Frame-Based Slip Detection for an Underactuated Robotic Gripper for Assistance of Users with Disabilities

Marx, Lennard; Pálsdóttir, Asgerdur Arna; Struijk, Lotte N. S. Andreasen

Published in:
Applied Sciences

DOI (link to publication from Publisher):
10.3390/app13158620

Publication date:
2023

Document Version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Marx, L., Pálsdóttir, A. A., & Struijk, L. N. S. A. (2023). Frame-Based Slip Detection for an Underactuated Robotic Gripper for Assistance of Users with Disabilities. *Applied Sciences*, *13*(15), Article 8620. https://doi.org/10.3390/app13158620

applied sciences

Article

Frame-Based Slip Detection for an Underactuated Robotic Gripper for Assistance of Users with Disabilities

Lennard Marx [1,*], Ásgerdur Arna Pálsdóttir [2] and Lotte N. S. Andreasen Struijk [2]

[1] Robotics and Mechatronics, Faculty of Electrical Engineering, Mathematics and Computer Science (EEMCS), University of Twente, 7522 NB Enschede, The Netherlands

[2] Center of Rehabilitation Robotics, Department of Health Science and Technology, Aalborg University, 9220 Aalborg, Denmark; aapa@hst.aau.dk (Á.A.P.); naja@hst.aau.dk (L.N.S.A.S.)

* Correspondence: l.marx@student.utwente.nl

Abstract: Stable grasping is essential for assistive robots aiding individuals with severe motor–sensory disabilities in their everyday lives. Slip detection can prevent unstably grasped objects from falling out of the gripper and causing accidents. Recent research on slip detection focuses on tactile sensing; however, not every robot arm can be equipped with such sensors. In this paper, we propose a slip detection method solely based on data collected by a RealSense D435 Red Green Blue-Depth (RGBd) camera. By utilizing Farneback optical flow (OF) to estimate the motion field of the grasped object relative to the gripper, while also removing potential background noise, the algorithm can perform in a multitude of environments. The algorithm was evaluated on a dataset of 28 daily objects that were lifted 30 times each, resulting in a total of 840 frame sequences. Our proposed slip detection method achieves an accuracy of up to 82.38% and a recall of up to 87.14%, which is comparable to state-of-the-art approaches when only using camera data. When excluding objects for which movements are challenging for vision-based methods to detect, such as untextured or transparent objects, the proposed method performs even better, with an accuracy of up to 87.19% and a recall of up to 95.09%.

Keywords: slip detection; assistive robots; stable grasping; semi-autonomous grasping; tongue-controlled robot

Citation: Marx, L.; Pálsdóttir, Á.A.; Andreasen Struijk, L.N.S. Frame-Based Slip Detection for an Underactuated Robotic Gripper for Assistance of Users with Disabilities. *Appl. Sci.* **2023**, *13*, 8620. https://doi.org/10.3390/app13158620

Academic Editors: Renato Vidoni, Andrea Giusti and Lorenzo Scalera

Received: 6 July 2023
Revised: 21 July 2023
Accepted: 22 July 2023
Published: 26 July 2023

1. Introduction

Reacting to slippage or adjusting unstable grasps of handheld objects is a subconscious task that happens automatically for most individuals [1]. Nondisabled individuals can use their sense of touch and vision to react to slippage or even predict when the held object is likely to slip. Making adjustments stabilizes the grip, and the grasp can be re-evaluated [2]. For individuals with severe sensory–motor disabilities, grasping tasks can be achieved with the help of assistive robot arms, which can be controlled via remaining bodily functions, such as tongue movements [3]. Developing such an interface for these robot manipulators is a complex task [4], and therefore, supporting the control with autonomy of the robot to reduce the complexity is of great importance. While grasping points for grasping tasks of robotic arms can be synthesized well with modern computer vision methods [5], the grasp might still be unstable, causing objects to slip out of the gripper during the pickup or midair during an executed task. Due to the nature of the setup and the user group, there is little to no chance for them to react in time to a slip event. This can be detrimental to the confidence and trust the user develops in the rehabilitative device and ultimately the desire to use the system for everyday aid [6]. Detecting these slip events and having the gripper react fully autonomously to those events can improve the robustness of—and increase the trust of the user in—the system.

In the last two decades, research on slip detection has primarily focused on industrial applications [7]. With the continuous improvement of machine learning methods, these strategies remain a current area of research [8,9]. However, it is essential to note that these methods have mainly been developed and trained on controlled environments for industrial settings. In contrast, in everyday life, assistive devices must cope with various outside influences and noises. As a result, assistive robot arms, although more lightweight and affordable, are often equipped with very few sensors.

The general research on slip detection mostly focuses on imitating the sense of touch of the human hand [7,10]. This sense of touch is replicated through sensors like multiaxial force sensors, sensors detecting microvibrations, thermal microflux sensors, and optical sensors [7,10]. With only visual data and a machine learning approach, Li et al. [8] achieve up to an 80.92% slip detection accuracy when providing the image difference sequence as input to their deep neural network (DNN) and 53.68% when giving the raw image sequence as input. In their work, Li et al. [8] equip a WSG-50 parallel gripper with a GelSight tactile sensor. They propose a method based on a DNN to classify and detect slip. When including the tactile data, the DNN achieves an accuracy of up to 88.03% using the pretrained Inception-V3 model. With the utilization of a CNN-LSTM (convolutional neural network–long short-term memory), Yan et al. [9] achieve up to 67.96% accuracy using only visual data. When including tactile data, they achieve a classification accuracy of up to 88.75%. Zhang et al. [11] reported an 88.03% detection accuracy with combined tactile and visual data. By utilizing interpolation of deformation fields detected by their custom-built sensors, they achieved an accuracy of 97.62%. In a recent study [12] by Gao et al., an accuracy of up to 96.96% was claimed. They performed visuo-tactile fusion using a multiscale temporal convolutional network (MS-TCN) to extract both visual and tactile data. It is noteworthy that all four of these studies used controlled environments with minimal background noise to train and evaluate their models for slip detection. Jiang et al. [13] developed a method for computer vision-guided robot arms to detect and grasp transparent objects. In their work, they also made use of the GelSight sensor and employed a poking strategy to help identify the transparent objects. With their strategy, Jiang et al. [13] achieved a grasping success rate of up to 85.2%.

