A Direct Reading Video Assessment Instrument for Repetitive Motion Stress

Final Report

August 31, 2020

Principal Investigator: Robert G. Radwin, PhD, University of Wisconsin-Madison, 1550 Engineering Drive, Madison, WI 53706, 608-263-6596, rradwin@wisc.edu

Co-Investigator: Yu Hen Hu, PhD, University of Wisconsin-Madison

Co-Investigator: Guanhua Chen, PhD, University of Wisconsin-Madison

Co-Investigator: David Rempel, MD, University of California, San Francisco

Co-Investigator: Carisa Harris Adamson, PhD, University of California, San Francisco

Co-Investigator: Stephen Bao, Washington State Department of Labor and Industries

Co-Investigator: Jia-Hua Lin, Washington State Department of Labor and Industries

Start and End Dates: 9/1/2016- 8/31/2020

Title: A Direct Reading Video Assessment Instrument for Repetitive Motion Stress
Investigator: Robert G. Radwin, University of Wisconsin-Madison, Madison, WI 53706 (Email: rradwin@wisc.edu)
Affiliation: University of Wisconsin-Madison
State: WI
Telephone: (608) 263-6596
Award Number: R01 11024
Start & End Dates: 9/1/2016- 8/31/2020
Sector Program: All
Cross-Sector Program Area: Musculoskeletal Disorders and Exposure Assessment

Final Report Abstract:

This research studies if computer vision can more effectively evaluate worker exposure and assess the associated risk for work related injuries than conventional methods. Previous methods involve either observations or measurements using instruments attached to a worker's hands or arms. Observation is often considered too subjective or inaccurate, and instruments too invasive or time consuming for routine applications in industry. Automated job analysis potentially offers a more objective, accurate, repeatable, and efficient exposure assessment tool than observational analysis. Computer vision uses less resources than instruments attached to workers and does not interfere with production; can quantify more exposure variables and interactions; is suitable for long-term, direct reading exposure assessment; and offers animated data visualizations synchronized with video for identifying aspects of jobs needing interventions.

This research leverages the research from coordinated multi-institutional prospective studies of upper limb work related MSD conducted between 2001 and 2010 that studied production and service workers from a variety of US industries, and used rigorous case-criteria and individual-level exposure assessments prospectively, including recording detailed videos of the work. Our study partners from the National Institute for Occupational Safety and Health, the Washington State Labor & Industries Safety & Health Assessment & Research for Prevention program and the University of California-San Francisco will provide task-level videos, associated exposure variable data, and prospective health outcomes for 1,649 workers.

This research refined and developed video algorithms and analyzed the videos to extract exposure measures for repetition, posture, exertions, and their interactions. In addition, new algorithms were developed for measuring and calculating wrist angle and parameters for manual lifting. The video extracted exposure measures were compared against conventional observational exposure measures made by our collaborators. Video and corresponding observational data were merged with the prospective health outcomes data to evaluate dose-response and to develop and validate parsimonious exposure risk models for an automated direct reading repetitive motion instrument. The completion of the modeling phase of this project is ongoing. This translational research is in concurrence with the Research to Practice (r2P) initiative by developing technology to disseminate knowledge from recent NIOSH sponsored prospective studies on MSDs.

Table of Contents

List of Terms and Abbreviations:

3D Static Strength Prediction Program (3DSSPP)

American Conference of Governmental Industrial Hygienists (ACGIH)

Bureau of Labor Statistics (BLS)

Decision tree (DT)

Duty cycle (DC)

Duty Cycle (D)

Feature vector training (FVT)

Frequency (F)

Frequency Independent Recommended Weight Limit (FIRWL)

Hand Activity Level (HAL)

k nearest neighbor (kNN)

National Institute for Occupational Safety and Health (NIOSH)

Multimedia video task analysis (MVTA)

National Institute for Occupational Safety and Health (NIOSH)

Region of Interest (ROI)

Revised NIOSH lifting equation (RNLE)

Recommended Weight Limit (RWL)

Threshold Limit Value (TLV)

Section 1:

Significant or Key Findings

A marker-less 2D video algorithm measured hand kinematics (location, velocity, and acceleration) in a paced repetitive laboratory task for varying hand activity levels (HAL). A decision tree (DT) algorithm identified the trajectory of the hand using spatiotemporal relationships during the exertion and rest states. A feature vector training (FVT) method utilized the k-nearest neighborhood classifier, trained using a set of samples or the first cycle. The average duty cycle error using the DT algorithm was 2.7%. The FVT algorithm had an average 3.3% error when trained using the first cycle sample of each repetitive task and had a 2.8% average error when trained using several representative repetitive cycles. Error for HAL was 0.1 for both algorithms, which was considered negligible. Elemental time, stratified by task and subject, were not statistically different from ground truth ($p < .05$). Both algorithms performed well for automatically measuring elapsed time, duty cycle and HAL.

Task level physical exposure variables measured using frame-by-frame multimedia video task analysis (MVTA) were compared with video computer vision measures for jobs selected from a selected subset of the Upper Limb MSD Consortium prospective study. The occurrence of each exertion was first identified in all the videos by human analysts for calculating the frequency (exertions/ second) and duty cycle (percent exertion time/ cycle time). The hands were tracked using marker-less video tracking and a feature vector algorithm was trained using the first cycle exertions identified by an analyst, for automatically estimating subsequent exertions in the videos. Frequency and duty cycle were estimated for 111 tasks. The average difference was 3.69% for duty cycle and 0.11 Hz for frequency. There was less than ±10% error for more than 75% of the tasks, resulting in a HAL error less than 1 unit, which is considered negligible.

Hand Activity Level (HAL) can be estimated from observations (HAL_O), calculated from exertion frequency, F and duty cycle D (HAL_F) or from speed, S and D (HAL_S). Data collected by prospective cohort studies were used to compare these methods. There was 75% agreement between HAL_O and HAL_S ($HAL_S=1.02\times HAL_O -0.2$, $R^2=0.78$, $F(1,1003)=43665$, $p <.001$), but only 30% agreement between HAL_O and HAL_F ($HAL_F=0.21\times HAL_O+2.2$, $R^2=0.04$, $F(1,1003)=71.71$, $p <.001$). HAL_S was more consistent with HAL_O since both are dependent on speed, and because HAL_S can be automated, it is more objective than HAL_F or HAL_O in this sample.

An effective computer vision method was developed to measure wrist flexion from 2D video for occupational health and safety research. A three-point feature tracking algorithm was used to estimate the wrist flexion angle between the back of the hand and the forearm. Based on the estimated angles, wrist posture was classified as flexion (palmar bending), neutral (no bending), or extension (dorsal bending) for each cycle of hand movement. Using videos of a simulated repetitive motion task, we demonstrated the feasibility of the proposed algorithms for assessing the frequency of hand activities during work. Tested on 1488 frames from 61 videos of repetitive tasks for 16 participants, the algorithm achieved performance of 73.12% average correct per-class accuracy. The sensitivity of flexion and extension are 77.33% and 74.62%. The specificity of flexion and extension are 73.00% and 74.37%.

A method for automatically classifying lifting postures from simple features in video recordings was developed and tested. We explored if an “elastic” rectangular bounding box, drawn tightly around the subject, can be used for classifying standing, stooping and squatting at the lift origin and destination. Current marker-less video tracking methods depend on a-priori skeletal human models, which are prone to error from poor illumination, obstructions and difficulty placing cameras in the field. Robust computer vision algorithms based on spatiotemporal features were previously applied for evaluating repetitive motion tasks, exertion frequency, and duty cycle. Mannequin poses were systematically generated using the Michigan 3DSSPP software for a wide range of hand locations and lifting postures. The stature-normalized height and width of a bounding box was measured in the sagittal plane and when rotated horizontally by 30º. After randomly ordering the data, a classification and regression tree algorithm was trained to classify the lifting postures. The resulting tree had

four levels and four splits, misclassifying 0.36% training-set cases. The algorithm was tested using 30 video clips of industrial lifting tasks, misclassifying 3.33% test-set cases. The sensitivity and specificity respectively, were 100.0% and 100.0% for squatting, 90.0% and 100.0% for stooping, and 100.0% and 95.0% for standing. The tree classification algorithm is capable of classifying lifting postures based only on dimensions of bounding boxes.

A widely used risk prediction tool, the revised NIOSH lifting equation (RNLE), provides the recommended weight limit (RWL), but is limited by analyst subjectivity, experience, and resources. A robust, non-intrusive, straightforward approach was developed to automatically extract spatial and temporal factors necessary for the RNLE using a single video camera in the sagittal plane. The participant's silhouette is segmented by motion information and the novel use of a ghosting effect provides accurate detection of lifting instances, and hand and feet location prediction. Laboratory tests using 6 participants each performing 36 lifts showed that a nominal 640 pixel × 480 pixel 2D video, in comparison to 3D motion capture, provided RWL estimations within 0.2 kg (SD=1.0 kg). The linear regression between the video and 3D tracking RWL was R^2=0.96 (slope=1.0, intercept=0.2 kg). Since low definition video was used in order to synchronize with motion capture, better performance is anticipated using high definition video.

Translation of Findings

A completely automated approach for measuring elapsed time and duty cycle was developed using marker-less video tracking and the tracked kinematic record. Such an approach is automatic, repeatable, objective, unobtrusive, and is suitable for evaluating repetitive exertions, muscle fatigue, and manual tasks. The algorithm was programmed, and a working prototype was developed for field usage. It is anticipated that this practical algorithm can be implemented on hand-held devices such as a smart phone, making it readily accessible to practitioners.

An algorithm for automatically calculating the revised NIOSH lifting equation using a single video camera was evaluated in comparison to laboratory 3D motion capture. The results indicate that this method has suitable accuracy for practical use and may be particularly useful when multiple lifts are evaluated. We have developed a computer vision lifting monitor for automatically quantifying parameters required for LBP risk analysis using digital video algorithms. The method identifies when every object is lifted, carried, and released from the hands and automatically measures the parameters for calculating the RNLE. This level of quantification is highly impractical and resource intensive for industry practitioners today. The technology addresses these issues by potentially saving practitioners substantial time and providing accuracy, objectivity, and consistency.

Research Outcomes/Impact

Impact was measured by scientific publications and patents applied for arising from inventions as an outcome of this research. Eight conference proceeding articles, four journal articles, and one doctoral thesis were published, to date. Since all of the data has not yet been fully analyzed, additional publications are anticipated in the near future. Two US patents have also been applied for, to date related to this research.

A database of exposure measures based on computer vision, observational video analysis, and related health outcomes was created. Data cleaning has been performed and currently data merging is still underway. Portions of this database has been part of publications and conference proceedings, while additional analysis is still underway.

Section 2

1. Measuring Elemental Time and Duty Cycle Using Automated Video Processing

1.1. Introduction

Duty cycle (DC) is one of the primary measures used in ergonomics for evaluating repetitive exertions, muscle fatigue, and manual materials handling tasks. It is defined as the proportion of time spent in actual task-related activities, and is typically calculated as the exertion time divided by the total time spent doing the task, including rest periods. DC has become an important metric for quantifying work activity and manual exertions in the workplace for optimizing work-rest periods for preventing fatigue and injuries.

DC has played a major role in predicting fatigue and its prevention (Rohmert 1973a; Rohmert, 1973b; Hagg and Milerad, 1997; Ding et al., 2002). Rohmert (1973a) described the phenomena of fatigue and recovery to determine work-rest cycles and their influence on workers' strain and stress. A study on continuous and intermittent isometric contractions linked load and work/rest ratio limits to indicators of localized muscle fatigue (Bystrom et al. 1994). Woods at al. (1997) considered the work-rest allowance effect on muscle fatigue and predicted an optimum work-rest schedule to minimize this effect. Potvin (2012) more recently created an equation based on DC from a meta-analysis of numerous studies in the literature to estimate maximum acceptable force in repetitive manual tasks.

DC has also been used for quantifying exposure to physical stress in repetitive manual work. The strain index, which is used for evaluating repetitive manual tasks, included DC as one of its parameters (Moore & Garg, 1995). A prospective study of biomechanical risk factors for carpal tunnel syndrome found DC for forceful hand exertions was a significant risk factor (Harris-Adamson et al., (2014). Latko at al. (1997) introduced a method for quantifying repetitive hand motion, the hand activity level (HAL), which was also related to DC.

HAL applies to mono-cycle tasks and is an observational visual-analogue metric which an observer subjectively assesses on a scale from 0 to 10, anchored between the "hand idle most of the time and no regular exertions" and "rapid, steady motions/exertion; difficulty keeping up." The HAL scale is part of the American Conference for Government Industrial Hygienists (ACGIH) threshold limit value (TLV™) for evaluating the risk of work related distal upper extremity musculoskeletal disorders (ACGIH Worldwide, 2001). The HAL rating can also be evaluated by directly measuring the exertion frequency and DC against a look-up table. Radwin et al. (2015) developed an equation for estimating HAL from these parameters as an alternative to the TLV™ table.

Common ways for measuring DC include time and motion studies, manual video coding, observational methods, and self-reports. Among these, times studies and video coding are the most accurate, whereas observations and self-reports lack consistency and reliability (Fan 2014). Bao et al. (2006) observed disagreement between observer rated frequency-DC estimates and detailed time study analyzes. In another study, Garg and Kapellusch (2011) discuss the lack of consistency between the methods of evaluating HAL, from observer rated assessment and table look-up values, and address the need for a consistent method of evaluation. Kapellusch at al. (2013) stated a similar need for a robust technique. Moreover Wells at. al. (2007) explains that estimating exposures related to time is difficult, in which self-reports are inconsistent while direct measurements are time and resource consuming. An objective, automated method for evaluating DC could help resolve these issues.

