Gait Detection and Analysis Using Image Processing

Giaan Nguyen, *Department of Electrical and Computer Engineering*, *Texas A&M University*

***Abstract*—The timed-up-and-go (TUG) test assesses a patient's mobility and stability and is used by physicians to diagnose elderly patients. Despite the onset of inertial measurement units (IMUs) in modern practices of the TUG test, there is a lack of standards in creating metrics for quantifying gait stability. A previous undergraduate project set out to quantify stability using the criteria of gait cadence and rotational steadiness (or minimal wobbliness) within IMU-recorded data. As a continuation, the author explores a cost-effective implementation of standardized metrics by implementing image processing and motion detection models on smartphone-recorded video samples rather than inertial sensor processing. Within the case study, stability is measured based on gait cadence and linear steadiness, with a future potential in evaluating other gait criteria.**

I. Introduction

The timed-up-and-go (TUG) test is a common gait assessment used by physicians to analyze a person's mobility and balance. The patient is asked to stand up from a chair, walk three meters forward, turn around, walk three meters back, and sit back down. Though simple, the TUG test is comprehensive since it tests multiple diagnostics such as gait cadence, wobbliness (or rotational unsteadiness), and the sitting-to-standing/standing-to-sitting time. Because of its comprehensiveness, often the TUG test can detect onset decline in health and disorders in elderly patients, particularly neurological disorders such as Parkinson's disease.

While the use of inertial measurement units (IMUs) – comprised of an accelerometer, a gyroscope, and a magnetometer – is not uncommon in modern practices of the TUG test, there does not exist a standardization for quantifying stability. As a member of a previous project, a capstone design group at the University of Houston (UH) implemented an IMU-based method for providing metrics [1]. For instance, the magnetometer can indicate when a person stands, turns around, and sits. The accelerometer can be used to determine when a person takes a step, which can be used for analyzing gait cadence. The gyroscope can assess the wobbliness of a person. (For a brief overview of the results of the IMU-based gait monitoring system, see the Appendix.)

However, while IMUs can be accurate since they can measure a person's linear acceleration, rotational velocity, and direction, the cost can be expensive depending on the degree of desired accuracy and precision. As of 2021, the cost of an IMU can range from 30-50 dollars on the lower end to hundreds of dollars. Additionally, while an IMU can show markers of unstable gait, the sensor is unable to pinpoint the source of such instability.

The author of this paper explores the plausibility of a cost-effective approach to TUG test standardization via image and data processing of mobile-recorded videos. Due to the ongoing COVID-19 pandemic and the allotted project timeline of one month, a case study with an emphasis on footwork is performed and analyzed, with a few peers assisting with running the physical TUG test.

II. Procedure

A. Experimental Setup

Using standard green duct tape with a width of approximately 50 mm, a length of three meters of tape is adhered to the floor as straight as possible, with enough open space on both sides of the tape such that no object is blocking the subject's path nor obstructing the camera's focus. A chair (preferably but optionally an armchair) is placed right before one end of the taped line so that the subject's feet is positioned before the line while they are sitting.

The observer filming the TUG test will be using their own smartphone, set to a reasonably low resolution and frame rate. (At the time, the lowest possible setting of 1280x720 pixel resolution and 30 fps was used.) Since the observer is recording the subject walking towards the camera lens and back, the observer is positioned at the other end of the line opposite to the chair. If possible, set up a tripod to reduce human error during filming due to any slight tremors. The phone should be set up such that the camera is capturing in portrait mode, with the furthest end of the line centered in the frame and the entire line vertically aligned with the frame; under this setup, at least the subject's lower body (if not the full body) should be observed during the entirety of the test. The subject may have to stand in front of the camera before the test to ensure the smartphone placement is appropriate.

The subject is asked to perform the TUG test in two different ways: normal walking (i.e., stable) and drunk walking (unstable). For normal walking, the subject is sitting in the chair at the start; they should then stand up, walk along the line towards the camera lens at their usual pace, turn around, walk back along the line towards the chair, and sit back down. Similar notions apply to drunk walking, except the subject is asked to exaggerate unstable motions; that is, swaying motions, inconsistent times between each step, and deviations from the line (or simply put, "walk like a drunk person").