We want to introduce a robust slip detection method for assistive applications in everyday tasks that makes use of minimal sensory data, is easy to implement, and only uses cheap, commercially available products. At the same time, the proposed method should be robust against noise such as movement in the background surrounding the object, from outside influences or from movements of the robot arm itself. This is crucial, as most current slip detection methods are developed in a controlled environment, which is not representative of the environments faced in everyday life of a user in need of robotic assistance, and thus might not be robust enough. The proposed method tackles these issues by removing background noise of any sorts. Further, we want to compare the proposed method with the performance of the state of the art. Section 2 describes the experimental setup as well as the methods used in the proposed algorithm. In Section 3, the results of the conducted experiment are presented. Section 4 discusses the results, as well as the limitations and potential improvements of the proposed method. The final Section 5 summarizes the work.

2. Methods

2.1. Setup

The JACO 2 (Kinova, Boisbriand, Quebec, Canada), a 6 degrees of freedom (DoF) robot arm, is attached to a motorized wheelchair and equipped with a RealSense d435 Red Green Blue-Depth (RGBd) camera (Intel, Santa Clara, CA, USA), as shown in Figure 1. The robot arm features an underactuated 2-Finger, 4-DoF gripper designed for grasping tasks. Each finger can be opened and closed with a single command, and the angle of the joint between the two links is dependent on the closure of the fingers and cannot be actuated manually. Control of the arm is facilitated through a fully integrated inductive tongue computer interface (TCI) [14]. The TCI provides semi-autonomous control, allowing the user to give inputs with their tongue via an inductive plate placed on the roof of the mouth, as illustrated in Figure 2.

The interface used is multimodal, allowing the user to control the end effector via a virtual joy-stick-like environment and switch modes to semi-autonomously grasp objects. Grasping intent prediction is accomplished by identifying the closest object to the palm-side of the gripper [5]. A grasping point is then determined by approximating the shape of the object with a virtual cylinder. While the user retains control over the grasping process and can signal to grasp or release via the TCI, they also have the ability to interrupt the process at any given time. The remaining path planning and grasping of the object are executed autonomously. Previous experiments revealed instances of slippage occurring during both the initial lifting of the object and while the object was already in-hand. With our proposed method, we aim to react to these slip events using the existing setup, avoiding the need to incorporate additional sensors and complexities into the system. This work aims to evaluate the proposed slip detection method, since there are already publicly available datasets published, we did not conduct experiments of our own, which would involve the TCI. To evaluate the performance of the proposed method, we conducted tests on a dataset consisting of 28 daily objects, each lifted 30 times [9], resulting in a total of 840 frame sequences.

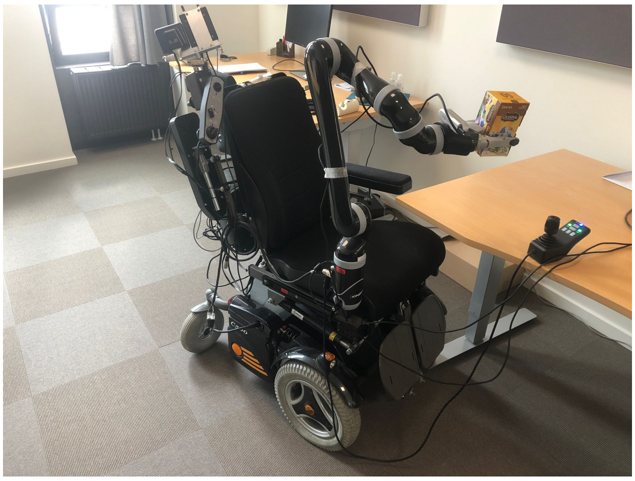

Figure 1. The JACO 2 robot arm attached to a wheelchair. A RealSense D435 Red Green Blue-Depth (RGBd) camera is attached to the end effector, which is holding a tea carton. The robot arm is controlled semi-autonomously by the user via a tongue–computer-interface (TCI).

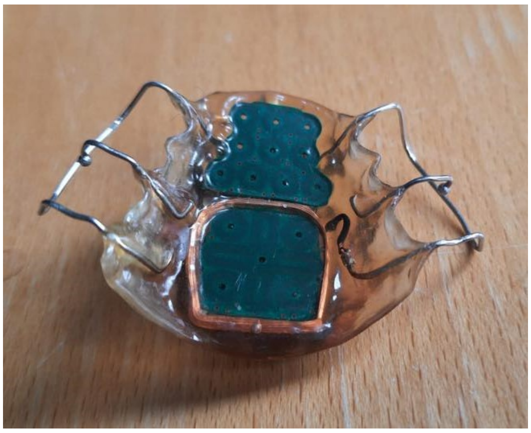

Figure 2. The tongue–computer interface individually fit for each user. Two conductive plates serve as interfaces. One to move the robot arm, the other to act as a menu to switch between manual and semi-autonomous mode. More detail on the TCI can be found in [14].

2.2. Object Recognition

In the presented setup, the RGBd camera was attached to the end effector, ensuring no relative movement of the grasped object was introduced by moving the robot arm. However, movement in the background and the robot arm itself could introduce noise that would negatively impact the performance of any displacement-based approach. To overcome this challenge, the grasped object was detected, effectively removing the background influences. This step was crucial for enhancing the system's robustness against environmental influences. Since immediate object-specific information was not required, a method was developed to approximate the outline of the grasped object, regardless of its size, shape, or texture.

The depth information from the RGBd camera was unable to be calculated accurately at distances closer than 30 cm [15]. As a result, the camera considered every object closer than 30 cm to be at a 0 cm distance. This characteristic was utilized to create a binary mask that removed all pixels at distances greater than 30 cm from the camera, effectively removing the background. To further enhance the method's robustness against background noise at the same distance, the gripper's metadata were used to interpolate the closure of the fingers to the visual equivalent in the raw RGB image. This step removed the background to the left and right of the grasped object. A script was developed to create a cubic spline, correlating the closure of each finger to the horizontal pixel position of the finger's contact point with the grasped object. Since the gripper in use was underactuated, the closure indicated by the gripper's metadata was not solely dependent on the size of the grasped object but also on the grasping force applied. A larger grasping force resulted in a greater closure readout, leading to a reduced estimation of the horizontal size of an object with increased grasping force. Considering the limited possibilities of the setup and no significant advantage in having extremely accurate estimates of the grasping force, the object's weight was used to correct for the larger closure readouts caused by higher grasping forces. This correction was achieved by estimating the grasped object's weight through the moment arms resulting from the joint torques and positions relative to the robot's base.

The remaining noise in the resulting binary mask (Figure 3) was further reduced by employing morphological image manipulation techniques [16]. Initially, the image was dilated to eliminate noise present inside the approximate boundaries of the object. Subsequently, erosion was applied to remove noise found outside the object's boundaries. Finally, another dilation step was performed to restore the mask's size to approximately match the object size. The boundaries of the resulting shapes were then calculated, and the largest of those boundaries was extracted. As these boundaries were found to be significantly noisy in terms of their shape, a decision was made to fit an ellipse to the estimated boundary (Figure 4).

