Scalable Distributed Sensor Network for Contact-less Gait Analysis
A Marker-less, Sensor-based System for Steering Rehabilitation Measures

Heinz-Josef Eikerling, Michael Uelschen and Florian Lutterbeck
University of Applied Sciences Osnabrueck, Faculty for Engineering and Computer Science, Barbarastr. 16, D-49076 Osnabrück, Germany

Keywords: Sensor Network, Distributed System, Wireless Transmission, Gait Analysis, Motion Capturing, Synchronisation.

Abstract: We describe the steps taken to evolve a system which enables gait analysis of persons recovering from illnesses affecting gait. The starting point is a centralized system which is turned into a fully distributed environment. The upgraded system is evaluated in terms of accuracy of the generated data by comparing it with the initial system and a marker-based reference system. Elementary gait metrics are used for these comparisons. Regarding the reference system the deviation (hence error) was found to be below 2%.

1 INTRODUCTION

We describe the steps taken to evolve a system supporting gait analysis of persons recovering from illnesses affecting gait from a centralized to a fully distributed environment.

Through prior work (Uelschen and Eikerling, 2015) it was shown that the initial version of the DYNMETRICS system is capable of monitoring the progress of rehabilitation measures being applied to persons recovering from surgical treatments in terms of general viability, setup and calibration of the sensor system and the precision of the derived gait characteristics: speed, step number and stride length, cadence and center of mass (CoM) displacement.

The DYNMETRICS system consists of a chain of adjacent sensors for recording and analysing human gait. In addition to the elementary temporal-spatial parameters the system recognize all 8 phases of gait according to the RLANC notion (Perry and Burnfield, 2010). This information allows the physician or the physiotherapist to tune the post-operative treatment in order to improve the overall rehabilitation process.

The general applicability of the system was demonstrated through a clinical trial with 54 patients being treated by the orthopaedic department of a rehabilitation centre. Figure 1 illustrates the gait pattern of a 74-year-old female patient with a nail fixation of the right femur. The outcome of DYNMETRICS reveals a shift of the center of mass to the contralateral leg in order to relieve the operated leg.

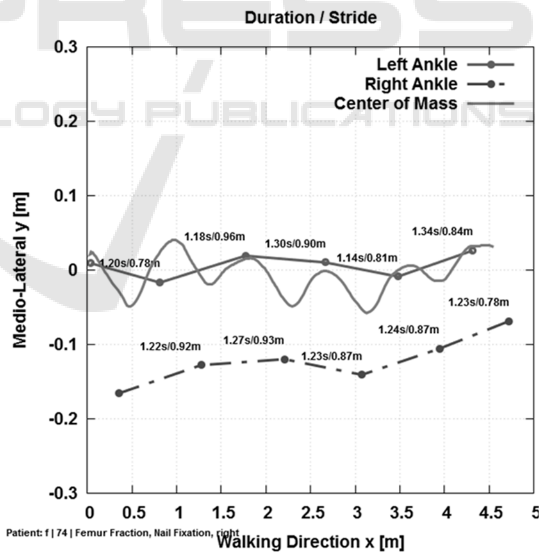

Figure 1: Gait Pattern of Rehabilitation Patient.

2 BACKGROUND

DYNMETRICS v1 features a set of low-cost but nevertheless performant MS Kinect (version 1) sensors and as such constitutes a mobile and marker-less tracking system. However, the initial version of

Eikerling, H-J., Uelschen, M. and Lutterbeck, F.
Scalable Distributed Sensor Network for Contact-less Gait Analysis - A Marker-less, Sensor-based System for Steering Rehabilitation Measures.
DOI: 10.5220/0005707801950202
In *Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2016) - Volume 1: BIODEVICES*, pages 195-202
ISBN: 978-989-758-170-0

the system had some shortcomings which to a certain extent have to be attributed to the Kinect v1:

