Abstract

We present artificial intelligence techniques employed in the development of an effective rehabilitation device, a “smart can”, that assesses hand function for individuals with stroke, spinal cord injury (SCI), traumatic brain injury (TBI), or other neurologic injuries. Currently, at the end of their formal therapies, many individuals with these pathologies are typically provided only with written home exercise programs prescribed by their therapists. It is not clear how performance changes are measured over time at home and it is therefore not surprising that these written exercises are commonly discontinued shortly after the formal therapies end. To this end, using AI-based methods, we have designed a rehabilitation device that specifically explores the use of a new type of performance measurement, “the jerk score”, so that the device can be eventually deployed in a home setting providing the user with a direct measure of the quality of the movements made by the affected hand. Therefore, using computer vision based techniques such as object detection, incremental visual tracking, activity recognition, and 3D virtual augmentation, we successfully demonstrate the efficacies of the system in the lab setting.

1 Introduction

The purpose of this paper is to highlight the key role that AI, especially computer vision and graphics play in the implementation and deployment of an innovative assistive technology, aimed at assessing hand function rehabilitation. Stroke\(^1\) alone is the 5th leading cause of death in the US, with one person dying every 4 minutes as a result(Kochanek et al. 2013). People who have suffered strokes and other neurological injuries often experience hand impairment (Seo, Rymer, and Kamper 2009), and this is demonstrated in (i) significant delays in how long it takes to grip, release and manipulate objects; and (ii) the quality of movement when manipulating the object.

Rehabilitation services are aimed at reducing impairment after a stroke or other neurological disorder, and improving functional mobility. The current medical pathway for individuals with stroke provides intensive care in the acute stages, but at the end of their formal therapies, individuals are often provided only with written home exercise programs prescribed by their therapists. Unfortunately, these written home exercise programs are commonly soon discontinued (Novak 2011). But it has been well established that practice of movements that challenge the individual is required for motor learning to continue to improve (Wolf et al. 1989)(Schmidt 1991), hence active and regular participation in home exercise programs is imperative. Studies involving household tasks suggest that objective feedback assists the individual to more accurately judge their performance, which in turn can encourage alterations in movement and lead to improved performance(Liu and Chan 2014).

To this end, our goal in this work was to develop portable AI-based technology, affordable for home use, capable of quantifying quality of movement and providing user feedback on motor performance for rehabilitation beyond the acute stages. Although our proposed device can be used by a wide range of users to improve general upper extremity control and function, it has been designed primarily to improve arm and hand function in chronic disability conditions.

1.1 Related Work

A device somewhat similar to our proposed work is a sensor-based rehabilitation system by (Hussain et al. 2012), which instrumented objects, a movement tracking wrist band with an inertial measurement unit (IMU) and a Microsoft Kinect\(^2\) camera, and lastly, an interface with computer games. Another related work includes that by (Duff et al. 2010) where the authors developed a mixed reality rehabilitation system that measured the reaching movements of chronic stroke survivors. The system provided feedback based on the movement patterns of the subject’s affected arm and torso while reaching to grasp. An extensive system was strapped on the patient’s torso and arm and feedback
on performance was provided in visual and musical forms to the subjects. A system more geared towards home-use by (Baran et al. 2011) involved a home-based adaptive mixed reality system geared for stroke survivor rehabilitation. It consisted of an elaborate custom-designed media center involving a custom table and chair setup, along with three optitrack cameras, mounted above a computer to track a reflective marker worn on the participant’s wrist as he/she attempted to reach one of three pre-determined target locations on the table.

Other commercial rehabilitation systems include (a) MindMaze (http://www.mindmaze.ch/) which involves the use of 3D motion capture cameras with gesture and object/user recognition and augmented reality capabilities. The main drawback here is the absence of quantitative measures to assist an individual in tracking progress; and (b) LimbsAlive (http://www.limbsalive.com/) which involves the use of two grip devices that the individual uses to manipulate objects on a screen. Again, similar to MindMaze, the user is expected to be motivated by viewing his/her performance on a computer screen, but no trackable quantitative measures are computed.

In this work, we therefore focus on developing a portable, affordable and functional system that integrates with existing home-friendly technologies such as the tablet (with a camera), to provide quantitative and qualitative feedback to the users about their performance.

2 The End-to-End Solution

The proposed system is designed to be readily used in a home setting with little or no supervision by a rehabilitation specialist. The end-to-end system consists of (a) a portable physical device embedded with motion sensors (accelerometer and gyroscope - See Figure 1); (b) a camera; (c) a user interface and feedback system; and (d) a set of functional exercises to be performed with the device. In its current state, the camera, user interface and feedback system all exist on the same computing device, a tablet. The end-to-end system provides both quantitative and qualitative information regarding task performance.

