Computer Vision Algorithms to Drive Smart Wheelchairs

Malik Haddad[1], David Sanders[2], Giles Tewkesbury[2], Martin langner[3], Sanar Muhyaddin[4] and Mohammed Ibrahim[5]
[1]*Northeastern University – London, St. Katharine's Way, London, U.K.*
[2]*Faculty of Technology, University of Portsmouth, Anglesea Road, Portsmouth, U.K.*
[3]*Chailey Heritage Foundation, North Chailey, Lewes, U.K.*
[4]*North Wales Business School, Wrexham Glyndwr University, Wrexham, U.K.*
[5]*Ministry of Communications, Abo Nawas Street, Bagdad, Iraq*

Keywords: Computer Vision, Powered Wheelchair, Assistive Technology.

Abstract: This paper presents a novel approach to driving smart wheelchairs using Computer Vision algorithms. When a user makes a movement, the new approach identifies that movement and utilises it to control a smart wheelchair. An electronic circuit is created to connect a camera, a set of relays and a microcomputer. A programme was created using Python programming language. The program detects the movement of the user. Three algorithms for Computer-Vision algorithms are used: Background Subtraction, Python Imaging Library and Open Source Computer Vision (OpenCV) algorithm. The camera was pointed towards the user a body part used for operating the smart wheelchair. The new approach will detect and identify the movement, the programme will analyse the movement and control the smart wheelchair accordingly. Two User Interfaces were built: a simple User Interface to control the architecture and a technical interface to adjust sensitivity, movement detection settings and operation mode. Testing revealed that OpenCV produced the highest sensitivity and accuracy compared to the other algorithms considered in this paper. The new approach effectively identified voluntary movements and interpreted movements to commands used to drive a smart wheelchair. Future clinical tests will be performed at Chailey Heritage Foundation.

1 INTRODUCTION

A novel approach for operating a smart wheelchair using an image processing algorithm and a Raspberry Pi is presented. The work presented is part of research conducted by the authors at the University of Portsmouth and Chailey Heritage Foundation funded by the Engineering and Physical Sciences Council (EPSRC) (Sanders and Gegov, 20180. The main aim of the research is to apply AI techniques to powered mobility to improve the quality of life of powered mobility users.

Around 15% of the world's population has been experiencing some type of disability with 2 to 4% of the population with disability bieng diagnosed with major problems in mobility (Haddad and Sanders, 2020). Population ageing and the spread of chronic disease have increased these numbers (Haddad and Sanders, 2020; Krops et al., 2018). The type of disability is shifting from mostly physical to a more complex mix of physical/cognitive disabilities. New systems, transducers and controllers are required to address that shift. People with impairment often had poorer quality of life than others (Bos et al., 2019). New Smart input devices are required that use the dynamic movement of body-parts using new contactless transducers to consider users' level of functionality rather than the type of disability and determine user intentions.

George Klein created the first powered wheelchair in collaboration with the National Research Council of Canada to help wounded users during the Second World War (Frank and De Souza, 2018). Since then, powered mobility often became a preferred option for people with disability (Frank and De Souza, 2018). Many researchers worked on improving navigation and steering of powered mobility by creating new systems. Sanders et al. (2010; 2021a) used a sensor structure to control wheelchair-veer and enhance driving. Many researchers used zero-forced sensing switches to operate powered wheelchairs (Haddad et al., 2021a). Sanders and Bausch (2015) used an expert

