufacturer	Motion Analysis
	# of cameras	7
	Camera resolution	1M pixel
	Frame Rate	60 Hz
Video Capture	Manufacturer	Point Grey
	Camera model	Bumblebee-II
	# of cameras	3
	Camera resolution	1024 × 768
	Frame rate	10 or 20 Hz for stereo, 15 Hz for multi-view

Table 2: The hardware system for data collection.

frame rate used to capture the video (10 fps for Tai-chi and Karate and 20 fps for jazz, hip-hop and sports), resulting in motion blur.

A total of 5 actors were used for data collection, with each actor performing one action category. We asked two martial arts masters to perform pre-arranged series of moves, called "forms" in Tai-chi or "katas" in Karate, two professional dancers to perform dances for jazz and hip-hop, and an athlete to perform sports actions. The subjects signed an informed consent form that allows the distribution of the data for academic use. The subjects wore natural clothing. The multiview and stereo-depth videos were captured separately. Since the stereo-depth videos are recorded from only one viewpoint, the subjects were instructed to modify their action sequences so that they face the camera as much as possible. In particular, the subjects were asked to avoid having their back to the camera when there are body rotations in the movements. For example, if the movement involved a 180 degree body rotation, which at the end leaves the subject with the back to the camera, then we asked the subject to perform a full 360 degree rotation. In another example, if the subject has their right-side facing the camera and the movement is a rotation to the left, leaving the back to the camera, we asked the subject to rotate to the right to face the camera. Hence, the action sequences in the multiview and stereo videos are different for some actions.

3.2. Capture setup

The capture space is shown in Table 2 and Fig. 1. The data was recorded in a studio environment with some background clutter. The video data was recorded with Point Grey Bumblebee-II cameras. The multi-view data was collected with 3 cameras placed around the capture space, while the stereo images were collected from one viewpoint. The multi-view data was captured at 15 fps, and the cameras were synchronized automatically when connected to the same hub. The depth data (stereo image) was captured at 10 fps or 20 fps.[2] The baseline of the stereo camera is 12 cm. The resolution of the images are 1024 × 768. The ground-truth pose data was captured using a MOCAP system by Motion Analysis. Seven MOCAP cameras are placed on the walls around the capture space to record the positions of markers on the human body. The MOCAP system works at 60 fps.

[2]A IEEE1394 hub with higher bandwidth was used when capturing the dancing and sports videos, resulting in a higher frame rate.

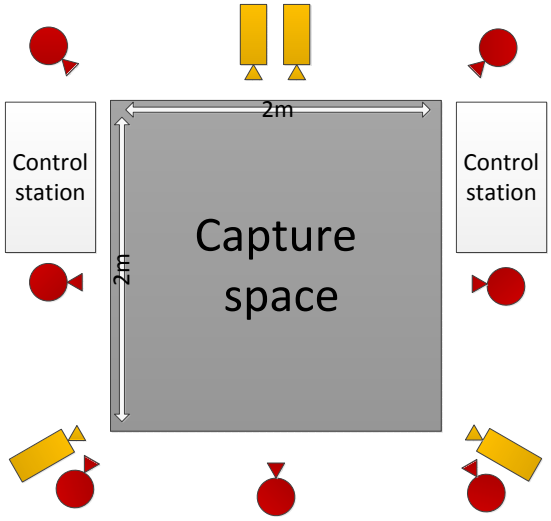


Figure 1: The layout of capture space. The red cameras are the infrared cameras of the MOCAP system, while the yellow cameras are RGB cameras used for capturing depth and multi-view videos.

3.3. Calibration and synchronization

For the calibration process of both the camera intrinsic parameters (transforming camera coordinates into image coordinates) and the extrinsic parameters (transforming MOCAP coordinates into the camera coordinates), we use the same strategy as HumanEva [3]. The intrinsic parameters (focal length, principle point, and distortion) for each camera c of the video capture system was estimated using a standard chessboard with the Calibration Toolbox for MATLAB [59]. To transform camera coordinates into MOCAP coordinates, the extrinsic parameters, which contain rotation R_c and translation T_c, are required. We collected a large number of 3D locations $\{\Gamma_i^M\}_{i=1}^N$ ($N > 300$) for one MOCAP marker. Next, in each camera, the corresponding points Γ_i^c (Γ^c is 2D for multi-view videos and 3D for depth videos) were marked manually.

Our setup does not have hardware synchronization equipment. We assume that the frame rate for both the video capture and motion capture systems are fixed during data collection, and then down-sample the rate of the motion capture data to match the video data after data collection. We then estimated the temporal offset α between the video capture and motion capture systems during calibration. Finally, the rotation and translation parameters and the temporal offset were estimated by minimizing

$$\min_{R_c,T_c,\alpha} \sum_{i=1}^{N} \|\Gamma_i^c - f(\Gamma_{i+\alpha}^M; R_c, T_c)\|^2, \quad (1)$$

where $f(\cdot)$ is the function for rotation and translation of 3D points. The motion capture system was calibrated with protocol from Motion Analysis.

3.4. Dataset

The MADS dataset contains 5 actions categories, totalling about 53,000 frames. Each action category consists of 6 se-

Table 3: Action sequences in MADS.

Action category	No.	Action Sequence	No. of multi-view frames	No. of stereo frames
Tai-chi	1.	"Commencing Form", "Buddha's Warrior Attendant Pounds Mortar (I)" ("Qishi","Jin Gang Dao Dui (1)")	800	550
	2.	"Single Whip", "Lazy about Tying Coat" ("Dan bian Xia Shi","Lan Zha Yi")	800	540
	3.	"White Crane Spreads its Wings", "Buddha's Warrior Attendant Pounds Mortar (II)" ("Bai He Liang Chi","Jin Gang Dao Dui (2)")	800	400
	4.	"Walk Obliquely", "Twist Step (I)" ("Xie Xing Ao Bu (1)")	800	540
	5.	"Walk Obliquely", "Twist Step (II)" ("Xie Xing Ao Bu (2)")	800	550
	6.	"Fist of Covering Hand and Arm", "Crossing Hand" ("Yan Shou Gong Quan","Shi Zi Shou")	800	500
		Total	**4000**	**2680**
Karate	1.	"Second Basic Form" ("Fukyugata Ni")	600	1320
	2.	"Third Basic Form" ("Fukyugata Sandan")	600	1400
	3.	"Third Peace Form" ("Pinan Sandan")	600	1450
	4.	"Fifth Peace Form" ("Pinan Godan")	600	1400
	5.	"Straddle Stance" ("Naihanchi Nidan")	600	1400
	6.	"Horse Riding Stance" ("Naihanchi Nidan Sandan")	600	1400
		Total	**3000**	**8370**
Jazz	1.	"Jazz action1"	753	1000
	2.	"Jazz action2"	776	981
	3.	"Jazz action3"	787	1000
	4.	"Jazz action4"	838	979
	5.	"Jazz action5"	738	961
	6.	"Jazz action6"	804	1000
		Total	**4696**	**5921**
Hip-hop	1.	"Hip-hop action1"	818	1330
	2.	"Hip-hop action2"	928	1000
	3.	"Hip-hop action3"	831	920
	4.	"Hip-hop action4"	1064	1000
	5.	"Hip-hop action5"	897	1000
	6.	"Hip-hop action6"	897	1000
		Total	**5435**	**6250**
Sports	1.	"Badminton"	779	970
	2.	"Basketball"	982	1000
	3.	"Football"	761	1000
	4.	"Rugby"	844	958
	5.	"Tennis"	704	957
	6.	"Volleyball"	990	970
		Total	**5060**	**5855**

quences. Details on the number of frames in each action category and for each sequence are shown in Table 3. All types of actions contain many self-occlusions. Fig. 2 shows a few sample images and ground-truth poses. For training and testing of discriminative models, we suggest a leave-one-out protocol, where one action sequence is held out for testing, and the remaining five actions of the same action category are used for training. Note that the poses in the test sequence may not always be present in the training sequences, as they are from different parts of the action sequence (e.g., Karate kata). This provides an important test for discriminative models on how well such methods can extrapolate to unseen poses.

3.4.1. Multi-view data

Each multi-view frame consists of three RGB images around the subject. To obtain the person's silhouette, we used a Gaussian mixture model (GMM) [60] and shadow detection [61] to remove the background. We also extracted color cues in the YUV domain inside the silhouette. Fig. 3 shows an example of a multi-view frame and its extracted silhouette.

3.4.2. Stereo-depth data

The stereo camera provides 2 parallel RGB views of the subject (see Fig. 4a). A depth image is calculated using the following procedure. First, the stereo images were rectified to remove distortions (Fig. 4b). Next, the disparity map (Fig. 4c) is obtained using the stereo matching algorithm from the 3D-Vision Toolbox in MATLAB [53]. These depth maps had a z-axis resolution of 64. To obtain the observed point cloud, the background is first removed using the background subtraction and shadow detection of [61] (Fig. 4d). Finally, the observed point cloud is reconstructed from the disparity map and the camera calibration parameters (Fig. 4e).

Note that the depth maps estimated from stereo images are noisier than those from ToF or Kinect sensors. In particular, using stereo images, the depth values are blurred at the body contour, and the depth values inside the body may not be accurate since there might be insufficient features to find stereo correspondences. In addition, the depth values for the feet are merged with those of the floor. Although stereo cameras produce noiser depth videos, they also work in less constrained environments than ToF/Kinect sensors. By providing noisier

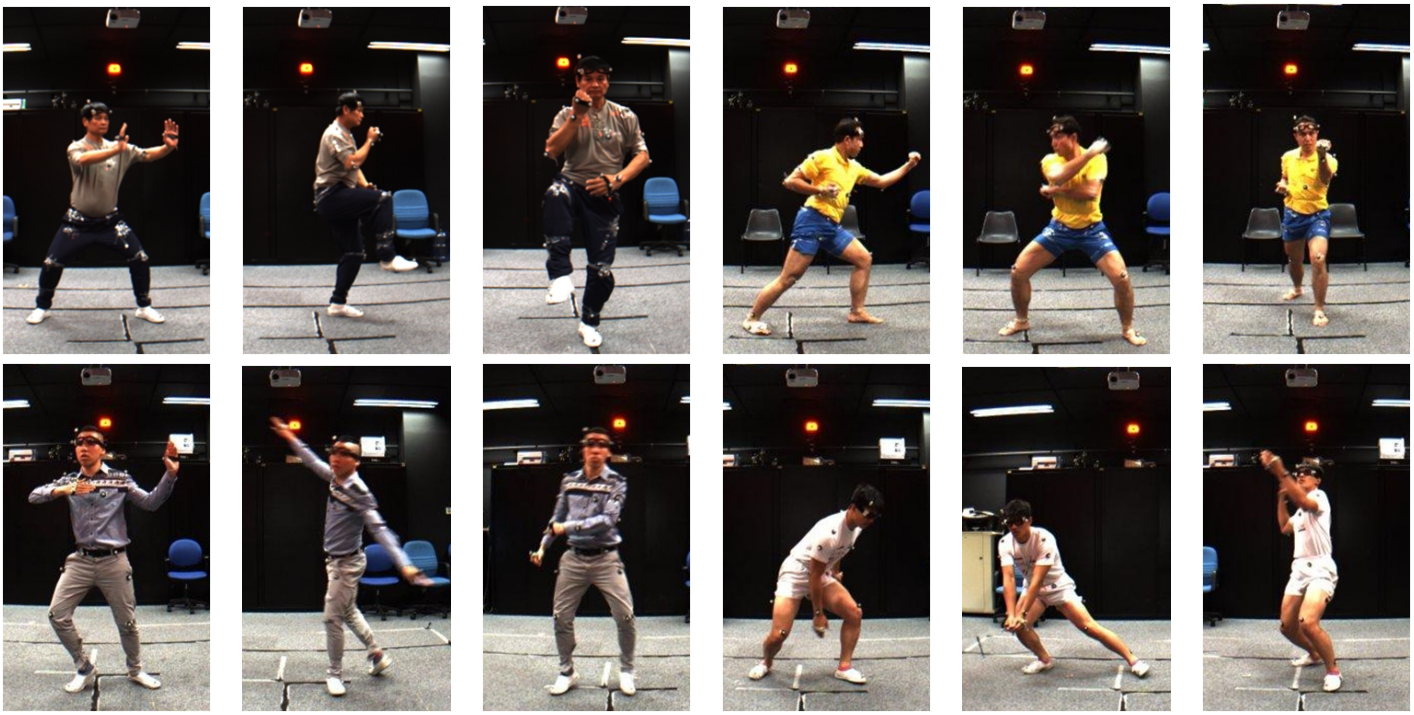


Figure 2: Sample poses in MADS: (top-left) Tai-chi; (top-right) Karate; (bottom-left) Hip-Hop dance; (bottom-right) Sports

