Seamlessly Extracting Simple Gait Parameters By Using RGB Camera Mounted On Rollator

3 authors, including:

Cagatay Odabasi
Fraunhofer Institute for Manufacturing Engineering and Automation IPA

Birgit Graf
Fraunhofer Institute for Manufacturing Engineering and Automation IPA

Seamlessly Extracting Simple Gait Parameters By Using RGB Camera Mounted On Rollator

Çağatay Odabaşı, Truong Giang Vo, Birgit Graf
Fraunhofer IPA, Stuttgart

***Abstract*— The demand for gait parameters collection for analysis in medical settings has been increasing in recent years due to the aging population. However, the available devices required for such collection are expensive due to RGBD sensors which should be robust to sunlight and require medical personnel to operate. Moreover, current vision systems generally have an implicit assumption that the user's legs are observed clearly. It is a limiting factor for user's life, because the user may want to wear a long skirt. Hence, in this paper, we propose a novel and low-cost solution for the problem, as a visual add-on for standard rollators which only uses the motion of the feet. This add-on device, without constant attention from medical personnel, can provide important measurements for health monitoring, such as the number of steps, cadence, time spent between each step. The device, which mainly consists of an RGB camera and a computational unit, makes use of the algorithm to capture the dense optical flow of only the forward motion of the feet. From this filtered optical flow, the active pixels are counted and inputed into a peak detector to determine the actual steps which occur. Other parameters would be determined based on the steps detected, consequently. The experimental results showed that the device can capture the gait parameters in real time and within reasonable error rate.**

I. INTRODUCTION

Due to various reasons, the median age in the population has been increasing in recent decades, and the trend will continue in the future [1] [9]. Consequently, there is a growing need for more devices and services for elderly care. One of such devices is the rollator (also called wheeled walker), which can provide aid for elderly people as well as patients with walking difficulties. A great number of elderly people are using rollators as walking aids. Additionally, gait parameters are key information for medical personel to monitor and analyze health conditions of elderly people and patients in general. However, collecting such parameters is costly, due to the requirement of complex systems and standby operators. Furthermore, with currently available equipment, it might affect the patients' quality of life (or even infeasible) to collect the parameters over a long stretch of time. Therefore, we explored the idea of integrating a low-cost device onto commonly used aid like rollator for gait parameter collection.

The intelligent rollator walker is not a new idea. In general, previous works offer navigation, safety, fall prevention, sit-to-stand assistance and health monitoring[10]. In the market, there are some products which extend the help capability of the walker by giving extra torque to wheels. [8] provided a navigation function to user by using the sensors placed in the environment. [14] used a sensor of sonar array to avoid obstacles. Similarly, [5] used IR sensors to avoid obstacles so that the falls are avoided. [6] proposed to use torque sensors to prevent falls of elderly people. The examples can be populated, but our focus in this paper is health monitoring using visual sensors.

In literature, there have been works implementing gait analysis on rollator. [2] used laser sensor to track leg movements by using different type of trackers. [3] again used a laser sensor to observe the body for fall detection. Gait analysis was also done by using the skeletal model extracted

Fig. 1. A rollator with propose add-on camera

The authors are with Fraunhofer IPA, 70569 Stuttgart, Germany cagatay.odabasi@ipa.fraunhofer.de

from laser sensor data. [15] [7] [13] used RGBD sensors mounted on rollators to extract gait parameters. There are also some approaches which only used depth information of the feet to estimate gait parameters[12] [11]. In general, these approaches relied on segmentation of feet. However, while they could extract high level features such as feet position and orientation, the methods either used highly complex systems, or restrictive equipements in term of cost (laser scanner) or in term of robustness (indoor sensors like RGBD cameras).