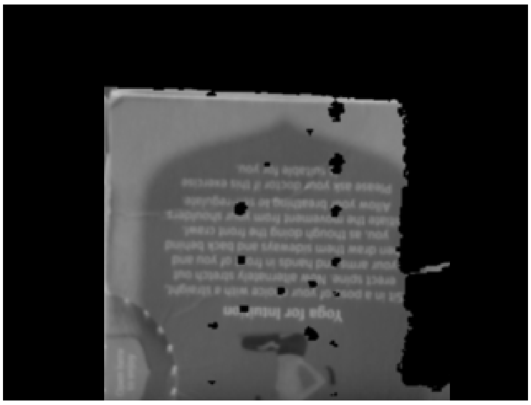

Figure 3. Example of a still noisy grayscale image of an object with the background mostly removed. Here, objects and movements close to the grasped object could cause the boundary of the objects to extend and include some of the background in the boundary estimation.

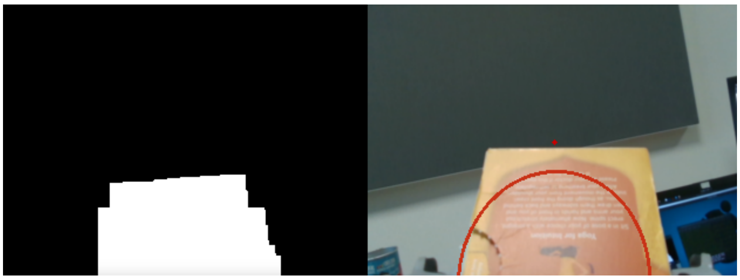

Figure 4. The resulting binary mask (**left**) and the original RGB image with the ellipse estimating its size (**right**). The ellipse results from fitting it to the largest boundary of the binary mask. For this specific setup, since the gripper is at the very bottom of the frame, the assumption can be made that the object is also always at the bottom. Hence, a half-dome shape was used by point rotation of the mask by 180° and fitting the ellipse to it.

The accuracy of the object boundary estimation did not need to be extremely precise for the proposed slip detection method to function effectively; the primary objective was the robust removal of the background. As the fitted ellipse slightly varied in shape for each frame, the resulting ellipse mask was utilized for optical flow (OF) estimation. Both the current frame and the last frame were compared using this ellipse mask to facilitate the comparison of frames. This prevented the changing shape of the mask from influencing the optical flow estimation.

2.3. Slip Detection

2.3.1. Optical Flow

The slip detection process relied on optical flow estimation between two frames, which were masked by the ellipse to approximate the general outline of the object. The magnitude and direction of the flow were then calculated to approximate the motion field of the object. For the optical flow calculations, the open-source computer vision library, OpenCV [17], was utilized. To accommodate less textured objects, the Farneback optical flow algorithm [18] was chosen over alternative methods like Lucas–Kanade [19], which would have been computationally less intensive. The Farneback algorithm is described in more detail in the next paragraph. Feature-based methods work effectively on printed or textured surfaces; however, many everyday objects lack stable features, especially since the exact borders of the object are often not included after the masking. In contrast, for textured objects, feature-based flow estimation is a viable alternative that is computationally less intensive. Nonetheless, the Farneback algorithm for dense optical flow estimation proved to be fast enough to run in near real time. The chosen parameters for the Farneback optical flow are as follows :

- Pyramid Scale: 0.5;
- Level: 3;
- Window Size: 15;
- Iterations: 3;
- Poly n: 5;
- Poly σ: 1.2.

2.3.2. Farneback Optical Flow

To estimate the motion field of a scene from two consecutive images, one approach is to make the assumption that the pixel intensities do not change from one frame to the other (Equation (1)).

$$I(x,y,t) = i(x+dx, y+dy, t+dt) \tag{1}$$

Another assumption made is that neighboring pixels have the same motion, indicating that they belong to the same object. To capture motion at different scales, an image pyramid of downsampled versions of the original image is created. Through polynomial expansion, the algorithm fits a quadratic function to the pixel intensities between the two frames and calculates coefficients that minimize the difference for each layer. The coarser layers aid in estimating the optical flow of the less coarse layers. After upsampling the result to the original image size, the flow direction can be estimated for each pixel, resulting in a good approximation of the actual motion field of the scene. However, this method has some limitations. Changes in lighting can refute the first assumption, as they may cause optical flow to be detected even when there is no actual motion. Conversely, when objects with smooth surfaces are moved (e.g., a ball is rotated), the pixel intensities do not change, and no optical flow is detected, despite actual motion occurring.

2.3.3. Optical Path Occlusion and Remaining Background Noise

To address potential optical path occlusion, a voting system was implemented (Algorithm 1). Pixels within the ellipse that estimates the object dimensions and have a magnitude of zero were excluded from consideration. This step was necessary to enable the voting system to function effectively on minimally textured objects, where large portions of the objects might not exhibit visual changes, even during slip events. These regions needed to be excluded to prevent false positives. Subsequently, the pixels were sorted based on their optical flow (OF) magnitude, and a threshold was set. The majority of the pixels needed to surpass this threshold for a slip event to be qualified as such. The voting system allowed for optical path occlusion by up to 50% of the grasped object's size. Even with such occlusions, the slip detection algorithm continued to work efficiently and still detected slip events accurately (Figure 5).

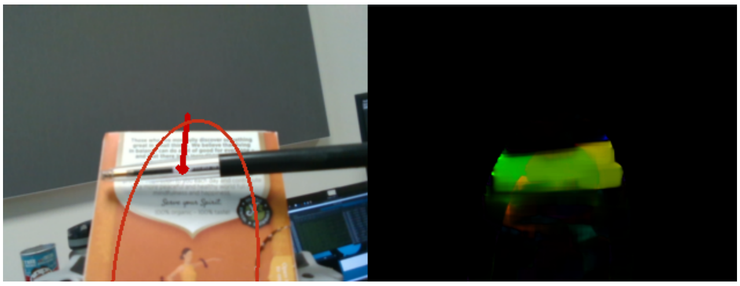

Figure 5. A pencil occluding the optical path of the camera to the object without causing slip to be detected. If an actual slip case would occur, more than 50% of the non-zero-magnitude pixel would show optical flow (OF) above the threshold, triggering slip to be detected even if the obstructing object is still.