Yen and Radwin (1999) looked at using signal pattern recognition for automatically quantifying cyclical tasks from wrist electrogoniometer signals and concluded that such an approach may be useful, using interactive fine-tuning. Recent advances in computer vision enable HAL to be measured non-invasively without instruments, using automated video processing that employs semi-automatic marker-less tracking of a region

of interest (ROI) located near the hand in order to measure frequency and duty cycle (Chen et al., 2013). Such an approach is automatic, repeatable, objective, unobtrusive, and is suitable for a real-time, direct reading assessment.

We previously demonstrated that hand root mean square (RMS) speed while exerting force, and DC measures were well suited for automatically estimating HAL and had good agreement with independent observational ratings of videos for actual industry jobs (Akkas et al., 2015) using DC obtained manually using Multimedia Video Task Analysis™ (MVTA™) frame-by-frame analysis (Yen and Radwin, 1995). We hypothesize that a completely automated approach for measuring DC could be achieved using the tracked kinematic record. The current study advances an automatic method to measure DC using marker-less video tracking.

Chen et al. (2013) previously calculated DC for a repetitive load transfer task performed in the laboratory. In this method, the local minima of absolute velocity values were first identified. If the acceleration between successive local minima points exceeded a preset threshold, it was determined that the hand was loaded during the period between the pair of local minima points. Such a definition of hand loading was based on observations made for the specific load-transfer task (moving a lead filled bottle from a tray to a rotating turntable). The DC values obtained using that approach were 1.27 times greater than those measured manually using MVTA (R^2 = 0.63).

We advance automatic measure of DC by studying two different approaches: (1) a decision tree (DT) algorithm and (2) a feature vector training (FVT) algorithm. Both methods utilized kinematic properties (i.e. location, velocity and acceleration) of a video marker-less tracked ROI. Each method was developed and used for estimating DC from a simulated repetitive hand intensive task performed in the laboratory. Ground truth time measurements of DC were ascertained utilizing manual frame-by-frame MVTA analysis and compared.

1.2. Methods

1.2.1 Task Simulation

In order to develop and test the algorithms for automatically measuring DC, we simulated a prototypical repetitive motion task in the laboratory. A subject grasps a ball from one location, moves it to a specified location, releases it, and reaches for another ball. An apparatus (Figure 1) was fabricated using an electromechanical linear actuator for indexing the balls that are obtained and deposited for a paced sequence. The device is comprised of an 840 mm travel length linear belt drive actuator (Misumi MSS-625) driven by a bi-polar stepper motor (ElectroCraft Model TPP34 with 560 N cm torque) and controlled by a stepper motor controller (IMS MX-CS101-401). The device was capable of moving a 2 kg object every 0.5 s across the actuator length of travel.

The participant stands in front of the apparatus, gets a ball at point A, and transfers it to another specified location (Figure 2). The balls weighed 59 g and had a 6.5 cm diameter. Distance was calibrated against a measured grid in the image (Figure 2).

A fourth generation i7 quad-core computer recorded video of the task performed. A Logitech C920 fixed focal length web camera was used for imaging the color video stream that was stored as an AVI video file (Xvid compressed) with 640 $\times$ 480 pixels resolution at the frame rate of 30 frames per second.

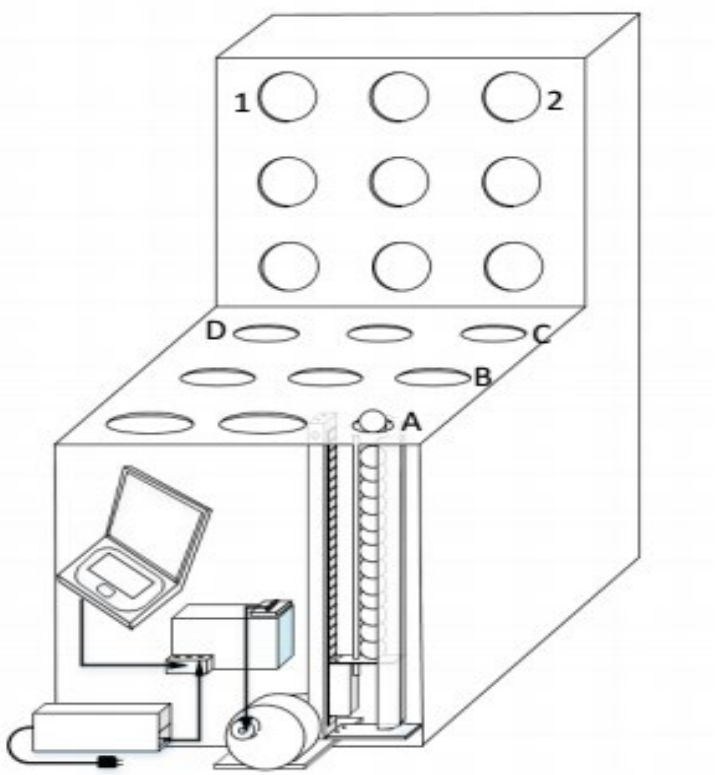


Figure 1.1. Laboratory task simulation apparatus. The participant grasps a ball from location A and moves it to either locations B, D, 1, or 2, depending on the specified task.

Estimation of DC requires identifying instances when the hand is loaded and exerts force against an object in a work cycle. In this simulated task, exertions occur during the time elapsed after the ball is grasped, moved, and until it is released. The sequence of this task can be described as Reach-Grasp-Move-Release- (Figure 3). After the subject grasps a ball, the Move element starts and continues until the ball is released. We further define the sequence Move--Release as “Put”, and the sequence Reach--Grasp as “Get” (Figure 3). Thus Put time represents exertion time, Get time represents rest time when the hand is not interacting with the object, while the total task time cycle time is the sum of the elapsed Put and Get times. The DC for the task is therefore the percentage time: Put / (Put + Get).

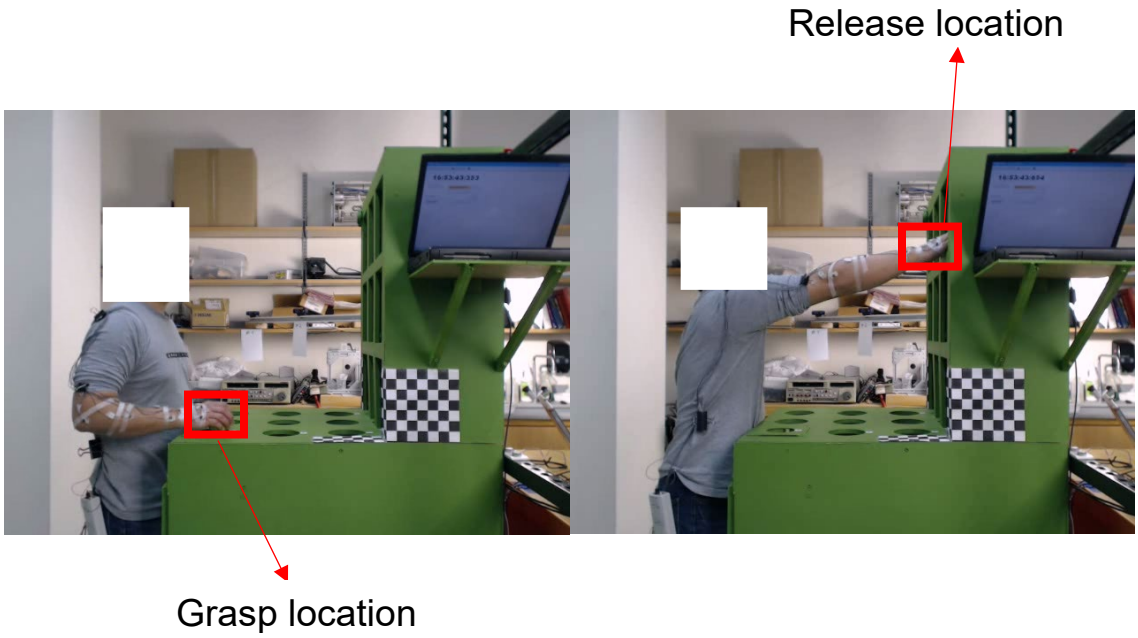


Figure 1.2. Representation of grasp and release location

We recruited 19 university student volunteers (6 males and 13 females) with informed consent and IRB approval. They were each video recorded while performing 15 cycles of the Get-Put task paced at various

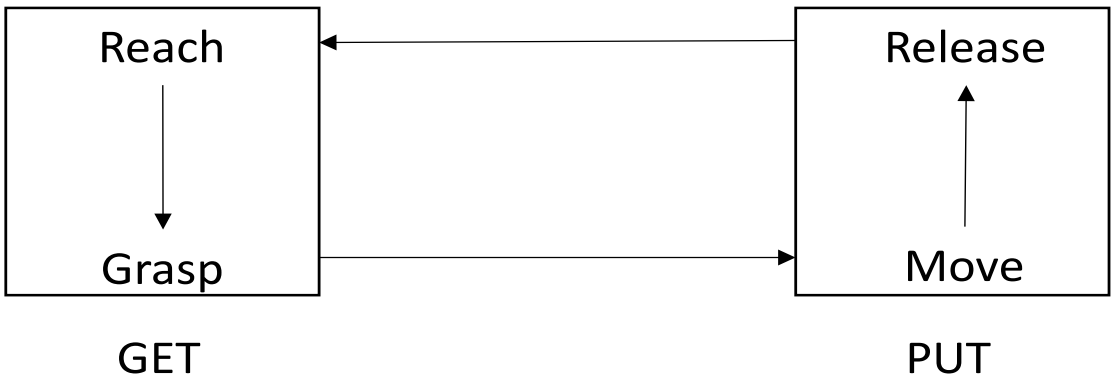


Figure 1.3. Graphical representation of the simulated task

frequencies; exertion and rest times were controlled using an auditory cue. The paced frequencies and DC for each task are given in Table 1. We calculated the paced HAL values for each task based on these frequencies and DC using the equation for HAL in Radwin, et al. (2015), which are also provided in Table 1. The observed HAL for each participant in every task was calculated from frequency and DC measured using MVTA (Table 1). These HAL values were used as ground truth measures for testing the computer vision algorithms.

Each subject performed 10-paced tasks in random order. A set of practice tests were provided for each condition. A one-minute rest was provided between each condition to prevent fatigue. Each video clip had a length ranging from 10 seconds to 80 seconds, and consisted of 15 cycles of task execution. Some participants were unable to accomplish every pace and those cases were excluded from the analysis. This

was due to combination of challenging experimental conditions or failures. Failures occurred when the subject missed or dropped the ball. The total number of error-free video clips are listed in Table 2.

Table 1.1. Summary of tasks performed listed by location, pacing frequency and DC

Location	Frequency	DC	Paced HAL	Observed Ground Truth HAL		Number of Video Clips
	(Hz)	(%)		Mean	SD	
A to 1	0.25	65	2.9	2.9	0.1	18
A to 1	0.75	60	5.8	5.6	0.2	19
A to 2	0.50	20	3.5	3.9	0.2	16
A to 2	1.25	47	6.4	6.5	0.2	18
A to D	1.25	47	6.4	6.5	0.2	19
A to D	1.50	25	5.6	6.7	0.3	23
A to C	1.00	15	4.2	5.6	0.4	17
A to C	1.50	25	5.6	6.7	0.3	22
A to C	1.75	25	5.8	6.9	0.2	18
A to B	1.75	25	5.8	6.8	0.3	19

1.2.2 Training and Test Data

In developing the algorithms, the entire video data were divided into non-overlapping training and test data sets. A total of 41 video clips of the repetitive laboratory task performed by the first 5 subjects (nine clips were excluded due to failures) were reserved for the training data and used for training and validating the developed algorithms. There were 87 video clips available for the remaining 14 subjects (53 clips were excluded due to task incompleteness or failures) were reserved as the test data set. The same data partitioning was applied to both the DT algorithm and the FVT algorithm. The training videos were used for the DT algorithm to facilitate manually tuning the threshold parameters in order to achieve the best performance. The training videos for the FVT algorithm were first used to identify features and then they were used to develop the algorithm.

Actual factory jobs involve different workflows so in practice few training video clips may be available for training the algorithm. Therefore, in developing the FVT method, we also experimented with a *first cycle* data partition method. Specifically, we manually labeled the Get and Put elements for the first cycle of the repetitive task in the video and used it for training the FVT algorithm. Then, we tested the algorithm on remaining cycles in the video as the test data.

1.2.3 Ground Truth Data

Trained analysts extracted ground truth DC measures from the videos of each task using singe frame video coding and MVTA software. They marked each frame when they identified a change from Get to Put or from Put to Get. After a frame was marked, all the remaining frames were marked using the same label until the

next change occurred. The start of an exertion (i.e. start of Put) was identified as the instant when the hands contacted the ball while the end of and exertion (i.e. start of Get) was identified as the instant when the ball no longer made contact with the hand.

1.2.4 Feature Vectors

The hand location on each video frame was tracked using the marker-less video tracking algorithm described in Chen et al. (2013; 2015) and Chen, Hu & Radwin (2014). The analyst initially identifies a rectangular region of interest (ROI) covering the image of the entire hand, which was tracked for each frame by the computer. A cross-correlation template matching tracking algorithm tracks the ROI center trajectory (x_i, y_i) over subsequent video frames.

Based on the trajectory, other kinematic features may be derived:

Velocity $v_{x,i} = (x_{i+1} - x_{i-1})/2\Delta$, $v_{y,i} = (y_{i+1} - y_{i-1})/2\Delta$,

Speed $|v_i| = \sqrt{(v_{x,i})^2 + (v_{y,i})^2}$;

Acceleration $a_{x,i} = (x_{i+1} - 2x_i + x_{i-1})/\Delta^2$, $a_{y,i} = (y_{i+1} - 2y_i + y_{i-1})/\Delta^2$

Acceleration Magnitude $|a_i| = \sqrt{(a_{x,i})^2 + (a_{y,i})^2}$.