As aforementioned, this paper proposes a novel device, which can be attached on majority of rollators in the market. This device, shown in Figure 1, is independent and allows patients to use the rollators as usual, without any interferences. Our system can monitor and generate simple, but not trivial parameters which are cadence, time spent between each step, total time spent on rollator and total number of steps. The advantages of our system can be grouped into four:

- Cheap - Just an RGB camera and a proccessing unit like Raspberry Pi.
- Simple - Can be easily mounted on the rollator.
- Novel - To the best of our knowledge, there is no such system with such simplicity.
- Independent - It does not need a special staff to operate it and can be used both indoor and outdoor. Elderly people can just use it as they use their rollators.

It should be mentioned that the proposed device would not replace clinical grade gait analysis, due to the fact that our system cannot collect body parts information such as speed of leg or motion of hip, etc. Instead, this device should be able to monitor simple parameters of patients during their daily life actitives, and such parameters can be further analyzed by professional medical personel afterward.

The paper is organized as follows. First, we explain the methodology and error calculation in the Section II. Then, the results are given to user with extended discussions in Section III. Lastly, the paper is concluded briefly in Section IV.

II. METHODOLOGY

The idea is to use dense optical flow information to infer the steps, because it is expected that a camera looking towards floor and towards feet can capture the steps easily. Let's call incoming gray scale frames at time k as $I^k \in R^2$. Let's also define a function $f_o(I^k, I^{k-1}) : R^2 \rightarrow R^2$ which calculates the dense optical flow. So, its input is I^k and I^{k-1} and the output is dense optical flow[4] D^k. Also, it is important that it is defined in polar coordinates.

D^k contains flow of every directions; however, we just need the direction towards the camera, because the only object which makes forward motion will be the active foot. Here, we define a filter function $f_f(r, \theta) : R^2 \rightarrow R^2$ which only allows one direction of flow. So, it applies the filtering operations below to each element of D^k.

$$f_f(r, \theta) = \begin{cases} if \ \|\theta - \theta_{forward}\| < \epsilon, & r = r \\ otherwise, & r = 0 \end{cases} \tag{1}$$

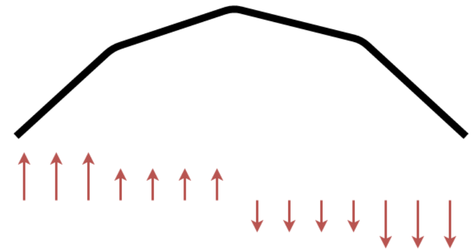

Fig. 2. Example of a peak. The black lines represents a toy signal and the red arrows are representing the first derivative of this signal. At peak point, the derivative changes its sign from positive to negative. It can be seen from the direction of arrow. Downward arrows represent negative values and upwards ones represent positive values.

where ϵ is the small neighborhood around the $\theta_{forward}$ which represents the angle lies parallel to the feet's forward moving direction.

For the computational efficiency, rather than detecting the locations of active pixels, we just count them with a counting function

$$f_c(D^k) = \sum_{i=0}^{N_c} \sum_{j=0}^{N_r} \begin{cases} if \|D^k(i,j)\| > 0, & 1 \\ otherwise, & 0 \end{cases} \tag{2}$$

where N_c and N_r are number of columns and rows of D^k respectively. So, the output is a scalar for each time instance k. The set of these scalars can be written as $S^k = \{x \in \mathcal{Z} \mid \{f_c(D^0), f_c(D^1), ..., f_c(D^k)\}\}$ which is a set of scalar values. So, it becomes the signal that we need to process to detect the steps.

In our experiments, we observed that during a step, S^k starts increasing, makes a peak and starts decreasing as visualized in Figure 3. So, these peaks are the step locations. To find the peak locations, we use Lucas Hermann Negri's implementation of peak detection algorithm. It simply gets first derivative of the signal and search for some points where the first derivative changes sign from positive to negative. The explanation can be seen in Figure 2. In the figure, the black line represents a toy signal and red arrows represent its derivatives. At peak point, the sign of derivative changes from positive to negative. It also uses minimum height and minimum distance thresholds to eliminate false positive peaks.

The peaks found by the algorithm can be collected in a set $P^j = \{x_j\}$ where $j = 1, 2, 3, .., N$, x_j is the frame number of j^{th} peak and N is the number of peaks found.