For simplistic purposes, we will refer to the end of the tape closest to the observer as the "front of the tape" and the end of the tape furthest away from the observer as the "back of the tape."

B. Image Processing: Extracting the Line

When processing a recorded video sample, the indices for the taped line should be extracted and stored. First, apply

ensemble averaging to reduce noise by taking the mean of the first ten frames of the video. Specifically, since the tape is green, isolate only the green component layers of the first ten frames and apply ensemble averaging to the green components. Since the camera was positioned so that the back of the tape was around the center of the frame, a positional mask can be applied, zeroing out intensities in the top half of the portrait frame as well as the left third and right third of the frame.

Then apply an intensity threshold mask such that the green tape is still present, but its neighboring objects are diminished if not zeroed; a threshold green intensity value between 110 and 130 is appropriate. To isolate the tape, a mathematical morphological operation is considered: by opening the image using a long rectangle (50x5 pixels in portrait mode), one can maintain the tape's shape while removing the neighboring pixels. Lastly, Canny edge detection is used to extract the edges of the tape. The final output resembles a long trapezoid; since the camera can only capture in the 2D-plane, the lack of depth recording can be seen by the pixel width of the tape in the image, in which the width of the tape commonsensically decreases as the point along the taped line moves away from the camera lens.

The results following each mask and operation can be seen in Fig. 1.

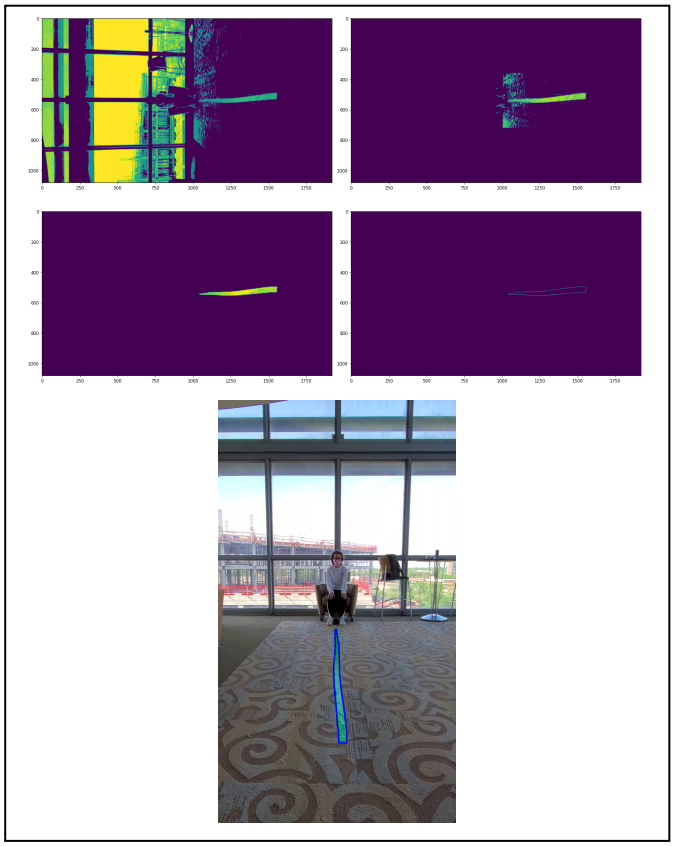

Fig. 1. Line Extraction. *The top left image shows the image following ensemble averaging of the green components. The top right shows the result of applying positional and intensity threshold masks. The center left shows the output of opening. The center right shows the output of the Canny edge detection. The bottom image shows the initial frame with the outline of the line from the Canny edge detection superimposed onto the image.*

The stored pixel indices will later be used to assess whether a person's footstep falls within the line or outside. Now will need to find a motion detection model to track the subject's movements, particularly their footsteps. Three different methods were considered for motion detection. To reduce the computational time, an output frame rate of 10 fps is set, i.e., one-third of the original frame rate of the recorded video.