We also computed the spatiotemporal curvature. The curvature function *k* is sensitive to the change of the direction of a curve relative to its arc length. It is defined as;

$$k = x\frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}}$$

where (x(t), y(t)) is a planar curve (Flanders et al., 1970; Flanders et al., 1974). This function is useful for detecting discontinuities in a movement based on velocity, acceleration and position (Rubin and Richards, 1985). The spatiotemporal curvature index peaks when there is a significant change in the action, such as changing direction, stopping or starting to move. These changes correspond to instances in motions such as grasps or releases (Rao et al., 2002). The curvature index K_i is:

$$K_i = \frac{\sqrt{a_{x,i}^2 + a_{y,i}^2 + (v_{x,i}v_{y,i} - a_{x,i}a_{y,i})^2}}{(\sqrt{a_{x,i}^2 + a_{y,i}^2 + 1})^3}.$$

A representative set of feature vectors for the laboratory task are shown in Figure 4, including corresponding location, velocity, acceleration, and spatiotemporal curvature.

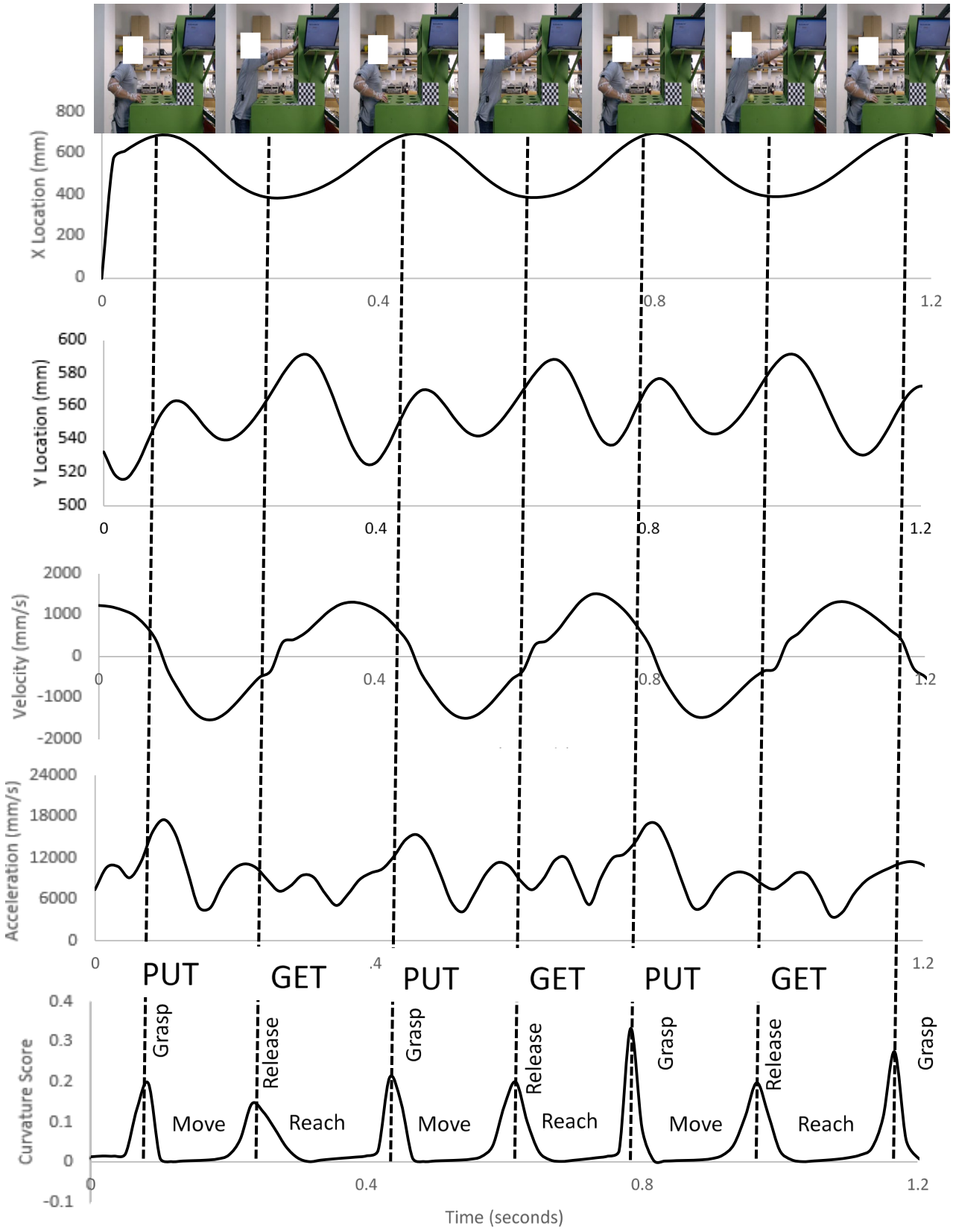


Figure 1.4. Representative hand kinematic feature curves for location, velocity, and acceleration aligned with the spatiotemporal curvature score measured using ROI marker-less video tracking.

1.2.5 Duty Cycle and HAL Measurement

The DC estimation process was divided into two phases. In the first phase, the hand movement in each video frame was classified as one of two states (Get and Put). The DC for the n^{th} cycle DC_n was estimated as:

$$DC_n = \frac{No.\,frames\ in\ Put\ State\ (\ n^{th}\ cycle)}{No.\,frames\ in\ Put\ State\ (\ n^{th}\ cycle) + No.\,Frames\ in\ Get\ State\ (\ n^{th}\ cycle)}$$

Since each video consisted of multiple cycles, multiple estimates of the DC were measured. The overall DC estimates were evaluated using the average of these individually estimated DC.

After the average DC estimate was obtained for a given task and participant, the corresponding *HAL* can be calculated using the equation from Akkas at. al. (2014).

$$HAL = 10 \cdot \left[\frac{e^{-15.87+0.02\,DC+2.25\ln S}}{1+e^{-15.87+0.02\,DC+2.25\ln S}}\right]$$

where *s* is RMS speed in mm/s.

1.2.6 Decision Tree Algorithm

The DT algorithm first used the frame-by-frame curvature score to determine the transition from Move to Release or Reach to Grasp. In order to accomplish this, we identified the peak curvature scores and their associated frame numbers for a given video clip and then determined if the state changed to Reach or Move by utilizing the velocity and location signal (Figure 4). Note that in Figure 4 the *x* trajectory clearly exhibits a periodic sinusoidal waveform while the *y* trajectory, the speed, and the magnitude of acceleration contains multiple peaks in one cycle. Also note that the curvature score peaks when the hand motion changes directions. Hence, the curvature and *x* trajectory signals used as characteristic features in the DT algorithm.

The algorithm was developed by first testing several pilot videos in order to determine the threshold velocity for state changes between Get and Put, identified by the experimenter using the 41 training video clips. The least error was gained for 200 mm/s velocity and the same velocity is used for all the test data. The transition point (Grasp or Release), velocity was close to zero (Figure 4). But for some faster tasks we observed that velocity was close to zero but never reached zero. Given the video sampling rate was 30 frames per second and the velocity calculation involved the central difference approximation using a previous and next frame, it may not be possible to always observe zero velocity, especially for high speed tasks. Additional noise might also be introduced from the tracking algorithm.

To overcome this, we added location criteria to the algorithm. The goal was to predict locations where the subject grasps the ball and releases the ball. This involves finding densest area within the tracked locations. To do this we divided the region into bins and counted the number of points within each bin. We automatically determined the bin width and length per video, based on the size of the tracking region. The next step was to determine if the hand is inside one of these locations.

Knowing that the first state is a Grasp we were able to detect transition states (e.g. Release, Grasp) using the above logic. The state remains the same until the next transition occurs when a subject grasps a ball, and the Move state starts. All frames are annotated as Move until a Release state is observed. After the Release state is reached, all frames until the next Grasp are annotated as Reach. The algorithm continues to annotate each frame until it reaches the end of the clip.

We then calculated the time elapsed between each transition, providing both cycle time (Grasp to Grasp) and exertion time (Move to Release). The DC was obtained by dividing exertion time (time elapsed between Move and Release) by total cycle time to calculate the percent time in an exertion. The steps are described in

Algorithm 1. This algorithm was applied directly to the 87 test video clips from the remaining 14 subjects without further adjustment.

inputs: feature vectors, video frame number from hand tracking

output: states (Get or Put) for each video frame

1. Find local maxima curvature index values and associated frames.
2. For each frame, check if the frame has local maxima curvature index point.
3. If a frame is identified as a local maxima, then check the location that corresponds to the frame number to see if the location is inside a grasp region or release region. If the location is inside one of these regions then use the previous state to annotate as next state (Put if previous state was Get, or Get if previous state was Put).
4. If above criteria do not apply, check the corresponding velocity and compare to the velocity threshold. If the current velocity is under treshold, then use the previous state (Put if previous state was Get, or Get if previous state was Put).
5. If none of above criteria apply, then annotate as same state as the previous one.
6. The estimated duty cycle is computed by (# of Put)/((# of Get) + (# of Put))

Algorithm 1. Decision Tree Algorithm

1.2.7 Feature Vector Training Algorithm (FVT)

We have developed a FVT algorithm to automatically select a subset of features in order to optimize DC estimation performance. The FVT algorithm employed machine-learning procedures to systematically develop the duty cycle estimator and hence potentially demands less manual tuning efforts. This is desirable when large amounts of repetitive hand movement videos from different factory work environments are to be analyzed automatically.

We first used cross-validation and the maximum error of DC estimation as the performance criterion to judge whether a specific subset of nine features (Table 2) was most desirable. Maximum error was the absolute difference between the estimated DC and ground truth DC. Utilizing the 41 training videos, we employed a feature selection procedure to train the algorithm on 40 videos, and then tested the result on the remaining video in the set. We rotated this process for different video clips 41 times and averaged the performance evaluated on the testing video. Since there were $2^9 = 512$ combinations of the 9 features described above, it would be too tedious to try every combination exhaustively. Instead, we opted to use a greedy backward subset selection method described in Couvreur, (1999).

We start with evaluating the performance (maximum DC estimation error) using the entire set of nine features. Then we evaluate the performance of the nine different subsets of eight features each with one feature

excluded from the currently selected nine features. And we select the eight-feature subset that produces the best DC estimate. We then repeated this process in order to select the best seven feature subset out of the previously selected best eight feature subset. This process is repeated until there is only one feature remaining. We compared the accuracies of the estimated DC values from these selected subset of features and choose the one that yielded the best performance.

Using this process, the nine features are ranked from one to nine, with nine being the most important feature appearing in all nine subsets, and one being the least important feature discarded first when eight features were selected among the nine. The ranking of these nine features are given in Table 2. The performance (maximum error of DC estimates) of nine different feature subsets are plotted in Figure 5 where smaller DC estimation error indicates better performance. Based on these results, we selected the following subset of six features (x, y, v_x, v_y, $|v|$, K) to be used in the FVT training algorithm. We observed that v_x (rank = 9) was the most important feature and a_x (rank = 1) was the least important feature.

Table 1.2. Ranking of Selected Features

Feature*	x	y	v_x	v_y	$\|v\|$	a_x	a_y	$\|a\|$	K
Rank	8	5	9	7	6	1	2	3	4

* x and y are coordinates, v_x, and v_y are velocities, $|v| = \sqrt{v_x^2 + v_y^2}$, a_x and a_y are accelerations, $|a| = \sqrt{a_x^2 + a_y^2}$, and K is curvature index.

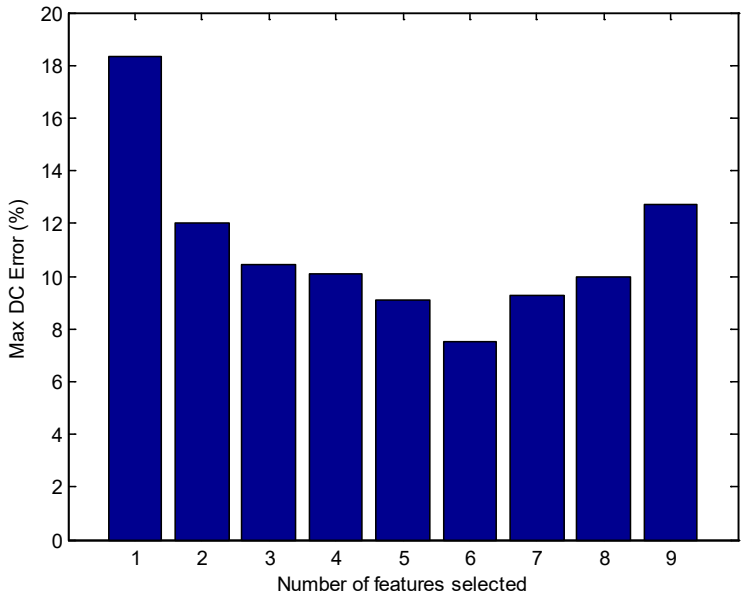


Figure1. 5. The performance of nine different feature sets based on maximum DC error, max |ground truth DC – estimated DC|.

Since these features had quite different ranges, in order to avoid one feature with large magnitudes dominating another feature, a normalization step was applied to ensure that the extracted feature values were approximately zero mean with a sample variance equal to unity.

In this work, we use the k nearest neighbor (kNN) classifier as the state estimator. In a kNN classifier, all training data (feature vectors) and corresponding labels (Get, Put) are stored in the classifier. For each test feature vector, a set of *k* (usually chosen as an odd number to facilitate majority voting) training vectors that are most similar to the test vector are identified using a similarity metric. Then the training vector dominant label (by majority voting of the *k* labels) is assigned to the test vector. In these experiments, we choose $k = 5$ when training with the 41 training videos, and used the Euclidean distance between the training and the test feature vectors as the similarity metric. When training with the first cycle samples, we choose $k = 1$ since in practice there may be few training samples available other than the first cycle.

As previously discussed, two different approaches of training and testing were applied to the FVT based algorithm. The first used the 41 video clip training data set and testing the kNN classifier with the 87 testing data set (kNN_r). The second approach, called first cycle training, used the feature vectors and corresponding labels of the first cycle of a test video clip as the training data and estimated the DC of the remaining cycles of the same test video clip (kNN_f). The 41 training video clips were not used by the kNN classifier in the first cycle training approach. Detailed steps of the training based DC estimation algorithm are summarized in Algorithm 2.