The parameters we want to extract are cadence, total number of steps and time between each step.

Cadence C means number of steps taken in 60 seconds. So, we calculate it after the each run of the algorithm by

$$C = \frac{N}{T} * 60 \tag{3}$$

where T is the total time spent during an experiment. Also, it is possible to calculate C in a regular basis, for instance, every 10 steps, but we did not do it in this work. The time

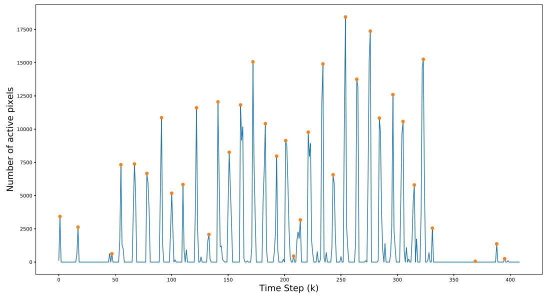

Fig. 3. S^k during an experiment. The blue line is indicating the magnitude of S^k and the orange points are detected peaks. These peaks are considered as step locations.

calculation of CPU can be accepted as perfect, so the only error propagates from the number of steps calculation.

Our system continually measures time of each loop of the algorithm which corresponds to processing time of single frame. After a frame is processed, a new frame is taken from camera and it is being processed.

Major source of error is coming from the error of total number of steps which can be calculated as follows, because we cannot do anything with time calculation of CPU. We depend on hardware for it.

$$error = \frac{\|N - N_a\|}{N_a} \tag{4}$$

where N_a is the actual number of steps during an experiment and N is the number found from algorithm as we stated before. It is manually counted by a human. The results will be given in Section III.

III. Results

One of the advantages of our system is that it can be implemented on a new rollator in minutes. Just stick an RGB camera along with a CPU, install the software and the system is ready to run. Even though, the experiments are done on 'ello from eMovements' with a camera mounted to it which can be seen in Figure 1, it is possible to transfer it to another rollator. The camera is RGBD camera, but Depth channel is not used. Also, we used a laptop with Intel(R) Core(TM) i7-6820HK CPU @ 2.70GHz; however, since it is implemented in ROS, the system can be easily deployed to any type of cheap single board computer. The code is completely written in Python along with Scipy, Numpy, matplotlib and OpenCV libraries.

First, we tested the speed of the algorithm. To do this, time spend between each loop of algorithm is plotted in Figure 4. The average timing is 0.07 seconds which is nearly 14 Hz. Considering that we used a laptop, this number will reduce, if you want to use a small CPU on the rollator. However, it is important that there is no optimization on neither software nor platform. The computationally intensive part of the algorithm is dense optical flow. We also did experiments with optical flow with feature tracking, but in that case the algorithm may not find any feature point, so we may not calculate the motion.

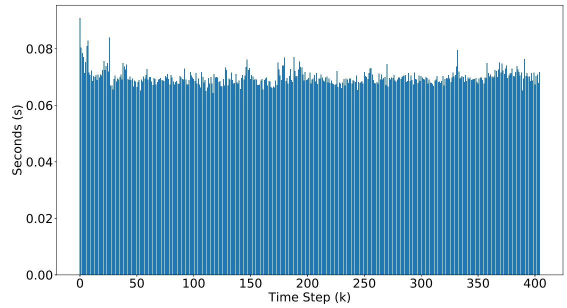

Fig. 4. Each time step (k) corresponds to one loop of algorithm. The values on the y axis show how much time is spent in each k

To better visualize the results, a sequence of RGB images I^k and corresponding processed optical flows D^k are presented in Figure 5. In the figure, when the foot moves forward, the optical flow image becomes activated. The active leg is, in this case, on the left. The forward motion begins in the first frame, so the optical flow image is barely activated. In the rest of the frames, there is a clear forward motion and the optical flow images are highly activated. So, as it is expected the number of activated pixels $\|S^{1,2,3}\| > \|S^0\|$.