C. Method I: Modified Frame Differencing

A modified version of the frame differencing method proposed by Gopal Thapa, Kalpana Sharma, and M.K. Ghose is considered since it still utilizes each input frame while still maintaining the 10-fps output [2]. First, a background model is acquired; while usually an initial image without the subject is captured, to simulate real-time scenarios, the median of 50 randomly selected frames is calculated and set as the background. While the mean could have been used, the median does not incorporate outlier intensities into the background and maintains integer values unlike the mean.

Since the frame rate is reduced to one-third of the original frame rate, an analysis of three consecutive frames instead of the proposed eight is done. Three frames are subtracted by the background and then summed together before being applied a morphological operation such as dilation. A rectangular contour is drawn, keeping track of the center of the bottom edge of the bottommost contour. The new image is then outputted as one frame of the output video.

D. Method II: Haar Cascade Classifier

The (Haar) cascade classifier uses Haar-like features akin to kernels to extract features of a certain image. Introduced by Paul Viola and Michael Jones, the premise behind the cascade classifier is to reduce computational time by discarding an image upon failure [3].

First, a large set of positive images and negative images are used to trained multiple classifiers, with positive images being the desired images (like images of the lower body) and negative images depicting otherwise (like images of faces or rooms). After training each classifier using the images, a classifier is selected for each feature of the positive image. For instance, a classifier that performs best for left leg cuffs is chosen, while another classifier that detects hip joints well is chosen. Effectively, by combining these weaker classifiers which alone can only detect one feature well, together they make a stronger classifier that can classify an image as positive or negative; this notion is called boosting.

Then during testing, a window of an image is tested one weak classifier at a time. If the window fails the first classifier, it gets discarded. Otherwise, it gets tested against the second classifier; if it fails, it gets discarded. In short, the window gets discarded upon first failure in a sequence of classifiers.

For testing, only every third frame of the input is processed to maintain the 10-fps frame rate for the output video. Since OpenCV already has a pre-trained cascade classifier for lower body detection, the pre-trained model is used [4]. As with the frame differencing method, the center of the bottom edge of the bottommost rectangular contour drawn on each frame is tracked and stored.

E. Method III: OpenPose Model

OpenPose is a pre-trained, free-for-non-commercial-use neural network capable of real-time 2D pose estimation, as authored by Ginés Hidalgo, Zhe Cao, Tomas Simon, Shih-En Wei, Yaadhav Raaj, Hanbyul Joo, and Yaser Sheikh at Carnegie Mellon University [5-8]. While there are different versions of the model, this project will only focus on the latest model BODY_25 as there is only one person within the camera frame, and extra information regarding the feet may prove to be useful, especially for future experiments.

Basically, the neural network utilizes two different branches, where one branch estimates the position of each of the 25 joints, and the second branch estimates the connections between joints (i.e., appendage pairs). Unlike the previous two methods, OpenPose is expected to be the most accurate with its ability to capture joints instead of general motion; however, it is also expected to be computationally expensive.

As with the cascade classifier, only every third frame of the input is processed to maintain the 10-fps frame rate of the output video. Since all 25 body points are drawn for each frame, only the points corresponding to the ankles are tracked, without the need for rectangular contours. An additional metric of averaging the camera left and right ankles (called the center) is computed.

F. Normalization

As previously discussed in Section II.B, while humans can perceive depth in 2D images, machines cannot capture depth within the 2D plane. This is evident by the apparent pixel width of the tape decreasing as one reads up the image. Theoretically, while a horizontal distance d pixels from the front of the tape should be the same as another horizontal distance d pixels from the back of the tape in the 2D plane as seen in Fig. 2, the ground truth is that the distance d from the back of the tape is actually longer than the distance d from the front of the tape.

