Human Joint Angle Estimation and Gesture Recognition for Assistive Robotic Vision

Alp Guler, Nikolaos Kardaris, Siddhartha Chandra, Vassilis Pitsikalis, Christian Werner, Klaus Hauer, Costas Tzsafestas, Petros Maragos, Iasonas Kokkinos

Human Joint Angle Estimation and Gesture Recognition for Assistive Robotic Vision

Alp Guler[1], Nikolaos Kardaris[2], Siddhartha Chandra[1], Vassilis Pitsikalis[2], Christian Werner[3], Klaus Hauer[3], Costas Tzafestas[2], Petros Maragos[2], Iasonas Kokkinos[1]

(1) INRIA GALEN & Centrale Supélec Paris,
(2) National Technical University of Athens, (3) University of Heidelberg

Abstract. We explore new directions for automatic human gesture recognition and human joint angle estimation as applied for human-robot interaction in the context of an actual challenging task of assistive living for real-life elderly subjects. Our contributions include state-of-the-art approaches for both low- and mid-level vision, as well as for higher level action and gesture recognition. The first direction investigates a deep learning based framework for the challenging task of human joint angle estimation on noisy real world RGB-D images. The second direction includes the employment of dense trajectory features for online processing of videos for automatic gesture recognition with real-time performance. Our approaches are evaluated both qualitative and quantitatively on a newly acquired dataset that is constructed on a challenging real-life scenario on assistive living for elderly subjects.

1 Introduction

The increase in elderly population is a fact worldwide [1]. In this context computer vision and machine learning research applied on human-robot-interaction from the perspective of assistive living has both scientific interest and social benefits. In this work we focus on two prominent directions and apply the respective methods in the context of a challenging assistive living human-robot interaction scenario. This involves a robotic rollator that interacts with the elderly subjects using visual sensors, assisting them in every-day activities. These directions involve the use of state of the art deep learning based approaches for human joint angle estimation for the future goal of subject stability estimation, as well as the application of action recognition methods to enable elderly subjects interact with the robot by means of manual gestures. Herein we focus on the visual processing pipelines of this interface, and show a variety of rich applications and experiments.

There has been a furore of activity on the pose estimation front in recent years. Pose estimation usually involves inferring the locations of landmarks or body parts and the quality of the prediction is measured by metrics that involve comparing the predicted and the ground truth locations in the image plane. In this work we address the problem of estimating human joint angles. The joint angle estimation task involves estimating the angles made by segments of the human body at the joint landmarks in world coordinates. More specifically, we are interested in estimating (a) the knee angles, that is, the angles between by the thigh and the shank segments of the left and right legs,

and (b) the hip angles, or the angles made by the torso and thigh segments. These angles will later be used to determine if the user's pose is unstable. In case of instability, the robot can assist the user achieve a stable position by exerting a physical force of the optimal magnitude in the optimal direction.

Human action recognition is a very active research area with a multitude of applications, such as video retrieval, health monitoring as well as human-computer interaction for assistive robotics. Proposed approaches on action recognition span several directions, such as deep architectures [2] and local spatio-temporal features with the popular Bag-of-Visual-Words (BoVW) paradigm [3–5], as well as one of the top performing approaches, the dense trajectories [6, 7]. Application on human-robot interaction [8] has received relatively little attention mainly due to the increased computational requirements of most action recognition systems, which prevents them from performing in real-time. We target the rocognition of elderly subjects' gestural commands by employing dense trajectories and exploring alternative encoding methods. We have implemented a real-time version of our gesture recognition system that uses an activity detector and is currently integrated in the robotic platform.

In this work we present our contributions on applying these approaches on data containing actual elderly subjects. We are guided by a vision whereby assistive human-robot interaction is advanced by state-of-the-art results in mid and higher level vision applications. The assistive scenario involves a robotic rollator equiped with multiple sensors, as shown in Fig. 2, that is capable of estimating the pose and the joint angles, as well as recognizing manual gestures using the proposed approaches. We utilize a deep learning based 2D joint localization approach fused with 3D information acquired from RGB-D sensors to localize joints in 3D and estimate the angles between line segments(see Sec.3). Moreover, we apply our action-gesture recognition approach based on the recent dense trajectories features [6], employing a variety of encoding schemes. We also feed mid-level vision information [9] to the higher level of action-gesture recognition. All cases are evaluated on rich scenarios of a new dataset and task with elderly subjects showing promising results. Finally, we examine practical issues concerning an online version of the system that is integrated in the robotic platform and has close-to-real-time performance.

2 The human-robot interaction assistive dataset

The experimental prototype used for data acquisition [10] consists of a robotic rollator equipped with sensors such as laser range sensors that scan the walking area for environment mapping, obstacle detection and lower limbs movement detection, force/torque handle sensors and visual sensors: an HD camera to record patient's upper body movements and two Kinect sensors. The first Kinect captures the torso, waist and hips and the second faces downwards towards the lower limbs. The recording process involved acquiring RGB-D videos using the open-source Robotics Operating System (ROS) software capturing human subjects in a set of predefined use-cases and scenarios.