Based on a small focus group involving six clinicians (occupational and physical therapists) and three stroke patients, the specific measurements of interest to be obtained from the device and system are: (i) completion of pre-specified exercises or activities, (ii) time taken to complete activity, (iii) smoothness of movement (iv) efficiency of the movement path (determining trajectory of movement) and (v) error monitoring.

The focus group participants contributed extensively to the overall design of this system and as we made progress on the development of the device, we intermittently checked back with them. The system is designed to be used by a patient at home after the initial set of formal therapies end. A clinician will be able to access the data recorded from the patient and can intermittently review it, give feedback and readjust the exercises. In its current state, only a small number of tasks are hardcoded but in the future the system will be made more flexible.

2.1 The physical device

The physical device implemented is an embedded “soda can”, shown in Figure 1. This smart can was designed to be a familiar and non-threatening 3D object, and using digital fabrication facilities, we developed the weight adjustable 3D-printed smart can. To enable weight adjustability, the can was designed to consist of a hollow cavity which could be filled with different weights, thus varying its total overall weight. The smart can was fitted with a 9-axis IMU sensor to track movement, specifically, acceleration, orientation and smoothness. The measurements from the sensor were incorporated into an augmented reality interface, to provide visual feedback to the users.

Sensors for motion estimation The sensor system of the smart can consists of an accelerometer and gyroscope (specifications of the IMU sensors are provided at https://www.sparkfun.com/datasheets/Sensors/Accelerometer/ADXL345.pdf). Although it is possible to obtain estimates for position and velocity using low-cost accelerometers, these estimates are often very poor, highly unstable and in general, unusable. While the accelerometer measurements by themselves are quite reliable, the orientation of the sensor must be known with a high degree of accuracy so that gravity measurements can be distinguished from the physical acceleration of the sensor. It is therefore imperative to involve a camera system along with a visual tracking algorithm in order to accurately measure position and time.

2.2 The camera

Due to the limitations of the IMU sensors, we augmented the system with a general-purpose camera located on the computing device (tablet or PC) on which the user interface resides and computer vision techniques were employed on videos of the tasks being implemented. The camera used for our experimentation was the in-built camera on the Apple

---

2As individuals improve their motor functions, the challenge levels of their exercises can be increased with added weights
Workbook computer, the iMac SD Webcam that captured video in standard-definition (640-by-480).

2.3 The user interface and feedback system

We developed a user interface which includes an augmented reality environment, implemented using Panda3D, an open-source, python-based, game engine that includes graphics, audio, I/O, collision detection, and other functionalities relevant to the creation of 3D games. The interface interacts with the motion sensors on the smart can via pySerial, a python module that encapsulates the access for the serial port. This interface has the dual role of (i) being a control system with which the user powers the system, selects new tasks to perform (when needed), etc.; and (ii) providing the user with information such as instructions on how to perform the exercises, feedback in the form of visual and quantitative measurements, etc.

2.4 The set of functional exercises

For this proof-of-concept phase of our project, we selected three functional exercises:

1. Move-Can - Moving laterally from one position to another on a hard supporting surface such as a table or wheelchair mat. More functional users could grasp and lift the can to perform this task while less functional ones can simply push the can along the supporting surface, to the target.

2. Drink-from-Can - Grasping and lifting the smart can upwards from the supporting surface towards the mouth; and

3. Pour-from-Can - Grasping, lifting and moving the smart can towards a target and then rotating it as though pouring from the can onto a target spot on the supporting surface.

Task difficulty can be graded to provide the appropriate level of challenge for the participant. Examples of these tasks are shown in Figure 2

3 Vision and Graphics Augmentation

In this section we discuss the specific contributions of computer vision and graphics to the overall project. In order to use a generic camera to obtain real world position readings in a natural, uncontrolled environment, we calibrated the camera to obtain its intrinsic and extrinsic parameters using the GML C++ Camera Calibration Toolbox4 (Zhang 1999).

3.1 Auto-detecting the smart can

By treating various poses of our smart can template as the input images, we can detect the presence of the smart can in a cluttered, noisy image. We employ one-shot learning and detection algorithms via the python version of Open Source Computer Vision (opencV5), an open-source library of computer vision programming functions. The steps involved in the one-shot learning approach are as follows:

(i) Ten template images of the smart can are obtained by rotating the can in increments of about 30 degrees and taking a high resolution image of the can. The images are cropped and stored,

(ii) 500 SURF features (Bay et al. 2008) are extracted on the template images and also stored.

(iii) When the user videos are recorded and converted into frames at the rate of 30 frames per second, SURF features are extracted from the first frame.

(iv) A point matching algorithm is used to find the descriptors in the template that are closest to those in the first frame.