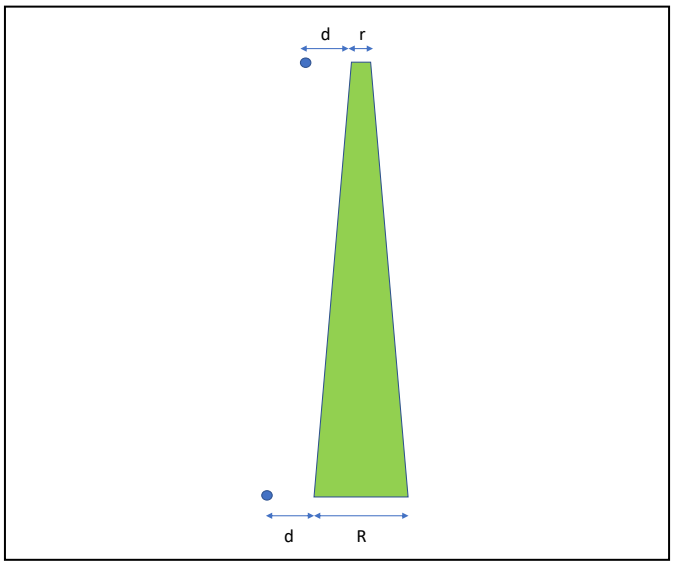

Fig. 2. Perceived Distance. *In the 2D plane, both distances d above are equal. When compared to the corresponding line widths r and R, the human brain perceives that the two distances are not equal.*

A normalization technique using the pixel width R of the front of the tape as the baseline is proposed. Let r be the pixel width of the tape at any given point along the line. Let d be the horizontal pixel distance from the tape at the point where r is measured. Then the normalization equation is given by

$$d_{new} = \frac{d}{\left(\frac{r+R}{2}\right)} = \frac{2d}{r+R} \quad (1)$$

Effectively, the distance is scaled more as the pixel width decreases. The choice of inversely scaling by the average of r and R is done to avoid over-penalizing the distances near the back of the tape. After normalization, each d_{new} can then be multiplied by the actual width of the tape, which is usually around 5 cm, to obtain meaningful distances in terms of centimeters. For ease of understanding, all figures shown in the results section are converted to inches.

III. Results

Following the implementation of each of the three models, the stored center and/or ankle index values are used to determine the distance from the tape throughout the duration of the recorded TUG test. If the indices are parallel to the line, then a program checks whether or not the indices fall within the tape's edge indices; if not, then the program calculates the minimum pixel distance of the indices from the line, implements the normalization technique introduced in Section II.F, and stores the estimates of the true distances for further analysis.

A. Runtime

The runtime of each of the three models were computed during implementation. The computation times are shown in Table 1 as well as the average for a 30-MB video input for each model. Notably, OpenPose requires at least ten times the amount of time compared to the other two models, averaging at about 26.5 min for a 30-MB input. Although frame differencing is the fastest method at 0.83 min for a 30-MB input, the average wait for a 30-MB input during cascade classifier implementation is still relatively manageable at 2.69 min.

TABLE I. Comparison of Runtimes

Video Input	**Motion Detection Model**		
	Frame Differencing	***Cascade Classifier***	***OpenPose***
Walk1 (19.2 MB)	0.53 min	1.71 min	19.42 min
Drunk1 (26.6 MB)	0.67 min	1.97 min	20.86 min
Walk2 (31.8 MB)	0.76 min	3.24 min	31.66 min
Drunk2 (36.1 MB)	1.21 min	3.38 min	26.59 min
Average Runtime for a 30-MB Video	0.8266 min	2.6898 min	26.4587 min

B. Performance

Fig. 3 compares the performance of each of the three models for both normal and drunk walking. Interestingly, the curves for the cascade classifier and the OpenPose center are similar for both modes of walking. Evidently, the frame differencing method is the least accurate. This may be the result of generating a background model from the median of randomly selected frames; normal walking has more frames with the subject aligned with the tape, so randomly selected frames will capture

the alignment as the background model and greatly emphasize any slight deviations from the tape. Conversely, for drunk walking, more erratic movements within frames are selected as the background, so measurements indicating unstable gait are dampened since the erratic motions are normalized by the method.