The dataset we use for estimating joint angles (Fig. 2) consists of (a) colour and depth images captured by the ROS-Kinect system, and (b) human-joint landmark point

Fig. 1: Sample gestures from Task-6a and illustration of the visual processing with dense trajectories for gesture recognition (Sec. 4). First row: "Come here". Second row: "I want to stand up".

trajectories captured by a *Qualisys* motion capture (*MoCap*) system. This dataset has $9K$ images coming from 27 video recordings.

The data acquired for gesture recognition ("Task-6a") comprises recordings of the patient sitting in front of the rollator, which is placed at a distance of 2.5 meters. Task-6a includes 19 different gestural and verbal commands. Each command is performed by the 13 patients, $3-6$ times. Sample gestures are depicted in Fig. 1. The task is challenging, as mobility disabilities seriously impede the performance ability of a verbal and/or gestural command. Moreover, due to the cognitive disabilities of some users, in some cases we observe different pronunciations of a command even among multiple performances of the same user. In essence, we miss the consistency that is sometimes assumed in other datasets [11]. In addition, background noise and other people moving in the scene make the recognition task even harder.

3 Human Joint Angle Estimation from Kinect RGB-D Images

Our first contribution is a deep learning framework for human pose estimation. More precisely, we are interested in estimating the hip and the knee angles. We define (a) the hip angle as the angle made at the hip joint by the shoulders and the knees, and (b) the knee angle as the angle made at the knee by the hips and the ankles. These angles give a fair indication of the human pose, and can be exploited to determine if the user's pose is unstable.

3.1 Pose Estimation Dataset

Our objective is to use the RGB-D images to estimate the joint-angles in the world coordinate system, and evaluate them against the angles computed from the point-trajectories. We assume that any deviation between the actual joint angles and those computed from the point-trajectories, due to an offset between the motion capture markers and the human body, is neglegible. Our dataset suffers from several severe limitations. Firstly, the upper, lower kinect and MoCap systems are not aligned in time. Secondly, the upper and lower kinect sensors look at parts of the human body individually,

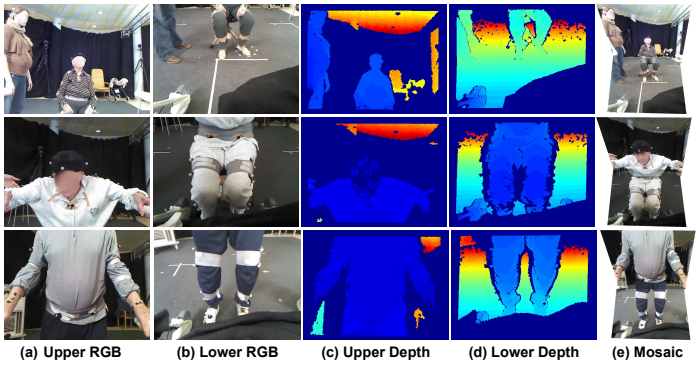

Fig. 2: Example images from our dataset. Notice the scale changes, and noise in the depth images. Our dataset also suffers from occlusion and truncation of body parts. We also have multiple people in the videos.

and do not see the entire context. Finally, the depth images from the kinect are sparse and noisy (fig. 2). We alleviate the first two limitations by using preprocessing procedures described in the rest of this section. To cope with noisy depth estimates, we use a sparse coding based strategy to denoise the predictions. This is described in section 3.6.

Data Alignment As mentioned, our input data and our ground truth are recorded by two different systems, and are not aligned. To evaluate the accuracy of our predictions, we require that the images are aligned-in-time with the joint-angle trajectories. More specifically, we need to know which image-frame corresponds to which time-stamp in the MoCap trajectory. This alignment problem is non-trivial due to several factors.

Firstly, the two recording systems do not start at the same time. Secondly, the two systems capture data at different frequencies. Besides, while the MoCap system has a constant frequency, the frequency of the ROS-kinect system varies with the system load. Due to this, the upper-body frames and lower-body frames coming from the two kinect sensors are not aligned. To fix the variable frame rate issue, we re-sample the kinect captured videos at 10 *fps* via nearest neighbour interpolation. To align the upper and lower body frames, we only use images from the time-stamps when both upper and lower frames are available, discarding the remaining frames. Finally, to align the MoCap trajectories with the kinect frames, we design an optimization framework. More specifically, we define variation-functions for the images and trajectories. The variation functions can be understood as how the data changes over time. Once we define these functions, we can align the data by ensuring that these variations are correlated in time, since each variation results from the same cause, namely the human subject moving.

We denote by $f_x(t)$ the variation function of the point trajectories. $f_x(t)$ is defined to be simply the L_2 norm of the position of center of the human body at each timestamp t in the point trajectories,

$$f_x(t) = ||b(t)||_2^2, \tag{1}$$

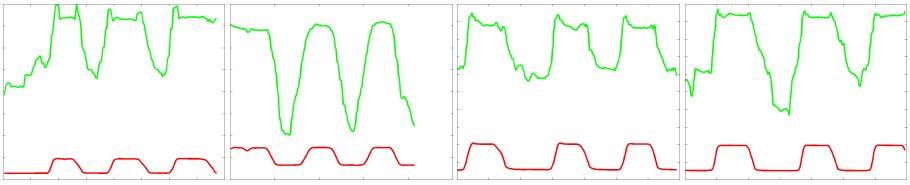

Fig. 3: Results of our alignment procedure for 4 videos. The green curve shows the image variation function $f_x(t)$, the red curve shows the trajectory variation function $f_x(t)$. The human subjects in these videos performed the sit-stand exercise 3 times.