(v) A homography is computed from the corresponding points in the two images and an affine transformation is used to warp the template boundaries onto the user image, thus identifying the boundaries of the can in the user image(Imani and Anandan 1998). The result is the smart can detection.

3.2 Tracking the can incrementally

Many existing visual tracking algorithms are able to track objects well in short durations and in well controlled environments(Black and Jepson 1996)(Matthews, Ishikawa, and Baker 2004)(Isard and Blake 1996)(Jepson, Fleet, and El-Maraghi 2003)(Georgescu et al. 2004)(Comaniciu, Ramesh, and Meer 2003), but these algorithms tend to fail by losing track of the object’s appearance, thus drifting over time. The main difficulty of visual tracking can be attributed to the challenge involved in handling the appearance variability of the target object over time. Ross et al. (Ross et al. 2008) proposed a method that efficiently learned and continually updated a low dimensional subspace representation of the target object.

The tracking problem is cast as an inference problem using a Markov model with hidden state variables. The state variable $X_t$ defines the affine motion parameters of the target at time $t$. Given a set of observed images $I_t = I_1, \ldots, I_t$, the goal is to estimate the value of the hidden variable $X_t$:

\[
p(X_t | I_t) \propto \int p(I_t | X_{t-1}) p(X_{t-1} | I_{t-1}) dX_{t-1}
\]

$p(I_t | X_{t})$ is the observation model where the likelihood of $X_t$ is estimated by observing $I_t$ and $p(X_{t} | X_{t-1})$ is the dynamical model represented by an affine image warp of the location of the target object in an image frame.

We found this appearance-based incremental tracker to perform well for the tracking of the smart can, especially because the can goes through several orientation and appearance changes during the functional exercises.

3.3 Recognizing pre-specified tasks

In order to identify the task performed by the user, we use a robust point matching algorithm on the trajectory data recorded from the smart can. We define the trajectory as the path that the smart can follows through 3-dimensional space as a function of time; the three dimensions we use are $y$, $z$ and $\theta$ - roll angle. We drop the use of the $x$ dimension when constructing trajectories because we found it to be insufficiently discriminatory - two tasks (Move-Can and Pour-from-Can) cover the similar amounts of $x$-axis distance. A

\[\text{http://opencv.org}\]


\[\text{http://opencv.org}\]
A high-level overview of the integration process is shown in Figure 3 and described in the steps below:

1. The process is initiated by the user via the user interface; the video and sensors are initialized.

2. Using openCV video-to-frames functionalities for the images and pySerial calls for the raw sensors, data from the two sources are simultaneously recorded in a log file.

3. The smart can detector is employed in tandem with the incremental tracker on all newly acquired frames and for each frame, the \(xy\) position of the center of the can along with the width of the can in the image are recorded in a log file. The real world depth is then calculated using the image can width. The noisy, raw orientation measurements are stabilized and converted to quaternions to better handle changes in orientation. The two log files are then merged to get time-aligned measurements for position, orientation and acceleration for the smart can.

4. The data from the IMU sensors and the visual tracker are combined to provide quantitative results in form of measurement scores (time, smoothness score etc.) as well as qualitative results in form of a playback of the executed task in an augmented environment. These are presented to the user as feedback via the interface.

**Smoothness score computation:** The reverse smoothness (or jerk) score is one of the main quantitative measures to be reported by the smart can. This score is defined as the time derivative of acceleration, used to quantify smoothness and coordination in sensory-motor performance studies (Hogan and Sternad 2009). We use the dimensionless squared smoothness measure (normalized jerk score - NJS) below (Takada, Yashiro, and Takagi 2006):
4 Experiments and Results

For experimental analysis, we recorded a total of 39 videos. The first 9 videos were used for development and early functionality testing in the lab. Task templates were generated from this first set of videos. The remaining 30 videos were later collected in a more formal and systemic manner, and were used to test the efficacies of different components of the smart can device.

For the more formal data collection exercise, we recorded videos of five consenting non-stroke participants performing the three pre-specified tasks twice, thus resulting in a total of 30 videos. The first time around, each participant performed the three tasks with normal smoothness movements as they would typically do. For the next round of collections, we showed the participants a youtube online video of actual stroke patients performing similar rehabilitation therapy exercises so that they could incorporate jerkiness into their movements. The results obtained from testing are presented and discussed in the rest of this section.

One-shot object detection

For testing the auto-detecting component, we placed the smart can at a pre-set position and then set up a series of target points at every 100mm on a table surface in a well lit room. Starting at 300mm, we placed the camera at the target point and took snapshot images of the can. The feature matching based auto-detection algorithm was applied on the images and if the can was successfully detected by the algorithm, the camera was moved to the next target point, another 100mm away from the can. We continued this exercise until the algorithm could no longer successfully and reliably detect the can in the set of images taken at that distance. We refer to this as the distance limit which we found to be 0.70m for this particular camera and algorithm combination.