- The *range* of Kinect v1 is limited.
- *Precision* of the sensor is lower when compared to Kinect v2.
- The v1 sensor is susceptible to *interference* with sun light or light originating from other IR sources.
- The latter applies to the sensor itself: the IR fields of two different Kinect v1 sensors interfere. Hence skeleton tracking can be significantly marred. This also limits the options to set up tracking environments comprising more than 1 sensor. According to own experimentation reported in (Uelschen and Eikerling, 2015) the line-up of sensors each twisted by a 42 degree angle (+/- 3 degrees) with respect to the walking trajectory works best.
- *Scalability* is reduced since a maximum of 4 sensors can be connected to a single machine.
- Certain hardware components (e.g., motor to adjust the inclination) lack *robustness*.

A decision was made to upgrade the initial system which was built on top of the Kinect v1 sensor to Kinect v2.

3 SYSTEM ARCHITECTURE

3.1 Prerequisites & Hardware

A comparison of Kinect v1 & v2 regarding the key features is shown in the table below:

Table 1: Comparison Kinect v1 & v2 features.

Feature	Kinect v1	Kinect v2
RGB camera	640 x 480 @30fps	1920 x 1080 @30 fps
Depth camera	320 x 240	512 x 424
Nominal range	0.4m – 4m	0.5m – 4,5m
Real range	0.8m – 3.5m	1m – 4m
Tracked joints	20	25
Max. no. tracked skeletons	2	6
Hand tracking	No	Yes
Face tracking	[Yes]	Yes
USB	2.0	3.0
Tilt motor	Yes	No
Max. #Kinects per host computer	4	1
Supported OS	Win 7, Win 8 / 8.1	Win 8 / 8.1 (64-Bit)

The sensor version upgrade requires major design modifications and imposes several challenges:

- Kinect v2 implicates much higher *computing* and *networking resources* which can hardly be matched with a single machine. According to the table above, a 64 Bit system is mandatory (M1).
- Thus a fully distributed approach to capturing the sensor feeds has to be implemented (M2).
- Since the sensors have to be deployed to one machine each, for gathering the produced data *synchronisation* issues have to be resolved (M3).
- Due to the reduced interference of Kinect v2 new *geometric arrangements* of the sensor network are thinkable which on the other side require to modify system calibration / configuration procedures (M4).

With respect to M1, a host machine with a 64 Bit quad core processor and 8 GB of main memory running Windows 8 is recommended by Microsoft. The Kinect SDK makes heavily use of parallel computations which is done on a graphics board supporting at least DirectX v11. The official requirements concerning the computing requirements are a bit fuzzy, since tasks are swapped between CPU and GPU and vice versa. According to own experimentation a dual core system with 4 GB of memory was found to serve the purpose. Factually an Intel i5 CPU @ 2.0…3.0 GHz processor with on-board HD 4600 graphics constitutes somewhat a lower bound. The distance between sensor and host machine is limited by the USB 3 connection, i.e. 11m.

Figure 2: Sensor mounted on mini PC via clamp.

In a feasibility study we checked the CPU load (min., max., mean) of the sensor nodes and the server for the configuration described above. As can be seen in Table 2, performance can be fulfilled by the proposed configuration.

Table 2: CPU load for v2 server and sensor nodes.

CPU load	Server	Node #1	Node #2	Node #3
Minimum	0.7%	10.0%	7.3%	10.5%
Maximum	29.0%	0.5%	33.0%	25.1%
Average	3.2%	14.6%	12.8%	15.3%

3.2 Design

In order to implement modifications M1 and M2 we configured a sensor node comprising the Kinect v2 and the host machine consisting of a mini PC running the raw data processing. In order to keep the effort for wiring the components as low as possible, the nodes transmit the collected tracking via wireless links, thus realising a Wireless Sensor Network.