where $b(t)$ denotes the center of the body (the pelvis). We define, $f_y(t)$, the variation function of the images I as the L_2 distance, in the HOG feature space, of an image at timestamp t from the first image ($t = 0$) in the sequence.

$$f_y(t) = ||I_t^{hog} - I_0^{hog}||_2^2 \tag{2}$$

The alignment problem can now be solved by aligning the f_x and f_y signals. To achieve this, we minimize the cross-correlation between them by exhaustively sampling a *delay* parameter between them. The exhaustive search takes less than $2-3$ minutes per video. Figure 3 shows the variation functions as the result of our alignment procedure for two different videos.

Image Stitching. In our experiments, we observed that images containing only the lower or the upper body are very hard samples for pose estimation methods because these lack a lot of informative context about the pose. To cope with this issue, we stitch the images from the lower and upper kinect to obtain a single image containing the full human body. This stitching is done by estimating a planar homography transformation between manually selected keypoints. Using this geometric transformation we overlay the upper and lower sensor images to get the *mosaic* image. However, the mosaic images look unrealistic due to perspective distortion. We fix this by performing perspective correction on the mosaic image by estimating the necessary affine transformation. Since the kinect sensors are stationary, this procedure is done only once, we use the same transformation matrices for all images in our dataset. Results of image stitching can be seen in fig. 2. In the following sections, we describe our approach.

3.2 Related Work

In the recent years, deep learning has emerged as the gold standard of machine learning on nearly all benchmarks, including the task of human pose estimation. A number of deep learning methods solve the pose estimation problem by first estimating the spatial locations of interesting landmarks, and then inferring the pose from these landmark locations. In a broad sense, these approaches can be classified into two categories. The first category consists of methods that directly recover spatial locations of interesting landmarks via regression. Inspired by the deep convolutional cascaded network of Sun et al. [12], the *Deep Pose* approach by Toshev et al. [13] treats pose estimation as a regression task, refining the predictions iteratively via a cascaded network architecture.

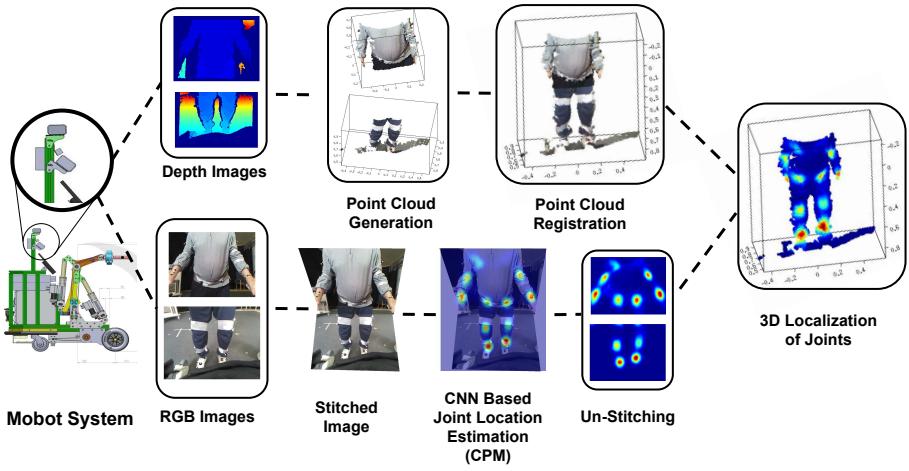

Fig. 4: A schematic representation of our joint angle estimation pipeline. We have two stationary RGBD sensors (shown in the circle in the first column) on a moving rollator. We stitch the upper and lower RGB images, and perform part localization on the mosaic image using Convolutional Pose Machines. The mosaic image is then unstitched and the part beliefs are transferred to the registered point cloud. This gives us the spatial locations of the body parts in the world space. Given these part locations, we estimate the joint angles using vector algebra.

The second class of methods estimate probability distributions of the presence of these landmarks, which can then be treated as unaries in an inference procedure that enforces spatial constraints to recover the likelihoods of these landmarks ([14–18]). These approaches typically use fully convolutional neural networks to estimate a per pixel likelihood of joint locations. A number of these approaches[19, 20] employ *iterative error feedback*, that is, a cascaded system to repeatedly refine the per pixel likelihoods, similar to recurrent networks. The *DeeperCut* approach [21] exploits a very deep residual neural network and achieves promising results without using a sequential refinement scheme. More recently, Haque et al. [18] proposed a viewpoint invariant pose estimation framework for RGB-D images that utilizes Long Short Term Memory (LSTM) [22] to incorporate the idea of error feedback.

3.3 Joint Angle Estimation from RGB-D Images

The joint angle estimation task can be addressed by either (a) directly regressing the angles from the images, or (b) first estimating the spatial locations of the human parts, and then estimating the joint angles using geometry. As described in section 3.2, we have two broad classes of methods that address the problem of spatial localization of landmarks. An advantage of methods that directly regress landmark locations from the image data is that they can learn pairwise interactions in a fully connected graph efficiently via fully connected layers, thereby exploiting the holistic context in the image.