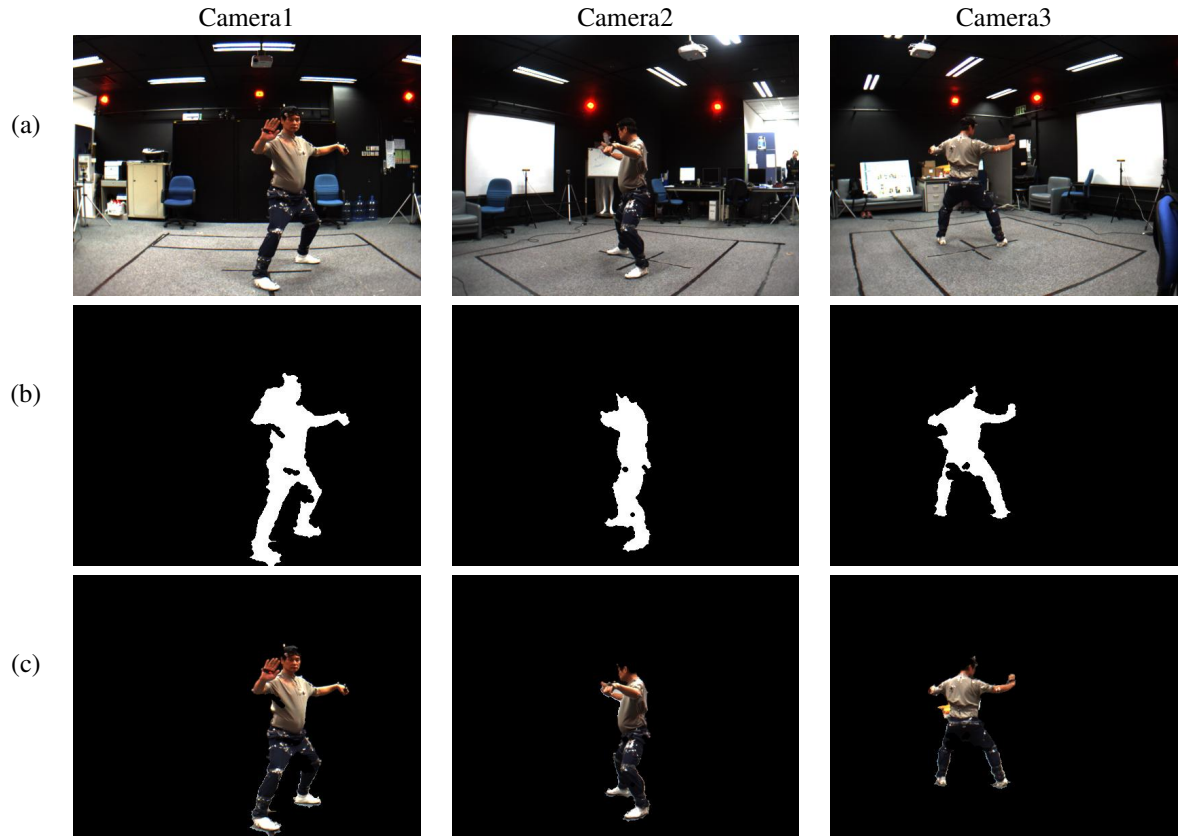


Figure 3: Multi-view frame processing: (a) input images from multiple cameras; (b) the silhouettes for each view; (c) the color cues of silhouettes.

depth data, we hope to provide an opportunity for the community to develop robust algorithms for human pose estimation that are not limited to only "clean" ToF/Kinect depth data.

3.4.3. MOCAP ground-truth data

The MOCAP system captures the positions of 35 markers on the body surface. We selected 19 out of 35 markers as the joints of the human body, similar to HumanEva [3]. The selected joints are torso, pelvis, left hip, left knee, left ankle, left toe, right hip, right knee, right ankle, right toe, left shoulder, left elbow, left wrist, left hand, right shoulder, right elbow, right wrist, right hand, and head (top-right forehead). The MOCAP data was cleaned manually when markers could not be recovered automatically by the system. In cases where the position of a marker could not be corrected in a frame, we set a flag in the ground-truth data and that frame is not used in the evaluation process.

For the evaluation, we use the mean per joint position error (MPJPE) [5] as the evaluation metric, which is also used for HumanEva [3]. To reduce the influence of ground-truth bias introduced from the MOCAP system, we use the displacement strategy of the PDT dataset [2]. An average local offset for each joint to its corresponding ground-truth is calculated, $\bar{j}_i = \frac{1}{F}\sum_{t=1}^{F}(\hat{j}_i^{(t)} - j_i^{(t)})$, where $\hat{j}_i^{(t)}$ and $j_i^{(t)}$ are the predicted and ground-truth positions of joint i in frame t, and F is the number of frames. The corrected joints are obtained by subtracting the local offset, $\tilde{j}_i^{(t)} = \hat{j}_i^{(t)} - \bar{j}_i$. Finally, the evaluation metric is the MPJPE of the corrected joints,

$$E_{avg} = \frac{1}{J}\sum_{i=1}^{J}\frac{1}{F}\sum_{t=1}^{F}\|\tilde{j}_i^{(t)} - j_i^{(t)}\|, \quad (2)$$

where J is the number of joints.

4. Baseline algorithms

In addition to the dataset, we also provide baseline results using representative tracking algorithms. For multi-view tracking, we use a robust Bayesian tracker [25] and twin Gaussian processes (TGP) [1] as the baselines for generative and discriminative methods, respectively. We also asked the author of [39] to evaluate our MADS multi-view videos using their code as the baseline for hybrid methods. For the depth-based tracking, in addition to the robust Bayesian tracking algorithm and TGP algorithm, we also provide results for the PDT tracker [2] and a Gaussian Mixture Model (GMM) pose estimation algorithm [44]. In this section we present the various baseline methods.

4.1. Robust Bayesian tracking algorithm

In Bayesian tracking, the tracking is treated as a problem of estimating the *posterior* $p(\theta_t|y_{1:t})$ probability distribution of the pose parameters θ_t at time t conditioned on the image observations $y_{1:t}$. With the assumption of a first-order Markov chain, the posterior can be derived [62] as

$$p(\theta_t|y_{1:t}) \propto p(y_t|\theta_t)p(\theta_t|y_{1:t-1}), \quad (3)$$

where $p(y_t|\theta_t)$ is the observation *likelihood* function, and $p(\theta_t|y_{1:t-1})$ is the *prediction* of the current pose using the previous posterior,

$$p(\theta_t|y_{1:t-1}) = \int p(\theta_t|\theta_{t-1})p(\theta_{t-1}|y_{1:t-1})d\theta_{t-1}. \quad (4)$$

The prediction from the previous pose is based on the diffusion model $p(\theta_t|\theta_{t-1})$. In (3), the current posterior is the prediction weighted by the likelihood $p(y_t|\theta_t)$.

The maximum of the posterior (Eq. 3) in each frame is found using the annealing particle filtering (APF) [16]. The APF is a layer-based particle filtering framework. Samples are diffused layer by layer with a Gaussian diffusion model, with the diffusion covariance gradually reduced from the top layer to the bottom layer. At the bottom layer, the samples converge to the local maximums of the likelihood function. The estimated pose state is weighted mean of the samples, $\hat{\theta} = \sum_{i=1}^{S}\omega_i^{(t)}\theta_i^{(t)}$, where the weight $\omega_i^{(t)}$ is the normalized likelihood of sample $p(y^{(t)}|\theta_i^{(t)})$.

We used a body model consisting of 15 cylinder parts, implemented by HumanEva [3]. Most joints are modeled as socket joints with 3 DoFs, while knees, clavicles and elbows are allowed 2 DoFs. The ankles and wrists are assumed to be only 1 DoF. With an additional 3 parameters representing the global position of the pelvis, the whole human body is modeled by a 40-dimension parameter. We next describe the likelihood functions used for multi-view tracking and depth tracking.

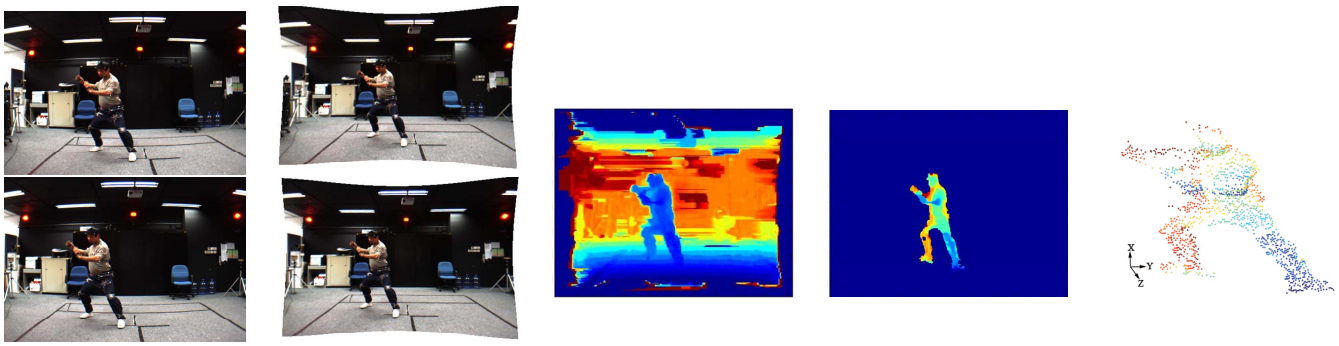
4.1.1. Robust likelihood on multi-view tracking

For multi-view tracking, we use the exponential Chamfer part-based likelihood (ECPBL) function from [25]. Although ECPBL can use silhouette, color, and edge descriptors, here we only use the silhouette and color as image descriptors, as the edge descriptors in MADS are much noisier and do not improve the tracking performance. The silhouette is a widely used cue for human pose tracking, since it represents the outline of a human body and can be extracted easily using background subtraction. To better localized self-occluded limbs, ECPBL segments the silhouette into parts, which are then compared with the projections of the corresponding parts in the body model. The silhouette is segmented using a GMM color model, which is learned using the initial pose and initial video frame. For the projection of body model parts, the portions of the projection that are occluded by other limbs are removed. The parts are then compared using the part-based matching term

$$\ell_{pECD} = \frac{1}{\sum_j |P_j^\theta|}\sum_j \sum_{\{i|P_j^\theta(i)=1\}} [1 - f(D_j(i))], \quad (5)$$

where P_j^θ is the visible portion of the j-th projected body part using pose parameters θ, and D_j is the Chamfer distance transform of the silhouette segment for the jth part, S_j.

$$f(x) = \exp(-(\frac{\|x\|}{\alpha})^\beta) \quad (6)$$



(a) (b) (c) (d) (e)

Figure 4: Stereo-based frame processing: (a) input stereo images captured with a stereo camera; (b) the rectified stereo images; (c) the depth map; (d) depth map with background removed; (e) observed point cloud.