There was a trade-off in determining the necessary majority of pixels that must vote for slip detection. Setting a lower percentage made the algorithm more sensitive to occlusions, potentially leading to false positives triggered by occlusions themselves. However, this lower percentage also required less of the object to be unoccluded in order to detect slip.

Algorithm 1 Optical flow detection algorithm

```
$MAJORITY = 0.5$
while True do
    $flow = calcOpticalFlowFarneback$
    $non\_zero\_OF = flow > 0$
    $inv(sort(non\_zero\_OF))$
    $majority = sorted[0 : length * MAJORITY]$
    if $majority[end] > THRESHOLD$ then
        $slip = True$
    else if $majority[end] < THRESHOLD$ then
        $slip = False$
    end if
end while
```

On the other hand, a higher percentage reduced the possibility of false positives caused by occlusions, as a larger portion of the object needed to be unoccluded for slip to be detected. However, this also made slip detection more challenging in cases where the occlusion in front of the object was relatively small.

The approach described has its inherent limitations, which are typical for vision-based methods. One potential solution to address the occlusion problem could involve using multiple cameras from different angles to obtain a more comprehensive view of the scene. However, in the specific use case described, the use of multiple cameras from different angles might be impractical or infeasible.

2.3.4. Dealing with Compliant Objects

One of the major challenges of using visual data and optical flow for slip detection lies in dealing with compliant objects. When grasping compliant objects, the grippers deform the object, leading to additional optical flow, which may trigger false positives in the algorithm. The grippers move only in one dimension relative to the camera, causing the optical flow resulting from grasp-induced deformations to be primarily horizontal, while the optical flow caused by slip (translational and rotational) is mostly vertical (Figure 6).

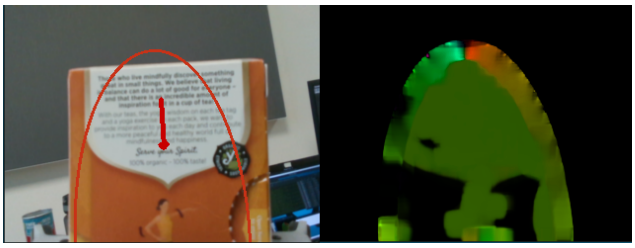

Figure 6. The original image with the approximated image boundaries and slip direction (**left**) and the HSV representation of the OF direction (**right**). The hue represents the direction of the flow, and the value represents the magnitude. A typical slip case: the green color indicates an angle of 90°, which in this case shows the object slipping downwards, which is indicated by the arrow on the left image.

The Farneback optical flow (OF) method not only estimates the OF magnitude for each pixel but also the angle of the flow. These flow angles were utilized to calculate the mean direction of the slip that triggered the detection. To achieve this, the angles were first converted from polar coordinates to unit vectors. This conversion was necessary to prevent false calculations of the mean angle around 0 or 2π. Without converting to unit vectors, the general direction of OF angles pointing to the right could erroneously result in the mean direction pointing to the left. Furthermore, OF angles falling within the horizontal alignment range of ±25° (Figure 7) were excluded from being classified as slip. This exclusion allowed the algorithm to detect possible slip cases, both translational and rotational, while effectively avoiding the classification of OF caused by the deformation of the object due to gripper action as slip.

The conversion from polar coordinates to a unit vector for all pixels was computationally expensive and unnecessary for estimating the general direction of the flow. Instead, a more efficient approach was adopted, wherein only the 10,000 pixels with the highest magnitude of optical flow were used to calculate the flow direction (Algorithm 2).

Algorithm 2 Flow direction algorithm

```
$MAX\_CALCULATIONS = 10{,}000$
$sort\_by\_magnitude(OF\_angles)$
$limit\_size(OF\_angles, MAX\_CALCULATIONS)$
$unit\_vectors = (cos(OF\_angles); sin(OF\_angles))$
$mean = mean(unit\_vectors)$
$direction = normalize(mean)$
```

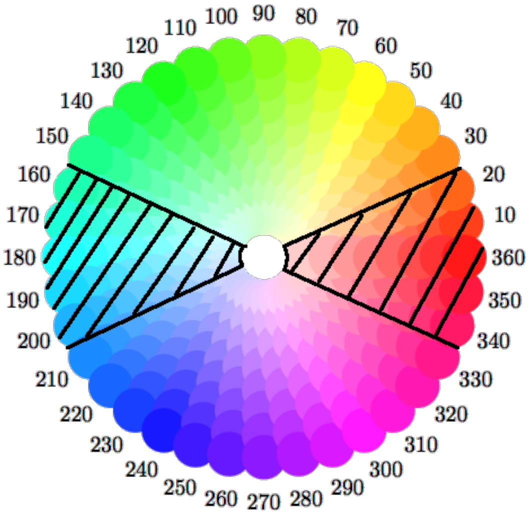

Figure 7. Optical flow angles when represented as an HSV image; the marked areas are the angles of OF that do not trigger slip detection to prevent the deformation of compliant objects from triggering the detection of optical flow.

2.3.5. Dealing with False Positives

An essential aspect of slip detection is the subsequent reaction of the robot to prevent the slip of the object and maintain a stable grasp, thereby preventing the object from falling out of the gripper. Achieving the fastest possible reaction time requires careful consideration of several factors. Minimizing computation delay is crucial, which means avoiding processes that introduce delays, such as temporal filtering. Addressing noise introduced by external influences is also important. To strike a balance, the proposed method applied a threshold that had to be chosen carefully. It needed to be low enough to detect slip cases while remaining high enough to avoid classifying noise as slip, reducing the occurrence of false positives. However, setting such a threshold might prevent the detection of slip cases with low magnitudes, as they could be lower in magnitude than the noise. To address these conflicting requirements, the algorithm utilized multiple temporal layers of slip detection. Instead of solely comparing the most recent frame with the previous one, the most recent frame was also compared with past frames at increasing temporal distances. This approach amplified the magnitude of slow slip cases, allowing them to reach the threshold. However, this amplification also affected noise. To mitigate this effect, the magnitude of the optical flow was filtered with a first-order Butterworth filter. The filter had a sampling frequency of 1000 Hz and a cutoff frequency of 20 Hz. It introduced a delay dependent on the camera's frame rate (Equation (2)). In the case of the RealSense D435 camera operating at 30 fps [15], this resulted in an additional delay of approximately 0.033 s for the extra layers.

$$\eta = \frac{temporal\ distance}{frame\ rate} \tag{2}$$

The layering approach introduced an additional $0.0\bar{3}$ s of delay per frame in temporal distance. Although the added delay was undesirable, it was primarily introduced for the detection of slip cases with very low magnitudes. In this context, the accuracy of slip detection took precedence over the reaction time. For the proposed method, a total of four layers of optical flow calculation were added to the algorithm, comparing the current frame to frames at 1, 2, 4, and 8 time steps in the past. This temporal layering allowed the algorithm to effectively detect slip events of varying magnitudes while accounting for potential noise and occlusions.