Evaluating runtime and performance, frame differencing is not the best choice for motion detection. For real-time performance, cascade classifiers are appropriate. Otherwise, for more accuracy, OpenPose is an appropriate selection. For gait criteria, the footwork centers of the OpenPose model will be analyzed.

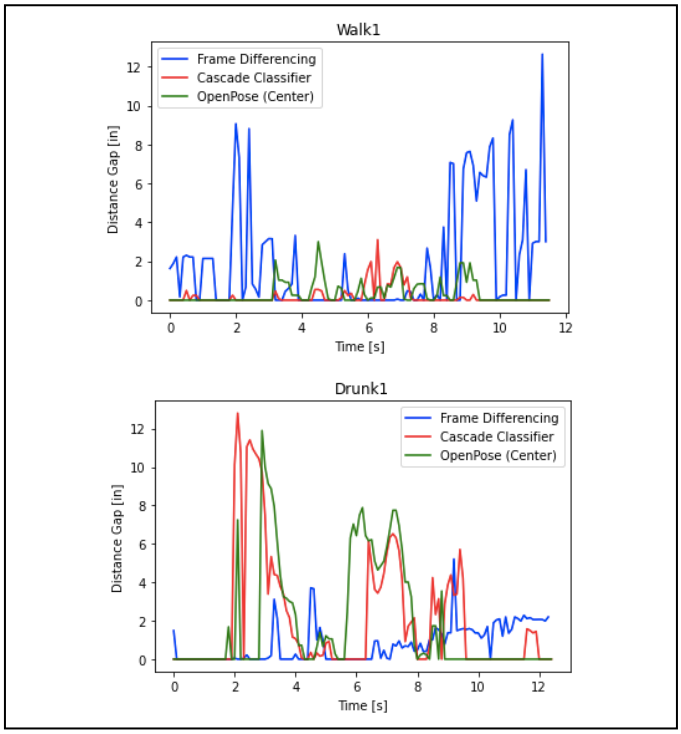

Fig. 3. Performance of Motion Detection Models. *The estimated distances for both normal walking (top) and drunk walking (bottom) are shown for each of the three motion detection models.*

C. Gait Criterion: Gait Cadence

Interestingly, when asked to walk straight along a line, subjects have a tendency to sweep their feet between steps. That is, from one step to the next, the moving foot tends to make an arc with the line to the next step. While this behavior was unforeseen, the foot-sweeps prove to be useful in examining gait cadence.

For analysis, the step time consistency ratio (STCR) proposed by the UH group is revisited [1]. Rather than using acceleration magnitude peaks, the foot-sweeps are used to estimate step times, or the times between each step. Peak detection is applied to the OpenPose footwork-center curve by locating relative maxima. The step times are then calculated by taking the index difference of the next step and the current step.

A step time t_{step} is considered consistent if

$$t_{step} \in \left[(1-\alpha)\mu_{step}, (1+\mu)\mu_{step}\right], \tag{2}$$

where $\mu_{step} = E\left[t_{step}\right]$ and α is a tweaking factor, usually set to a low value like 0.2. The STCR then is ratio of the number of consistent steps to the total number of steps. As discovered by the UH group, if the gait cadence criterion $STCR \geq 0.6$ is satisfied, then the corresponding gait is stable on the basis of consistency.

In Fig. 4, the OpenPose footwork-center curves for normal and drunk walking are plotted, with peaks indicating foot-sweeps marked. For normal gait, the step times appear more consistent as the peak-to-peak gaps are more evenly spaced out than the drunk gait patterns. For this particular case, normal walking has an expected step time of 0.59 seconds and a 0.60 STCR. By the gait cadence criterion, the normal walk is corroborated. On the contrary, drunk walking has an expected step time of 0.63 seconds and a 0.09 STCR. The extremely low STCR indicates that the drunk gait indeed is unstable.