(6) is an exponential transformation of the Chamfer distance measurement, which makes it more robust to small deviations in rotation and translation, and limits the penalty for large errors due to outliers.

The likelihood term in (5) favors poses where all the body parts are described by the silhouette. We also include a second likelihood term to make sure that the whole silhouette is used completely [63],

$$\ell_b = \frac{1}{|S|} \sum_{\{i|S(i)=1\}} [1 - P^\theta(i)], \tag{7}$$

where S is the binary silhouette image, and P^θ is the projection of the body for the pose parameter θ.

Finally, the two terms are combined to form a bi-directional silhouette matching term,

$$-\log p(z_t|\theta_t) \propto \gamma_1 \ell_{pECD} + \gamma_2 \ell_b, \tag{8}$$

where $\{\gamma_1, \gamma_2\}$ are weights for the likelihood items. The likelihood, Eq. 8, is calculated for each view and then summed together as the overall likelihood function. Compared to the standard silhouette matching between image observation and body projection, the exponential Chamfer distance (ECD) transform reduces the number of local maximums of the likelihood, resulting in a posterior that is easier to optimize using APF.

4.1.2. Robust likelihood function on depth-based tracking

We also adapt the robust likelihood function from [25] for tracking via depth images. Denote the observation point cloud as $\mathcal{P}_I = \{x_i\}_{i=1}^N$, and the model point cloud as $\mathcal{P}_M(\theta) = \{v_m^\theta\}_{m=1}^M$, for pose θ. The distance between each observation point in $\mathcal{P}_I$ to the nearest model point in $\mathcal{P}_M(\theta)$ is calculated as $d_{\mathcal{I}\to\mathcal{M}}(x_i) = \min_{v_m^\theta \in \mathcal{P}_M(\theta)} \|x_i - v_m^\theta\|^2$. Similarly, the distance between each model point and the nearest observation point is calculated as $d_{\mathcal{M}\to\mathcal{I}}(v_m^\theta) = \min_{x_i \in \mathcal{P}_I} \|x_i - v_m^\theta\|^2$.

Finally a bi-directional likelihood function is formed by applying the exponential transformation $f(x)$ to the distance measurements,

$$\begin{aligned} -\log p(\mathcal{P}_\mathcal{I}|\theta) = &\frac{1}{N} \sum_{i=1}^{N} [1 - f(d_{\mathcal{I}\to\mathcal{M}}(x_i))] \\ &+ \frac{1}{M} \sum_{m=1}^{M} \left[1 - f(d_{\mathcal{M}\to\mathcal{I}}(v_m^\theta))\right]. \end{aligned} \tag{9}$$

Similar to multi-view tracking, the upper bound on the penalty removes the effect of outliers and avoids sharp modes in the likelihood, while the exponential transformation make it less sensitive to small errors in the hypothesis. This allows the APF to focus on reducing larger errors.

4.2. Twin Gaussian processes

Gaussian process regression (GPR) is nonparametric regression method. For human pose estimation, the input is the feature vector extracted from the image, while the output is the vector of joint locations (i.e., the pose). Let (r_i, x_i) denote the i-th training instance, consisting of an input vector r_i and an output vector x_i, and define the matrices of inputs and outputs as $R = [r_1, \cdots, r_N]$ and $X = [x_1, \cdots, x_N]$. Likewise, let (r_*, x_*) be a test input vector and output vector. GPR assumes the regression function $f(r)$ is a zero-mean Gaussian process with covariance function (kernel function) $k(r_i, r_j)$, which encodes the relationships between input variables [1]. The kernel function used is the radial basis function (RBF). For the d-th output dimension, the joint distribution of the d-th dimension of the training outputs $X^{(d)}$ (i.e., the d-th row of X) and the d-th dimension of an unknown test output $x_*^{(d)}$, according to GPR, is

$$\begin{bmatrix} (X^{(d)})^T \\ x_*^{(d)} \end{bmatrix} \sim \mathcal{N}_R \left(0, \begin{bmatrix} K & (k_*)^T \\ k_* & k_{**} \end{bmatrix} \right), \tag{10}$$

where K is the $N \times N$ kernel matrix with entries $[K]_{ij} = k(r_i, r_j)$, k_* is a $1 \times N$ row vector with $[k_*]_i = k(r_i, r_*)$ and $k_{**} = k(r_*, r_*)$.

The joint distribution of the output vector $[X^{(d)}, x_*]$ is also

given by

$$\mathcal{N}_X\left(0, \begin{bmatrix} (X^{(d)})^T X^{(d)} & (X^{(d)})^T x_*^{(d)} \\ X^{(d)} x_*^{(d)} & x_*^{(d)} x_*^{(d)} \end{bmatrix}\right) \tag{11}$$

Twin Gaussian processes (TGP) [1] estimates the output target as the one that minimizes the difference between the input and output Gaussian distributions, $\mathcal{N}_X$ and $\mathcal{N}_R$, as measured by the Kullback-Leibler divergence,

$$\hat{x}_* = \underset{x_*}{\operatorname{argmin}} D_{KL}(\mathcal{N}_X \| \mathcal{N}_R). \tag{12}$$

To reduce the computational complexity, [1] predicts each test output using a reduced training set consisting of the K nearest neighbors to the test input (denoted as TGP-KNN). In [1], the input image features are HMAX and HOG extracted from the silhouette bounding box, while the output is a vector of pose parameter. Here we only use HOG as the image features. For multi-view pose estimation, we extract the HOG features from each view and concatenate them. For depth-based estimation, we extract HOG features directly from the depth map.

4.3. Personalized Depth Tracker

Helten, et al. [2] proposed a personalized depth tracker (PDT) to jointly estimate the body shape parameters and pose parameters. Denote $\mathcal{P}_{\varphi,\theta} \in \mathbb{R}^{3P}$ as the point set of the mesh model, where φ is the shape parameter vector and θ is the pose parameter vector, and P is the number of vertices of the mesh model. They define an eigenvector matrix $\phi \in \mathbb{R}^{3P \times \|\varphi\|}$ to describe the statistical variations from the shape parameter φ to the mesh model $\mathcal{M}$,

$$\mathcal{P}_{\varphi,\theta_0} = \mathcal{P}_{0,\theta_0} + \phi\dot{\varphi}. \tag{13}$$

Using the twisted exponential transformation [64], $\mathcal{P}_{\varphi,\theta}$ has an approximate linear relationship to the pose parameter θ.

Given the initial pose and shape, the mesh model is generated and compared to the observation point cloud $\mathcal{P}_I$ using distances to nearest points. To be more robust to noise, PDT considers both point-to-point distances, $d_{\text{point}}(x_i) = \|x_i - v(x_i)\|^2$, and point-to-plane distances, $d_{\text{normal}}(x_i) = \langle x_i - v(x_i), N(i)\rangle$, where $v(x_i)$ is the closest point in the mesh to x_i, and $N(i)$ is the normal vector of the mesh $\mathcal{M}_{\varphi,\theta}(x)$. The two distances are combined using a fixed threshold τ,

$$d_{\mathcal{I}\to\mathcal{M}}(x_i) := \begin{cases} d_{\text{point}}(x_i), & \|x_i - v(x_i)\|^2 > \tau \\ d_{\text{normal}}(x_i), & \text{otherwise.} \end{cases} \tag{14}$$

Finally, the bi-directional matching is used to construct the energy function,

$$E(\varphi, \theta | \mathcal{P}_I) = \sum_{i=1}^{N} d_{\mathcal{I}\to\mathcal{M}}(x_i) + \sum_{i=1}^{M} d_{\mathcal{M}\to\mathcal{I}}(v_i^\theta). \tag{15}$$

The energy function in (15) has analytical partial derivatives with respect to the shape parameter φ and the pose parameter θ. Hence, a gradient descent solver is used to update parameters after each iteration until convergence. Finally, the initial pose for PDT is the pose corresponding to the closest depth-map retrieved from the database in [9]. Hence, PDT is a hybrid approach that combines discriminative prediction with generative optimization.

4.4. A GMM-based pose and shape estimation algorithm

Ye and Yang [44] proposed a GMM-based joint pose and shape estimation algorithm. Instead of searching for explicit point correspondences, they used a GMM to model the relationship between body-model mesh points and the observation point clouds. The algorithm assumes that the distribution of the observation point cloud $\mathcal{P}_I$ follows a GMM whose centroids are form the generated mesh model $\mathcal{P}_M^\theta$. The likelihood of each observation point is

$$p(x_n) = (1-u)\sum_{m=1}^{M} p(v_m^\theta) p(x_n | v_m^\theta) + u\frac{1}{N}, \tag{16}$$

where

$$p(x_n | v_m^\theta) = \frac{1}{(2\pi\sigma^2)^{d/2}} \exp\left(\frac{-1}{2\sigma^2}\|x_n - v_m^\theta\|^2\right) \tag{17}$$

is a d-dimensional isotropic Gaussian and d is the dimensionality of a point, and u is the weight of uniform distribution for modeling outliers. The prior is assumed to be uniform $p(v_m^\theta) = 1/M$. The negative-log likelihood function is

$$E(\theta, \sigma^2) = -\sum_{n=1}^{N} \log\left(\sum_{m=1}^{M} \frac{1-u}{M} p(x_n | v_m^\theta) + \frac{u}{N}\right). \tag{18}$$

The expectation-maximization (EM) algorithm [65] is used to estimate the parameters θ and σ. During the E-step, the assignment probability of point n to centroid m, $p_{mn} = p(v_m^\theta | x_n)$, is calculated using parameters estimated in previous iteration,

$$\hat{p}_{mn} = \frac{\exp(\frac{-\|x_n - v_m^\theta\|^2}{2\sigma^2})}{\sum_{m=1}^{M} \exp(\frac{-\|x_n - v_m^\theta\|^2}{2\sigma^2}) + u_c}, \tag{19}$$

where $u_c = \frac{(2\pi\sigma^2)^{d/2} uM}{(1-u)N}$. For the M-step, parameters are estimated by minimizing the negative-log likelihood function,

$$Q(\theta, \sigma) = -\sum_{m,n} \hat{p}_{mn}\left(\log\left(\sum_{m=1}^{M} \frac{1-u}{M} p(x_n | v_m^\theta)\right) + \log\frac{u}{N}\right). \tag{20}$$

To make optimization analytically solvable, the body point cloud $\mathcal{P}_M^\theta$ is generated by the skinning mesh model weights α and the pose parameter θ using the twisted exponential transformation on the pose parameters. For more derivation details about mesh model representation and EM algorithm, please check [44].