The layers are visually represented in Figure 8. The initial comparison of the current frame with the preceding frame introduced the minimally necessary reaction delay, which was dependent on the frame rate and resulted in approximately $0.0\bar{3}$ s of delay, along with the time for the algorithm's calculations, which were negligible. The subsequent layers, representing frame distances of −2, −4, and −8, respectively, introduced additional delays of approximately $0.0\bar{9}$ s, $0.1\bar{6}$ s, and $0.2\bar{9}$ s. This layering approach allowed for a fast reaction time with minimal delay for most slip cases while ensuring robustness to noise and providing a timely reaction for slip cases with low magnitudes.

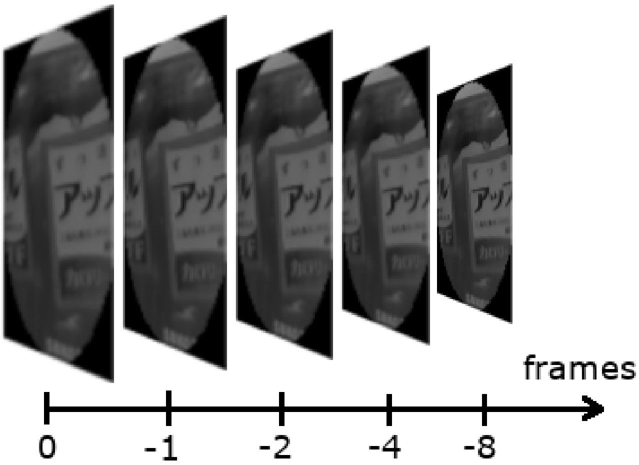

Figure 8. Frames stacked at increasing temporal distances, each being compared with the current frame (0) for optical flow calculation. For every step in temporal distance, extra delay is added; however, the reaction time does not have to be fast. The example frames are taken from the dataset published under open access in Ref. [20].

2.4. Evaluation

The algorithm's performance was evaluated using a dataset specifically designed for slip detection research, which was previously utilized in a paper on slip detection via machine learning by a research group at Waseda University, Tokyo [9]. Unfortunately, a similar dataset from the Massachusetts Institute of Technology (MIT), which could have provided additional comparisons, was not available, and the researchers did not respond to inquiries at the time of writing this paper. The dataset used for evaluation consisted of 35 everyday items that were lifted by a "Nicebot" robot arm equipped with a WSG-50 parallel gripper.

The objects (Figure 9) were subjected to 30 grasping and lifting trials, each with different opening distances, resulting in a total of 1050 image sequences. These image sequences were then labeled to indicate slip or nonslip cases. It is important to note that all objects used in the experiments were within the gripper's maximum opening distance in size. The gripper was equipped with a 6 × 4 matrix uSkin tactile sensor, and a RealSense D345i RGBd camera was positioned in front of the gripper.

As the camera was not directly attached to the end effector, the captured image data needed to be stabilized relative to the gripper's movement for comparability. To achieve this, template matching was applied to a portion of the gripper in each frame (Figure 10). However, rounding errors resulting from the template matching process introduced high-frequency noise. To mitigate this noise, a temporal filter was applied. The filter utilized a second-order Butterworth filter [21] with a sampling frequency of 1000 Hz and a cutoff frequency of 10 Hz. The resulting transfer function and difference equation can be seen in Equations (3) and (4).

$$H(z) = \frac{0.0026 + 0.0052z^{-1} + 0.0026z^{-2}}{1 - 1.7402z^{-1} + 0.8008z^{-1}} \tag{3}$$

$$\begin{aligned} y[n] = {} & 0.0026x[n] + 0.0052x[n-1] + 0.0026x[n-2] \\ & - 1.7402y[n-1] + 0.8008y[n-2] \end{aligned} \tag{4}$$

Seven of the objects in the dataset had instances where the gripper left the frame during some or all of the image sequences, making it impossible to stabilize the frames relative to the gripper's movement. Consequently, these objects were excluded from the evaluation, leaving a total of 840 image sequences used for the algorithm's evaluation.

Regarding the slip detection algorithm's threshold, it was initially chosen through trial and error. However, after the evaluation, the threshold was optimized using a two-step process. First, a grid search was conducted over a range of thresholds. Then, a gradient descent was performed on the three thresholds that resulted in the highest classification rates.

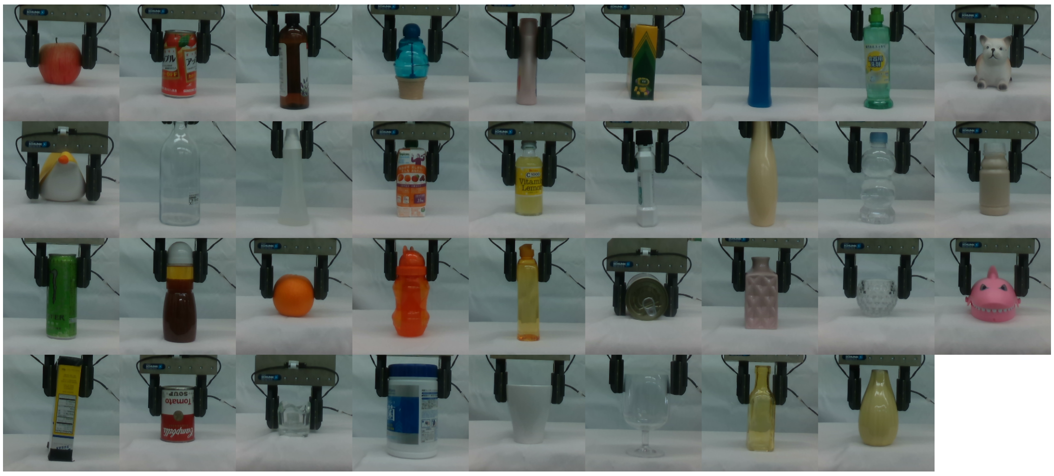

Figure 9. The objects of the open access dataset [20] used for the evaluation. Seven objects had to be excluded, since the gripper left the frame during the lifting task, making it impossible to stabilize the frames relative to the gripper.