The node as shown in Figure 3 can be mounted on a tripod via a special fixation (see Figure 2) incorporating a metal clamp to alleviate effects of shock (e.g., caused by unintentionally dragging the cables attached to the sensor). This fixation also eases the accurate alignment of the sensors with respect to the horizontal and frontal plane. For the latter it is needed to painstakingly control the tilt angle.

Figure 3: Fixation of Kinect v2 on top of mini PC.

Since the *central* approach (i.e., all sensors directly connected to one central hub) in DYNMETRICS v1 is not feasible any more, different topologies for connecting the nodes have to be studied. Aside from the prevalent models (tree, chain, cluster, flat) we are in favour of a simple star topology in which one dedicated node (*concentrator* node) is responsible for collecting and storing the data being generated by the *sensor* nodes. The transmitted data consists of the position of the joints according to the sensor space (x/y/z coordinates).

The concentrator node stores the collected data and serves it to applications for processing, analysing and for visualising the recorded or live data. Within the analysis the *sensor data fusion* is done. We follow a central-level (Klein, 2012), cooperative (Brooks and Iyengar, 1998) approach in which the data originating from a pair of adjacent sensors is used to compute the position of a body joint pertinent to the global coordinate space. The implemented procedure is an adaptation of *Umeyama's* algorithm (Umeyama, 1991).

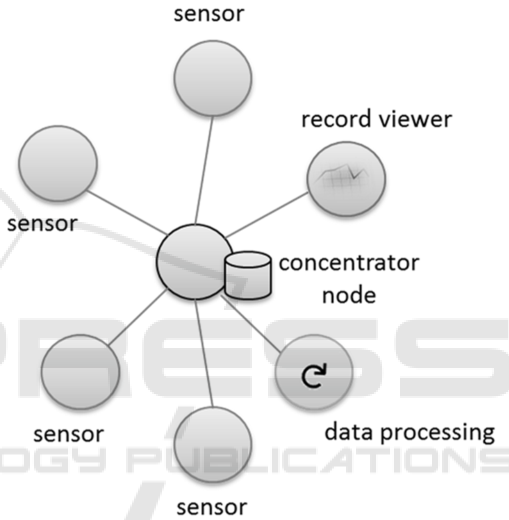

Figure 4: Topology of DYNMETRICS v2 sensor network.

3.3 System Interfaces

Assuming sensor and concentrator node to reside in the same network, a simple client / server interface for registering sensor nodes and for collecting the tracking data was designed. Moreover, through a proprietary protocol implemented in a lean management layer, client software updates can be deployed to the sensor nodes as follows:

1. The concentrator broadcasts a port under which the server software expects the tracking data to be delivered.
2. The sensor node as client connects to the server.
3. Through transmitting the checksum of the installed client software package the need for an update can be detected. The package can be replaced if necessary.
4. The server then asks for the ID of the client

and retrieves locally stored configuration file of that client / sensor node. Subsequently the configuration commands are sent to the client.

5. Afterwards the sensor transmits tracking data as time-stamped records to the server.

The records are delivered in a binary format of fixed size (b_s = 392 Byte). For a rate of f = 30 captures / s this results in a transmission rate of $w'_{min} = b_s \cdot f$ = 12 kByte/s per sensor. The overall bandwidth requirement in a setup with n sensors is therefore $w_{min}(n) = n \cdot w'_{min}$, e.g. $w_{min}(4)$ = 48 kByte/s. A time-stamp $t_c(i)$ marking the arrival of a record i at the concentrator is stored along with it.

Additional to the tracking data, images (low resolution, approx. 328 kByte per image) are captured by the sensor node at a configurable rate (default: 7 images / s). These images are tentatively stored in the sensor node during tracking. They can be used to check the plausibility of the data and rectify artefacts being detected particularly during the off-line data analysis. Once the recording is stopped, the images are offloaded from the clients to the server. This is done in order to as much as possible discharge the communication channel during the recording. The local time-stamp of the capture precedes the transmitted image.