4.5. 3D pictorial structures model

Belagiannis et al. [39] introduces a 3D pictorial structures (3DPS) model to estimate multiple human skeletons from 3D human body part hypotheses. The 3DPS model utilizes a conditional random field (CRF) with multiple potential functions. The unary potentials functions contain the 2D body part detection confidences (ϕ^{conf}), the reprojection error (ϕ^{repr}) of a body part from different views, the body part visibility from multiple views (ϕ^{vis}), and the body part length constraint (ϕ^{len}). The pairwise potential functions are built by imposing the kinematic constraints on body part translation (ϕ^{tran}) and rotation (ϕ^{rot}) with a collision potential function to constrain body parts to not collide with each other (ϕ^{col}). To find out a body configuration y with multiple body part hypotheses x, a posterior probability is calculated.

$$p(y|x) = \frac{1}{Z(x)} \left[\prod_i \phi_i^{conf}(y_i, x) \phi_i^{repr}(y_i, x) \phi_i^{vis}(y_i, x) \phi_i^{len}(y_i, x) \right] \cdot \left[\prod_{(i,j) \in E_{kin}} \phi_{i,j}^{tran}(y_i, y_j) \phi_{i,j}^{rot}(y_i, y_j) \phi_{i,j}^{col}(y_i, y_j) \right].$$

To infer the 3D human body skeletons, belief propagation is used to estimate the marginal distributions of the body parts, while the number of individuals is found with an object detector. The body parts for each individual are sampled from the marginal distribution and projected to each view. Then, all 3D poses are parsed based on the body part detection results. The inference process for a single person is the same as the multi-person case.

5. Experiments

In this section, we present experiments testing the baseline algorithms on the proposed MADS. Videos of the results can be found in the supplemental and online.[3] We test state-of-the-art methods on our MADS dataset. For multi-view videos, we test a generative ECPBL [25] tracker and a discriminative TGP [1] tracker as the baseline algorithm. Also, we test the 3DPS method [39] as the baseline for hybrid methods. For depth videos, we asked the authors of the PDT tracker [2] and the GMM-based tracker [44] to run our Tai-Chi videos using their methods. We also implement baselines by extending the APF and TGP to run on depth videos.

5.1. Experiments setup for multi-view videos

In our multi-view experiments, we down-sampled all images to 512×384, and used the three color views for tracking. We tested generative Bayesian tracking with ECPBL [25]. We set the exponential transformation parameters to $\alpha = 6$, $\beta = 2$, and the likelihood weights are $[0.4, 0.6]$. The APF used 200 particles and 5 layers. We used the constraint set by HumanEva [3], and also add limb intersection and angle range constraints. To show the influence of each element of the ECPBL algorithm, we also tested basic likelihood functions, including the bi-directional silhouette (BiS) [3], as well as BiS using exponential Chamfer distance (BiS+EC) or the part-based model (BiS+PB).

We also test the discriminative method TGP [1]. We use the TGP-KNN variant, which performs best in [1], and the parameters are set the same as in [1]. We extract HoG features from the gray-level image inside the bounding box of the silhouette. For each view, we extract a 324-dimensional HoG vector, yielding a 972-dimensional feature vector for the frame. TGP is trained using a leave-one-action out protocol within each action category. Since the TGP outputs are relative joint positions (all joints position minus the root position), we do not apply the joint offset displacement step (see in Section 3.4.3) on the TGP result. We used the TGP code provided by the authors.

3DPS [39] was evaluated using the authors' code and their own pre-trained body part detector. Their model only contains 15 joints, so we evaluate the mean error of pose estimation with the corresponding 15 joints in our dataset.

5.2. Results on multi-view videos

The average tracking errors on multi-view sequences are shown in Table 4, while the tracking error curves over time are shown in Fig. 5. Comparing the generative tracking methods on the 5 action categories, the Tai-chi and sports sequences are the easiest to track (overall average error ∼140 mm), compared to the Karate and hip-hop sequences, which are the hardest to track (∼200 mm error). The Tai-chi and sports actions do not contain many torso rotations and are performed slowly, while the Karate and dances (hip-hop and jazz) are very quick and consist of many body spinning actions. For generative trackers, the quick motions cannot be covered by the simple Gaussian diffusion model used with APF. In addition, generative trackers with the Gaussian diffusion model have difficulty recovering from tracking failures, even though the current frame may be easy to track. For the body spinning actions, where the torso rotates together with the movements of the limbs, APF has difficulty because it has a limited number of particles to simultaneously optimize both the torso orientation, which is a root parameter of the body kinematic-chain, and the limb parameters.

For generative tracking, ECPBL [25] outperforms other generative likelihood functions on both simple actions (Tai-chi and sports) and challenging actions (Karate, hip-hop, jazz). In addition, each component of ECPBL (EC and PB) improves the performance of the basic BiS likelihood, with EC improving about 3% and PB improving 12.5%. Using both EC and PB (i.e., ECPBL) yields an improvement of 20% over BiS.

Table 4 also shows the results of the baseline discriminative algorithm, TGP [1]. TGP performs similarly to ECPBL on Tai-chi, Karate and sports actions (within 10% error difference), while performing significantly better than ECPBL on the two dancing actions (38% better on jazz, 48% better on hip-hop). The performance of TGP depends mainly on whether the test poses are similar to those in the training frames, and not necessarily on the complexity of the actions. For example, TGP performs best on the jazz and hip-hop sequences, which

[3] http://visal.cs.cityu.edu.hk/static/images/MAPD/

Table 4: Average tracking error (mm) on multi-view sequences in MADS. The standard deviation of the tracking error is in parenthesis.

Tai-chi multi-view

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
BiS [3]	99.2(34.3)	123.3(27.9)	124.2(33.7)	226.8(79.0)	291.6(113.6)	147.6(46.7)	168.8(55.9)
BiS + EC	103.6(27.4)	97.7(25.5)	112.1(30.6)	160.5(42.5)	296.2(107.8)	159.6(43.9)	155.0(46.3)
BiS + PB	**60.7(13.9)**	**83.9(21.0)**	117.9(31.7)	178.6(45.7)	305.8(122.5)	**144.7(52.7)**	148.6(47.9)
ECPBL [25]	64.2(16.1)	96.8(30.3)	**106.6(33.6)**	145.6(52.4)	244.4(85.6)	145.1(56.2)	133.8(32.9)
TGP-KNN [1]	146.0(55.6)	99.0(29.3)	131.1(56.2)	**82.0(25.3)**	**87.6(37.7)**	183.8(134.7)	121.6(56.5)
3DPS [39]	179.7(73.7)	200.3(72.0)	257.5(88.6)	263.0(114.8)	293.4(155.5)	294.6(162.2)	248.1(111.1)

Karate multi-view

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
BiS [3]	290.0(99.6)	161.4(29.9)	199.4(70.1)	343.5(85.6)	350.2(88.5)	170.7(37.6)	252.5(68.6)
BiS + EC	330.8(94.4)	161.2(39.5)	279.4(83.7)	335.1(106.3)	255.8(86.6)	140.2(47.7)	250.4(76.4)
BiS + PB	289.8(124.2)	**156.6(34.2)**	**167.9(66.6)**	319.9(125.4)	**250.8(91.1)**	134.8(50.8)	220.0(82.0)
ECPBL [25]	**260.8(108.5)**	158.8(35.2)	183.8(76.6)	326.6(120.2)	252.0(78.3)	**125.3(30.4)**	217.9(74.8)
TGP-KNN [1]	269.2(128.4)	187.5(101.8)	183.8(96.7)	**226.7(95.7)**	250.8(121.2)	196.4(82.1)	219.1(104.3)
3DPS [39]	330.5(146.0)	260.9(111.9)	230.3(92.0))	289.2(143.3)	314.5(131.9)	197.6(124.1)	270.5(124.9)

Jazz multi-view

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
BiS [3]	270.3(66.3)	249.8(102.2)	217.2(78.2)	169.2(38.0)	184.1(66.8)	223.7(57.5)	219.1(68.2)
BiS + EC	219.9(73.6)	199.7(57.7)	248.4(57.7)	259.1(55.9)	156.7(32.0)	234.8(66.9)	219.8(57.3)
BiS + PB	209.4(62.5)	180.2(41.5)	165.3(49.3)	153.7(37.3)	182.2(80.8)	220.6(63.6)	185.2(55.8)
ECPBL [25]	208.4(77.2)	177.3(42.1)	158.6(36.7)	161.6(48.4)	147.9(34.5)	182.9(49.5)	172.8(48.1)
TGP-KNN [1]	**118.2(51.8)**	**125.1(53.7)**	**101.0(57.3)**	**92.7(45.9)**	**89.3(35.8)**	**118.5(61.6)**	107.4(51.0)
3DPS [39]	245.0(151.3)	273.3(140.8)	235.3164.2)	219.1(137.1)	218.8(155.0)	255.5(142.3)	241.2(148.4)

Hip-hop multi-view

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
BiS [3]	253.9(55.6)	248.5(58.2)	256.1(53.1)	199.4(75.8)	149.7(48.8)	202.2(55.3)	218.3(57.8)
BiS + EC	216.0(77.2)	162.0(49.2)	240.9(48.3)	232.3(63.4)	173.9(93.0)	194.0(80.8)	203.2(68.6)
BiS + PB	248.6(60.4)	165.2(53.5)	245.6(45.4)	155.4(78.4)	170.1(78.4)	199.1(88.9)	197.3(62.8)
ECPBL [25]	210.8(74.2)	159.9(48.6)	234.4(40.3)	157.9(48.3)	161.6(80.1)	233.0(97.3)	192.9(64.8)
TGP-KNN [1]	**114.1(49.8)**	**103.6(45.3)**	**95.5(67.3)**	**87.4(32.1)**	**90.7(56.7)**	**112.1(73.5)**	100.6(54.1)
3DPS [39]	188.8(85.3)	210.5(94.2)	182.3(80.9)	179.2(69.4)	180.9(75.9)	199.7(79.8)	190.2(80.9)

Sports multi-view

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
BiS [3]	150.6(46.3)	160.1(35.4)	158.8(51.0)	136.7(34.2)	183.6(43.3)	222.1(56.6)	168.7(44.5)
BiS + EC	152.6(39.7)	171.1(51.1)	195.5(91.4)	141.1(32.7)	156.0(34.3)	188.6(52.8)	167.5(50.3)
BiS + PB	125.0(26.7)	149.6(33.8)	159.4(42.5)	129.8(26.5)	157.4(37.9)	168.2(47.9)	148.2(35.9)
ECPBL [25]	124.4(25.6)	135.0(30.9)	**142.3(37.1)**	**122.5(27.8)**	140.9(33.8)	157.6(47.1)	137.1(33.7)
TGP-KNN [1]	**121.0(74.8)**	**131.3(64.7)**	185.4(68.9)	123.2(62.2)	**125.2(66.2)**	**148.8(109.7)**	139.1(74.4)
3DPS [39]	200.8(149.9)	180.3(112.4)	235.9(132.5)	221.7(178.1)	243.3(151.0)	243.8(160.3)	221.0(143.4)