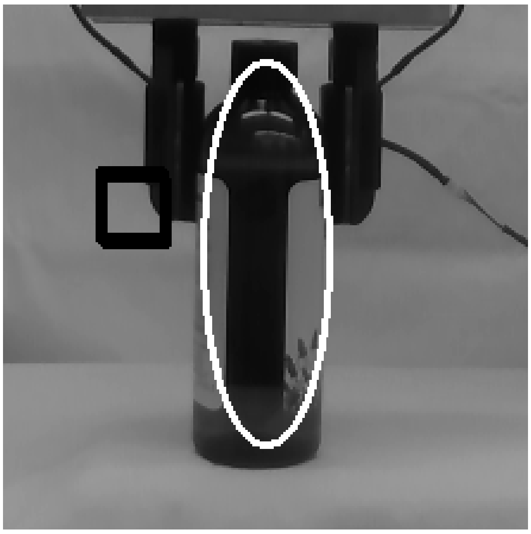

Figure 10. The ellipse estimating the object boundaries is stabilized relative to the gripper by matching a template. The black rectangles represent where in the image the template was found. The image is part of the dataset published under open access in Ref. [20].

3. Experimental Results

The results of the evaluation, along with threshold optimization, are summarized in Table 1. The algorithm's accuracy on the entire dataset with the optimized threshold is 82.38%. However, it is important to consider the limitations of optical flow when evaluating its performance. Specifically, transparent objects like glasses, bottles, and vases can lead to false positives, as background movements behind the object can be registered as slip events. On the other hand, objects without texture, such as plain mugs, opaque bottles, and unprinted cardboard, present the opposite challenge, as no change in pixel values is detected even when the object slips. To address these challenges, the dataset was split into two groups based on the presence of texture. A total of 19 objects were categorized as textured, while 9 objects were categorized as untextured, based on subjective judgment by the conductor of the experiment. When transparent objects and objects without much texture were excluded, the algorithm's accuracy improved to 87.19%. On the excluded objects, an accuracy of 67.67% was achieved.

Table 1. Performance when optimized on Accuracy.

Objects	All	Textured	Untextured
Accuracy	82.38%	87.19%	67.67%
Precision	87.99%	83.13%	76.77%
Recall	75.00%	93.33%	50.67%
F1 Score	80.98%	87.93%	61.04%

An additional argument can be made regarding the cost of false positives and false negatives in the slip detection algorithm. While false positives may not have a significant impact as they result in the gripper grasping the object tighter, for very compliant objects, this could be problematic. On the other hand, false negatives have a much greater cost, as they can lead to the object being dropped and potentially damaged or cause the contents of the held object, such as water in a mug, to be spilled. Therefore, it is crucial to examine the recall, which indicates the percentage of actual slip cases that were correctly identified.

The algorithm achieves a recall of 75.00% on the entire dataset. However, when evaluated with only textured objects, the recall increases to 93.33%. On the other hand, when tested with untextured or transparent objects, the recall drops to 50.67%. To improve the recall and reduce the occurrence of false negatives, a further optimization of the algorithm's threshold was performed. A weighted optimization was used to strike a balance between recall and accuracy. The recall was given twice the weight of accuracy, and the threshold was adjusted accordingly. The result of the weighted optimization can be found in Table 2. After optimization, the recall of the algorithm increased by 13.14% to 87.14%, while the accuracy decreased by 3.57%, resulting in an accuracy of 78.81%. The greatest gain in recall was observed for untextured objects, with an increase of 12.66%. For textured objects, a smaller gain of 1.76% was achieved.

Table 2. Performance with a weighted optimization on Recall and Accuracy at a 2:1 ratio

Objects	All	Textured	Untextured
Accuracy	78.81%	81.23%	66.33%
Precision	74.69%	74.45%	67.38%
Recall	87.14%	95.09%	63.33%
F1 Score	80.44%	83.51%	65.29%

4. Discussion

The proposed slip detection algorithm based on classical optical flow calculations using only visual data addresses common issues like optical path occlusion and compliant objects, while also being robust to outside influences through background removal. The results of the evaluation demonstrate that the performance of the proposed method is comparable to the state-of-the-art approaches (Table 3). While other methods achieve slightly higher accuracy when including tactile data, attaching tactile sensors to the gripper might not always be feasible. Many existing methods utilize bulky parallel grippers to attach tactile sensors, which can add additional thickness to the gripper, making it less suitable for some applications. It is evident from the evaluation that the algorithm's performance is highly dependent on the object being grasped. Two main issues are apparent: First, objects with smooth surfaces and no printed texture do not show any change in pixel value when they move relative to the camera. Although a motion field exists on the object's surface, it cannot be detected visually. This makes it challenging to detect slip events solely through visual information, especially if there is no background or gripper to relate the motion to. This distinguishes the proposed method from others, as it applies background removal for increased robustness, but it may sacrifice some information about relative motion to the background. Secondly, transparent objects present an additional challenge. They not only lack a detectable motion field even though one exists but also allow the background of the scene to be visible through the object itself. This causes false positives to be detected when the background moves due to environmental movements or robot arm motions. The low recall of 50.67% for untextured and transparent objects indicates a significant number of false negatives in the results. In conclusion, the proposed algorithm offers a viable approach for slip detection with minimal sensory data and is effective in many scenarios. However, further improvements may be necessary to address challenges related to objects with smooth surfaces and transparent objects to enhance the algorithm's overall performance and reliability.

Table 3. Comparison of the results of the proposed method with the current literature. These are the results achieved with only visual data, except for [11], which also includes tactile data, as no experiment with visual data only was conducted in their work. Ref. [13] achieves a grasping success rate of 85.20%, which is not directly comparable.

Article	**Proposed**	[8]	[9]	[11]	[12]
Accuracy	78.81%	80.92%	67.96%	88.03%	96.00%
Precision	74.69%	N/A	N/A	N/A	92.69%
Recall	87.14%	N/A	N/A	N/A	100.00%
F1 Score	80.44%	N/A	N/A	N/A	96.21%

Generally, the algorithm appears to have an easier time detecting slips than detecting nonslip cases. This can be observed in the recall and accuracy of both optimization approaches (see Tables 1 and 2). The recall is significantly higher than the accuracy, indicating that false positives occur more frequently than false negatives. In the context of the application, this is generally acceptable, as a false positive would only require an adjustment or tightening of the grasp, whereas a false negative could result in dropping the object. By optimizing the algorithm for higher recall, false negative cases can be minimized without sacrificing too much accuracy. Since the works of the research groups we are comparing our proposed method with all use learning methods that process image data but are trained in a controlled environment [8,9,11–13], it is possible that the proposed method performs better in uncontrolled environments. The performance of DNNs or convolutional neural networks (CNNs) that are trained on images with a consistent white background and tested in the same environment might be affected by a changing background. For industrial applications, this limitation might not be relevant, but for using the robot as an assistive device in daily life, the ability to perform regardless of the environment is essential. Our work proposes the first method tailored to this specific use case in assistive robots. Despite the limitations of our proposed method, it still performs at a comparable level to the state of the art, regardless of the environment (see Table 3).