However, these methods typically use an object detector at the testing stage [13] and then regress from the detected bounding boxes to the landmark locations. Object detection is necessary to prevent noise in the context around the object of interest from corrupting the features. While this is a reasonable strategy, this two step procedure can be undesirable for datasets containing low resolution images, or images captured in the wild where object detection can be inaccurate. Methods that estimate the probability distributions can work around this requirement by estimating the probability distributions over the entire image, and then inferring the landmark locations by enforcing spatial constraints. Our dataset has noisy, low resolution images, with significant occlusion, and dramatic scale changes. This motivates us to employ a fully convolutional network to estimate the spatial locations of the human parts in the image plane.

For our task of 3D localization of the human parts, it is natural to use the depth cues from the depth images. The depth field of view of the Microsoft Kinect sensor ranges from 25 centimeters to a few meters. This leads to a noisy and sparse reconstruction of the depth field in cases where the object of interest is close to the sensor. Our dataset has numerous instances where the human is too close to the kinect sensor. Consequently, the depth information is unreliable (or absent) in these cases. The RGB sensors do not suffer from this limitation. This motivates the usage of the clean RGB image for the estimation of joint locations, later to be combined with the depth information to reconstruct the 3d joint locations. We therefore, first estimate the part positions from the RGB images, then use the depth images to reconstruct the part positions in the world system, and finally estimate the joint angles via geometry. Our pipeline is described in Fig. 4. For estimating the part positions in 2D, we use the *Convolutional Pose Machines*[15] approach which achieves the state of the art results on the challenging MPII dataset[23]. This approach was trained using thirty thousand images from the MPII and Leeds Sports[23, 24] datasets.

3.4 Convolutional Pose Machines

As mentioned in section 3.3, our method is based on the *convolutional pose machines* [15] approach, that uses sequential prediction convolutional blocks that operate on image and intermediate belief maps and learn implicit image-dependent spatial models of the relationships between parts. We now briefly describe the prediction strategy.

The human part localization task is done in several stages. The first stage of the method predicts part beliefs from local image evidence via a fully convolutional network. The network has seven convolutional layers, trained to minimize the L_2 distance between the predicted beliefs and the ground truth beliefs. The ground truth beliefs are synthesized by putting Gaussian peaks at ground truth locations of parts. The second stage learns convolutional features from the image, and combines them with the beliefs from the first stage (via concatenation), and again learns a classifier to predict part beliefs from this combination of features and beliefs using the L_2 loss. The third stage predicts the part beliefs by combining features and the beliefs from the second stage and so on. While the network in [15] uses six stages, our network uses a cascade of four stages. The network is designed to learn interactions within increasingly larger neighbourhoods by repeatedly increasing the receptive fields of the convolutions in each subsequent stage of the cascade. Larger receptive fields are able to model long-range

spatial interactions between the landmarks, and help impose geometrical constraints on the locations of the body parts, such as the knee, the hip, the shoulder, etc. The cascaded network predicts the per-pixel beliefs for fourteen landmarks, namely the neck, head, and the left and right ankle, knee, hip, wrist, elbow, shoulder and wrist.

3.5 Joint Angles from Part Locations

The method described in section 3.4 predicts human part locations in the image plane. However, our objective is to estimate joint angles in the world space. After the part beliefs are estimated in the mosaic image, we unstitch the mosaic image, and transfer the part beliefs to the upper and lower images. We then construct the 3D point cloud using the depth images and the intrinsic parameters of the kinect sensors, and transfer the part beliefs to the point clouds. This is followed by registering the point clouds of the upper and lower kinect sensors so that they are in the same space, and transferring the part beliefs to the common system. For each part, we choose the location to be the point with the maximum belief. If the maximum belief for a part is below a threshold ϵ, we conclude that this part is either occluded or truncated. This gives us 3D spatial locations of the body parts. Given the part locations in the world coordinate system, we compute the joint angles using vector algebra.

3.6 Denoising via Sparse Coding

As described in section 3.3, our dataset suffers from drastic scale changes causing the depth to be sparse and noisy. This causes the angle estimates to be noisy. To overcome this difficulty, we pose reconstruction of our predictions as a classic sparse coding based denoising approach. We estimate a dictionary $\mathcal{D}$ of codewords from the ground truth angles $\hat{x}$ by solving the following optimization

$$\mathcal{D}^* = \underset{\mathcal{D},w}{\operatorname{argmin}} ||\hat{x} - \mathcal{D}w||_2^2 + \lambda_1||w||_1 + \lambda_2||w||_2^2. \tag{3}$$

The reconstruction of noisy predictions x is done by estimating a sparse linear combination w^* of the dictionary as

$$\begin{aligned} w^* = \underset{w}{\operatorname{argmin}} ||x - \mathcal{D}^*w||_2^2 + \lambda_1||w||_1 + \lambda_2||w||_2^2 \\ y = \mathcal{D}^*w^* \end{aligned} \tag{4}$$

This reconstruction suggests that the predictions can be represented sparsely over the dictionary, and helps tilt the distribution of the predicted angles towards the ground truth angles. We use this technique to denoise our hip angle predictions. We report results of denoising in section 3.7.