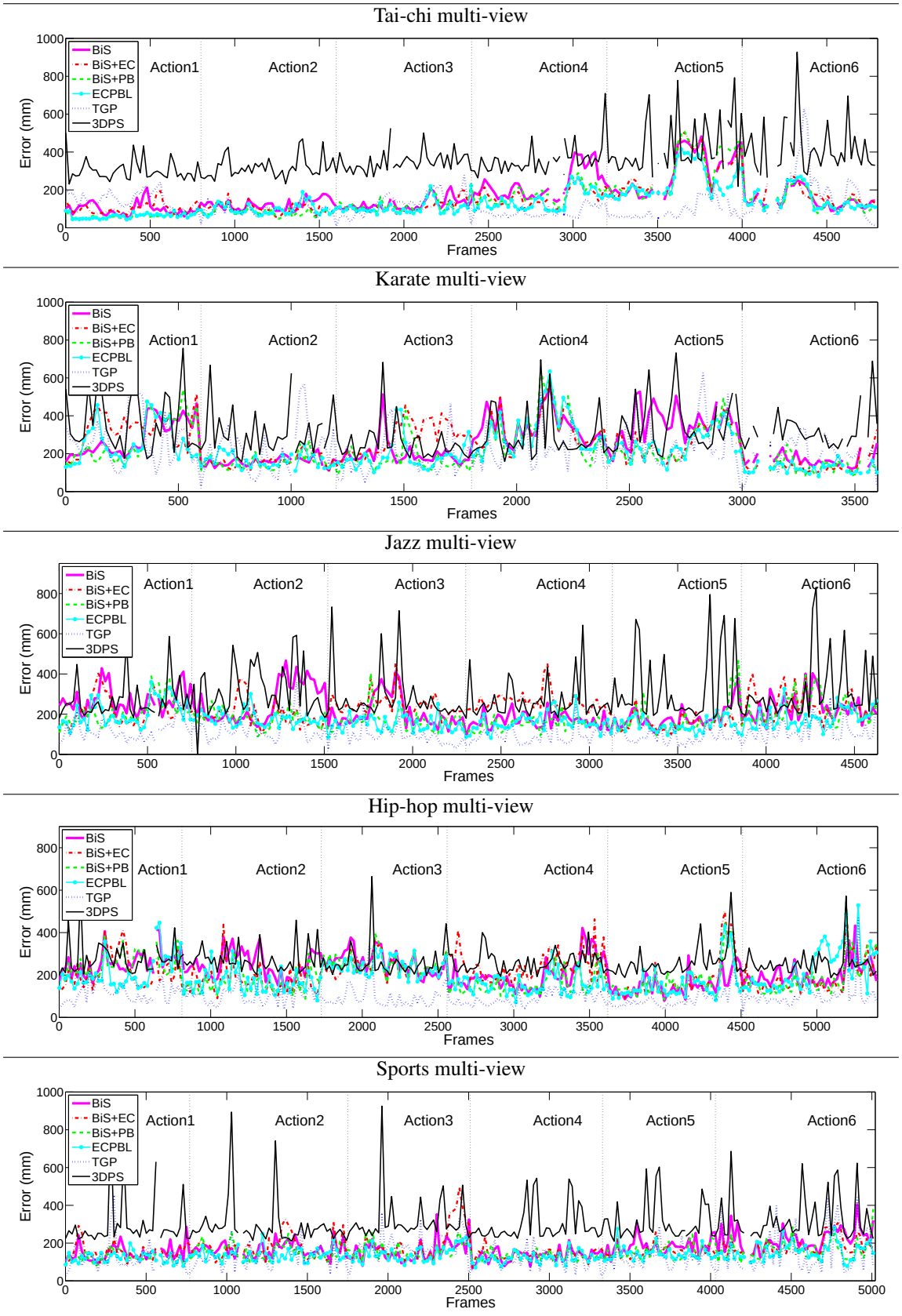


Figure 5: Tracking error (mm) over time on multi-view sequences in MADS. Sequences for each action category are concatenated into a single plot. Gaps in the error plots are due to frames with invalid ground-truth.

contain similar spinning and rotation movements among different sequences. On the other hand, TGP performs worse on the Karate sequences because the action sequences are chosen from different Karate forms. On the sports sequences, TGP error is similar to that of ECPBL, but with higher standard deviation in the error (see Table 4). This is because TGP performs well on similar poses that are common between the different sports, resulting in low errors on some frames, but loses track on unseen poses yielding higher errors on other frames (see Fig. 5).

The 3D pictorial structures model [39] performs worst among baselines. The main reason for poor performance could be that the human body part detector is not that accurate, since the author use their own body part detector trained from the MPII dataset [66]. Sometimes the left body parts may be recognized as right body parts, which introduced very large error. Hence, the standard deviation of 3DPS method is pretty large and the error curve changes rapidly.

5.3. Experiments setup for depth videos

For human pose tracking using depth images, we test the robust likelihood function described in Section 4.1.2 (APF), which compares the depth image and the model via their point clouds. The APF tracker used the same setting as the experiments on multi-view videos. We also consider two other likelihood functions: uni-directional matching with exponential transformation (Uni+Exp); bi-directional matching with linear distance measure (Bi+Lin).

For discriminative methods, we test TGP [1] where the inputs are HoG features extracted from the depth image inside the bounding box of the silhouette. We also tested the PDT [2] and the GMM-based pose algorithm [44], which are recent state-of-the-art hybrid methods that are based on point clouds.

5.4. Results on depth videos

The average error on each stereo action sequence is shown in Table 5, and the tracking error curves over time are shown in Fig. 6. Comparing the action categories, the Tai-chi sequences have the lowest overall error for all the baseline methods. This is because Tai-chi motion is slow and the master is usually facing to the camera, which makes tracking easier. On the other hand, for Tai-chi Action 3, most of the methods perform poorly because the actions are performed with the right side to the camera, making the left limbs invisible. Similar to the multi-view case, Karate and dancing actions (jazz and hip-hop) are very challenging because of their quick motions and body spinning actions. When there is single depth view, the body spinning actions cause more totally occluded limbs, which cannot be localized by the tracker.

Comparing the generative tracking methods, Bi+Exp usually performs better overall than Uni+Exp and Bi+Lin, except on Sports. Bi+Exp has lower error than Uni+Exp (about 12%) because the bi-directional matching of Bi+Exp encourages the model to cover the observations, but also constrains the model to not be too far away from the observations. Compared to using a linear penalty (Bi+Lin), the exponential transformation (Bi+Exp) smoothens the objective function, resulting in better localization of the limbs (about 7% lower error over all action categories). Note that Bi+Lin performs better than Bi+Exp on Sports and a few other actions (e.g. Karate Action 5, Hip-hop Action 5). The linear penalty performs better when the torso is rotating because the exponential transformation blurs the sides of the torso, making it appear to be at a different orientation.

The TGP algorithm [1] does not perform as well on the depth videos as on the multi-view videos. TGP still has lower error rate on Karate and Hip-hop actions compared to the generative methods, because these action categories have similar poses between the action sequences. For jazz and sports actions, TGP performs poorly, possibly because the HOG feature on depth map is not robust enough for human pose estimation, compared to HOG on multi-view RGB.

Finally, we also compare PDT [2] and the GMM-based algorithm [44] on the Tai-chi depth sequences. Overall PDT performs worse than the GMM-based algorithm and Bi+Exp. The discriminative stage of the PDT algorithm predicts candidate poses from their collected database, which sometimes does not contain poses similar to the Tai-chi sequences (especially for Action 3). This results in poor initialization for the subsequent generative tracking stage. Overall, the GMM-based algorithm also tends to perform worse than Bi+Exp. Failures of the GMM-based method are typically due to noise in the stereo depth image; the body shape could not be estimated correctly resulting in poor estimation of the pose.

5.5. Tracking error analysis

In this section we analyze the errors made by the generative APF-tracker and the discriminative Twin GP tracker.

5.5.1. Generative tracker analysis

To further study the failures of the generative APF-based tracker on our MADS dataset, we run another trial of experiments. We divide the each sequence into several sub-sequences, and re-initialize the tracker using the ground-truth pose on the first frame of each sub-sequence. The sequences are divided by the action type, e.g. spinning and arm waving for dancing sequences. The sports sequences are divided by the particular sport actions, e.g. serving, shooting and dribbling. We run the ECPBL tracker on the multi-view sequences, and run the Bi-exp tracker on depth sequences. Several tracking curves and tracking examples are shown in Fig. 7 and Fig. 8.

Fig. 7 shows that, in the dancing sequences, the spinning action always has large accumulated errors. This is because the root angle and position parameters need to change quickly in a spinning action, which makes the APF optimization process more difficult, since the search space is larger. Another reason is that the orientation of the body is estimated mainly by the torso size of the subject and the model, whereas the exponential transformed cost function of (6) blurs the size of observation, which confounds the estimation of the body orientation. For depth sequences, the task is also difficult due to many self-occlusions in monocular depth images, e.g., see the tracking examples of Jazz Action2 depth spinning actions in Fig. 8. For both depth videos or RGB videos, the low frame rate and

Table 5: Average tracking error (mm) on depth sequences in MADS. The standard deviation of the tracking error is in parenthesis.

Tai-chi depth

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
Uni+Exp	59.9(11.7)	70.6(16.6)	102.0(20.4)	70.4(14.5)	**68.1(14.1)**	100.8(30.3)	78.6(17.9)
Bi+Lin	61.4(10.8)	69.5(15.0)	111.0(27.5)	102.7(26.9)	70.3(11.9)	126.6(48.90)	80.0(23.5)
Bi+Exp	**61.2(11.8)**	**66.0(13.1)**	**96.0(27.2)**	**69.5(15.6)**	68.6(13.5)	**86.6(23.3)**	74.6(17.4)
TGP-KNN [1]	119.4(49.6)	114.2(78.7)	214.5(119.5)	113.2(36.8)	98.4(28.9)	133.7(46.6)	132.2(60.0)
PDT [2]	79.8(30.1)	79.7(25.0)	229.0(48.0)	105.6(31.6)	108.8(39.4)	140.0(54.7)	123.8(38.1)
GMM-based [44]	88.0(29.7)	66.1(20.9)	127.7(36.9)	80.2(19.0)	113.7(29.1)	114.0(30.1)	98.4(27.6)

Karate depth

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
Uni+Exp	344.5(113.9)	249.7(71.4)	328.2(88.1)	178.7(30.0)	148.0(52.6)	351.7(67.4)	266.8(70.6)
Bi+Lin	327.8(70.5)	271.3(103.4)	286.0(101.9)	194.2(31.0)	**117.5(35.8)**	306.5(49.9)	250.5(65.4)
Bi+Exp	**202.3(60.3)**	230.6(78.8)	**217.9(62.0)**	156.8(34.9)	235.1(49.6)	237.7(49.8)	213.4(55.9)
TGP-KNN [1]	214.4(116.8)	**167.4(61.6)**	246.7(71.5)	**149.3(55.9)**	119.5(41.7)	**197.2(63.2)**	182.4(68.5)