The dataset we used to evaluate the performance of our algorithm is not perfect, as the camera setup differed from ours. In our setup, the camera was attached to the end effector, but in the dataset, it was stationary and placed in front of the setup. This necessitated stabilizing the frames relative to the end effector movements, which could potentially have influenced the results. However, the fact that our algorithm performed well on artificially stabilized image data suggests that potential applications with a stationary or non-end-effector-attached camera, like the EXOTIC Exoskeleton [22], are feasible. The EXOTIC Exoskeleton is a 5-DoF upper limb exoskeleton, controlled by a tongue–computer interface, where the camera is located on the user's shoulder, rather than the end effector or wrist. To make the slip detection algorithm applicable to this setup, the collected camera data would need to be stabilized. One approach could involve template matching a part of the exoskeleton close to the object in each recorded frame. Another option could be calculating the end effector position from the joint angles via forward kinematics and projecting it onto the two-dimensional camera plane to obtain its position in pixel coordinates. Combining both methods might lead to the most accurate estimation of the grasped object in the recorded frame.

The method we employed to separate the object from the background is somewhat specific to our setup. With the continuous improvement of AI, segmentation models have become increasingly effective. Utilizing such segmentation models could improve the accuracy of extracting the object held by the robot arm and separating it from the background more precisely. However, some of the more sophisticated models may have longer calculation times, making them unsuitable for real-time applications. For our algorithm to work effectively, the segmentation accuracy needs to be extremely high to avoid false positives. Conversely, if the object extraction is overly accurate, comparing two cut-out subframes at the same coordinates becomes challenging. When estimating the motion of the grasped object with optical flow, perfectly extracted subframes around the object in the current frame would have to be applied to the last frame to calculate the optical flow between the frames. In the event of a slip, part of the background would appear in the past frame, which could potentially cause issues, although the voting part of the algorithm should minimize this effect.

4.1. Limitations

Apart from the limitations related to untextured and transparent objects, the algorithm has several other constraints. Changes in lighting conditions could cause the algorithm to detect optical flow, even when no actual motion of the object is occurring, leading to false positives. However, this should not significantly impact the recall, which is considered more crucial in the application of our device. A fundamental drawback of our proposed method is that it solely relies on data from the RGBd camera and is based on frame-to-frame comparisons, lacking the ability to predict incipient slip events. Unfortunately, this is an inherent limitation of our approach. While the algorithm includes a mechanism to handle optical path occlusion, it is limited to approximately 50% occlusion of the object, and the frame itself can experience more occlusion. This limitation arises due to the nature of purely vision-based approaches and can only be addressed to some extent.

Regarding the handling of compliant objects, our proposed method is simple but effective. However, there is considerable room for improvement, as we currently make strong assumptions about the optical flow direction resulting from the deformation of the grasped object by the gripper. Instead of directly using the flow direction, we could consider the coherency of the flow direction. Slip can generally be categorized into two types: translational and rotational slip. For purely translational slip, the estimated motion field by the optical flow exhibits high coherence, with all pixels having the same flow direction. In rotational slip cases, the motion field's coherence depends on the distance of the pixels from the point of rotation, which is determined by the gripper's contact point, whose position is known. The motion field of compliant objects being deformed is less coherent, or at least it shows a different pattern. By excluding these patterns instead of solely relying on horizontal optical flow, we might be able to improve the handling of compliant objects.

4.2. Future Work

Improving the algorithm would involve optimizing the parameters used in optical flow estimation, flow direction calculation, and temporal layering of the optical flow. A multidimensional grip search followed by gradient descent could be performed on these parameters, leading to a re-evaluation of the algorithm using the dataset presented above. Furthermore, evaluating the algorithm on similar datasets from comparable studies would provide a better understanding of its performance and enhance the comparison with those studies. Regarding the optical path occlusion strategy, merely eliminating pixels with zero optical flow might not be sufficient. Pixels with very small optical flow magnitudes could still skew the majority vote toward false negatives. This issue is particularly relevant for objects with high texture at the image borders but little texture inside, such as some packaging materials. To address this, an additional threshold on the lower end of the optical flow magnitude spectrum could be implemented to remove these pixels, allowing more "relevant" pixels to vote on slip cases. This enhancement would improve the algorithm's robustness in handling the optical path occlusion of such objects. To obtain a more accurate estimation of the grasping force, we could utilize not only the weight estimation of the grasped object but also the grasping force estimated from the current applied to the underactuated gripper during the grasping process. This additional information could be incorporated into a more sophisticated grasping strategy that uses force feedback to adjust the robot arm's grip, leading to improved grasping performance.

5. Conclusions

In this paper, we introduce a slip detection method for assistive robotic arms that offers several advantages, including minimal resource usage, reliance on classical and easy-to-implement methods, and the use of affordable and readily available materials. The approach leverages optical flow to estimate the motion field of grasped objects and addresses common challenges encountered in vision-based approaches. While the method is not as effective as tactile sensors, it achieves accuracy comparable to state-of-the-art learning-based methods. The proposed method effectively handles common issues associated with purely vision-based slip detection methods, such as optical path occlusion or compliant objects. By manipulating the optical flow estimation resulting from frame comparisons, a voting system of pixels enables successful slip detection, even with up to 50% occlusion of the grasped object's optical path. Additionally, the algorithm discriminates specific optical flow patterns to prevent false triggers caused by deformations of compliant objects resulting in optical flow changes. However, predictive or incipient slip detection is not feasible due to the nature of the method, which relies on comparing two frames. An important advantage of the proposed method is its robustness against changes in the environment and robot arm movements achieved by removing the background of the camera-recorded scene. This feature proves crucial in everyday tasks that take place in diverse and dynamic environments. To further enhance the algorithm's performance, future improvements could focus on parameter optimization and the development of strategies to address challenges related to untextured and transparent objects.