4 IMPLEMENTATION ISSUES

4.1 Software Modifications

For implementing the above concepts, several modifications had to be applied to the DYNMETRICS v1 software package. In the Kinect SDK 2.0 the tracking data is stored in the body object instead of the skeleton in previous releases. Also the changes (naming and virtual placement on the body) with respect to the sensed joints were implemented. In order to keep the previous version with Kinect v1 sensors functional, the new release of the DYNMETRICS software contains a switch to toggle between a Kinect v1 and v2 mode.

4.2 Wireless Connectivity

The transmission of recorded data between sensor and concentrator node is done via WLAN. Hence we had to deal with the intricacies of wireless communication, i.e. reduced bandwidth plus increased instability of the connection and higher latency.

Special attention had to be paid to the latter. Whereas in an Ethernet LAN the latency is and rather constantly below 5 ms, according values in a WLAN range between 8 and 40 ms. Additionally the Jitter (i.e., the fluctuation of the latency) is much higher. This can deteriorate the precision of the measurements since at a rate of 30 captures / s (capture interval approx. 33 ms) the capture time interval is below the latency. Since otherwise the order of events can become wrong, the sensor time-stamp $t_s(i)$ for record i received by the concentrator node is analyzed and a new correction value (= skew) $\sigma(i)$ is computed and applied to the sensor-generated time-stamp $t_s(i)$, thus yielding a corrected time-stamp $t'_s(i)$. This value is also contained in the record stored at the concentrator node.

In the current setup we feature the 802.11 ac standard running in the 5 GHz band for minimizing interference with other wireless networks.

Figure 5: Setup for comparison v1 & v2.

4.3 Time Stamping & Control

For keeping the timing skew between the sensor and concentrator node clocks within certain boundaries, a mechanism for controlling the clocks is required. It has to be remarked that the drift of the local clock ranges between 25….50 ms / h. We aim at a maximum deviation of 10 ms which is well below the capture time interval.

In order to achieve this we could not use standard time synchronization mechanisms (e.g., NTP) since the time needed for synchronization varies and be up to 2^{13} s ≈ 2 h. Instead we worked out a proprietary mechanism: the concentrator node hosts a time server which is continuously accessed by the sensor nodes with a refresh interval of 5s. The client computes the clock skew $\sigma(i)$ at the i-th interval between contractor node and reference node. The update interval is set to 5s. If for two subsequent skew values $|\sigma(i) - \sigma(i-1)| > 10$ ms applies, an error message is thrown and the tracking data of the current recording is discarded. The local clock of the sensor node is not corrected since the latency at the time of propagation is not known.

We tested the fault rates for different network configurations. The fault rate for a 1 Gbit Ethernet is 0,035% / h. For 802.11 n (2.4 GHz) and 802.11 ac (5 GHz) WLANs the values are 0,5% / h and 0,06% / h respectively. The latter is the value for the preferred network configuration and appears to be acceptable.

5 VALIDATION

5.1 Comparison System V1 Vs. V2

In order to judge the accuracy of the modified system a comparative measurement of the two sensor systems was organized. Both systems were run in parallel by stacking the v1 sensor on top of v2 as shown in Figure 5. The joint positions of the tracked skeletons will later on be corrected with respect to the constant vertical offset. The measurement was done for 3 test subjects. In Table 3 the essential statistics are shown. Due to the higher sensing range the duration and hence the number of the tracked skeletons is noticeably higher for the v2 recordings.

Table 3: DYNMETRICS v1 & v2 tracking statistics.