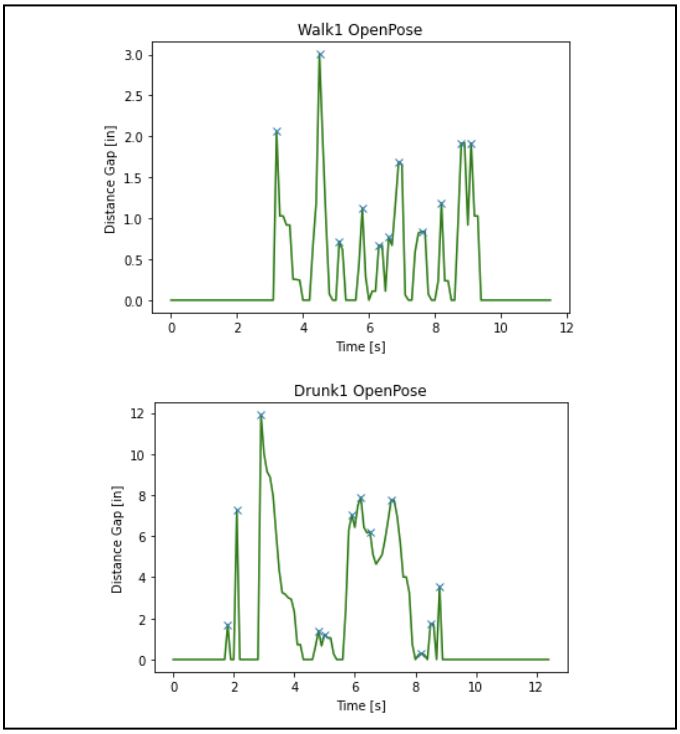

Fig. 4. Peak Detection for Foot-Sweeps. *The peaks resembling the foot-sweeps during normal (top) and drunk (bottom) gait curves are marked.*

D. Gait Criterion: Linear Steadiness

A new criterion not yet explored in the undergraduate project is examined. With imaging, the distance from the line per step can be recorded and analyzed.

Recall that even a person normal-walking along a line will still have subtle foot-sweeps, though not as exaggerated as unstable gait. To compensate for the foot-sweeps, a threshold of 8 cm (or 3.14 in) is set such that

$$d_{new} = \begin{cases} 0, & d < thresh \\ d, & d \geq thresh \end{cases} \tag{3}$$

Additionally, since machines cannot perceive depth within a 2D image like the human brain can, it is harder to determine

whether a raised foot mid-step will progress to a step forward or backward. Therefore, instead of using step times, instances of time will be used. A steady instance is then defined as a point in time in which the footwork-center falls within the mean of the distances d_{new} defined by Eq. 3. Therefore, the linear steadiness criterion can be given by the steady-to-total-instances ratio (STIR).

As seen in Fig. 5, the thresholded normal gait curve for OpenPose footwork-center essentially goes to zero, indicating that the gait is indicative of high stability. In fact, the calculated STIR is 1, which essentially is perfect. On the other hand, a significant portion of the thresholded drunk gait curve is still well over the mean thresholded-distance gaps, which is an indicator of unstable motion. This is corroborated by its calculated STIR of 0.744, which is not high enough.