Haddad, M., Sanders, D., Tewkesbury, G., langner, M., Muhyaddin, S. and Ibrahim, M.
Computer Vision Algorithms to Drive Smart Wheelchairs.
DOI: 10.5220/0011903100003612
In *Proceedings of the 3rd International Symposium on Automation, Information and Computing (ISAIC 2022)*, pages 80-85
ISBN: 978-989-758-622-4; ISSN: 2975-9463

system that analysed users' hand tremors to improve driving. Sanders (2016) considered self-reliance factors to develop a system that blended control between powered wheelchair drivers and an intelligent ultrasonic sensor system. Haddad et al. (2019; 2020a; 2020b) used readings from ultrasonic sensor arrays as inputs to a Multiple Attribute Decision Making (MADM) system and blended the outcome from the MCDM system with desired input from a user to deliver a collision-free steering direction for a powered wheelchair. Haddad and Sanders (2019) used a MADM method, PROMETHEE II, to propose a safe path. Haddad et al. (2020c; 2020d) utilized microcomputers to develop intelligent Human-Machine Interfaces (HMI) that safely steered powered wheelchairs. Many researchers created intelligent systems to collect drivers' data for analysis (Haddad et al., 2020e; Haddad et al., 2020f; Sanders et al., 2020a; Sanders et al., 2020b; Sanders et al 2021b) and applied AI techniques to powered mobility problems, deep learning to safely steer a powered wheelchair (Haddad and Sanders, 2020), rule-based systems to deliver a safe route for powered wheelchairs (Sanders et al., 2018), intelligent control and Human-Computer Interfaces based on expert systems and ultrasonic sensors (Sanders et al., 2020c, image processing algorithms and facial recognition to identify powered mobility drivers (Tewkesbury, 2021; Haddad et al., 2021b) The system created aimed at enhancing the quality of life of powered wheelchair users and increasing their mobility.

Powered wheelchair users often produced a voluntary movement to activate an input device used to drive their wheelchair. Users often used a joystick or a switch to indicate the desired direction and speed. Examples of other input tools included foot control, head or chin controllers or sip-tubes and lever switches.

Discussions with Occupational Therapists (OTs), carers and helpers at Chailey Heritage Foundation/School showed that some students lacked the ability to provide enough hand movement to use a joystick or a switch and the click noise produced from closing switches disturbed the attention of some of the young wheelchair drivers diagnosed with a cognitive and physical disability.

The approach presented in this paper is considered a new way to operate a smart wheelchair using a minimal amount of limb/thumb movement and zero force sensing. The new approach aimed to detect users' hand/thumb movement and use that movement to operate a powered wheelchair.

2 THE NEW APPROACH

The new approach used an electronic circuit and a Python program to detect movement and operate a powered wheelchair. The circuit connected a camera with a Raspberry Pi and a relay. The camera was directed to the user's body part responsible to generate a voluntary movement.

The program was installed onto the Raspberry Pi and triggered the camera to continuously take images. Three Computer-Vision algorithms were considered. A simple UI with four buttons was created to operate the new system as shown in figure 1. The simple UI was designed to match the level of functionality of potential users who had an intellectual disability. It had a straightforward operation and offered an appropriate match between desired commands and user abilities (Lewis, 0221).

To detect movement, the Start button shown in figure 1 would be pressed. Due to safety concerns, the system would not start detecting movement until the Start button was pressed.

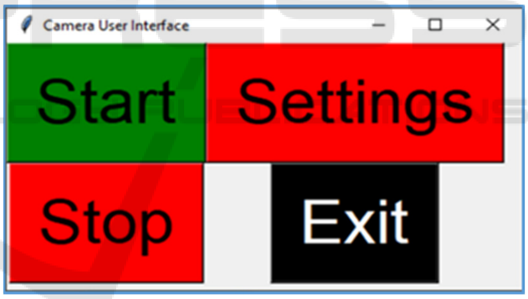

Figure 1: UI used to operate new approach.

2.1 Background-Subtraction Algorithm

The first algorithm considered in the new approach was a Background-subtraction method. The method compared consecutive image frames. A Technical Interface (TI) was created to adjust image and movement detection settings as shown in figure 2.

The UI had 8 track-bars used to modify the image and movement detection settings:

- Display Video: Display detected movement on a display screen.
- Motion Speed: The number of consecutive frames containing motion.