Training Phase

inputs: feature vectors, state labels (Get and Put)

1. Nomalize all features into zero mean and unit variance, and store the normalization factors for testing phase.
2. Train the k-nearest neighborhood classifier by using the chosen feature vectors and the MVTA labels of the training data.

Testing Phase

inputs: frame number, feature vectors

output: states (Get or Put) for each frame

1. Normalize all features with the normalizartion factors from training phase.
2. Input the feature vectors into the classifier trained in the training phase frame by frame to classify each frame to be Get or Put.
3. The estimated duty cycle is computed by (# of Put)/((# of Get) + (# of Put))

Algorithm 2. Feature Vector Training Algorithm

In addition to using the DT and FVT algorithms, we also implemented the DC estimation algorithm previously used in Chen et al. (2013). This algorithm (CH) used threshold velocity and acceleration values to estimate the loaded duration in a cycle for a given task and then estimated the DC. We applied this algorithm to the 87 test video clips. We experimented with different settings and by trial and error chose a 90% acceleration threshold value.

1.3. Results

1.3.1 Put and Get Element Time

We summarize the error distribution of Get and Put state estimations for the three methods: DT, FVT with the 41 video training set (kNN_r), and FVT trained with first cycle (kNN_f) in Table 3. The first column represents the ground truth get and put times calculated from manual single frame coding. The sample mean and standard deviation (in parentheses) of the Get and Put duration estimation errors, which were the absolute difference between estimated and ground truth time, are listed in Table 3.

A two tailed *t*-test was performed to examine if ground truth Get and Put times were statistically different from predicted Get and Put times. All three algorithms predicted Get and Put times that were not significantly different from ground truth Get and Put times ($p > .05$).

Table 1.3. Mean (SD) of Estimated Element Time (seconds) Error for Get and Put using the Decision Tree (DT), kth Nearest Neighbor (41 video training set) (kNN_r), and kth Nearest Neighbor (First cycle training set) (kNN_f) Algorithms

		Duration Estimation Error		
Element	Ground Truth Time	DT	kNN_r (41 video training set)	kNN_f (First cycle training set)
Get	5.0 (1.8)	0.4 (0.6)	0.2 (0.2)	0.3 (0.3)
Put	4.6 (1.7)	0.4 (0.6)	0.2 (0.2)	0.3 (0.3)

1.3.2 DC and Corresponding HAL Estimates

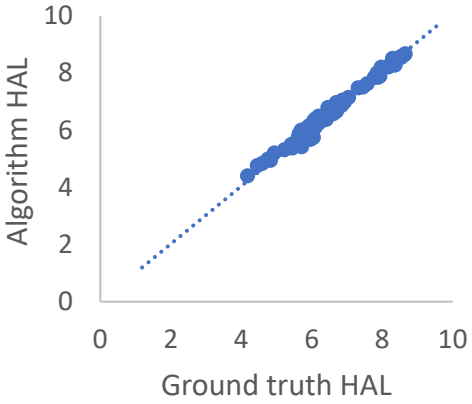
DC estimate error (%), was calculated as the average absolute difference between ground truth DC (%) and predicted DC (%) for the four algorithms (CH, DT, kNN_r, kNN_f). We provide a summary of the DC estimation errors of these four methods in Table 4. The Max and Min give the maximum and minimum absolute values of estimate errors. The Mean and SD give the mean and standard deviation of the estimate errors. We also calculated the HAL estimation errors in Table 5. To better understand the HAL estimation outcome, we also provide scatter plots of the HAL estimation outcome by the DT, kNN_r, kNN_f algorithms versus the ground truth HAL values, as well as the coefficient of determination, and summarize the results in Figure 6.

Table 1.4 Duty Cycle Estimation Error Summary (percent) for the Chen et al. (2013) (CH), Decision Tree (DT), kth Nearest Neighbor (41 video training set) (kNN_r), and kth Nearest Neighbor (First cycle training set) (kNN_f) Algorithms

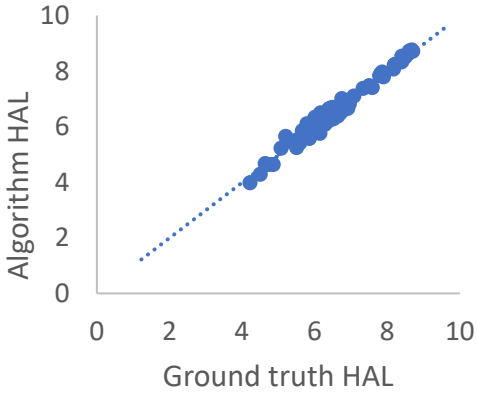
	CH	DT	kNN_r	kNN_f
Max	48.5	7.3	8.8	10.9
Min	0.0	0.1	0.1	0.2
Mean	15.5	2.6	2.8	3.3
SD	10.1	2.0	2.1	2.5

Table 1.5 HAL Estimation Error Summary (HAL units) for the Chen et al. (2013) (CH), Decision Tree (DT), kth Nearest Neighbor (8417 video training set) (kNN_r), and kth Nearest Neighbor (First cycle training set) (kNN_f) Algorithms

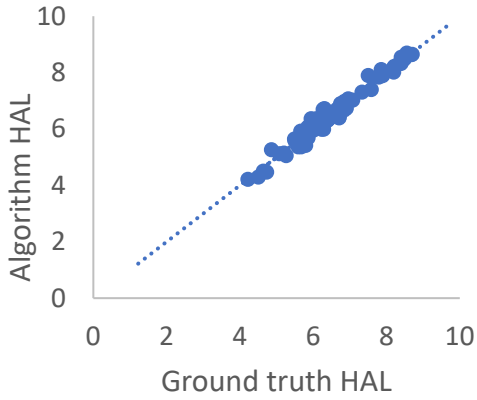
	CH	DT	kNN_r	kNN_f
Max	2.3	0.3	0.4	0.4
Min	0.0	0.0	0.0	0.0
Mean	0.6	0.1	0.1	0.1
SD	0.4	0.1	0.1	0.1



A. DT Algorithm (R^2=.99)



B. kNN_r Algorithm (R^2=.99)



B. kNN_f Algorithm (R^2=.99)

Figure 1.6. Predicted HAL versus ground truth HAL for the Decision Tree (DT), kth Nearest Neighbor (41 video training set) (kNN_r), and kth Nearest Neighbor (First cycle training set) (kNN_f) Algorithms, N=87.

1.4. Discussion

In this study we utilized marker-less video tracked hand kinematics data to automatically estimate DC. We started with a simple laboratory task consisting of four basic elements, Grasp-Move-Release-Reach. Our aim was to automatically measure exertion time (i.e. time elapsed between Move and Release) in order to calculate DC. The algorithms were used to predict DC for 87 cases. The small average errors of 2.7% for DT algorithm and 2.8% for FVT algorithm appear promising.

An exertion occurs when the hands are loaded and is defined as an action associated with muscle contractions in the hands and arms. The objective is to identify when exertions are made by using kinematic data measured from the tracked ROI. The algorithms in this study identify the distinctive activity of the ROI (i.e. location, velocity and acceleration) that occur during exertions when the muscles are contracting and the hands are loaded, and affect the ROI kinematic properties. This works because repetitive tasks are cyclical. Since they repeat over and over, distinct characteristics in their kinematic properties can be used to ascertain when exertions occur and when they do not occur in a cycle.

We also calculated HAL based on the predicted duty cycle and RMS speed. The comparison of HAL ratings against ground truth calculated HAL ratings had an average error of 0.1 for both algorithms which may be considered negligible.

Chen et al. (2013) reported an algorithm for estimating DC based on threshold crossings for speed and acceleration in a laboratory load transfer task. The task used in Chen et al. (2013) was elementally similar to our task where subjects get a bottle and put it on a rotating table. The algorithm reported in Chen, et al (2013) was implemented here in order to serve as a baseline comparison against the new algorithms developed in this work. We conclude that the new algorithms performed considerably better.

The Chen et al. (2013) threshold algorithm detected peak acceleration and associated local minima velocity points to identify the hand loaded elements. However that method was specific to the particular load transfer laboratory task kinematics and may not be applicable to a more arbitrary tasks. Our improvement in the current study used more global features in order to detect hand loaded and unloaded elements for various tasks, provided that the tasks correspond to simple Get and Put elements. Since these algorithms are more generic, we anticipate that they can be extended to more generalized tasks. This will be tested in future studies.

The selection of simple kinematic features measured using robust marker-less tracking methods should be applicable to tasks found in industrial settings. Since these algorithms are not dependent on accurately tracking multiple markers and body linkages, they have the potential to be useful in workplace environments where the tracking may present more challenges, such as obstructions, burred images, and movements out of the video window.

The current algorithm was developed using only Get and Put elements. Get involves movement while unloaded whereas Put involves movement while loaded. In order to estimate DC for more generalized tasks in future studies, we propose using a four-state transition model to describe the repetitive sequence of hand motion. Additional elements such as Wait and Hold were not considered. Wait involves no movement while unladed whereas Hold involves no movement while loaded (i.e. static forces). The current study is a proof of concept. Future research will consider these additional elements. While Get and Put elements present the different stages of the simulated repetitive hand motion task performed in the laboratory, this model may not always be applicable to more complex tasks. The transition from one state to another is boded by the previous state. For instance a Hold or Put can only come after a Get. In the case of the current laboratory task, after Get ball, a subject can only Move it or Hold it, but cannot Wait. The distinguishing feature between Move and Hold is the motion or velocity of the region of interest (ROI). Similarly, Wait or Reach can follow Put, but Hold cannot. The Wait and Reach state are also distinguished by the velocity of the ROI. In the next phase of this research we plan to use this logic into to predict states of Wait and Hold.

The HAL measure is intended for mono tasks and our study only considers mono tasks. This is typical of many industrial tasks. The DT algorithm and FVT algorithms were tasked to classify each video frame as Get or Put states. Hence the performance of each classification impacts directly on the subsequent duty cycle estimation and HAL estimation results. These algorithms will undoubtedly perform best when movements are regularly repeating rather than arbitrary movements. Since our laboratory participants were not experienced manual workers and were trained for only a few trials, their hand motion was not always as rhythmic as it might be for an experienced worker doing a repetitive tasks day in and day out. We anticipate that using better trained subjects exhibiting regular motions may in some cases result in as good or better performance.

Conceivably the DT algorithm does not require training other than selecting a prototypical cycle for establishing the exertion and non-exertion elements. The FVT algorithm requires training but might be more adaptive to different tasks. Future research will next test the algorithms using actual industrial tasks. While the feature vector machine learning algorithm may be useful for more arbitrary tasks, the amount of learning required for the FVT algorithm might be a limitation. Consequently machine learning using the first cycle method should be less dependent on training data and although performance was poorer for the laboratory task, the potential advantage is its flexibility in adapting to videos of different manufacturing jobs.

Automatic detection of duty cycle from hand kinematics using computer video processing is not only useful for calculating elemental time, but may also useful for estimating maximum acceptable exertion levels (i.e. ratio of force to strength). Potvin (2012) modeled maximum acceptable exertion in repetitive manual tasks based on DC by doing a meta-analysis from numerous studies in the literature. The automatic detection of loaded and unloaded states (i.e. DC) may be applied to the Potvin equation for establishing maximum acceptable exertion levels as well as measuring acceptable rest and work periods based on automatic video processing. The application of Potvin's equation if practice necessitates knowledge of exertion forces as well as an estimate of strength for a specific exertion.

The current study was limited to a simple laboratory simulation in order to explore the potential of automatic duty cycle processing. The task was performed across a confined distance across the sagittal plane, was very stereotypical, highly repeatable, and the loads were based solely on inertial loads, with no static loading. Since the paced HAL varied a relatively narrow range from 2.9 to 6.4, it is not known how the algorithms will perform for work having comparably very slow, or very fast tasks with HALs of much less than 3 or greater than 6. Tasks involving more complex activities, static loads, and movement across the sagittal plane would likely result in greater errors.

Factory videos will inevitably be more challenging due to numerous factors, such as quality of the videos as well as irregular motion and interruptions. Adding new feature vectors like object positions or object sizes might be useful. In the current simulated task, we used tennis balls as the object and once a subject grasps the ball, it is hard to detect it, and thus we could not utilize the above features in our algorithms. The methods developed in this study are intended for a single video camera because in industrial applications locating a camera at a specific vantage point or locating multiple cameras is not always practical.

The goal of proving the concept, in principle, was accomplished. Future research will consider more complex tasks as well as more challenging spatial parameters. The methods developed were a very promising first step in the development of a comprehensive and flexible method, that could accurately quantify duty cycle for a larger set of task characteristics.

2. How Do Computer Vision Upper Extremity Exposure Measures Compare Against Manual Measures?

2.1 Background

Various quantification methods have been used to measure exposure to risk factors for musculoskeletal injuries, including self-reports, observation, video-based frame-by-frame analysis, and direct measurements. Each technique has advantages and disadvantages. Self-reports are the easiest to implement and are applicable to a wide range of working situations. Observational techniques are also practical to use in a wide

range of working situations. Direct measurement methods are the most accurate and reliable since these do not depend on subjective judgement but have the disadvantage of more time and cost (David, 2005).