The accuracy of total number of steps N is also an important evaluation parameter. We created 9 different experiments in which the person is walking around rooms and corridors of our institute. In these experiments, the users are entering some rooms, are turning with rollators and sometimes stop. The error is calculated as in Equation 4 and it is visualized in Figure 6. Our maximum error is 17.5% and it is achieved at 40 steps. From the figure, it is seen that the error tends to decrease with increasing number of steps. The major error is caused by bad illumination and double peaks. For the bad illumination case like a dark room, the system cannot see the foot, in contrary, for the double peak case, because of

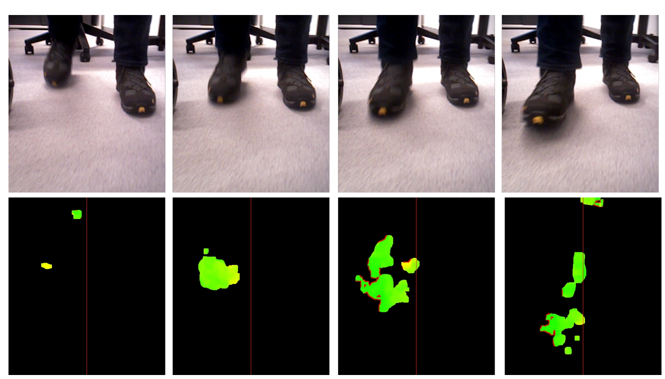

Fig. 5. RGB and processed optical flow of 4 consecutive frames, Top row: RGB frames $I^k, k = 0, 1, 2, 3$ which show a movement of the leg on the left. Bottom row: Corresponding processed optical flow $D^k, k = 0, 1, 2, 3$. The areas which have forward motion towards the camera activated and represented with green color on the images.

the noise in optical flow calculation makes double peaks and the system counts those peaks as two different steps.

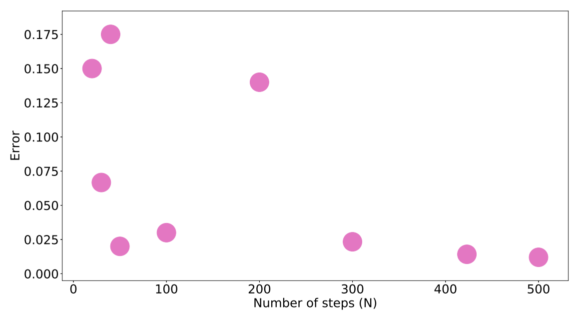

Fig. 6. Error versus Number of Steps

During the experiments, we realized that some of the steps can be seen as two different peak which are close to each other, since the dense optical flow is noisy. Normally, our system should count those peaks as different steps; however, we put a parameter $\sigma_s = 5$ which is a threshold between consecutive peaks. The value of it is chosen as 5, because our system runs at 14 Hz and such $14/5 = 2.8Hz$ step frequency is nearly impossible for an elderly people. However, this parameter should be adjusted to different systems, because the running time would be different in other systems.

We also present the cadence and total number of steps of the 4 different users. The calculated values can be seen in Table I. The cadence values are not deviating too much, because their ages are deviating between 22-31. However, in future, we want to extensively test this method on elderly people.

TABLE I

CADENCE AND TOTAL NUMBER OF STEPS OF FOUR DIFFERENT USERS

	User 1	User 2	User 3	User 4
Cadence	75	73	68	65
Total Number of Steps	30	26	26	35

IV. CONCLUSION

In this paper, we presented a novel rollator visual add-on which can infer cadence, total number of steps, time spent between each step and total usage time of rollator. This device enables cheap monitoring of patients without any additional equipment other than a single board computer and an RGB camera mounted on patient's rollator. In term of visual capability, while RGBD cameras can collect more parameters with higher accuracy, they are in general more expensive, and can only work in limited lighting conditions. On the other hand, simple RGB camera can provide robustness for the system in various illumination settings without suffering noticeable drop in accuracy.
In brief, the proposed device can provide essential parameters for gait analysis and patient monitoring while remains simple and low-cost in term of both hardware and software.

In the future works, we also want to compare our method with RGBD based methods in different environments.