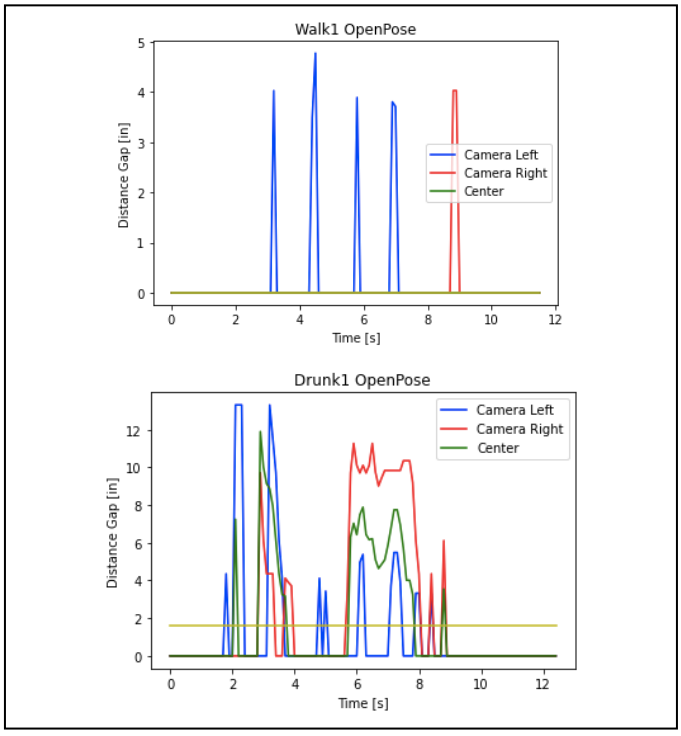

Fig. 5. Distance Gaps Post-Thresholding. *The yellow line represents the mean thresholded distance gaps for normal (top) and drunk (bottom) gait. Thresholded camera left, camera right, and footwork-center curves are plotted.*

IV. Discussion

When comparing the performance of the motion detection models, it is evident that OpenPose performs best in accuracy. However, the Haar cascade classifier is an appropriate medium in the sense that it has potential for fast (and possibly real-time) detection. On the other hand, since physicians already send patients' tests for lab work, using OpenPose for analysis might not be too farfetched, considering some lab work takes hours or even days to complete.

In fact, since OpenPose already isolates appendages in its model, other body parts can be tracked. For instance, tracking the hip joints and knees could be beneficial in analyzing rotational steadiness. Examination of the spine in the model could be used for assessment of balance; ideally, a well-balanced person would have the spinal line segment deviate at zero degrees from the direction of the tape line. Theoretically, by using the tape a baseline, one can measure the angle from the vertical for the hip joints, knees, and spine to assess the wobbliness of a patient and determine the source of such instability.

Of course, a case study may not be enough to demonstrate the efficacy of image-based standardizations for the TUG test. More subjects would need to be studied, preferably during a non-pandemic situation when there is less wary due to social distancing. Additionally, a better method for normalizing the distance gaps as discussed in Section II.F could be proposed, perhaps based on the lens length since it is known that the tape must be 3 meters long. To minimize foot-sweep detections, parallel tape lines can be used so that footsteps fall within the two lines; however, foot-sweeps proved to be beneficial in gait cadence calculations, so such a modification may not be necessary.

Previously, inertial data analysis can quantify a patient's stability based on the criteria of gait cadence and rotational steadiness in near instantaneous time. The drawback is that the method cannot pinpoint the exact source for instability. Image analysis shows a promise in quantifying gait cadence and linear steadiness and potentially rotational steadiness in the future, should other appendages be isolated and tracked. Of course, the drawback is the lack of depth perception within 2D estimation as well as the slow processing time. Naturally, combining the two methods of analysis can lead to the best of both worlds, in which the collective information from both methods can reveal more about a subject's stability and, one day, their health.

Appendix

The IMU-based device as proposed at UH is worn on the center of the waist of the subject. The magnetometer alone can determine when the subject is turning around during the TUG test. As seen in Fig. 6, the valley within the curve represents the moment the person turns around, whereas the plateaus show the instances when they are walking along the same direction.