3.7 Experiments and Discussions

In this section we report empirical results for the estimation of hip, and knee angles. As described in section 3.1, our dataset has 27 videos, containing about $9K$ image frames.

We use the point trajectories of the shoulders, hips, knees and angles from the MoCap data to compute the ground truth angles. Our prediction pipeline is described in section 3.4. Our networks are trained on $30K$ images from the *MPII* and *Leeds Action Dataset*, as mentioned in section 3.4, and are the state of the art on these challenging datasets. The evaluation criterion is the Root Mean Square Error (RMSE) in degrees between the predicted angles and the ground truth angles. We also report the detection rate for each angle. An angle is said to be *detected* if the locations of all three parts constructing the angle are available. The detection rate for an angle is defined to be the ratio of frames in which the angle is detected.

We first study the effect of changing the threshold ϵ described in section 3.5. This threshold is applied to the maximum beliefs of parts, and introduces a measure of confidence in the part localization. If the maximum belief of a part is below ϵ we conclude that the part is absent. This absence occurs because of occlusion or truncation of parts of interest. Fig. 5 shows the variation of RMSE and detection rate for different values of ϵ. As the minimum accepted confidence ϵ increases, RMSE decreases substantially and naturally the detection rate drops due to increasing number of rejected angles. High confidence in part locations, which mainly depends on visibility of parts, leads to better estimates of joint angles. This increase in performance with large ϵ is observed despite the remaining erroneous joint locations, which would have been corrected if more context (and in many cases joints themselves) were visible. This provides evidence that a better placement of the cameras such that full body context is provided to the algorithm would lead to an increase in the performance of the algorithm. To further emphasize this, we exemplify estimated angles and groundtruth angles as a function of time for the same subject and same task under two different camera configurations in Fig.6. In the configuration where the patient is closer to the sensors and the ankles and shoulders are occluded the algorithm is not able to estimate the angles, whereas in the setting where the subject is distanced from the sensor, the angle estimation is more accurate.

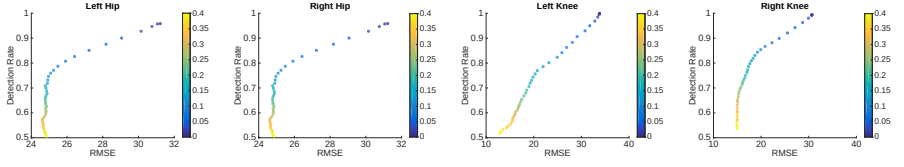

Fig. 5: Variation of *Root Mean Square Error* in degrees and detection rate with confidence threshold ϵ. As the confidence threshold increases (as the colours become hotter), the detection rate and the angle estimation errors decrease.

For our quantitative results, we choose $\epsilon = 0.2$, which gives a detection rate of 0.75 (figure 5). Our results are reported in table 1. We report the RMSE values for the estimation of the left and right knee and hip angles, averaged over the entire dataset. It can be seen that the errors in estimation of the hip angles are higher than those corresponding to the knee angles. This is due to two factors: (a) the shoulders are truncated when the human subject to too close to the camera (fig. 2, row 3), and (b) the hip joints are oc-

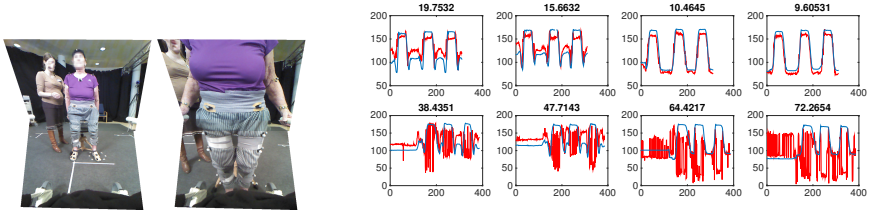

Fig. 6: Angle estimation on same subject and same task(sitting - standing) under different camera configurations. The left image corresponds to a time instance in the top row time series and the right image corresponds to an instance in the bottom row time series: Estimated (red) and the groundtruth angle (blue). The columns portray Left Hips, Right Hips, Left Knee and Right Knee angles respectively. Corresponding RMSE values are displayed on top of each time series.

cluded by the elbows and hands (fig. 2, row 1). Table 1 also reports the estimation errors after the denoising process described in section 3.6. To estimate the dictionary (eq. 3), we use a leave-one-out strategy. The dictionary for denoising of a video is constructed using the dataset, leaving out this particular video. The parameters for the dictionary learning procedure are as follows: $\lambda_1 = 10$, $\lambda_2 = 0.001$, and the size of the codebook $K = 200$. We see an improvement in estimation of both the knee and hip angles after denoising, however the improvement in case of the hip angles is more prominent.

Method	Left Knee	Right Knee	Left Hip	Right Hip	Average
CPM	16.16	15.24	24.51	25.27	20.29
CPM + Denoising	13.66	14.23	15.06	16.48	14.86

Table 1: Quantitative results of our Convolutional Pose Machines pipeline for the estimation of joint angles. The evaluation metric is the Root Mean Square Error in degrees. Denoising using a sparse coding strategy improves average performance by 5.4 degrees.