Jazz depth

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
Uni+Exp	282.6(100.4)	262.3(69.3)	297.5(76.6)	195.0(69.5)	236.3(118.7)	233.6(67.8)	251.2(83.7)
Bi+Lin	225.7(60.3)	241.0(76.6)	202.1(91.2)	**151.6(39.4)**	259.3(98.0)	**183.2(56.7)**	210.5(70.4)
Bi+Exp	237.3(74.0)	**237.1(74.5)**	191.1(80.8)	160.5(34.2)	**201.9(98.4)**	194.7(98.4)	203.8(69.8)
TGP-KNN [1]	**184.1(80.5)**	244.3(89.9)	**172.4(77.6)**	226.8(108.8)	249.8(110.9)	199.2(90.8)	216.5(99.6)

Hip-hop depth

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
Uni+Exp	257.9(67.4)	232.4(93.9)	288.7(80.8)	141.3(34.6)	209.3(80.4)	141.5(41.2)	211.8(66.4)
Bi+Lin	254.7(58.4)	240.1(78.2)	254.8(54.5)	214.3(67.7)	**152.0(63.0)**	227.6(69.1)	223.9(65.1)
Bi+Exp	181.2(68.2)	221.4(87.4)	251.7(51.9)	211.1(75.1)	190.4(77.7)	**132.2(40.8)**	198.0(66.8)
TGP-KNN [1]	**144.6(58.8)**	**170.6(72.8)**	**218.7(76.4)**	**140.9(61.5)**	184.2(109.1)	141.0(76.9)	166.2(75.9)

Sports depth

	Action1	Action2	Action3	Action4	Action5	Action6	Overall
Uni+Exp	123.2(27.1)	131.5(35.5)	300.7(82.5)	145.3(40.4)	242.2(87.1)	167.0(52.2)	184.7(54.1)
Bi+Lin	**122.5(31.0)**	**124.9(36.2)**	264.7(76.5)	155.1(38.2)	165.8(44.9)	**150.6(51.3)**	163.9(46.3)
Bi+Exp	156.7(51.6)	130.3(36.9)	276.9(68.8)	**134.6(32.8)**	**146.8(37.8)**	166.0(44.2)	168.5(45.4)
TGP-KNN [1]	254.2(83.1)	215.4(81.4)	**248.1(89.4)**	212.6(96.9)	236.9(88.2)	239.1(99.0)	234.4(89.7)

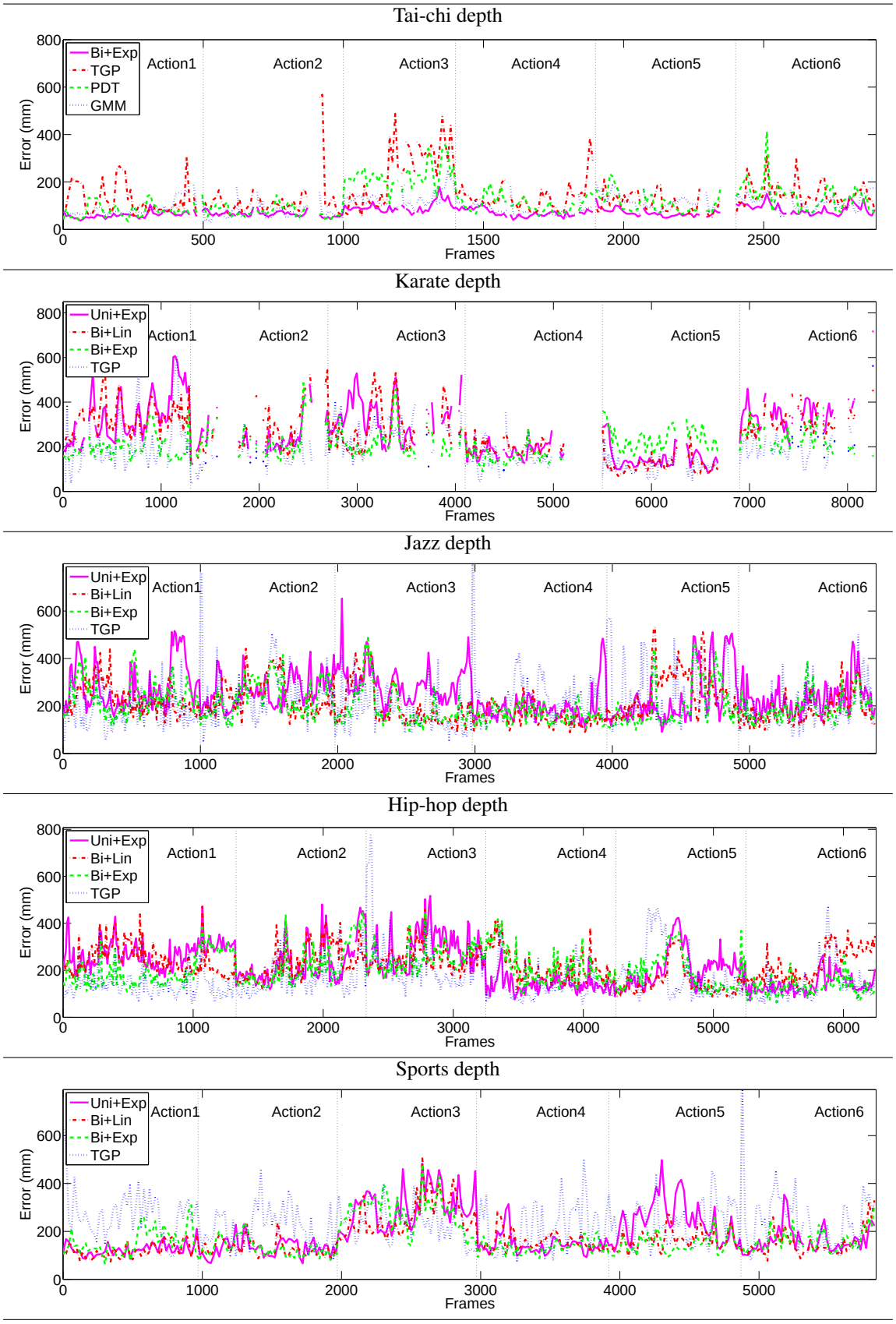
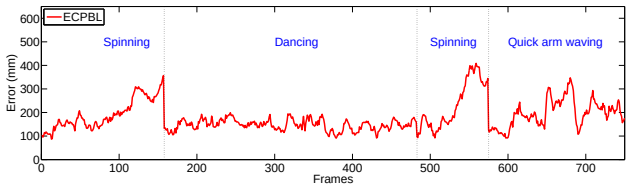
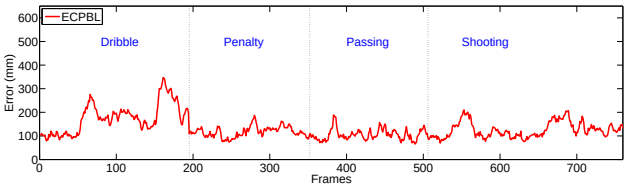


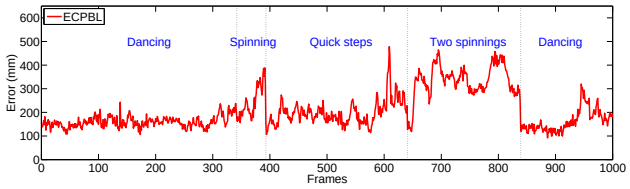
Figure 6: Tracking error (mm) over time on depth sequences in MADS. Sequences for each action category are concatenated into a single plot. Gaps in the error plots are due to frames with invalid ground-truth.



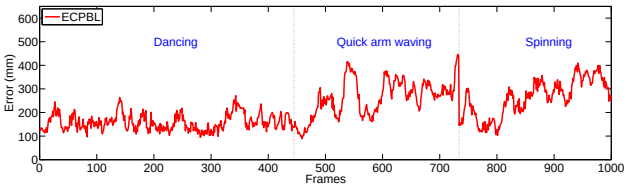
Jazz Action1 multi-view



Sports Action3 multi-view sequence



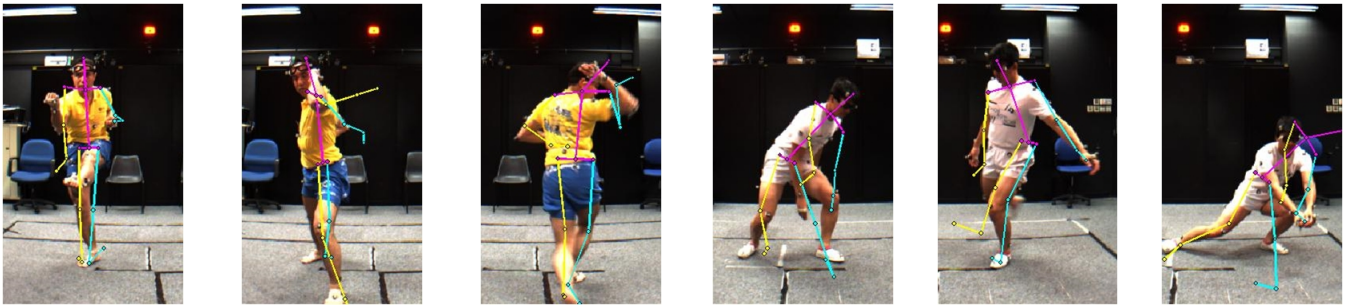
HipHop Action2 depth



Jazz Action6 depth sequence

Figure 7: Tracking errors over time on re-initialization experiments. Each action sequence is divided into several sub-sequences by the movement type (denoted by the black dashed line). At the beginning of each sub-sequence, the APF-based tracker is re-initialized with the ground-truth pose.

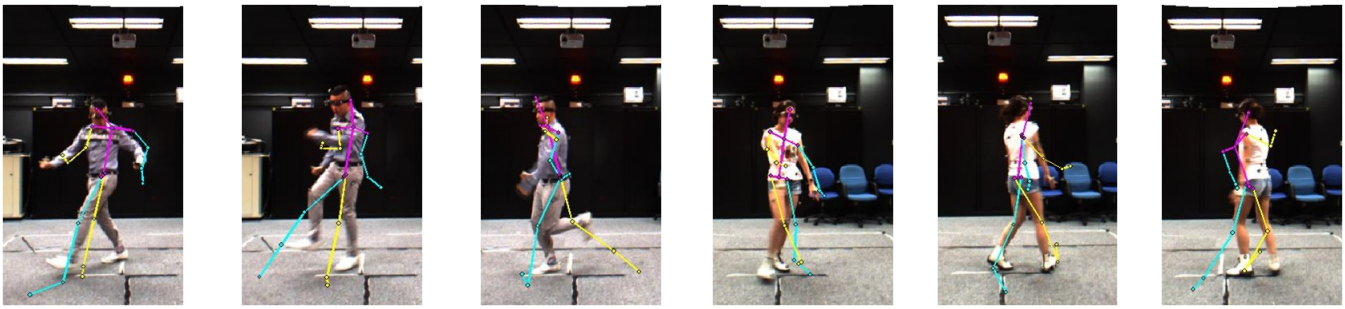
Tracking examples for self-occlusions



Karate Action1 frame 306 | Karate Action1 frame 370 | Karate Action4 frame 80 | Sports Action2 frame 231 | Sports Action3 frame 280 | Sports Action6 frame 948

Tracking examples for quick steps

Tracking examples for spinning



Hip-hop Action2 frame 575 | Hip-hop Action2 frame 585 | Hip-hop Action2 frame 595 | Jazz Action2 frame 480 | Jazz Action2 frame 490 | Jazz Action2 frame 500

Figure 8: Examples of poor tracking results by APF-based tracker on MADS due to self-occlusion, quick steps, and spinning.