The American Conference of Government Industrial Hygienists (2017) Threshold Limit Value® (TLV®) uses the hand activity level (HAL) rating scale, a 10-point visual analog scale based on hand speed and rest pauses. HAL may be determined subjectively by an observer or from a lookup table, or an equation by measuring exertion frequency (F) and percent duty cycle (D) where:

$$F = \left(\frac{\text{exertions}}{\text{work time}}\right) \text{ and} \quad (1)$$

$$D = 100\left(\frac{\text{work time}}{\text{work time+rest time}}\right). \quad (2)$$

The observational method for ascertaining the HAL rating for a job depends on a trained observer viewing workers performing the job on site or observing a video of the job off site. The process is often labor intensive and there may be differences in ratings between observers. Since HAL can be determined from frequency and duty cycle (Radwin, et al., 2014), manually coding video, based on frame-by-frame analysis is a widely used technique. Software such as Multimedia Video Task Analysis (MVTA) (Yen and Radwin, 1995; Yen and Radwin, 2006) can be used to manually annotate each exertion. (Bao, et. al, 2006; Paquet et. al, 2006; Lu, et. al, 2008; Dartt, et al., 2009; Chang, et. al, 2010; 2010; Burt, et al., 2013; Coelho, et. al, 2013; Harris-Adamson, et al, 2013; McGaha, et. al, 2014; Estember, et. al, 2015; Dale, et al., 2016). The frequency is calculated by counting number of exertions, divided by total time and the D is calculated as the time spent exerting, divided by total time (Radwin, et. al, 2015).

Bao et al. (2006) observed poor correlation between HAL observations, self-reports and direct-measurements for several reasons, including using different definitions of repetitive exertions. Wurzelbacher, et al. (2010) compared > 700 tasks for 484 workers and found that while correlated, the agreement (within±1 unit) between observations and calculated HAL was 61%.

A semi-automated computer vision method was developed using cross-correlation template matching for measuring the spatial-temporal properties of the hand motions using conventional video (Chen, et al., 2013). This process was used for measuring HAL, based on hand speed and D (Akkas, et al. 2015). Akkas et al. (2016) utilized the tracked motions in algorithms to identify exertions in simple repetitive motion tasks using pattern recognition machine learning.

In the current study we compare exposure variables *F* and *D*, calculated both using manual single-frame MVTA analysis, and video marker-less tracking (Akkas et al., 2015, Akkas et al, 2016, Akkas et al, 2017, Greene et al., 2017).

2.2 Methods

Pooled data from the upper limb MSD consortium studies

This study utilized exposure data from prospective studies conducted by the National Institute for Occupational Safety and Health (NIOSH), the Safety & Health Assessment & Research for Prevention (SHARP) in the State of Washington, and the University of California -San Francisco (UCSF). Some data from these prospective cohort studies had been previously pooled and analyzed as part of the Upper Limb MSD Consortium, a group of seven prospective cohort studies (Bao et al., 2015; Harris-Adamson et. al., 2013a, 2013b; Harris-Adamson et. al., 2014; Kappellusch et al., 2013, 2014; Fan et al., 2015). All three sites recorded and digitized videos for every task and collected exposure data using video-based time study, which were used for the current study.

Selected videos for automated analysis

Because the videos were created for a different purpose, not all were suitable for computer vision analysis. Examples of non-suitable videos include occlusion of the hands for most of the time by other body parts, tools

or other workers, or non-stable camera recording, which introduces an artificial signal when a hand is tracked. The criteria for inclusion were videos containing at least five contiguous cycles, no breaks in the video, limited camera motion, and no visual obstructions. Videos with a partially obstructed view were deemed suitable for the automated analysis if the hand was blocked, but there was a sufficient view of the arm to determine the location of the hand. Videos with small breaks were deemed suitable if the break is outside of the primary task.

A preliminary feasibility test of randomly selected 30 videos resulted in 90% of inclusion using above inclusion criteria. Using the inclusion criteria, we serially selected 111 videos. The videos included were the available videos where we applied hand tracking and data checking to date. Thus, not all study sites are equally represented. Summary statistics of *D* and *F* are given in Table 1. After the selection process, the analyst identified the dominant hand and initialized hand tracking starting from the first frame that captures the most active hand fully. The analyst manually corrected the tracked location of the hands when required.

Table 2.1. Summary statistics for the tasks studied

	Duty Cycle (%)	Frequency (Hz)
Min	0.00	0.00
Max	100.00	1.48
Median	89.37	0.29

Computer vision exposure measures

A feature vector training (FVT) pattern recognition algorithm was based on the k-nearest neighborhood (KNN) method to identify exertions. The FVT algorithm used the first cycle of each task for training and the remaining cycles were predicted (Akkas et al., 2016). When tested against an independent set of 50 selected factory videos (not included in the 111 sample), the FVT algorithm had an average 1.4% difference error (Akkas et al., 2017). We applied the same FVT algorithm to the 111 tasks in the current study and identified video frames representing exertions of the dominant hand. As a result, we counted total frames of exertions as well as the total number of exertions in order to calculate *F* and *D*.

2.3 Results

The calculated *F* (Hz) and *D* (%) errors were the average difference between the manual frame-by-frame and the computer vision estimates. Summary statistics of the estimate errors are provided in Table 2. A histogram of the estimated *D* and *F* are shown in Figures 1 and 2. More than 75% of the cases had *D* estimation error less than ±10%

Table 2.2. Estimation error for D and F

	Duty Cycle (%)	Frequency (Hz)
Min	-35.87	-1.66
Max	56.43	1.35
Mean	3.69	0.11
SD	17.10	0.41

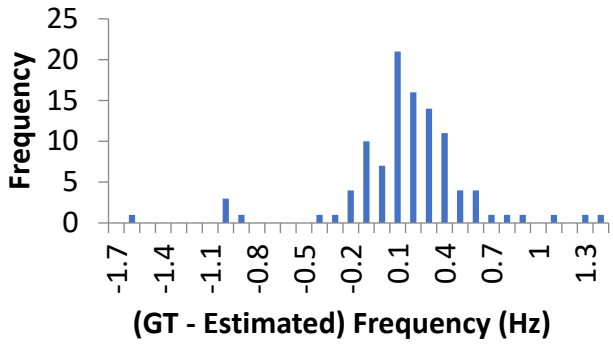


Figure 2.1. Histogram of frequency error (Ground truth frequency – estimated frequency).

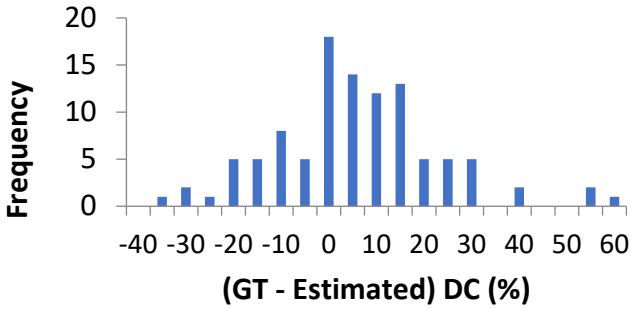


Figure 2.2. Histogram of duty cycle error (Ground truth duty cycle – estimated duty cycle).

2.4 Discussion

In this study we estimated exposure variables used to quantify risk for upper extremity MSD using computer vision. The results indicate that computer vision can reliably estimate important exposure variables for most tasks.

In Akkas et al. (2017) we showed that ±5% D error will result in 0.5 HAL error when the speed is fixed. Moreover, a HAL difference of 1 is considered as negligible (Latko et al., 1997). Computer vision was more reliable than the human observers in Wurzelbacher, et al. (2010) that found agreement (within±1 unit) between observations and calculated HAL was 61% while computer vision achieved comparable error for more than 75% of the tasks.

Labeling of an exertion varied considerably depending on the exertion definition. For example, if a power hand tool was operated several times and the entire time holding the tool was counted as a single exertion, the resulting frequency was small. However, if every instance of exertion were counted, a larger *F* would be obtained. More consistent definitions of exertions should result in better outcomes.

The FVT algorithm was developed for cyclic tasks and identifies patterns where an exertion period is followed by a rest period. A review of cases where it failed (i.e. estate error > ±10%) found that this occurred when the first cycle was not representative of the remaining cycles, for example, if a worker did something unusual. The estimate can thus be improved by selecting a more representative cycle for training. Furthermore, non-repetitive tasks are more challenging since they do not have a cyclic pattern. Future research will consider not only using features for kinematic signals such as speed, displacement, and location, but also take into account when non-repetitive activities are performed. Since the videos used in this study were taken for a different purpose, we anticipate the algorithms will perform better when videos are recorded specifically for computer vision analysis.

3. The Speed Calculated Hand Activity Level (HAL) Matches Observer Estimates Better than the Frequency Calculated HAL

3.1 Introduction

The Hand Activity Level (HAL) of the ACGIH TLV is estimated on a 10-point visual-analog scale by trained observers rating exertion speed and hand pauses. Only exertions greater than 10% of posture-specific strength are counted. It can also be estimated from measurements of exertion frequency (*F*) and percent duty cycle (*D*) using a lookup table or an equation [1]. HAL can automatically be estimated using computer vision [2] from videos by tracking hand speed (*S*) and *D* [3], [4].

Bao et al. [5] observed poor correlation between HAL observations, self-reports and direct-measurements for several reasons, including using different definitions of repetitive exertions. Wurzelbacher, et al. [6] compared more than 700 tasks for 484 workers and found that while correlated, the agreement (within±1point) between observations and calculated HAL was 61%. The current study compares observer estimated HAL (HAL_O), frequency *F* and *D* calculated HAL (HAL_F) estimated from frame-by-frame analysis of videos, and speed *S* and *D* calculated HAL (HAL_S) using computer vision. The effect of exertion definition was also considered.

3.2 Methods

Video and exposure data from prospective cohort studies [7], [8], [9], [10], [11] were used. A total of 1004 videos containing HAL_O values were selected in the order of subject ID, containing at least five contiguous cycles, no breaks, limited camera motion, and no visual obstructions. Hand motion, including *S*, was measured from the videos using computer vision. The estimates of total exertion *F* and *D* from the data were used to calculate HAL_F, and *S* and *D* were used to estimate HAL_S.

3.3 Results

There was 75% agreement (within±1point) between HAL_O and HAL_S (HAL_S=1.02×HAL_O -0.2, R^2=0.78, $F(1,1003)=43665$, p <.001), however, there was only 30% agreement (within±1point) between HAL_O and HAL_F (HAL_F=0.21×HAL_O+2.2, R^2=0.04, $F(1,1003)=71.71$, p <.001).We selected a subset of 111 videos and checked the exertion annotation. After modifying *F* to count every forceful finger, hand or wrist exertion, as recommended in Bao et al. [5], agreement was 57% between HAL_F and HAL_O (HAL_F=0.71×HAL_O+1.21, R^2=0.54, $F(1,110)=128.4$, p<.001).

3.4 Discussion

HAL_S was more comparable to HAL_O than HAL_F. HAL_F varied considerably depending on the exertion definition. For example, if a power hand tool was operated several times and the entire time holding the tool was counted as a single exertion, the resulting HAL_F was small. However, if every instance of exertion was counted, a larger *F* and thus a larger HAL_F was obtained. When recommended rules for *F* were applied, HAL_F and HAL_O were more closely aligned, and therefore these rules should be used for more reliability among measures. HAL_S was more consistent with HAL_O since both are dependent on speed, and because HAL_S can be automated, it was more objective than HAL_F or HAL_O in this sample.

4. Defining repetitive hand exertions for exposure assessment

Numerous criteria have been used for hand exertions (Stetson, et al., 1991; Wiktorin et al., 1993, Marshall and Armstrong, 2004; Bao, et al., 2006). Similarly, repetition has been quantified in various ways (Radwin et al., 1993; Moore and Garg 1995, Latko et al. 1997, ACGIH Worldwide, 2001). The Threshold Limit Value for Hand Activity Level (ACGIH Worldwide, 2018) exposure guideline has revised the criteria for hand exertions from the original guideline published in 2001, potentially leading to inconsistent interpretations of hand activity level (HAL). Repetition rate and duty cycle are therefore dependent on the category of forces considered. Bao, et

al. (2006) observed that that different definitions of repetitive exertions lead to measuring different physical exposure phenomena and produce very different results. Furthermore, there were poor correlations between different measures of repetitiveness estimated using different methods. We compared observed HALs obtained for the NIOSH supported upper extremity consortium study against frequency and duty cycle computed HALs based on video records for the same task and applied different criteria for negligible, non-negligible, and forceful exertions. We defined an *exertion* as a visible hand or forearm muscular effort while grasping an object or applying a force (e.g. hold, manipulate, trigger, push, pull, or handle an object) during task performance, regardless of the force required. Exertions were categorized by their magnitude of force, ranging from negligible exertions (i.e. Borg CR10<1 or MVC≤10% level of force), to non-negligible exertions (i.e. Borg CR10≥1 or MVC>10% level of force) and forceful exertions (i.e. pinch force ≥ 9N or power grip force ≥ 45 N, or Borg CR10≥2). The ***total repetition* rate** is the rate of all exertions qualifying as Category 1 or Category 2, expressed as exertions/unit time. The ***forceful repetition*** rate is the rate of all exertions qualifying as Category 2, expressed as exertions/unit time. Count every qualifying exertion, even if they are not followed by a pause (e.g. when a subsequent exertion occurs while control of an object is retained, or when grasp of an object is released immediately followed by another grasp).Do not count pauses, rests, or non-qualifying exertions. Count coordinated exertions, when they are used together, as the same exertion, such as ulnar-radial deviation, flexion-extension, or push-pull. A coordinated exertion is a hand or forearm muscular effort that uses one or more agonist or antagonist muscle sequences for performing a task (e.g. swiping a paint back and forth, striking a key).

Category 0	Negligible Exertions Borg CR10<1 or MVC≤10% level of force A pause or rest occurs when exertions are negligible for a measurable period of time (i.e. reach for an object, hold a sheet of paper)
Category 1	Perceptible Exertions Borg CR10≥1 or MVC>10% level of force
Category 2	A subset of Category 1, limited to forceful exertions Borg CR10≥2 Pinch force ≥ 9N or power grip force ≥ 45 N

Figure 4.1 Exertion categories.

5. Video-based Automatic Hand Posture Classification

The wrist is always in one of three states including neutral, flexion and extension. The predefined categorization (PDC) strategy, which was designed based on several widely used posture observational methods, classifies the wrist posture

as neutral when the angle is between -15 to 15 degree (Bao 2009). In our experiment, these states are classified by the angle θ between hand and elbow, if $\theta > 15$, it is an extension. If $\theta < -15^{\circ}$, it is a flexion. Otherwise, it is in a neutral position.