Person ID	1		2		3	
Kinect Version	v1	v2	v1	v2	v1	v2
Time interval	7009 ms	7473 ms	6287 ms	6737 ms	6736 ms	7075 ms
# Skeletons	300	322	294	318	336	355
Skeletons / second	42.8	43.1	46.8	47.2	49.9	50.1

Afterwards the recorded raw data delivered by the sensors is fused and processed by the gait analysis software. Table 4 accounts for the computed essential metrics: number of detected steps, cadence, average stride (= 2 steps) length, average stride difference. Taking the v1 values as the reference, the deviation (yielding the assumed error for the v2 measurements) ranges between 0.66% and 6.23%. The differences in length were not taken into account here since those values come close to the overall precision of the Kinect sensor which is approx. +/-2% for the stride length measurement as reported in (Uelschen and Eikerling, 2015). Hence the average deviation / error is 2.8%.

Table 4: Kinect v1 & v2 gait metrics.

Person ID	1		2		3	
Kinect Version	v1	v2	v1	v2	v1	v2
# Steps	5	6	6	6	5	5
Cadence	75	76	85	84	67	65
∅ stride length	1.37 m	1.36 m	1.27 m	1.28 m	1.47 m	1.45 m
∅ stride difference	0.11 m	0.06 m	0.05 m	0.10 m	0.11 m	0.07 m
Deviation / error	6.23%		0.66%		1.49%	

Aside from comparing the accuracy with respect to gait metrics we also accounted for the different sensing ranges of v1 and v2 installations. According to prior experimentation a distance of 1.4 m of the sensor nodes was found to be optimal for v1. Starting with a value of 2.30 m for v2 sensor node distances this value was incrementally decreased in 10 cm steps so as to find the maximum distance permitting a faultless tracking. This optimal value was determined to be 2.00 m, thus outperforming the range of v1 by 42%.

5.2 Comparison DYNMETRICS V2 Vs. QualiSys

As mentioned in the introduction, conceptually the precision of the marker-less tracking systems is lower than that of marker-based systems. The advantage of marker-based systems comes with the drawback of a much higher setup time for starting a measurement which is needed to tag the subject (10 – 15 min per person). Figure 6 shows the markers used for comparing the DYNMETRICS v2 with the QualiSys tracking system.

For the comparison 4 subjects were examined and for each subject 5 to 7 runs were recorded (see Table 5). As can be seen in the table the range of DYNMETRICS (DM) as indicated by the average number of detected steps is higher when compared to QualiSys (QS).

Table 5: DYNMETRICS vs. QualiSys test configurations.

	# Records	# Steps	
		DM	QS
Person 1	7	7,7	4,9
Person 2	5	7,0	2,6
Person 3	5	6,4	4,2
Person 4	5	6,6	4,4

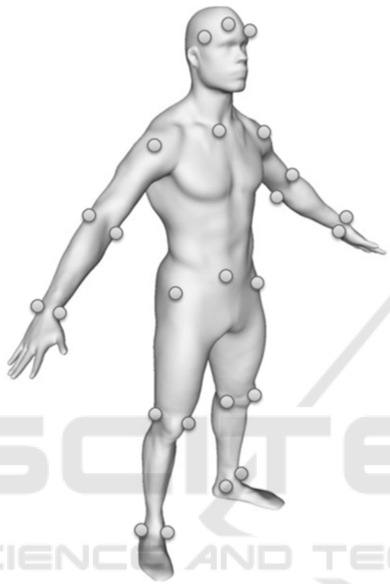

Figure 6: Markers used for QualiSys tracking.

Table 6 compares the computed essential gait metrics cadence and stride length as mean values for DYNMETRICS and QualiSys. As can be derived from the values, the average difference for stride length and cadence is 1.94% and 3.16% respectively. The latter can be mainly attributed to the calculated cadence of person 2 measured via QualiSys.

Table 6: DYNMETRICS vs. QualiSys computed metrics.