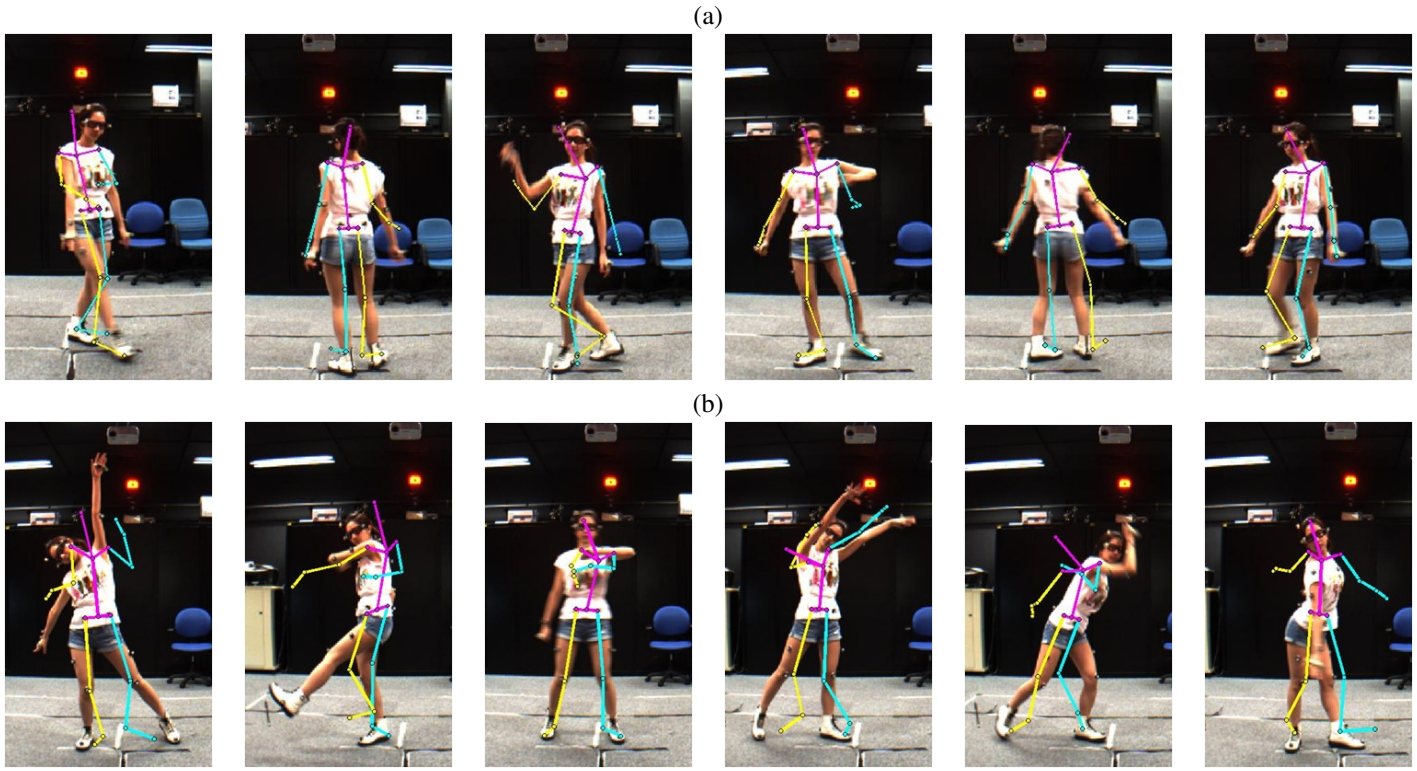


Figure 9: (a) Examples of TGP tracking results on Jazz multi-view sequences of spinning actions. (b) Examples of failure TGP tracking results.

quick motions produced large amounts of motion blur, which also makes localization of limbs harder.

Another cause of tracking errors is quick actions, e.g. quick arm waving or jumping in dancing sequences (see Fig. 7). The initialization of APF for the current frame is the tracked pose in the previous frame. When there is a large change between two frames, then the initialization will be far away from observation, and the Gaussian diffusion model of APF will not be able to cover such a large search space. An example of quick step in the Hip-hop Action2 depth sequence is shown in Fig. 8.

In our MADS dataset, there are many self-occlusions, which are also a typical problem for human pose tracking and estimation. In the martial arts videos, there are many arm and leg movements to defend and attack, while in the dance videos, arms are waved frequently in front of the torso. In the basketball and football videos, the dribble actions usually have self-occlusions, since the subject needs to protect the ball from being stolen by others. All these action categories introduce many self-occlusions that are different than other daily-life motions, e.g. walking. Examples of highly-occluded poses are shown in Fig. 8.

5.5.2. Discriminative tracker analysis

Compared to the generative trackers, the discriminative trackers does not need an initial pose and human 3D model. On the other hand, discriminative trackers require training data and a training process. The performance of discriminative trackers depends on the similarity between the training data and testing data. On our MADS dataset, TGP [1] outperforms generative APF-based trackers on the dancing sequences, since the spinning action appears in other sequences (Fig. 9 (a)) and thus can be learned by the TGP model. Some failure tracking results are shown in Fig. 9 (b).

To study when can the TGP tracker performs well, we compute the 3D mean joint error between the ground truth test poses and the nearest pose in the training set, and compare this to the TGP tracking error in Fig. 10. From the figure, we can find that the tracking error is rarely smaller than the error to the nearest pose, which indicates that the TGP tracker highly depends on the training set. When there is not a very similar pose in the training set, e.g. the peaks of the blue curve, the tracking error of TGP will have larger errors, which means the TGP fails to localize the body limbs correctly.

Fig. 11 shows a scatter plot of the error of the nearest training pose and the tracking error for TGP. Below 120mm, the TGP tracking error and the nearest pose error varies linearly, which demonstrates that TGP can predict the nearest pose in the training set or variations thereof. On the other hand, when the nearest pose has error larger than 120 mm, the TGP tracking result has much larger errors than 120 mm. This suggests that, for the TGP tracker to work well, the test poses should be similar to the training poses within 120mm, above which TGP will incur more significant errors.

6. Conclusions

In this paper, we have introduced the Martial Arts, Dancing and Sports dataset (MADS). The MADS dataset contains five

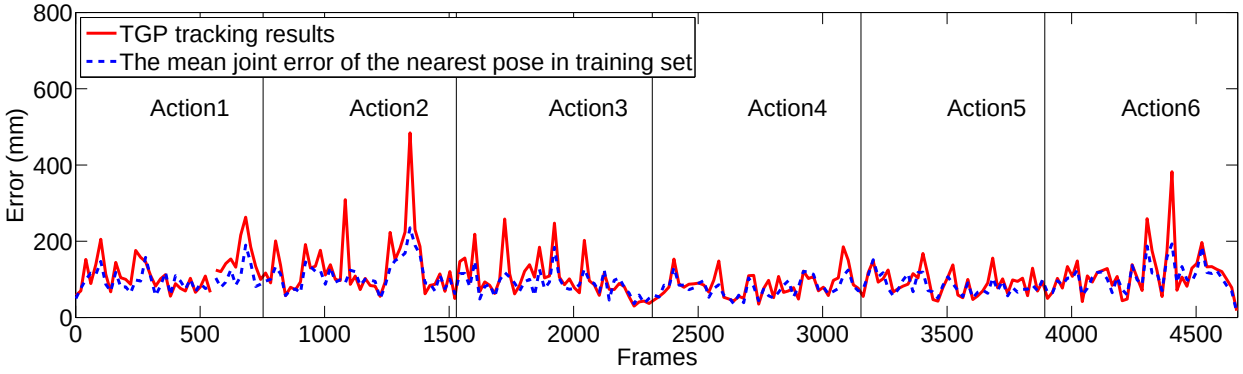


Figure 10: Error curve for the multi-view Jazz action category, with its action sequences concatenated into a single plot. The red curve is the tracking error of TGP tracker, while the blue curve is 3D joint mean error to its nearest pose in the training set.

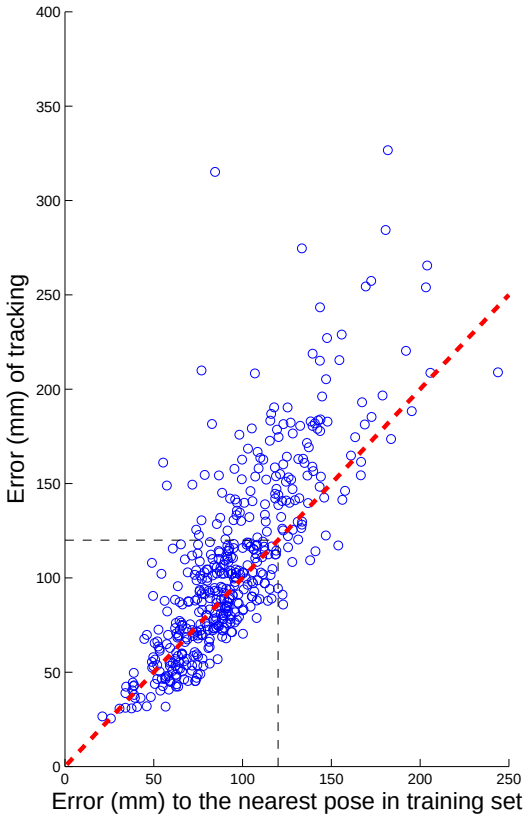


Figure 11: The scatter plot showing the error between the ground truth test pose and its nearest pose in the training set vs. the tracking error of TGP tracker. The red dashed line is when the tracking error is equal to the nearest pose error. The black dashed line indicates 120mm error.

categories of challenging actions (Tai-chi, Karate, jazz, hip-hop and a sports combo), which are not widely used by the computer vision community. These actions contain more complex poses that do not appear in typically designed repeatable actions (e.g. walking, gesture and boxing). The dataset consists of color multi-view sequences and color stereo sequences, with corresponding MOCAP data as ground truth. The stereo depth data contains more noise than Kinect/ToF active sensors, but provides an important opportunity to improve the robustness human pose estimation in unconstrained environments using stereo sensors. For benchmarking purposes, we evaluate several baseline algorithms on our MADS dataset, including an APF-based Bayesian tracker, twin Gaussian processes, PDT, and a GMM-based tracker. All data and associated code will be made freely available for academic use.

Acknowledgement

We thank L Sigal, AO Balan and MJ Black for providing the HumanEva baseline protocol. We thank L Bo and C Sminchisescu for their TGP code. We also thank Mao Ye from University of Kentucky for running his GMM-based algorithm on our Tai-chi depth data, Thomas Helten from MPI (now in Pixargus) for running his PDT tracker on our Tai-chi depth data, and Vasileios Belagiannis from TUM for running his 3DPS tracker on our multi-view data. We also thank Sensei Alan Lai and Ueshiro Karate Hong Kong for helping with the data collection. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (CityU 123212).

References

[1] L. Bo, C. Sminchisescu, Twin gaussian processes for structured prediction, International Journal of Computer Vision 87(1-2) (2010) 28–52.

[2] T. Helten, A. Baak, G. Bharaj, M. Muller, H.-P. Seidel, C. Theobalt, Personalization and evaluation of a real-time depth-based full body tracker, in: 3DTV-Conference, 2013 International Conference on, IEEE, 2013, pp. 279–286.