ACKNOWLEDGMENT

This project is supported by SOCRATES project which is an MSCA-ITN-2016 Innovative Training Networks funded by EC under grant agreement No 721619 and also supported by TechBridge program of Fraunhofer-Gesellschaft. Also, I want to thank all of my colleagues in Domestic and Personal Robotics Group in Fraunhofer IPA. Finally, I want to thank to eMovements to provide us their "ello" rollator.

REFERENCES

[1] Joaquin Ballesteros, Cristina Urdiales, Antonio B Martinez, and Marina Tirado. Online estimation of rollator user condition using spatiotemporal gait parameters. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 3180–3185. IEEE, 2016.

[2] Georgia Chalvatzaki, Xanthi S Papageorgiou, Costas S Tzafestas, and Petros Maragos. Comparative experimental validation of human gait tracking algorithms for an intelligent robotic rollator. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*, pages 6026–6031. IEEE, 2017.

[3] C Dune, P Gorce, and JP Merlet. Can smart rollators be used for gait monitoring and fall prevention? In *IEEE/RSJ Int. Conf. on Int. Rob. and Sys*, 2012.

[4] Gunnar Farnebäck. Two-frame motion estimation based on polynomial expansion. *Image analysis*, pages 363–370, 2003.

[5] Thomas Hellström, Olof Lindahl, Tomas Bäcklund, Marcus Karlsson, Peter Hohnloser, Anna Bråndal, Xiaolei Hu, and Per Wester. An intelligent rollator for mobility impaired persons, especially stroke patients. *Journal of medical engineering & technology*, 40(5):270–279, 2016.

[6] Stephan Irgenfried and Heinz Wörn. Motion control and fall prevention for an active walker mobility aid. In *New Advances in Mechanisms, Transmissions and Applications*, pages 157–164. Springer, 2014.

[7] Cyril Joly, Claire Dune, Philippe Gorce, and Patrick Rives. Feet and legs tracking using a smart rollator equipped with a kinect. In *Workshop on" Assistance and Service Robotics in a Human Environment" in conjonction with IEEE/RSJ Int. Conf. on Int. Rob. and Sys.(IROS)*, 2013.

[8] Vladimir Kulyukin, Aliasgar Kutiyanawala, Edmund LoPresti, Judith Matthews, and Richard Simpson. iwalker: Toward a rollator-mounted wayfinding system for the elderly. In *RFID, 2008 IEEE International Conference on*, pages 303–311. IEEE, 2008.

[9] Wolfgang Lutz, Warren Sanderson, and Sergei Scherbov. The coming acceleration of global population ageing. *Nature*, 451(7179):716–719, 2008.

[10] Maria Martins, Cristina Santos, Anselmo Frizera, and Ramón Ceres. A review of the functionalities of smart walkers. *Medical engineering & physics*, 37(10):917–928, 2015.

[11] Martins Martins, Cristina P Santos, Solenne Page, Ludovic Saint-Bauzel, Viviane Pasqui, and Anthony Mézière. Real-time gait assessment with an active depth sensor placed in a walker. In *Rehabilitation Robotics (ICORR), 2015 IEEE International Conference on*, pages 690–695. IEEE, 2015.

[12] Solenne Page, Maria M Martins, Ludovic Saint-Bauzel, Cristina P Santos, and Viviane Pasqui. Fast embedded feet pose estimation based on a depth camera for smart walker. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 4224–4229. IEEE, 2015.

[13] João Paulo, Paulo Peixoto, and Urbano J Nunes. Isr-aiwalker: Robotic walker for intuitive and safe mobility assistance and gait analysis. *IEEE Transactions on Human-Machine Systems*, 2017.

[14] Matthew Spenko, Haoyong Yu, and Steven Dubowsky. Robotic personal aids for mobility and monitoring for the elderly. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 14(3):344–351, 2006.

[15] He Zhang and Cang Ye. An rgb-d camera based walking pattern detection method for smart rollators. In *International Symposium on Visual Computing*, pages 624–633. Springer, 2015.