	∅ Cadence		∅ Stride	
	DM	QS	DM	QS
Person 1	104.4	107.4	1.34	1.36
Person 2	99.0	103.7	1.57	1.63
Person 3	93.4	97.1	1.38	1.41
Person 4	104.9	106.0	1.38	1.38
∅ Deviation	3.16%		1.94%	

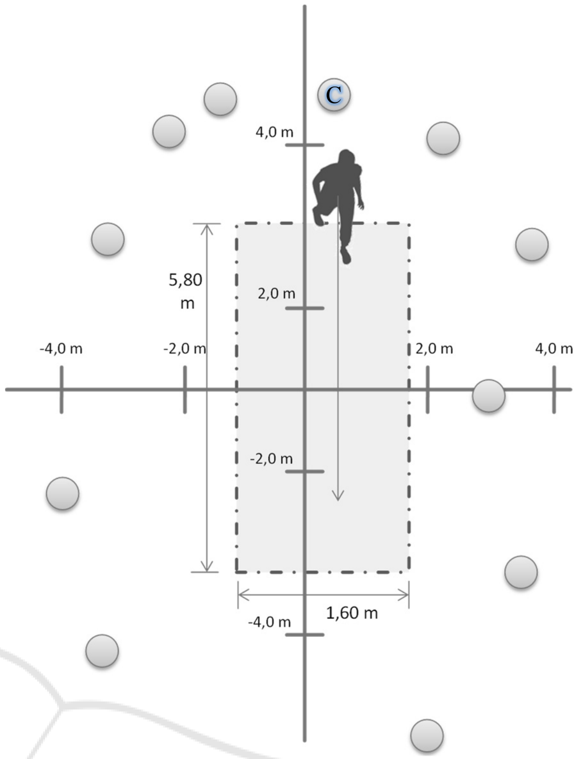

Figure 7: Placement of QualiSys sensors. Dedicated camera for taking control images (c) is marked.

Assuming a linear correlation between DM and QS concerning the examined metrics, we can also compute the *Pearson product-moment correlation coefficient* (Samuels, 2015):

$$r_c = \frac{\sum_{i=1}^{n}(c_{DM,i} - \overline{c_{DM}}) \cdot (c_{QS,i} - \overline{c_{QS}})}{\sqrt{\sum_{i=1}^{n}(c_{DM,i} - \overline{c_{DM}})^2} \cdot \sqrt{\sum_{i=1}^{n}(c_{QS,i} - \overline{c_{QS}})^2}}$$

, where $\overline{c_{DM}}$ and $\overline{c_{QS}}$ are the mean values for the cadence for all subjects and $c_{DM,i}$ and $c_{QS,i}$ are the recorded average cadence values for person i w.r.t. to the DYNMETRICS and QualiSys measurements. Similarly, we can compute the correlation coefficient for stride:

$$r_s = \frac{\sum_{i=1}^{n}(s_{DM,i} - \overline{s_{DM}}) \cdot (s_{QS,i} - \overline{s_{QS}})}{\sqrt{\sum_{i=1}^{n}(s_{DM,i} - \overline{s_{DM}})^2} \cdot \sqrt{\sum_{i=1}^{n}(s_{QS,i} - \overline{s_{QS}})^2}}$$

As a result, we get the values for $r_c = 0.968$ and $r_s = 0.991$.

5.3 Discussion

As can be seen in the previous chapter, the deviation of the elementary metrics determined by the upgraded system is below 4% for all metrics averaging 2.5% when compared to the v1 system and the QualiSys system. Through extensive testing reported in earlier work, the accuracy of DYNMETRICS v1 was found to be +/-2%. Therefore the deviation roughly equals the error of the reference systems.

The determined correlation coefficients show that there is a close and linear correlation between DYNMETRICS v2 and QualiSys measurements.

Aside from these accuracy considerations there are other differences. As pointed out above, the nominal path length of v2 is approx. 42% longer (2.0 m vs. 1.4 m per sensor) when compared to v1. For the comparison with the QualiSys tracking system the length after cropping was determined to be 7.05 m for DYNMETRICS v2 (see Figure 8) whereas the according value for QualiSys was found to be 5.80 m (see Figure 7). For achieving these lengths 4 and 11 sensors respectively were required.