- Sensitivity: The minimum absolute difference for a given pixel in two consecutive frames to be identified as changed.
- Height in Pixels: Height of image in pixels.
- Width in Pixels: Width of image in pixels.
- Frames per Second: The number of captured images per second.
- Detection Area in Pixels: The minimum number of adjacent changed pixels in two consecutive frames required to be identified as changed.
- Switch Activation Time: This track-bar allowed the new approach to function in two different modes: Switch mode and Time-Delay mode. Setting the Switch Activation Time to 0 would allow the system to function in switch mode, where the system would activate the relay when movement was detected and deactivate it when no movement was detected. Setting the Switch Activation Time to a value other than 0 would allow the system to function in Time-Delay mode, where if a movement was detected, the system would activate the relay for the value set by the track-bar (in seconds), then the relay would be deactivated if no further movement was detected.

Once track-bars were set to the desired values and Apply Settings button was clicked, the approach stored all values in a CSV file. These settings were used to detect movement for different users.

When the Start button from the UI was pressed, the approach used the settings stored in the CSV file and Background-subtraction to compare each pixel of the new frame with the previous frame. If a pixel in the new frame was changed by a value more than the Sensitivity value and a number of adjacent pixels identified as changed were larger than the Detection Area, the system would draw a green contour around the changed pixels and would identify the new frame as changed as shown in figure 3. Where figure 3a shows a user's thumb when no movement was detected and figure 3b shows the user's thumb when movement was detected and green contours identify the movement area.

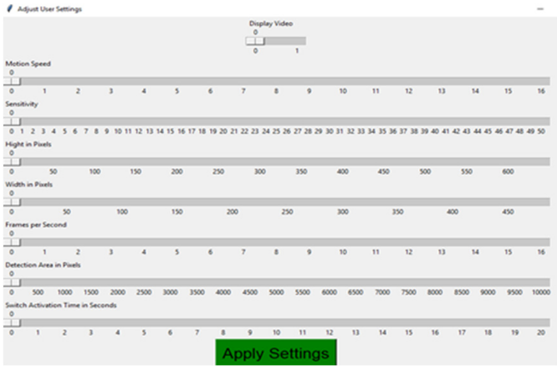

Figure 2: TI used to control Background-subtraction algorithm settings.

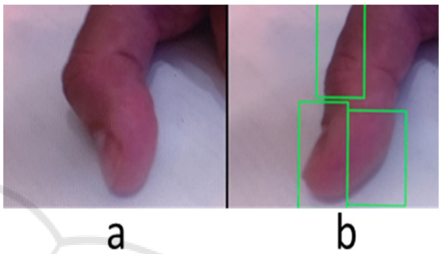

Figure 3: Video feed showing user's thumb; a: no movement detected, b: movement detected.

If the number of consecutive frames identified as changed was higher than the Motion Speed, the program would send logic high voltage to a specific pin. That logic high value would activate the relay.

2.2 Python Imaging Library Algorithm

A second Computer-Vision algorithm was used to improve sensitivity and accuracy. A similar procedure used in the previous Sub-section was followed. Python Imaging Library (PIL) was used instead of the Background-subtraction algorithm.

PIL compared each pixel of the new image with the previous image taken. If the new image was different from the previous image, the approach a high logic voltage to the output pin. The high logic value was used to trigger a relay.

Sensitivity to movement and the amount of movement required to trigger the relay could be adjusted using "Sensitivity" and "Threshold" parameters in the Python program. Sensitivity represented the number of pixels in the new image required to be different in order to detect a movement. The threshold represented the level of difference in the same pixels in two consecutive images to be considered as different.

2.3 Open Source Computer Vision Algorithm

The third attempt to improve sensitivity and accuracy considered an Open-Source-Computer-Vision (Open-CV) Algorithm. A similar procedure used in the previous Sub-section was followed. OpenCV was used to analyse consecutive images captured. Sensitivity and Detection Area values were assigned. OpenCV algorithm compared each pixel in the new image taken against pixels from the previous image. If a pixel in the new image was changed by a value greater than the Sensitivity value and the number of neighbouring pixels marked as changed was greater than the Detection Area, the system identified these pixels by drawing a green contour around them.