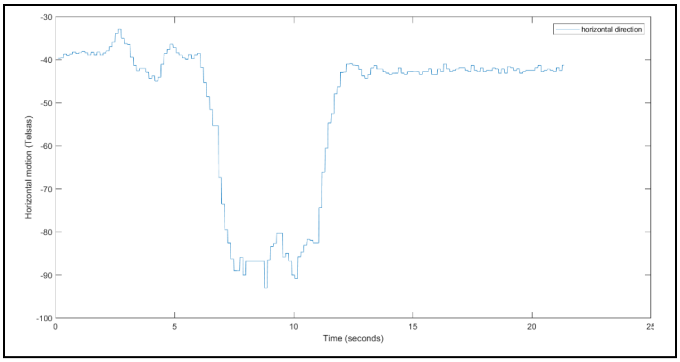

Fig. 6. Horizontal Direction. *The horizontal direction as measured by the magnetometer is depicted.*

The accelerometer detects changes in the three-axial linear accelerations. By plotting the acceleration magnitudes as seen in Fig. 7, footsteps are extracted by using peak detection of the curve. STCR analysis can then be done to determine stability based on gait cadence.

The accelerometer detects changes in the three-axial linear accelerations. By plotting the acceleration magnitudes as seen in Fig. 7, footsteps are extracted by using peak detection of the curve. STCR analysis can then be done to determine stability based on gait cadence.

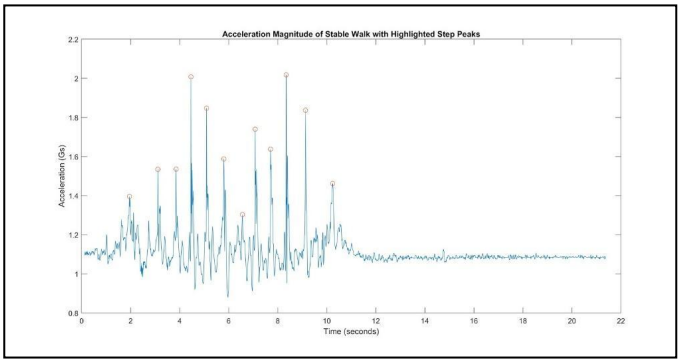

Fig. 7. Acceleration Magnitude Peak Detection. *The peaks of the magnitude acceleration are marked.*

Lastly, the gyroscope is able to determine rotational stability following processing. As seen in Fig. 8, a person's stable walk is indicative by the relatively constant rotational velocities along two axial directions, with an elevation in the third direction to indicate that the subject has turned around.

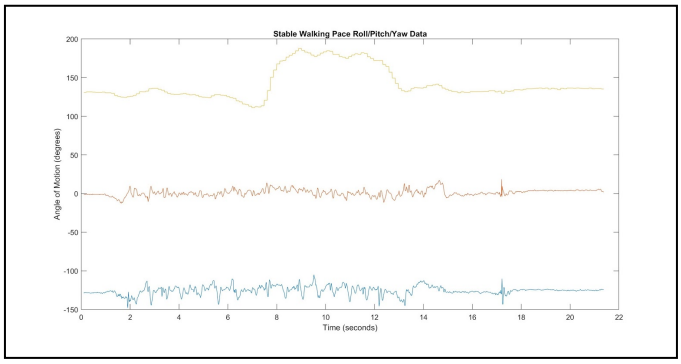

Fig. 8. Gyroscopic Steadiness. *The rotational velocities along the three-axial directions are depicted.*

ACKNOWLEDGMENT

The author would like to thank Rachel Glass and Nancy Nguyen for their assistance with filming the TUG test. This work was made possible by Dr. Zixiang Xiong and his Digital Image Processing & Computer Vision course at Texas A&M University.

REFERENCES

[1] M. Al-Subbagh, P. Comeaux, G. Nguyen, P. Replogle, "A Report on the Development of a Wireless Walking Pace Monitoring System," *In UH-BME*, unpublished, 2019.

[2] G. Thapa, K. Sharma, and M.K. Ghose, "Moving Object Detection and Segmentation using Frame Differencing and Summing Technique," *International Journal of Computer Applications*, vol. 102, pp. 20-25, Sept. 2014.

[3] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[4] G. Bradski, "The OpenCV Library," *Dr Dobb's Journal of Software Tools*, 2000.

[5] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[6] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand Keypoint Detection in Single Images using Multiview Bootstrapping," *In CVPR*, 2017.

[7] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," *In CVPR*, 2017.

[8] S. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," *In CVPR*, 2016.