5.1 Wrist Flexion Angle Estimation Algorithm

Denote the coordinates of three manually selected landmark points H, W and F as (x_h, y_h), (x_w, y_w) and (x_E, y_E) respectively. The wrist bending angle θ is the angle between line segments $\overrightarrow{WH}$ and $\overrightarrow{EW}$. An example of the three landmark points is shown in Figure 2.

The vectors $\overrightarrow{WH}$ and $\overrightarrow{EW}$ can be expressed as:

$$\overrightarrow{WH} = (x_h - x_w, y_h - y_w) \quad (1)$$

and

$$\overrightarrow{EW} = (x_w - x_E, y_w - y_E) \quad (2)$$

Using trigonometry, one has

$$|\theta| = \cos^{-1}\left(\frac{\overrightarrow{EW} \cdot \overrightarrow{WH}}{\|\overrightarrow{EW}\|\|\overrightarrow{WH}\|}\right) \quad (3)$$

The sign of θ may be determined by

$$sign(\theta) = \begin{cases} sign\left(\overrightarrow{EW} \times \overrightarrow{WH}\right) & if\ the\ palm\ is\ toward\ clockwise\ direction \\ -sign\left(\overrightarrow{EW} \times \overrightarrow{WH}\right) & if\ the\ palm\ is\ toward\ counterclockwise\ direction \end{cases} \quad (4)$$

Above formulation assumes the palm is facing downward to perform the task. For tasks that require the palm face upward, (I think the formula either is still applicable or can be made applicable with minor modifications. Please figure this out given the orientation of the palm). In practice, the orientation of the palm (upward or downward) will be a user input to this algorithm.

We applied an open source 2D human pose estimation algorithm Openpose (Cao et al. 2019) to estimate 2D coordinates of three body joints of the right arm, the elbow (E), wrist (W) and hand (H) respectively. Openpose is an open-source 2d human pose estimation algorithm. It takes an image as the input to a two-branch Convolutional Neural Networks and predicts a set of 2D confidence map of body part locations and a set of 2D vector fields of part affinities simultaneously. Finally, greedy inference is applied to parse the confidence map and affinity field to output 2D coordinates of 25 key points which are the body skeletal joints of the subject.

The diagram of the proposed wrist flexion angle estimation algorithm is summarized in Figure 3.

Figure 1. Three tracked points and the hand posture angle θ.

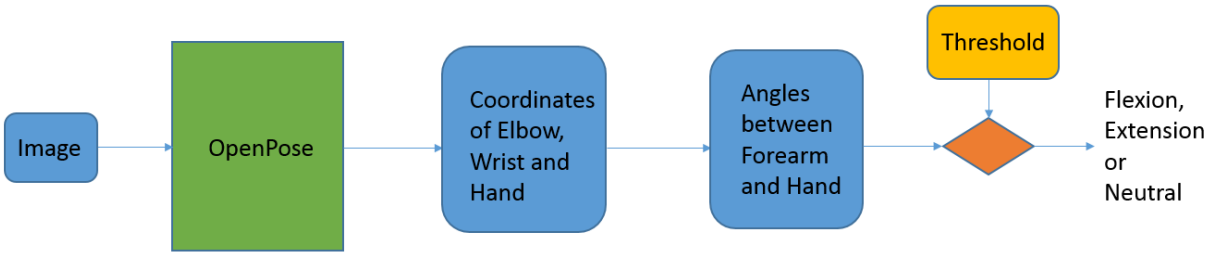


Figure 5.3. Diagram of our method.

5.2 Experiment

Data Collection

The data set consists of 61 videos of 16 different subjects (3 males and 13 females). These videos are selected from the simulation task videos recorded in laboratory (Akkas et al. 2016). The subjects were recruited with informed consent and IRB approval. In each video, the subject is asked to stand in front of an apparatus to perform 15 cycles of moving tennis balls from one location to another, as demonstrated in Figure 3.

The apparatus (Figure 4) is comprised of an 840 mm travel length linear belt drive actuator (Misumi MSS-625) driven by a bi-polar stepper motor (ElectroCraft Model TPP34 with 560 N cm torque) and controlled by a stepper motor controller (IMS MX-CS101-401). The device is capable of moving a 2 kg object every 0.5 s across the actuator length of travel.

The hands were videoed from the side view (sagittal plane) with the palm of the right hand facing downward. The selected videos include two kinds of tasks, which are moving the tennis balls from location A to C and A to 2 (as shown in Figure 4). We utilized an Optotrak 3D infrared motion tracking systems for tracking the hands and arms of the subjects and adopted the angles formed by these tracking points as the ground truth. Using the angles estimated from Optotrak, we can classify each frame of the video as one of three states: flexion, extension or neutral. These states are used as ground truth states to verify the performance of our algorithm.

The Optotrak 3D points here can be considered as in the global coordinate system, which cannot be directly compared with the 2D image coordinates estimated by Openpose. In order to compare the two kinds of coordinates in the same coordinate system, we need to find out the linear transformation between the Optotrak 3D coordinates $[X_i\ Y_i\ Z_i]^T$ and the 2D image coordinates $[x_i\ y_i]^T$, as shown in eq. (5).

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \cong \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \tag{5}$$

To be explicit,

$$x_i = \frac{m_{11}X_i + m_{12}Y_i + m_{13}Z_i + m_{14}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}} \tag{6}$$

$$y_i = \frac{m_{21}X_i + m_{22}Y_i + m_{23}Z_i + m_{24}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}} \tag{7}$$

Rearrange them, then we get,

$$m_{11}X_i + m_{12}Y_i + m_{13}Z_i + m_{14} - m_{31}x_iX_i - m_{32}x_iY_i - m_{33}x_iZ_i - m_{34}x_i = 0 \tag{8}$$

$$m_{21}X_i + m_{22}Y_i + m_{23}Z_i + m_{24} - m_{31}y_iX_i - m_{32}y_iY_i - m_{33}y_iZ_i - m_{34}y_i = 0 \tag{9}$$

where m_{jk}'s are the unknown parameters we want to figure out, so we can write them into the matrix form:

$$\begin{bmatrix} X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -x_iX_i & -x_iY_i & -x_iZ_i & -x_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -y_iX_i & -y_iY_i & -y_iZ_i & -y_i \end{bmatrix} \begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{10}$$

With only one pair of corresponding Optotrak 3D coordinates $[X_i\ Y_i\ Z_i]^T$ and 2D image coordinates $[x_i\ y_i]^T$, eq. (10) is still an underdetermined system. We need at least 6 pairs of them to solve m_{jk}'s.

After obtaining the transformation matrix, we can map the Optotrak 3D coordinates into the image 2D coordinates, and compute the ground truth angles using eq. (1)(2)(3)(4).

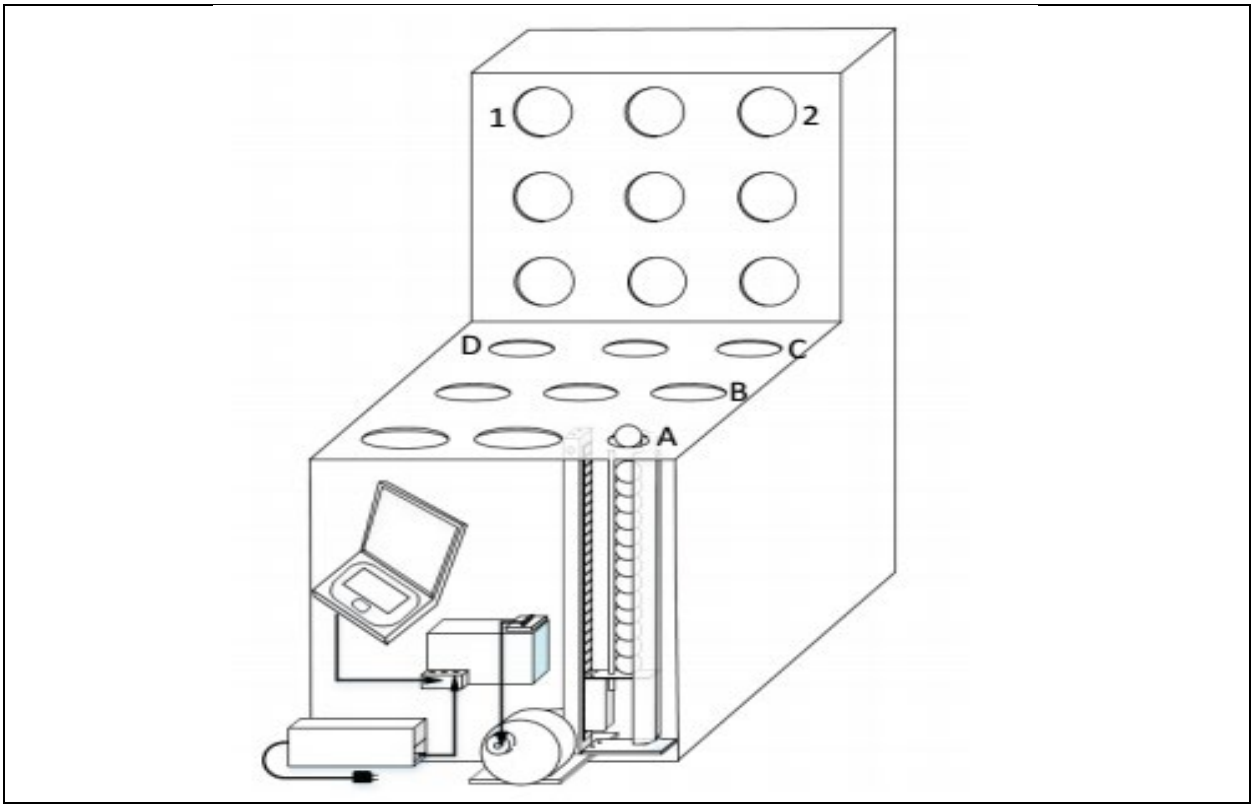


Figure 5.4. Laboratory task simulation apparatus. The participant grasps a ball from location A and moves it to either locations C or 2, depending on the specified task.

Synchronization

Because the initialization of the Optotrak and the video recorder were independent, there was a time delay between the signals recorded. To compensate this time difference and synchronize the signals, the cross-correlation of the relative displacement of the two signals was calculated with the displacement signals acquired through tracking one single point. We can obtain the time delay between two signals by searching the relative displacement corresponding to the global maximum of the cross-correlation.

A plot of position to time is shown in Figure 5(a). The solid line and dash line represent the Optotrak and video tracking results, respectively. We observed that these two signals have the same frequency, but they are not synchronized in the time domain. The cross-correlation between these two signals. Is demonstrated in Figure 5(b). The horizontal axis represents the relative displacement of these two signals. The relative displacement corresponding to the maximum cross-correlation, D, is the estimated time delay. After shifting the Optotrak signal by the time delay, the Optotrak signal and the video tracking signal were aligned better, as shown in Figure 5(c). The relationship between the two signals before and after shifting are shown in Figure 5(d).

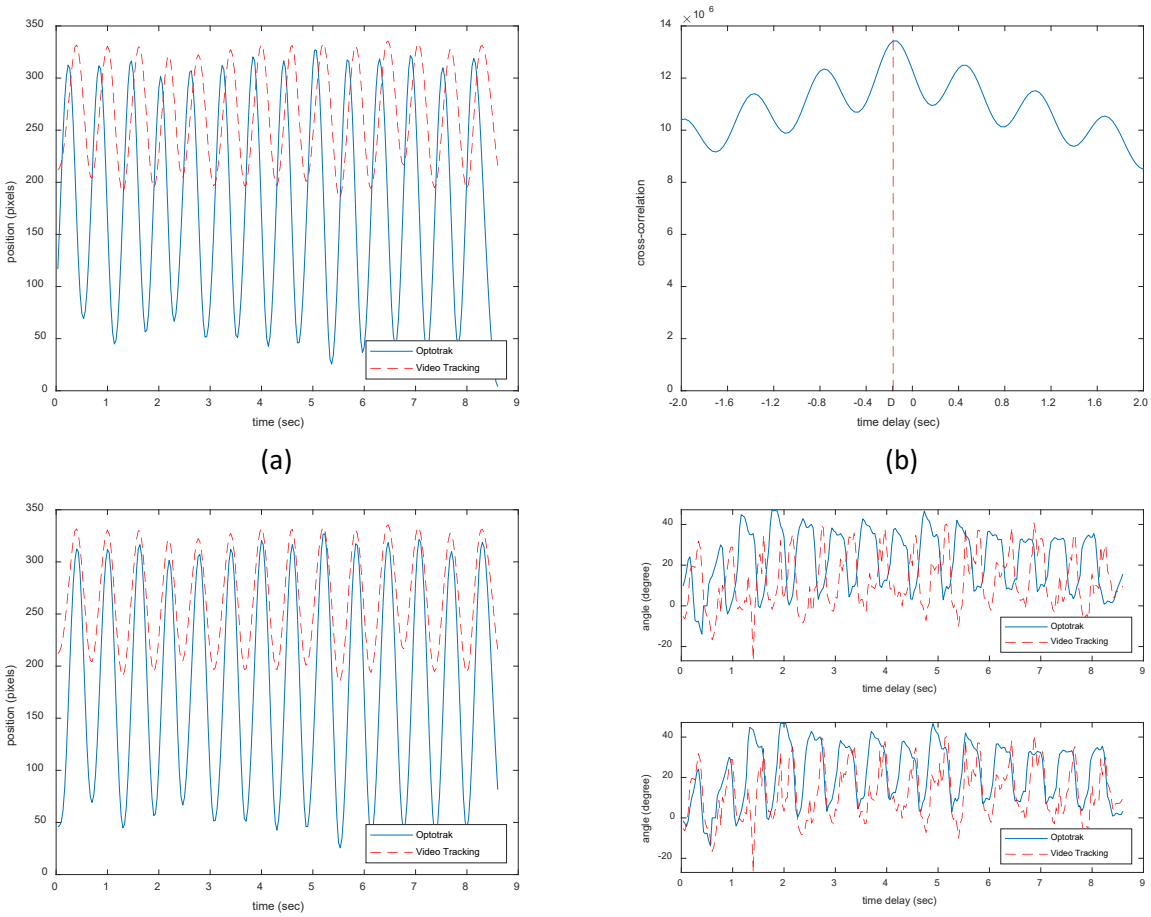


Figure 5. (a) Position-time curves of Optotrak and video tracking data; (b) Cross-Correlation of Optotrak and video tracking signals; (c) Aligned position-time curves of Optotrak and video tracking data; (d) (Top) Angle-time curves of Optotrak and video tracking data; (Bottom) Two curves are aligned after shifting one signal by the time delay D.