[3] L. Sigal, A. O. Balan, M. J. Black, Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion, International Journal of Computer Vision 87(1-2) (2010) 4–27.

[4] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, R. Bajcsy, Berkeley mhad: A comprehensive multimodal human action database, in: Applications of Computer Vision (WACV), 2013 IEEE Workshop on, IEEE, 2013, pp. 53–60.

[5] C. Ionescu, D. Papava, V. Olaru, C. Sminchisescu, Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments, IEEE Transactions on Pattern Analysis and Machine Intelligence 36 (7) (2014) 1325–1339.

[6] V. Ganapathi, C. Plagemann, D. Koller, S. Thrun, Real time motion capture using a single time-of-flight camera, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2010, pp. 755–762.

[7] V. Ganapathi, C. Plagemann, D. Koller, S. Thrun, Real-time human pose tracking from range data, in: Computer Vision–ECCV 2012, Springer, 2012, pp. 738–751.

[8] Microsoft, Kinectsdk, https://www.microsoft.com/en-us/kinectforwindows/ (2013).

[9] A. Baak, M. Müller, G. Bharaj, H.-P. Seidel, C. Theobalt, A data-driven approach for real-time full body pose reconstruction from a depth camera, in: Consumer Depth Cameras for Computer Vision, Springer, 2013, pp. 71–98.

[10] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, R. Moore, Real-time human pose recognition in parts from single depth images, Communications of the ACM 56 (1) (2013) 116–124.

[11] M. Ye, X. Wang, R. Yang, L. Ren, M. Pollefeys, Accurate 3d pose estimation from a single depth image, in: IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 731–738.

[12] T. B. Moeslund, A. Hilton, V. Krüger, L. Sigal, Visual Analysis of Humans, Springer, 2011.

[13] R. Poppe, Vision-based human motion analysis: An overview, Computer Vision and Image Understanding 108(1) (2007) 4–18.

[14] M. Isard, A. Blake, Condensation – conditional density propagation for visual tracking, International Journal of Computer Vision 29(1) (1998) 5–28.

[15] C. Sminchisescu, B. Triggs, Estimating articulated human motion with covariance scaled sampling, The International Journal of Robotics Research 22 (6) (2003) 371–391.

[16] J. Deutscher, I. Reid, Articulated body motion capture by stochastic search, International Journal of Computer Vision 61(2) (2005) 185–205.

[17] P. F. Felzenszwalb, D. P. Huttenlocher, Pictorial structures for object recognition, International Journal of Computer Vision 61(1) (2005) 55–79.

[18] J. Gall, B. Rosenhahn, T. Brox, H.-P. Seidel, Optimization and filtering for human motion capture, International Journal of Computer Vision 87(1-2) (2010) 75–92.

[19] J. Bandouch, O. C. Jenkins, M. Beetz, A self-training approach for visual tracking and recognition of complex human activity patterns, International Journal of Computer Vision 99(2) (2012) 166–189.

[20] P. Wang, J. M. Rehg, A modular approach to the analysis and evaluation of particle filters for figure tracking, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Vol. 1, 2006, pp. 790–797.

[21] V. John, E. Trucco, S. Ivekovic, Markerless human articulated tracking using hierarchical particle swarm optimisation, Image and Vision Computing 28 (11) (2010) 1530–1547.

[22] C. Menier, E. Boyer, B. Raffin, 3d skeleton-based body pose recovery, in: 3rd International symposium on 3D data processing, visualization and transmission (DPVT'06), IEEE Computer Society, 2006, pp. 389–396.

[23] C.-H. Huang, E. Boyer, N. Navab, S. Ilic, Human shape and pose tracking using keyframes, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 3446–3453.

[24] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, T. P. Andriacchi, Markerless motion capture through visual hull, articulated icp and subject specific model generation, International Journal of Computer Vision 87(1) (2010) 156–69.

[25] W. Zhang, L. Shang, A. B. Chan, A robust likelihood function for 3d human pose tracking, IEEE Transactions on Image Processing 23 (2014) 5374–5389.

[26] A. Agarwal, B. Triggs, Recovering 3d human pose from monocular images, Pattern Analysis and Machine Intelligence, IEEE Transactions on 28 (1) (2006) 44–58.

[27] C. Ionescu, F. Li, C. Sminchisescu, Latent structured models for human pose estimation, in: IEEE International Conference on Computer Vision (ICCV), 2011, pp. 2220–7.

[28] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2005, pp. 886–893.

[29] G. Pons-Moll, D. Fleet, B. Rosenhahn, Posebits for monocular human pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 1–8.

[30] C. Ionescu, J. Carreira, C. Sminchisescu, Iterated second-order label sensitive pooling for 3d human pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 1–8.

[31] A. Toshev, C. Szegedy, Deeppose: Human pose estimation via deep neural networks, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 1–8.

[32] S. Li, Z.-Q. Liu, A. B. Chan, Heterogeneous multi-task learning for human pose estimation with deep convolutional neural network, International Journal of Computer Vision (IJCV) 113 (1) (2015) 19–36.

[33] S. Li, A. B. Chan, 3d human pose estimation from monocular images with deep convolutional neural network, in: Asian Conference on Computer Vision (ACCV), 2014, pp. 1–8.

[34] N. Lawrence, Probabilistic non-linear principal component analysis with gaussian process latent variable models, The Journal of Machine Learning Research 6 (2005) 1783–1816.

[35] J. M. Wang, D. J. Fleet, A. Hertzmann, Gaussian process dynamical models for human motion, Pattern Analysis and Machine Intelligence, IEEE Transactions on 30 (2) (2008) 283–298.

[36] G. W. Taylor, L. Sigal, D. J. Fleet, G. E. Hinton, Dynamical binary latent variable models for 3d human pose tracking, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010, pp. 631–638.

[37] L. Sigal, M. Isard, H. Haussecker, M. J. Black, Loose-limbed people: Estimating 3d human pose and motion using non-parametric belief propagation, International journal of computer vision 98 (1) (2012) 15–48.

[38] M. Burenius, J. Sullivan, S. Carlsson, 3d pictorial structures for multiple view articulated pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013, pp. 3618–3625.

[39] V. Belagiannis, S. Amin, M. Andriluka, B. Schiele, N. Navab, S. Ilic, 3d pictorial structures for multiple human pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 1669–1676.

[40] H. Joo, H. Liu, L. Tan, L. Gui, B. Nabbe, I. Matthews, T. Kanade, S. Nobuhara, Y. Sheikh, Panoptic studio: A massively multiview system for social motion capture, in: IEEE International Conference on Computer Vision (ICCV), 2015, pp. 3334–3342.

[41] S. Knoop, S. Vacek, R. Dillmann, Fusion of 2d and 3d sensor data for articulated body tracking, Robotics and Autonomous Systems 57 (3) (2009) 321–329.

[42] S. Pellegrini, K. Schindler, D. Nardi, A generalisation of the icp algorithm for articulated bodies., in: BMVC, Vol. 3, Citeseer, 2008, p. 4.

[43] D. Droeschel, S. Behnke, 3d body pose estimation using an adaptive person model for articulated icp, in: Intelligent Robotics and Applications, Springer, 2011, pp. 157–167.

[44] M. Ye, R. Yang, Real-time simultaneous pose and shape estimation for articulated objects using a single depth camera, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 1–8.

[45] C. Qian, X. Sun, Y. Wei, X. Tang, J. Sun, Realtime and robust hand tracking from depth, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 1–8.

[46] C. Plagemann, V. Ganapathi, D. Koller, S. Thrun, Real-time identification and localization of body parts from depth images, in: IEEE International Conference on Robotics and Automation (ICRA), IEEE, 2010, pp. 3108–3113.

[47] J. Taylor, J. Shotton, T. Sharp, A. Fitzgibbon, The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2012, pp. 103–110.

[48] Y. Zhu, B. Dariush, K. Fujimura, Kinematic self retargeting: A framework for human pose estimation, Computer Vision and Image Understanding 114 (12) (2010) 1362–1375.

[49] J. Lallemand, M. Szczot, S. Ilic, Human pose estimation in stereo images, in: Articulated motion and deformable objects, Springer, 2014, pp. 10–19.

[50] J. Gall, A. Fossati, L. Van Gool, Functional categorization of objects using real-time markerless motion capture, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2011, pp. 1969–1976.

[51] X. Wei, P. Zhang, J. Chai, Accurate realtime full-body motion capture using a single depth camera, ACM Transactions on Graphics (TOG) 31 (6) (2012) 188.

[52] T. Helten, M. Muller, H.-P. Seidel, C. Theobalt, Real-time body tracking with one depth camera and inertial sensors, in: IEEE International Conference on Computer Vision (ICCV), IEEE, 2013, pp. 1105–1112.

[53] MathWorks, 3-d vision toolbox, http://www.mathworks.com/help/vision/stereo-vision.html (2014).

[54] F. Bogo, J. Romero, M. Loper, M. J. Black, Faust: Dataset and evaluation for 3d mesh registration, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2014, pp. 3794–3801.

[55] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis, I. Pitas, The i3dpost multi-view and 3d human action/interaction database, in: Visual Media Production, 2009. CVMP'09. Conference for, IEEE, 2009, pp. 159–168.

[56] T. Tung, T. Matsuyama, Topology dictionary for 3d video understanding, Pattern Analysis and Machine Intelligence, IEEE Transactions on 34 (8) (2012) 1645–1657.

[57] D. Vlasic, I. Baran, W. Matusik, J. Popović, Articulated mesh animation from multi-view silhouettes, in: ACM Transactions on Graphics (TOG), Vol. 27, ACM, 2008, p. 97.

[58] H. Hirschmuller, Accurate and efficient stereo processing by semi-global matching and mutual information, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2005, pp. 807–814.

[59] J.-Y. Bouguet, Camera calibration toolbox, http://www.vision.caltech.edu/bouguetj/calib_doc/ (2013).

[60] Z. Zivkovic, Improved adaptive gaussian mixture model for background subtraction, in: International Conference on Pattern Recognition (ICPR), Vol. 2, IEEE, 2004, pp. 28–31.

[61] T. Horprasert, D. Harwood, L. S. Davis, A robust background subtraction and shadow detection, in: Proc. ACCV, 2000, pp. 983–988.

[62] A. Doucet, S. Godsill, C. Andrieu, On sequential monte carlo sampling methods for bayesian filtering, Statistics and computing 10 (3) (2000) 197–208.

[63] C. Sminchisescu, A. Telea, et al., Human pose estimation from silhouettes. a consistent approach using distance level sets, in: 10th International Conference on Computer Graphics, Visualization and Computer Vision (WSCG'02), Vol. 10, 2002.

[64] C. Bregler, J. Malik, K. Pullen, Twist based acquisition and tracking of animal and human kinematics, International Journal of Computer Vision 56 (3) (2004) 179–194.

[65] A. P. Dempster, N. M. Laird, D. B. Rubin, Maximum likelihood from incomplete data via the em algorithm, Journal of the Royal Statistical Society. Series B (Methodological) (1977) 1–38.

[66] M. Andriluka, L. Pishchulin, P. Gehler, B. Schiele, 2d human pose estimation: New benchmark and state of the art analysis, in: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014, pp. 3686–3693.