4 Dense Trajectories for Gesture Recognition

Gesture recognition allows the interaction of the elderly subjects with the robotic platform through a predefined set of gestural commands. Our gesture classification pipeline, depicted in Fig. 7, employs Dense Trajectories features along with the Bag-of-Visual-Words framework. We briefly describe the individual steps involved and present our experimental results on Task-6a. Dense trajectories [25] consists in sampling feature points from each video frame on a regular grid and tracking them through time based on optical flow. Different descriptors are computed within space-time volumes along each trajectory. Descriptors are: the Trajectory descriptor, HOG [26], HOF [27] and MBH [26].

Features are encoded using BoVW, VLAD or Fisher vector to form a video representation. BoVW uses a codebook which is constructed by clustering a subset of ran-

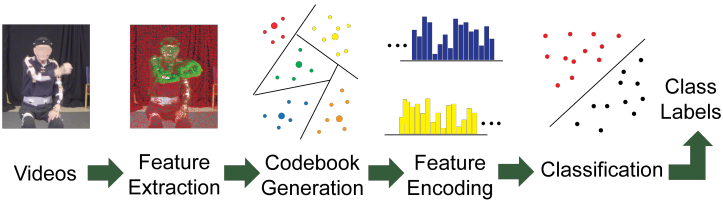

Fig. 7: Gesture classification pipeline.

domly selected training features into $K = 4000$ clusters. Each trajectory is assigned to its closest visual word and a histogram of visual word occurrences is computed, yielding a sparse K-dimensional representation. VLAD encodes first order statistics among features by computing differences between extracted features and the visual words. Fisher vector encodes first and second order statistics using a Gaussian Mixture Model (GMM) with $K = 256$ gaussians. To compensate for the high dimensionality of FV, we reduce the descriptors' dimensionality by a factor of 2 using Principal Component Analysis.

Videos encoded with VLAD or Fisher vector are classified using linear support vector machines (SVMs). Different descriptors are combined by concatenating their respective VLAD or FV vectors. In the BoVW case, videos are classified using SVMs with the χ^2 [26] kernel. Different descriptors are combined in a multichannel approach, by computing distances between BoVW histograms as:

$$K\left(\mathbf{h}_i, \mathbf{h}_j\right) = \exp\left(-\sum_c \frac{1}{A_c} D\left(\mathbf{h}_i^c, \mathbf{h}_j^c\right)\right), \tag{5}$$

where c is the c-th channel, i.e. $\mathbf{h}_i^c$ is the BoVW representation of the i-th video, computed for the c-th descriptor, and A_c is the mean value of χ^2 distances $D\left(\mathbf{h}_i^c, \mathbf{h}_j^c\right)$ between all pairs of training samples. Since we face multiclass classification problems, we follow the one-against-all approach and select the class with the highest score.

Experiments Gesture classification is carried out on a subset of the Task-6a dataset comprising 8 subjects and 8 gestures[1], without limiting the generality of results. Results are shown in Table 2. It is evident that the large variability of the gesture performance among patients has a great impact on performance. The combined descriptor performs consistently better, since it encodes complementary information extracted from the RGB channel. VLAD and Fisher vector further improve performance, since they encode rich information about the visual words' distribution. Fig. 8 depicts the mean confusion matrix over all patients computed for BoVW and the MBH descriptor. Naturally, gestures that are more easily confused by the elderly subjects are harder to classify correctly, e.g. "Help" and "PerformTask" both consist of a similar horizontal movement but in different height. To demonstrate the difficulty of the task, we use the same ges-

[1] The 8 selected gestures are: "Help", "WantStandUp", "PerformTask", "WantSitDown", "ComeCloser", "ComeHere", "LetsGo", "Park".

	BoW									VLAD	Fisher
	p1	p4	p7	p8	p9	p11	p12	p13	avg.	avg.	avg.
traject.	50.0	24.0	35.3	30.3	20.8	43.8	37.5	68.8	38.8	37.0	45.0
HOG	56.3	28.0	38.2	48.5	25.0	62.5	37.5	71.9	46.0	56.6	48.0
HOF	62.5	36.0	32.4	45.5	54.2	46.9	62.5	71.9	51.5	54.9	51.7
MBH	62.5	56.0	44.1	51.5	58.3	56.3	45.8	81.3	57.0	69.9	66.3
combined	75.0	52.0	52.9	57.6	58.3	65.6	62.5	75.0	**62.4**	**67.6**	**68.1**

Table 2: Classification accuracy (%) per patient on a subset of the Task-6a dataset that contains 8 gestures performed by 8 subjects. Results for different encoding methods are shown; "avg." stands for average accuracy over all patients.

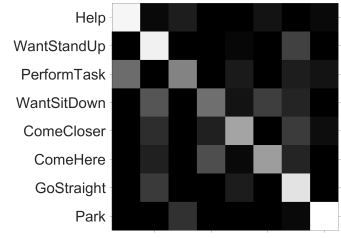

Fig. 8: Mean confusion matrix over all subjects of the Task-6a dataset. The results are obtained with the MBH descriptor and BoW encoding (6th row of Table 2).