All the experimental videos were therefore recorded with the camera being at 0.7m or less from the device.

Figure 4a shows a small number of the keypoints that successfully match features in a template to the portion of the user image containing the can. The pattern of the template that best matches the can in the image is shown at the top left corner.

Using this distance restriction, the smart can was successfully auto-detected in 100% of the first frames of test videos recorded.

\[
NJS = \sqrt{\frac{1}{2} \int_{t_1}^{t_2} \left( (\frac{d^3 y}{dt^3})^2 + (\frac{d^3 z}{dt^3})^2 + (\frac{d^3 x}{dt^3})^2 \right) \, dt} \frac{(\Delta t)^2}{A^2} \quad (1)
\]

where \( \frac{d^3}{dt^3} \) is obtained directly from the stabilized readings from the accelerometer, \( \Delta t \) is \( t_2 - t_1 \) or duration of the movement and \( A \) is the amplitude of displacement (or extent).

Visual tracker evaluation

In order to evaluate the incremental visual tracker described previously, as a qualitative baseline, we implemented the non-parametric particle filter-based tracker (Blanco, González, and Fernández-Madrigal 2008), and applied it on the sequences where the can was being displaced from one location to another. The incremental visual tracking method significantly outperformed the baseline particle filter for this displacement task. From qualitative analysis, we found that surprisingly, the Move-Can task was more challenging for the trackers than the other tasks because the stripped pattern on the can changed constantly in the course of moving it from one location to the next. The trackers thus tended to follow the pattern on the can, rather than the contour of the can. In spite of this, the incremental tracker still proved robust, continuing to track well long after the particle filtering tracker had failed by drifting off the target object early in the process.

Figure 5 shows the results of computing the root mean squared (RMS) error between the predicted locations of features on the smart can and the manually annotated ground-truth locations of the same features. One curve shows the errors from our tracker of choice while the other shows the errors from our baseline tracker.

Activity recognition

For activity recognition, using the dynamic time warping technique described previously, we computed the alignment costs between the 30 recorded videos and the previously collected task template trajectories. Task classification was accomplished by assigning a video to the class of the template with which it had the lowest alignment cost. We performed several different classification tests: (i) classifying all 30 videos irrespective of whether the movements were jerky or normal, using the smart can position and orientation data obtained from visual tracking; (ii) again, classifying all 30 videos but only using the smart can orientation values obtained from the sensors; (iii) classifying only the normal half (15) of the videos using the smart can position and orientation values obtained from visual tracking. The results are presented in Figure 6.
Figure 5: The RMS errors obtained from tracking feature points for each frame of one of the Move-Can task sequences. The errors obtained from the incremental visual tracker (red) are significantly lower than those obtained from the particle filter based tracker (blue), which begins to drift very early in the tracking process.

Figure 6: Resulting confusion matrices (rows represent predicted classes) are shown on the left and the bar chart of the diagonal of the corresponding confusion matrix is shown on the right. The classification accuracy is shown at the top of the bar chart. The top images show the results obtained from classifying all 30 videos using data from the visual tracker; the bottom images show the results obtained from classifying the normal 15 videos using data from the visual tracker.

**Smoothness comparisons**
We also investigated the efficacy of the reverse smoothness score (or normalized jerk score - NJS) defined in Equation 1. We recorded videos of each of the five participants performing each task two times, once with normal movements and next jerkily, however the participants chose to interpret the notion of jerkiness. Figure 7 demonstrates the clear separation between the videos recorded with normal movements and those recorded with jerkiness. In the figure, person-tasks 1, 6 and 11 were all movements made by the same person, similarly, person-tasks 2, 7 and 12 were made by the same participant, etc. It is therefore interesting to observe that the jerk scores cluster strongly by the individual performing the tasks rather than by the tasks being performed. The individual who performed person-tasks 1, 6, and 11 has unusually high jerk scores for normal movement, when compared to the other participants.

**Augmented reality environment**
On the completion of a task by a user, a video of the exercise is recorded and stored simultaneously with the sensor readings (orientation). The system subsequently runs the visual tracker to obtain position data of how the can moved during the course of the exercise. These combined readings are then used to manipulate a virtual can mirroring the real-life movement of the user.

Figure 8 shows an example of a real task performed during testing that was then augmented in a virtual reality environment.
5 Conclusion

In this paper, we have shown how a series of AI-based methods can be applied in home settings, for an innovative stroke rehabilitation assistive device. The main contribution of the work is the ability to now collect qualitative and quantitative measures on exercise tasks performed by post-stroke patients, in order to provide objective feedback for judging progress more accurately. Specifically, the normalized jerk score has been shown to be a quantitative measure that can be tracked over time to demonstrate patient improvements.

References