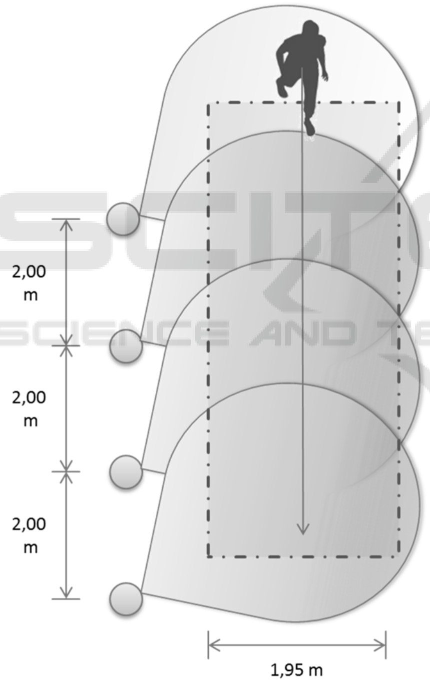

Figure 8: Setup for DYNMETRICS measurement.

During the comparison of DYNMETRICS v2 and QualiSys we also accounted for the time needed to prepare the 4 test subjects for a tracking by means of QualiSys. The time for this individual preparation ranged between 10 to 15 min whereas DYNMETRICS does not require a per-patient preparation time.

Beyond the basic gait metrics we also examined the derived advanced metrics. Figure 9 compares the computed CoM displacement of DYNMETRICS v2 and QualiSys for a data sample. Both periods of the recorded walking pattern are rather identical. The amplitude varies slightly due to the fact that the positions of the joints recognized by the Kinect sensor differ from the respective marker positions utilized by the QualiSys system. This particularly applies to the upper limb joints.

As a result of this the calculations of the center of mass for the two systems deviate from each other. This is also caused by our amendable setup in which

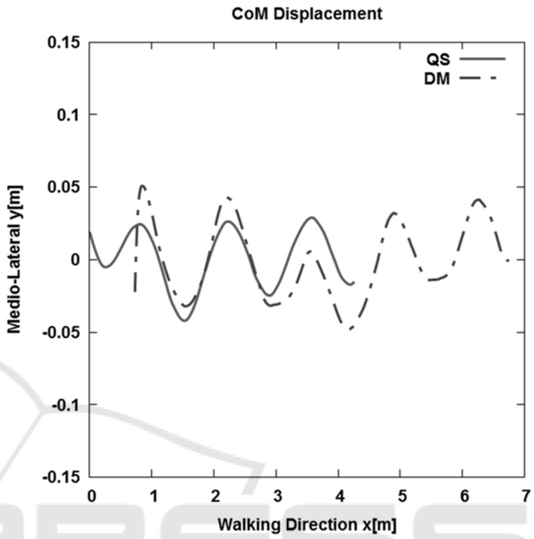

Figure 9: Comparison of CoM displacements.

the DYNMETRICS v2 system stretches a significantly longer tracking area. In an improved configuration the QualiSys tracking area would be prolonged by at least 2 meters and the overlapping area would be 6 meters in order to gain meaningful data.

6 SUMMARY

Though similar systems using Kinect v1 were described elsewhere, see e.g. (Gabel *et al.*, 2012), (Saiyi Li *et al.*, 2014) & (Qiu, J. W. et al., 2014), through this work we wanted to answer two pending questions pertaining to the use of marker-less sensor systems for gait analysis: (i) are such marker-less systems (DYNMETRICS and alike) in general applicable? and (ii) what precision can be achieved?

The results presented in this paper give rise to affirm the statement in the first question. With the applied modifications the robustness and the scalability could be increased without deteriorating precision. In response to the second question we found that the error with respect to a marker-based system can be assumed to be 2% concerning basic gait metrics. This is at least outweighed by the downside of marker-based systems regarding the overhead for setup and handling.