A new TI was created to adjust system sensitivity and the amount and duration of movement required to trigger the relay circuit. Figure 4 shows the new TI. The new TI had two buttons: Apply Settings and Exit. Four track-bars used to modify OpenCV settings:

- Duration of Motion: The number of consecutive images having motion.
- Sensitivity Threshold: The minimum difference required in a given pixel in two consecutive images to be marked as changed.
- Detection Area in Pixels: The area of neighbouring pixels marked as changed in two consecutive images.
- Switch Activation Time in Seconds: This track-bar permitted the new approach to work in two different functions: Switch and Time-Delay. Setting this track-bar to 0, the new system would operate as a switch (Switch mode), where the relay would be triggered once a movement was detected and deactivated when no movement was detected. Setting this track-bar to any other value would operate the system in a Time-Delay mode. When a movement was detected, the relay would remain triggered for that value of time (in seconds). Then the relay would be triggered back to off when no movement was detected.

Once track-bars were set to desired values and the Apply Settings button was clicked, the program would store the new values and update values used in the program. Figure 5 shows the new system detecting movement with different settings. Figure 5a shows the new system detecting movement with relatively low sensitivity and a large detection area, figure 5b with medium sensitivity and medium detection area and figure 5c with relatively high sensitivity and small detection area. These settings allowed the new system to be used by different users with different levels of functionality and types of disabilities.

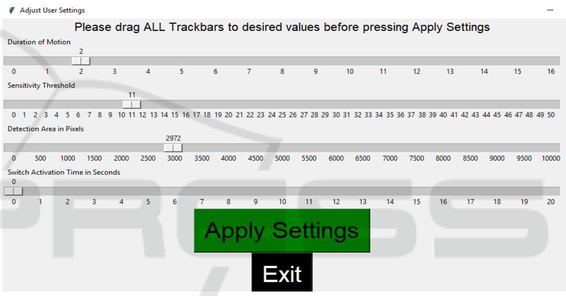

Figure 4: New TI used to modify OpenCV sensitivity and amount and duration of movement required.

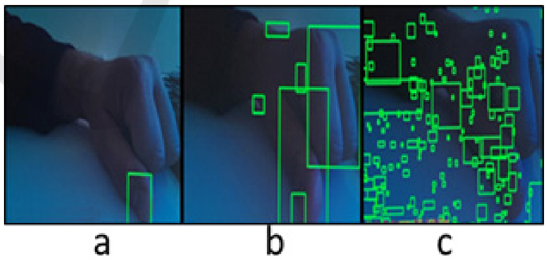

Figure 5: New approach detecting movement with different settings: a: low sensitivity and a large detection area, b: medium sensitivity and medium detection area and c: high sensitivity and small detection area.

3 DISCUSSION AND RESULTS

The work presented in this paper described a new approach to steering a smart wheelchair using zero-force sensing and Computer-Vision algorithms. The new approach used a circuit that connected a camera, a Raspberry Pi and a relay. A Python program was created and installed on Raspberry Pi. The program implemented three Computer-Vision algorithms to conduct image processing and control the camera and the relay.

The camera was directed toward a user's thumb/limb which generated voluntary movements that were used to control a powered wheelchair. A simple UI was used to operate the new approach and a TI was used to adjust image and movement detection settings. The level of sensitivity, speed and amount of movement required could be adjusted using track-bars in the TI. All three algorithms successfully detected all movements in front of the camera and surrounded it by a green contour. The new approach analysed that movement and compared it to stored settings. If the change detected in an image was greater than stored values, the new approach would identify that change and trigger a relay used to control a powered wheelchair.

The new approach was tested practically and it successfully detected movement. The new approach operated in silence and did not generate a clicking sound when operated.

The new approach could be used by multiple users. Specific values for "Sensitivity" and "Threshold" could be allocated for each new user according to their level of functionality using the trackbars in the TI.