5.3 Results

In our experiment, we use a total of 1488 frames extracted from 61 videos. For each frame, we have the ground truth angles by Optotrak and the estimated angles by Openpose. The state of each frame depends on which interval the corresponding angle locates in. Instead of using all frames as our samples, we only compare the target frames which have the extreme postures. As Figure 6 demonstrates, we find out the local maximum and local minimum of the Optotrak angles. The frame numbers corresponding to the blue circles are the target frames in which we have the extreme postures. Finally, there are totally 1488 target frames.

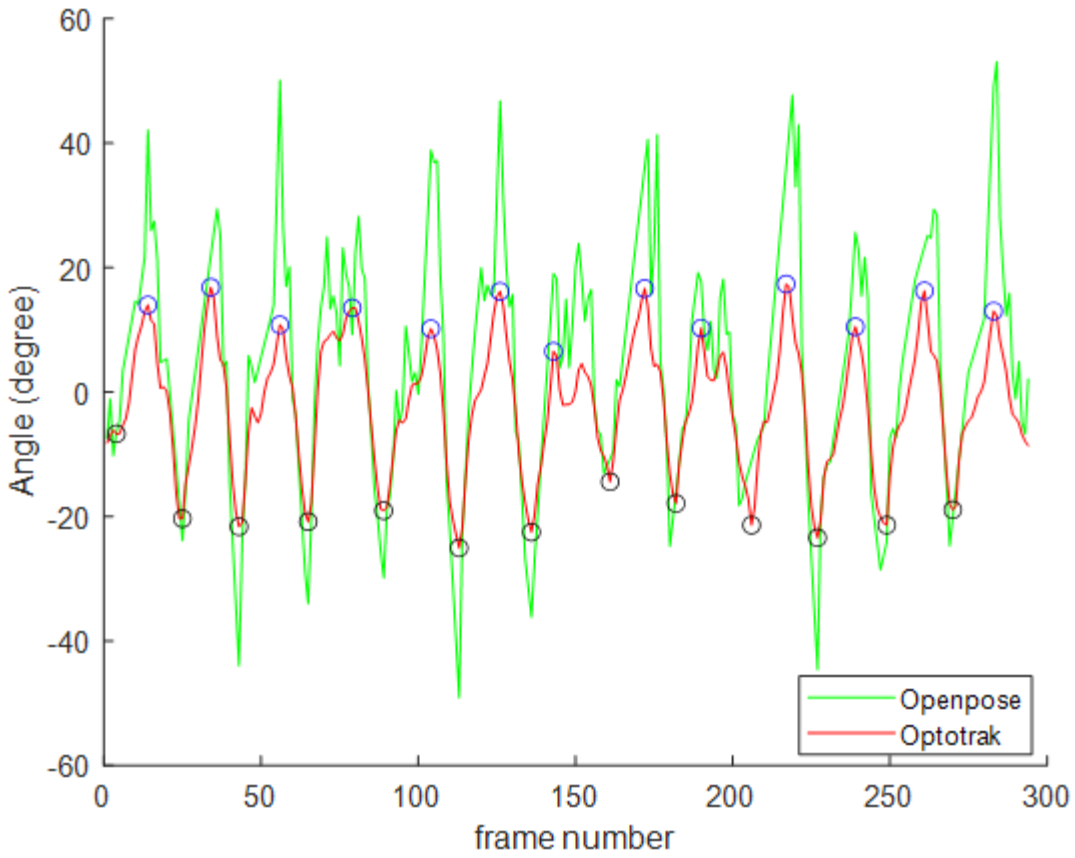


Figure 6. Angle Prediction by the Algorithm

Figure 5.7 shows the angle prediction by the algorithm for one of our videos. We can see that the angle changes periodically due to the repetitive motion. Table 1 is the confusion matrix for all target frames of the total of 61 videos. We adopt *average per-class accuracy* as our performance criteria.

$$Average\ per\ class\ accuracy = \frac{sum\ of\ accuracy\ for\ each\ class\ predicted}{number\ of\ classes} \quad (11)$$

It is more appropriate than the Classification rate when there are considerable differences between the amounts of samples in different classes. In our experiment the average per-class accuracy is 73.12%. The overall RMSE (root mean squared error) of the angle is 12.62 degrees.

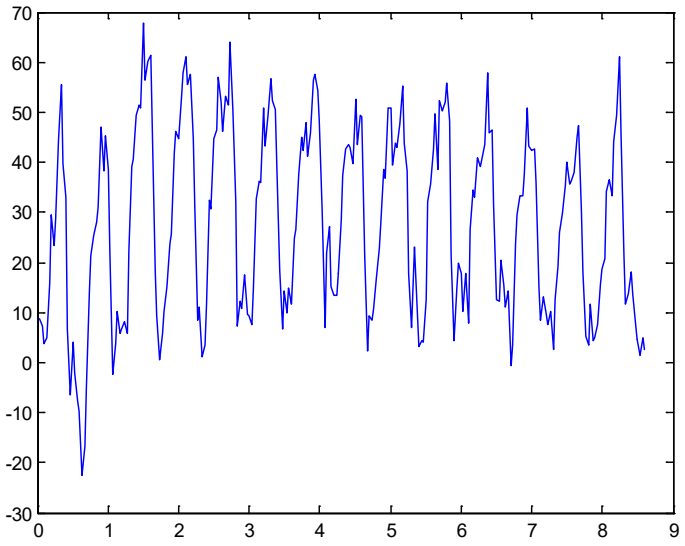


Figure 5.7. Angle Prediction by the Algorithm

Table5.1. Confusion Matrix

Prediction / Ground Truth	$\theta < -15^{\circ}$ **Flexion**	$15^{\circ} > \theta \geq -15^{\circ}$ **Neutral**	$\theta \geq 15^{\circ}$ **Extension**
$\theta < -15^{\circ}$ **Flexion**	**383 (25.74%)**	**116 (7.80%)**	**4 (0.27%)**
$15^{\circ} > \theta \geq -15^{\circ}$ **Neutral**	**114 (7.66%)**	**520 (34.95%)**	**95 (6.38%)**
$\theta \geq 15^{\circ}$ **Extension**	**2 (0.13%)**	**70 (4.70%)**	**184 (12.37%)**

5.4 Discussion

In this work, we present a computer vision-based method, utilizing the Openpose keypoints detection, to measure the joint angle of the hand posture. This measurement can be used as an objective indicator to assess the physical exposure to the work-related musculoskeletal risk of the workers in the factory. Our research makes the posture estimation measurement automatic, therefore we can collect more data and the result will be more consistent.

We achieve an overall performance of 12.62 degrees RMSE angle and 73.12% average per-class accuracy in a controlled laboratory environment. The sensitivity of flexion and extension are 77.33% and 74.62%. The specificity of flexion and extension are 73.00% and 74.37%. Applying the proposed method on the real environment might be promising. Not limited to the hand posture, our method can also be extended to the application of measuring neck, shoulder and elbow postures. For example, we can use the three landmarks: ear, center of shoulders and center of hips detected by

Openpose to replace the hand, wrist and elbow in this experiment. The new three landmarks viewed from the sagittal plane of the subject will form the angle of neck.

Due to the limitation of single-view videos, the assumption of the side view is necessary. As future work, the depth image videos might be applied to remove this restriction.

A study in MacKinnon 2020 compared the results between the video analysis and electrogoniometer method. They classified the wrist flexion-extension posture into five categories: $\theta \geq 45^{\circ}$, $45^{\circ} > \theta \geq 15^{\circ}$, $15^{\circ} > \theta \geq -15^{\circ}$, $-15^{\circ} > \theta \geq -45^{\circ}$ and $\theta < -45^{\circ}$, and achieve 57% agreement. To compare our results, we analog the Optotrak and Openpose in our experiment to the electrogoniometer method and video analysis. If we use the same five-class category, our confusion matrix will become what is shown in Table 2. The agreement between Optotrak and Openpose method is 66.20%, which is 9.2% higher than the agreement between the video analysis and electrogoniometer method. That indicates that our method has better performance than the tradition video analysis conducted by human observer.

Table 5.2. Confusion Matrix

	$\theta < -45^{\circ}$ **Flexion**	$-15^{\circ} > \theta \geq -45^{\circ}$ **Flexion**	$15^{\circ} > \theta \geq -15^{\circ}$ **Neutral**	$45^{\circ} > \theta \geq 15^{\circ}$ **Extension**	$\theta \geq 45^{\circ}$ **Extension**
$\theta < -45^{\circ}$ **Flexion**	**5 (0.34%)**	**12 (0.81%)**	**3 (0.20%)**	**0 (0.00%)**	**0 (0.00%)**
$-15^{\circ} > \theta \geq -45^{\circ}$ **Flexion**	**57 (3.83%)**	**309 (20.77%)**	**113 (7.59%)**	**3 (0.20%)**	**1 (0.07%)**
$15^{\circ} > \theta \geq -15^{\circ}$ **Neutral**	**5 (0.34%)**	**109 (7.33%)**	**520 (34.95%)**	**85 (5.71%)**	**10 (0.67%)**
$45^{\circ} > \theta \geq 15^{\circ}$ **Extension**	**0 (0.00%)**	**2 (0.13%)**	**70 (4.70%)**	**150 (10.08%)**	**26 (1.75%)**
$\theta \geq 45^{\circ}$ **Extension**	**0 (0.00%)**	**0 (0.00%)**	**0 (0.00%)**	**7 (0.47%)**	**1 (0.07%)**

We conclude that using a computer vision-based method to estimate the hand posture provides a quick assessment and requires no extra equipment on the subject. There is still a long way to go to create a safer and better working environment for the labors.

6. Predicting Sagittal Plane Lifting Postures from Image Bounding Box Dimensions

Back pain is among the most common workplace injuries and most costly to workers compensation, averaging over 20% of employer costs (Bhattacharya et al., 2012). The U. S. Bureau of Labor Statistics (BLS) reported that in 2016 back injuries were a leading injury that resulted in days away from work or job transfer or restriction in a variety of industries (BLS, 2018). In 2015, musculoskeletal injuries such as sprains and strains caused by overexertion in lifting accounted for 31% of all the cases of nonfatal occupational injuries and illnesses requiring days away from work, remaining relatively unchanged from previous years (BLS, 2016). In 2014, the number of cases affecting the lumbar region accounted for 1,580 of the total 7,750 musculoskeletal injuries in the state of Wisconsin, or just over 20% (NIOSH, 2014).

Manual materials handling and lifting are contributing factors of low-back pain and injuries (Bernard et al, 1997; Barondess et al., 2001). Extreme working postures are a risk factor for low-back disorders (Marras et al., 1995; Bernard et al., 1997; da Costa and Vieira, 2010). Numerous factors have been identified as greatly impacting the force on the lower vertebrae (L4/L5 compression) during a lift, such as critical joint angles of the hip and

knee that increase moment loading (Hwang et al., 2009). These joint angles are good indicators of the amount of stress placed on the lower back during a lift because they affect the moment at the low back vertebrae. Of the different lifting postures, stoop lift is often associated with greater lower back moments resulting from the combination of knee and trunk angles (straight knees and bent trunk). Previous studies have reported association between stoop lift and low back disorders (e.g. Holmström et al., 1992; Myers et al., 1999).

Video recordings have become an indispensable tool in occupational ergonomics. A survey of Certified Professional Ergonomists conducted by Dempsey et al. (2005) reported that video cameras were the most commonly used tool (91.6%), and that they were rated “very useful” by most of the participants. With the advantages of objectivity, low-cost, and little interference with work, video task evaluation tools have been drawing increasing attention. Furthermore, since video recordings are commonly used in industrial applications, new exposure analysis tools based on video for automatically extracting information should gain wide acceptance by both practitioners and workers.

Motion tracking tools include technologies that require markers are limited in their ability to measure parameters such as body angles in industrial practice. Camera placement is often challenging in the workplace while marker placement takes time to attach and can interfere with the work. Although many of these methods offer laboratory precision, such accuracy and precision may not be needed for occupational health and safety practice. Marker-less methods are similarly subject to variations in the observation and estimation techniques for projected angles and leads to less accuracy and high intra- and inter-observer variability (Plantard et al., 2016). To combat these issues, some researchers have developed error-correction models that improve the accuracy of collected data, though it is still difficult to evaluate the effectiveness of these models (Spector et al., 2014).

A variety of studies have explored computer vision-based methods for analyzing human motion and activities. Wang and Suter (2007) proposed a framework for recognizing human activities based on silhouette data extracted from videos. Marfia and Roccetti (2016) proposed a method that identifies stoop lifts based on lower back location in relation to the ground detected by the computer vision using silhouette information. Similarly, Seo et al. (2016) used silhouette information extracted using computer vision to identify postures in videos. Li and Lee (2011) developed a method that extract skeletal models of workers from 2D videos with joint locations which can provide abundant information for ergonomic applications. Mehrizi, et al. (2017, 2018a, 2018b) demonstrated a computer vision marker-less motion capture method to assess 3D pose, back moments and joint kinematics of lifting tasks. These methods possess great potential for application in task evaluation. However, they are dependent on a controlled environment for information extraction, and their ability to yield accurate conclusions in complex industrial settings where poor lighting, occlusion, and dynamic backgrounds are common.