Author Contributions: Conceptualization, Á.A.P., L.N.S.A.S. and L.M.; methodology, L.M.; software, L.M.; validation, L.M.; formal analysis, L.M.; investigation, L.M.; resources, Á.A.P. and L.N.S.A.S.; data curation, L.M.; writing—original draft preparation, L.M.; writing—review and editing, Á.A.P., L.N.S.A.S. and L.M.; visualization, L.M.; supervision, Á.A.P. and L.N.S.A.S.; project administration, L.N.S.A.S.; funding acquisition, L.N.S.A.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the independent research fund Denmark under project 8022-00234B and the Erasmus+ programme.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are available on request by contacting the corresponding author and under the sources cited in this paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

OF	Optical Flow
OpenCV	Open Computer Vision
DoF	Degrees of Freedom
TCI	Tongue–Computer Interface
RGBd	Red Green Blue Depth
DNN	Deep Neural Network
CNN	Convolutional Neural Network
MS-TCN	Multi-Scale Temporal Convolutional Network
LSTM	Long Short-Term Memory
MIT	Massachusetts Institute of Technology

References

1. Salimi, I.; Hollender, I.; Frazier, W.; Gordon, A.M. Specificity of internal representations underlying grasping. *J. Neurophysiol.* **2000**, *84*, 2390–2397. [CrossRef] [PubMed]
2. Hershkovitz, M.; Tasch, U.; Teboulle, M. Toeard a formulation of the human grasping quality sense. *J. Field Robot.* **1995**, *12*, 249–256. [CrossRef]
3. Struijk, L.N.S.A.; Egsgaard, L.L.; Lontis, R.; Gaihede, M.; Bentsen, B. Wireless intraoral tongue control of an assistive robotic arm for individuals with tetraplegia. *J. Neuroeng. Rehabil.* **2017**, *14*, 110. [CrossRef] [PubMed]
4. Pálsdóttir, Á.A.; Mohammadi, M.; Bentsen, B.; Struijk, L.N.S.A. A Dedicated Tool Frame Based Tongue Interface Layout Improves 2D Visual Guided Control of an Assistive Robotic Manipulator: A Design Parameter for Tele-Applications. *IEEE Sens. J.* **2022**, *22*, 9868–9880. [CrossRef]
5. Bengtson, S.H.; Thøgersen, M.B.; Mohammadi, M.; Kobbelgaard, F.V.; Gull, M.A.; Andreasen Struijk, L.N.S.; Bak, T.; Moeslund, T.B. Computer Vision-Based Adaptive Semi-Autonomous Control of an Upper Limb Exoskeleton for Individuals with Tetraplegia. *Appl. Sci.* **2022**, *12*, 4374. [CrossRef]
6. Pak, R.; Rovira, E.; McLaughlin, A.C.; Baldwin, N. Does the domain of technology impact user trust? Investigating trust in automation across different consumer-oriented domains in young adults, military, and older adults. *Theor. Issues Ergon. Sci.* **2017**, *18*, 199–220. [CrossRef]
7. Romeo, R.A.; Zollo, L. Methods and Sensors for Slip Detection in Robotics: A Survey. *IEEE Access* **2020**, *8*, 73027–73050. [CrossRef]
8. Li, J.; Dong, S.; Adelson, E. Slip Detection with Combined Tactile and Visual Information. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 21–25 May 2018; pp. 7772–7777. [CrossRef]
9. Yan, G.; Schmitz, A.; Tomo, T.P.; Somlor, S.; Funabashi, S.; Sugano, S. Detection of Slip from Vision and Touch. In Proceedings of the 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 23–27 May 2022; pp. 3537–3543. [CrossRef]
10. Francomano, M.T.; Accoto, D.; Guglielmelli, E. Artificial Sense of Slip—A Review. *IEEE Sens. J.* **2013**, *13*, 2489–2498. [CrossRef]
11. Zhang, Y.; Kan, Z.; Tse, Y.A.; Yang, Y.; Wang, M.Y. FingerVision Tactile Sensor Design and Slip Detection Using Convolutional LSTM Network. *arXiv* **2018**, arXiv:1810.02653.
12. Gao, J.; Huang, Z.; Tang, Z.; Song, H.; Liang, W. Visuo-Tactile-Based Slip Detection Using A Multi-Scale Temporal Convolution Network. *arXiv* **2023**, arXiv:2302.13564.
13. Jiang, J.; Cao, G.; Butterworth, A.; Do, T.T.; Luo, S. Where Shall I Touch? Vision-Guided Tactile Poking for Transparent Object Grasping. *IEEE/ASME Trans. Mechatron.* **2023**, *28*, 233–244. [CrossRef]
14. Struijk, L.N.S.A.; Lontis, E.R.; Bentsen, B.; Christensen, H.V.; Caltenco, H.A.; Lund, M.E. Fully integrated wireless inductive tongue computer interface for disabled people. *Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.* **2009**, *2009*, 547–550. [PubMed]
15. Intel RealSense, RealSense Depth Camera D435. Available online: https://www.intelrealsense.com/depth-camera-d435/ (accessed on 15 May 2023).
16. Chanda, B. Morphological Algorithms for Image Processing. *IETE Tech. Rev.* **2008**, *25*, 9–18.
17. Intel RealSense. High Speed Capture Mode of Intel Realsense Depth Camera d435. Available online: https://opencv.org/ (accessed on 15 May 2023).
18. Farnebäck, G. Two-Frame Motion Estimation Based on Polynomial Expansion. In Proceedings of the Image Analysis: 13th Scandinavian Conference, SCIA 2003, Halmstad, Sweden, 29 June–2 July 2003; Bigun, J., Gustavsson, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2003; pp. 363–370.
19. Patel, D.; Upadhyay, S. Optical Flow Measurement using Lucas Kanade Method. *Int. J. Comput. Appl.* **2013**, *61*, 6–10. [CrossRef]
20. Yan, G. Multimodality Slip Detection/Prediction Dataset Using uSkin Tactile Sensor from Sugano Lab, Waseda University. 2021. Available online: https://zenodo.org/record/4584809 (accessed on 15 May 2023). [CrossRef]

21. Shouran, M.; Elgamli, E. Design and Implementation of Butterworth Filter. *Int. J. Innov. Res. Sci. Eng. Technol.* **2020**, *9*, 7975.
22. Thøgersen, M.B.; Mohammadi, M.; Gull, M.A.; Bengtson, S.H.; Kobbelgaard, F.V.; Bentsen, B.; Khan, B.Y.A.; Severinsen, K.E.; Bai, S.; Bak, T.; et al. User Based Development and Test of the EXOTIC Exoskeleton: Empowering Individuals with Tetraplegia Using a Compact, Versatile, 5-DoF Upper Limb Exoskeleton Controlled through Intelligent Semi-Automated Shared Tongue Control. *Sensors* **2022**, *22*, 6919. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.