4 CONCLUSIONS AND FUTURE WORK

The new approach used a friendly User Interface, detected movements used to operate a smart wheelchair and needed less effort to operate a smart wheelchair.

The authors are planning to upload the program and schematic diagram to an open-access platform. Users will be able to download them free of charge.

Three different Computer-Vision algorithms were compared to improve sensitivity to movement and detection accuracy. Results showed that the new approach provided fast movement detection. Track-bars used in the new approach provided enhanced motion detection settings.

The speed of movement considered in this approach could be used to filter out unwanted movement including user hand tremors or undesired movement generated from driving a powered wheelchair on the unsettled ground.

OpenCV provided more accurate and sensitive image detection when compared to other Computer-Vision algorithms. PIL was easier to install and set up on the Raspberry Pi than other algorithms. The Background-subtraction algorithm approach provided faster movement detection.

Future work will further investigate general shifts in impairment from purely physical to more complex mixes of cognitive/physical. That will be addressed by considering levels of functionality rather than disability. New transducers and controllers that use dynamic inputs rather than static or fixed inputs will be investigated. Different AI techniques will be investigated and combined with the new types of controllers and transducers to interpret what users want to do. Smart Inputs that detect sounds and dynamic movement of body-parts using new contactless transducers and/or brain activity using EEG.

REFERENCES

Sanders, D., Gegov, A., 2018. *Using artificial intelligence to share control of a powered-wheelchair between a wheelchair user and an intelligent sensor system*, EPSRC Project, 2019.

Haddad, M., Sanders, D., 2020. *Deep Learning architecture to assist with steering a powered wheelchair*, IEEE Trans. Neur. Sys. Reh. 28 12 pp 2987-2994.

Krops, L., Hols, D., Folkertsma, N., Dijkstra, P. Geertzen, J., Dekker, R., 2018. *Requirements on a community-based intervention for stimulating physical activity in physically disabled people: a focus group study amongst experts*, Disabil. Rehabil. 40 20 pp 2400-2407.

Bos, I., Wynia, K., Almansa, J., Drost, G., Kremer, B., Kuks, J., 2019. *The prevalence and severity of disease-related disabilities and their impact on quality of life in neuromuscular diseases*, Disabil. Rehabil. 41 14 pp 1676-1681.

Frank, A., De Souza, L., 2018. *Clinical features of children and adults with a muscular dystrophy using powered indoor/outdoor wheelchairs: disease features,* *comorbidities and complicationsof disability*, Disabil. Rehabil. 40 9 pp 1007-1013.

Sanders, D., Langner, M., Tewkesbury, G., 2010. *Improving wheelchair-driving using a sensor system to control wheelchair-veer and variable-switches as an alternative to digital switches or joysticks*, Ind Rob: An int' jnl. 32 2 pp157-167.

Sanders, D., Haddad, M., Tewkesbury, G., 2021a. Intelligent control of a semi-autonomous Assistive Vehicle. In *SAI Intelligent Systems Conference*. Springer.

Haddad, M., Sanders, D., Tewkesbury, G., Langner, M., 2021a. Intelligent User Interface to Control a Powered Wheelchair using Infrared Sensors. In *SAI Intelligent Systems Conference*. Springer.

Sanders, D., Bausch, N., 2015. Improving steering of a powered wheelchair using an expert system to interpret hand tremor. In *the International Conference on Intelligent Robotics and Applications*. University of Portsmouth.

Sanders, D., 2016. Using self-reliance factors to decide how to share control between human powered wheelchair drivers and ultrasonic sensors, IEEE Trans. Neur. Sys. Rehab.25 8 pp 1221-1229.

Haddad, M., Sanders, D., Gegov, A., Hassan, M., Huang, Y., Al-Mosawi, M., 2019. Combining multiple criteria decision making with vector manipulation to decide on the direction for a powered wheelchair. In *SAI Intelligent Systems Conference.* Springer.