Fig. 9: Sample frames from the gesture dataset acquired in [28].

	Task-6a	dataset from [28]
traject.	38.8	74.0
HOG	46.0	53.8
HOF	51.5	77.3
MBH	57.0	82.5
combined	62.4	84.8

Table 3: Comparative results on Task-6a and the dataset acquired in [28]. Mean classification accuracy (%) over all subjects is reported.

ture classification pipeline on the dataset acquired in [28]. It includes 13 young subjects (Fig. 9) that perform the same 19 gestures as the ones in Task-6a under similar conditions. Training and testing is carried out with the same settings using BoW encoding. Mean classification over all subjects is reported. Comparative results shown in Table 3 illustrate that variability in the execution of the gestures among healthy and cognitively intact subjects is effectively handled by our pipeline. This highlights the challenge that the development of a gestural communication interface for elderly people presents.

4.1 Filtering background trajectories

To further improve our gesture recognition results, we have worked towards the integration of the body part segmentation method introduced in [9] into our gesture recognition

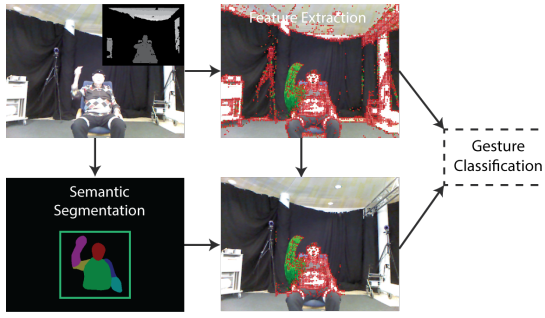

Fig. 10: Combining semantic segmentation with feature extraction for gesture classification on the Task-6a. A separate BoVW histogram is computed for the whole frame and the area enclosed by the bounding box. The two vectors are combined into a single video representation.

pipeline. Specifically, we use the subject's bounding box to reject noisy background trajectories (see Figure 10). We compute a separate BoVW histogram that corresponds to the area enclosed by the bounding box, which is used to augment the original BoVW vector. The mask shown in the Figure 10 is the output of the semantic segmentation algorithm introduced in [9] and applied on Task-6a data. Following the same experimental protocol as in the rest of the experiments described in this section, we obtained additional improvements on the related Task-6.a scenario, as shown in Table 4 below. Given the simplicity of the employed approach, results show remarkable improvement. A holistic approach for gesture recognition, such as dense trajectories, can greatly benefit from exploiting mid-level information to remove background noise. Our current research plans include more efficient exploitation of the rich information contained in the output of the semantic segmentation pipeline.

Feat. Descr.	GR	SS+GR	Impr. (%)
traject.	38.8	42.1	8.59
HOG	46.0	46.7	1.47
HOF	51.5	56.3	9.33
MBH	57.0	63.9	12.18
combined	62.4	65.6	5.22

Table 4: Average classification accuracy (%) over all 8 patients using our baseline method (first column) and employing the foreground-background mask (second column). Results show a consistent improvement (third column) over multiple feature descriptors. Results are obtained using the BoW encoding.

4.2 On-line processing

Towards the realisation of a user interface that enables elderly subjects interact with the robotic platform, we have implemented an on-line version of our system that performs

continuous gesture recognition. ROS is employed to provide the main software layer for interprocess communication.

Our overall system comprises two separate sub-systems: (a) the activity detector (*AD node*) that performs temporal localization of segments that contain visual activity and (b) the gesture classifier (*GC node*) that assigns each detected activity segment into a class. The AD node processes the RGB stream in a frame-by-frame basis and determines whether there is any visual activity in the scene, based on an activity "score", whose value is thresholded. The GC node is appropriately signaled at the beginning and the end of the activity segments. Small segments of activity are rejected to ensure that small spontaneous movements of the user are not processed.

The Gesture Classifier node processes video segments and assigns them to one of the pre-defined categories using the classification pipeline described previously. To reduce the total processing time we downsample RGB frames both in space and time by a factor of 2. The GC node caches input frames from the camera. When the appropriate signal is received, feature extraction begins immediately, starting from the indicated frame's timestamp. When the activity segment ends features are extracted from the remaining frames of the activity segment, the classification pipeline continues. The robot reacts to the recognized gestural command by either providing audio responses or moving in order to assist the elderly user. The system has been integrated in the robotic platform and used by actual elderly patients. It operates approximately at 20 fps on an Intel i7 CPU.

5 Conclusions

In this work we have examined multiple directions on an assistive human-robot interaction task, from the visual processing point of view. We have provided rich algorithmic and experimental evidence on how one may estimate joint angles of the elderly subjects with the given challenging setup and have signified the importance of the sensor localization that includes a full view of the subject. Additionally, we have shown that the adopted sparse coding based denoising approach increases performance. Another application concerns the automatic recognition of gesture commands, as well as a practical integrated system that online processes streaming video data, and accurately recognizes the commands with real-time performance. The systems are currently under validation studies conducted with elderly subjects in geriatric clinics, opening new perspectives in the field of robotic vision for assistive living. Ongoing and future plans involve the more deep integration of the multiple directions presented in this paper, by incorporating the information of the estimated pose and joint angles for gesture recognition, as well as the integration of gesture recognition in an overall network that jointly estimates the pose, and the higher level gesture concepts. Finally, further exploiting the computer vision interface applications with geriatric clinician experts can lead to a deeper understanding on how we should advance and jointly experiment with our approaches so that it would be for the best interest of the interdisciplinary research communities and above all for the benefit of the elderly population.