7 OUTLOOK

Though we have shown the general feasibility of our implementation of a marker-less tracking system, we also pointed to its limitation concerning precision and accuracy which to our mind is intrinsic can hardly be changed.

Future work will focus on including other metrics (partly derived out of the basic ones presented here) as described in (Perry and Burnfield, 2010) and the consideration of other bother parts impacting and characterizing gait.

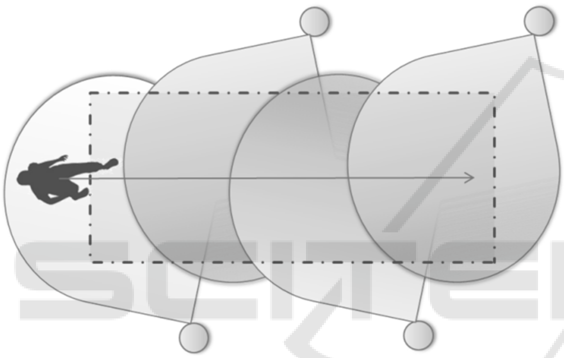

Figure 10: Alternative setup for full body tracking.

For instance arm swinging is rather symptomatic for a person (Meyns *et al.*, 2013). However, determining the exact positioning of the arms using a marker-less tracking system is rather challenging. It has to be particularly ensured that all segments of the arm are visible to the sensor system during the recording. Hence the geometrical arrangement needs to be revised such that sensors are deployed on both sides of the walking trajectory (see Figure 10). Due to interferences this turned out to be impossible with the Kinect v1.

With the new sensor release and the concepts presented in this paper new opportunities in this regard are looming, although essential parts of the software will have to revised: since the arms and legs of the tracked person may occasionally not visible to one particular sensor, the lacking data will have to be provided by one of the sensors in juxtaposition. Hence a revised fusion algorithm is demanded.

ACKNOWLEDGEMENTS

This work was supported by a grant of University of Applied Sciences Osnabrück.

REFERENCES

Brooks, R.R. and Iyengar, S.S. (1998), *Multi-sensor fusion: Fundamentals and applications with software*, Prentice Hall, Upper Saddle River, NJ.

Gabel, M., Gilad-Bachrach, R., Renshaw, E. and Schuster, A. (2012), “Full body gait analysis with Kinect”, *Conference proceedings ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, Vol. 2012, pp. 1964–1967.

Klein, L.A. (2012), *Sensor and Data Fusion: A Tool for Information Assessment and Decision Making,* 2nd ed., Society of Photo-Optical Instrumentation Engineers (SPIE), Bellingham.

Meyns, P., Bruijn, S.M. and Duysens, J. (2013), “The how and why of arm swing during human walking”, *Gait & Posture*, Vol. 38 No. 4, pp. 555–562.

Perry, J. and Burnfield, J.M. (2010), *Gait analysis: Normal and pathological function,* 2. ed., SLACK, Thorofare, NJ.

Qiu, J. W. et al. (2014), “Continuous Human Location and Posture Tracking by Multiple Depth Sensors”, *IEEE International Conference on, and Green Computing and Communications (GreenCom),*, pp. 155–160.

Saiyi Li, Pathirana, P.N. and Caelli, T. (2014), “Multi-kinect skeleton fusion for physical rehabilitation monitoring”, *Conference proceedings ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, Vol. 2014, pp. 5060–5063.

Samuels, M.L. (2015), *Statistics for the life sciences, global edition,* 5th edition, Pearson Education Limited, [Place of publication not identified].

Uelschen, M. and Eikerling, H.-J. (2015), “A Mobile Sensor System for Gait Analysis supporting the Assessment of Rehabilitation Measures”, September 9–12, 2015, Atlanta, GA, USA.

Umeyama, S. (1991), “Least-squares estimation of transformation parameters between two point patterns”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13 No. 4, pp. 376–380.