Haddad, M., Sanders, D., Langner, M., Ikwan, F., Tewkesbury, G., Gegov, A., 2020a. Steering direction for a powered-wheelchair using the Analytical Hierarchy Process. In *2020 IEEE 10th International Conference on Intelligent Systems-IS*. IEEE.

Haddad, M., Sanders, D., Thabet, M., Gegov, A., Ikwan, F., Omoarebun, P., Tewkesbury, G., Hassan, M., 2020b. Use of the Analytical Hierarchy Process to Determine the Steering Direction for a Powered Wheelchair. In *SAI Intelligent Systems Conference*. Springer.

Haddad, M., Sanders, D., 2019. *Selecting a best compromise direction for a powered wheelchair using PROMETHEE*, IEEE Trans. Neur. Sys. Rehab. 27 2 pp 228-235.

Haddad, M., Sanders, D., Ikwan, F., Thabet, M., Langner, M., Gegov, A., 2020c. Intelligent HMI and control for steering a powered wheelchair using a Raspberry Pi microcomputer. In *2020 IEEE 10th International Conference on Intelligent Systems-IS*. IEEE.

Haddad, M., Sanders, D., Langner, M., Bausch, N., Thabet, M., Gegov, A., Tewkesbury, G., Ikwan, F., 2020d. Intelligent control of the steering for a powered wheelchair using a microcomputer. In *SAI Intelligent Systems Conference*. Springer.

Haddad, M., Sanders, D., Langner, M., Omoarebun, P., Thabet, M., Gegov, A., 2020e. Initial results from using an intelligent system to analyse powered wheelchair users' data. In *the 2020 IEEE 10th International Conference on Intelligent Systems-IS*. IEEE.

Haddad, M., Sanders, D., Langner, M., Thabet, M., Omoarebun, P., Gegov, A., Bausch, N., Giasin, K., 2020f. Intelligent system to analyze data about powered wheelchair drivers. In *SAI Intelligent Systems Conference*. Springer.

Sanders, D., Haddad, M., Tewkesbury, G., Bausch, N., Rogers, I., Huang, Y., 2020a. Analysis of reaction times and time-delays introduced into an intelligent HCI for a smart wheelchair. In *the 2020 IEEE 10th International Conference on Intelligent Systems-IS*. IEEE.

Sanders, D., Haddad, M., Langner, M., Omoarebun, P., Chiverton, J., Hassan, M., Zhou, S., Vatchova, B., 2020b. Introducing time-delays to analyze driver reaction times when using a powered wheelchair. In *SAI Intelligent Systems Conference*. Springer

Sanders, D., Haddad, M., Tewkesbury, G., Gegov, A., Adda, M., 2021b. Are human drivers a liability or an asset?. In *SAI Intelligent Systems Conference*.Springer.

Sanders, D., Gegov, A., Haddad, M., Ikwan, F., Wiltshire, D., Tan, Y. C., 2018. A rule-based expert system to decide on direction and speed of a powered wheelchair. In *SAI Intelligent Systems Conference*. Springer.

Sanders, D., Haddad, M., Tewkesbury, G., Thabet, G., Omoarebun, P., Barker, T., 2020c. Simple expert system for intelligent control and HCI for a wheelchair fitted with ultrasonic sensors. In *the 2020 IEEE 10th International Conference on Intelligent Systems-IS*. IEEE.

Tewkesbury, G., Lifton, S., Haddad, M., Sanders, D., 2021. Facial recognition software for identification of powered wheelchair users. In *SAI Intelligent Systems Conference*. Springer.

Haddad, M., Sanders, D., Langner, M., Tewkesbury, G., 2021b. One Shot Learning Approach to Identify Drivers. In *SAI Intelligent Systems Conference*. Springer.

Lewis, C., 2007. *Simplicity in cognitive assistive technology: a framework and agenda for research*, Universal Access in the Information Society 5 4 pp 351-361.