References

1. OECD: Elderly population (indicator) (2016)
2. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in Neural Information Processing Systems. (2014) 568–576
3. Niebles, J.C., Fei-Fei, L.: A hierarchical model of shape and appearance for human action classification. In: Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on, IEEE (2007) 1–8
4. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. International journal of computer vision **79**(3) (2008) 299–318
5. Marszalek, M., Laptev, I., Schmid, C.: Actions in context. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009), IEEE (2009) 2929–2936
6. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, IEEE (2011) 3169–3176
7. Wang, H., Schmid, C.: Action recognition with improved trajectories. In: Computer Vision (ICCV), 2013 IEEE International Conference on, IEEE (2013) 3551–3558
8. Fanello, S.R., Gori, I., Metta, G., Odone, F.: Keep it simple and sparse: Real-time action recognition. Journal of Machine Learning Research **14** (2013) 2617–2640
9. Chandra, S., Tsogkas, S., Kokkinos, I.: Accurate human-limb segmentation in rgb-d images for intelligent mobility assistance robots. In: Proceedings of the IEEE International Conference on Computer Vision Workshops. (2015) 44–50
10. Fotinea, E.S., Efthimiou, E., Dimou, A.L., Goulas, T., Karioris, P., Peer, A., Maragos, P., Tzafestas, C., Kokkinos, I., Hauer, K., et al.: Data acquisition towards defining a multimodal interaction model for human–assistive robot communication. In: International Conference on Universal Access in Human-Computer Interaction, Springer (2014) 613–624
11. Escalera, S., Gonzàlez, J., Baró, X., Reyes, M., Guyon, I., Athitsos, V., Escalante, H., Sigal, L., Argyros, A., Sminchisescu, C., Bowden, R., Sclaroff, S.: Chalearn multi-modal gesture recognition 2013: grand challenge and workshop summary. In: Proc. of the 15th ACM on Int'l Conf. on Multimodal Interaction, ACM (2013) 365–368
12. Sun, Y., Wang, X., Tang, X.: Deep convolutional network cascade for facial point detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013) 3476–3483
13. Toshev, A., Szegedy, C.: Deeppose: Human pose estimation via deep neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2014) 1653–1660
14. Tompson, J., Jain, A., LeCun, Y., Bregler, C.: Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. In: NIPS. (2014)
15. Wei, S.E., Ramakrishna, V., Kanade, T., Sheikh, Y.: Convolutional pose machines. arXiv preprint arXiv:1602.00134 (2016)
16. Belagiannis, V., Zisserman, A.: Recurrent human pose estimation. arXiv preprint arXiv:1605.02914 (2016)
17. Lifshitz, I., Fetaya, E., Ullman, S.: Human pose estimation using deep consensus voting. arXiv preprint arXiv:1603.08212 (2016)
18. Haque, A., Peng, B., Luo, Z., Alahi, A., Yeung, S., Fei-Fei, L.: Viewpoint invariant 3d human pose estimation with recurrent error feedback. arXiv preprint arXiv:1603.07076 (2016)
19. Ramakrishna, V., Munoz, D., Hebert, M., Bagnell, J.A., Sheikh, Y.: Pose machines: Articulated pose estimation via inference machines. In: European Conference on Computer Vision, Springer (2014) 33–47

20. Carreira, J., Agrawal, P., Fragkiadaki, K., Malik, J.: Human pose estimation with iterative error feedback. arXiv preprint arXiv:1507.06550 (2015)
21. Insafutdinov, E., Pishchulin, L., Andres, B., Andriluka, M., Schiele, B.: Deepercut: A deeper, stronger, and faster multi-person pose estimation model. arXiv preprint arXiv:1605.03170 (2016)
22. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8) (Nov. 1997) 1735–1780
23. Andriluka, M., Pishchulin, L., Gehler, P., Schiele, B.: 2d human pose estimation: New benchmark and state of the art analysis. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2014)
24. Johnson, S., Everingham, M.: Clustered pose and nonlinear appearance models for human pose estimation. In: Proceedings of the British Machine Vision Conference. (2010) doi:10.5244/C.24.12.
25. Wang, H., Klaser, A., Schmid, C., Liu, C.L.: Action recognition by dense trajectories. In: 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (Jun. 2011) 3169–3176
26. Wang, H., Ullah, M.M., Klser, A., Laptev, I., Schmid, C.: Evaluation of local spatio-temporal features for action recognition. In: University of Central Florida, U.S.A. (2009)
27. Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: IEEE Conference on Computer Vision and Pattern Recognition, 2008. CVPR 2008. (Jun. 2008) 1–8
28. Rodomagoulakis, I., Kardaris, N., Pitsikalis, V., Arvanitakis, A., Maragos, P.: A multimedia gesture dataset for human-robot communication: Acquisition, tools and recognition results. In: IEEE International Conference on Image Processing (ICIP-16). (sep 2016)