Practical methods should be sufficiently robust to obtain reliable information without being compromised from various sources of noise. Advances in computer vision video technology have been applied in ergonomics for evaluating repetitive motion tasks (Chen, et al., 2013), exertion frequency and duty cycle (Akkas et al., 2016), and for visualizing repetitive motion task factors (Greene et al., 2017). This approach extracts key spatiotemporal features of motion in a semi-automatic manner from conventional video recordings whereby the analysts interactively select a region of interest such as a hand or arm relative to a stationary region. This method relaxes the need for high precision tracking rather than imposing an a priori model on the tracked activity. Consequently, the analysis complexity is greatly reduced and may be more tolerable of the numerous variations encountered in field video recordings of occupational tasks.

The current study explores how a practical computer vision approach can be applied to lifting tasks. While previous methods have focused on obtaining precise joint angles, a tool capable of classifying lifting postures that pose different levels of load to the low back is more pragmatic for ergonomics practice. We devise a practical method that is insensitive to challenging workplace conditions, such as poor illumination, poor vantage points, and obstructions, by relaxing the need for high precision and emphasizing extracting essential features from a video that are useful for assessing stress and strain on the lower back. This study explores if

the features of a simple “elastic” rectangular bounding box, continuously tracking the subject can be used for classifying three types of postures assumed while lifting or lowering an object at the origin or destination (i.e. standing, stooping and squatting). The ultimate objective is to develop computer vision software for automatically classifying lifting postures based solely on bounding box dimensions, without needing to measure joint angles or fit the data to skeletal models.

6.2 Methods

Our approach builds on the concept of extracting simple key features from the video image, rather than fitting complex skeleton models used in motion tracking. Background subtraction, a video processing technique that identifies moving regions by comparing every frame to the stationary background, is used to identify the moving foreground (i.e. the worker), thus not needing to track specific body parts. A bounding box is drawn tightly surrounding the foreground (Figure 1). Using this technique, we have devised a method for identifying the instance an object is first moved at the lift origin of the lift, or last stops moving after the object is released at the lift destination. The dimensions of the bounding box and the image it encloses at the origin and destination of the lift are measured (i.e. height and width).

Mannequin poses of various lifts for varying locations were generated using the 3D Static Strength Prediction Program Version 7.0.0 (3DSSPP) from the University of Michigan (2017). A bounding box was drawn around each mannequin and the corresponding box height (H) and width (W) were used as a training set to develop a decision tree algorithm to identify the lifting posture of a subject based on bounding box dimensions. In this paper, a specific lifting posture is classified as one of three states: squatting, stooping, or standing.

The resulting decision tree algorithm was tested using videos of actual industrial lifting tasks. The instance when an object in a video is lifted or released was identified, a bounding box was drawn around the subject, and the aspect ratio was entered into the algorithm for classifying the lifting posture assumed. Independently, the lifting posture was classified based on the subject’s hip and knee angles and compared against the algorithm classifications to measure its accuracy.

Algorithm Development

The 3DSSPP program allows the manipulation of several factors of the lifting task and subject, such the gender and anthropometry. A right-facing sagittal mannequin posed in the standing, stooping and squatting positions, with corresponding bounding boxes, is illustrated in Figure 2. Previous research has demonstrated that lifting posture states can be classified from angle measurements of the hip and knee (Burgess-Limerick et al., 1997; Anderson et al., 1985; Dysart et al., 1997). The definitions of torso and knee angles are illustrated in Figure 3.

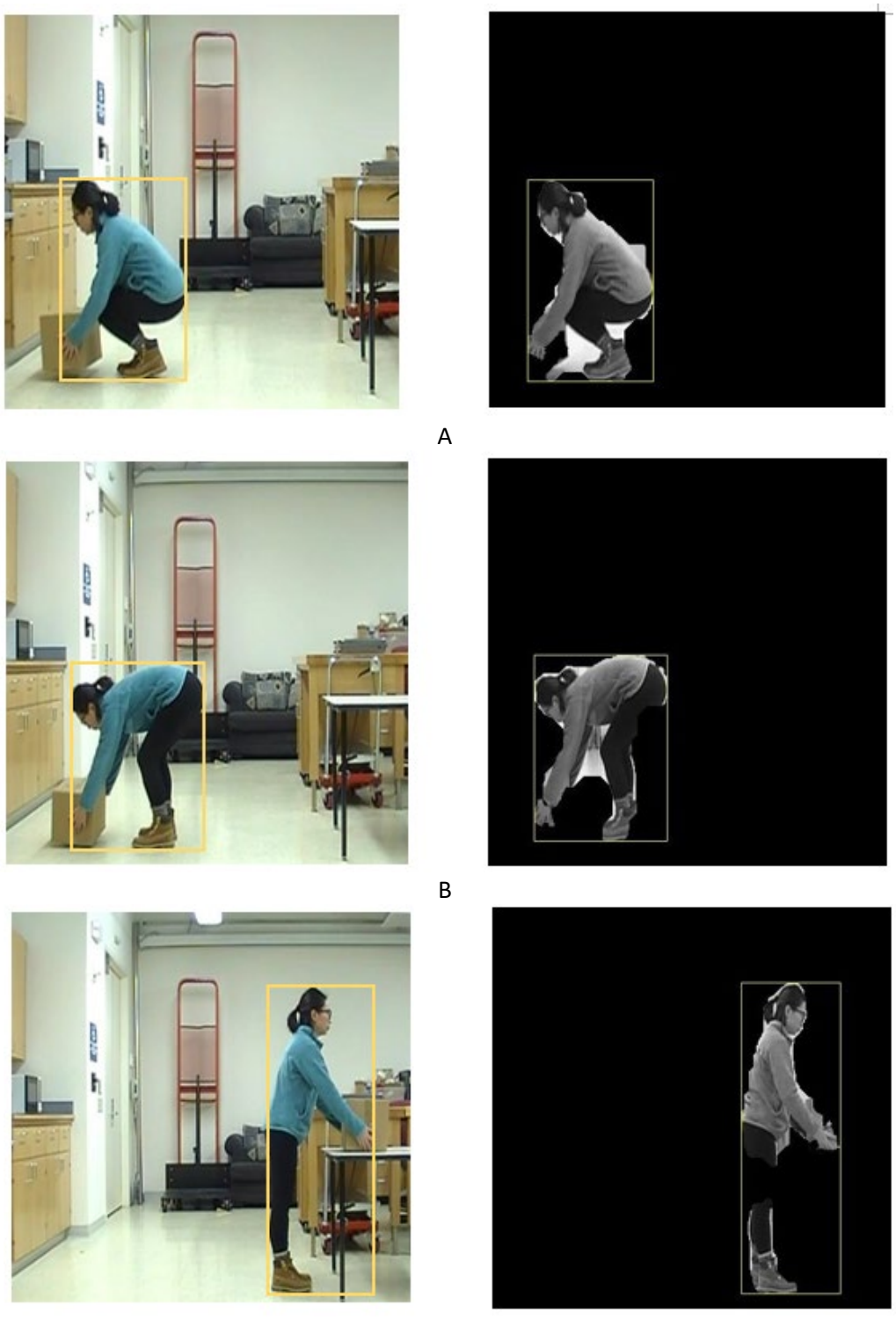
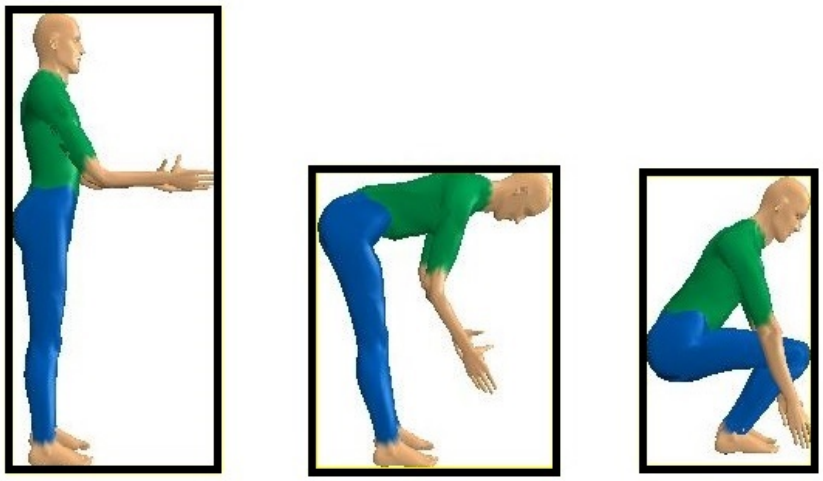
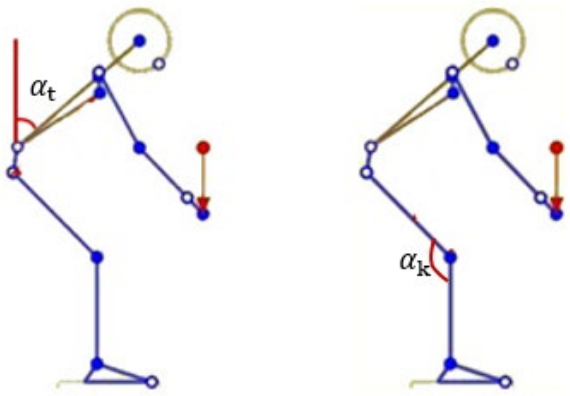


Figure 6.1: Original video frames (left) and background subtracted computer vision images (right) of a subject performing squatting (A), stooping (B) and standing (C) lifts during a lab-simulated task. In the task the subject picks up an object from the shelf on the left of the video frame, transfers and lowers it, and places it on the ground in the right portion of the video frame. A rectangular bounding box encloses the subject in each image, independent of the object lifted.



A B C

Figure 6.2: Examples from 3DSSPP of standard standing (A), stoop (B) and squat (C) lifts, and their associated bounding box dimensions.



A B

Figure 6.3: 3DSSPP models demonstrating torso angle (α_t) (A) and knee angle (α_k) (B) measured to classify lifting postures.

A squat is classified when the included knee angle (α_k) is $\alpha_k < 130$ degrees, and a stoop is classified when the torso angle (α_t) is $\alpha_t > 40$ degrees flexion from the vertical. If neither of these conditions are true, the lifting posture is classified as a stand. If both $\alpha_k < 130$ degrees and $\alpha_t > 40$ degrees, the lifting posture is classified as

a squat, since stoop lifts are intended to be done with relatively straight legs. This classification comes from the notion that a squatting subject may reach farther out to lift an object, requiring greater hip flexion greater while the knees are still bent. These classification rules are illustrated in Figure 4. The angles used to classify these lifting postures for the models can be determined from data provided in 3DSSPP.

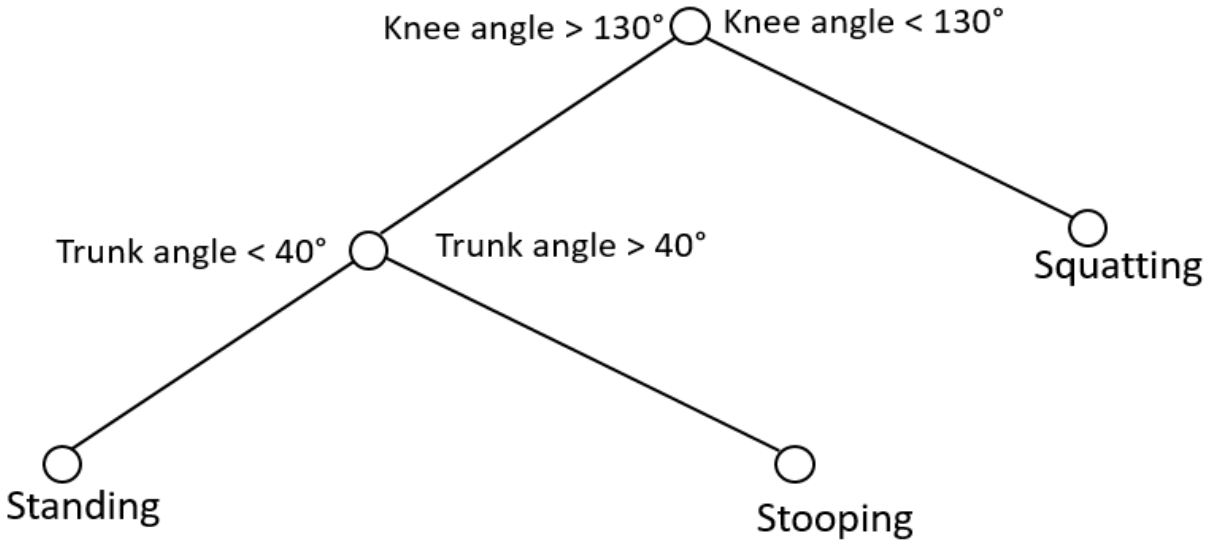


Figure 6.4: Rules of classifying standing, squatting and stooping lifting postures.

To consider the range of lifting locations, the manual lifting Threshold Limit Value (TLV) documentation developed by the American Conference of Governmental Industrial Hygienists (ACGIH) was used (ACGIH, 2017). The TLV classifies distinct zones (three horizontal and four vertical). The horizontal zones are defined by the horizontal hand distance from the center of the body, for 30 cm, 60 cm, and 80cm. The vertical zones are based on the anthropometry of the subject and use key landmarks including mid-shin height, knuckle height when the arms are completely extended, and 8 cm below shoulder height. These zones expand from the floor to the upper reach limits.

To reflect the differences in lifting posture between anthropometries at key zone boundary points, the 20 intersections of the horizontal and vertical zone markers on the TLV diagram were computed for each anthropometry using values from the 2012 Anthropometric Survey of U.S Army Personnel, and the hand location was entered into 3DSSPP to find the resulting bounding lifting posture state (Gordon, C.C, et al., 2014). A representation of the 20 zone intersections is shown in Figure 5.

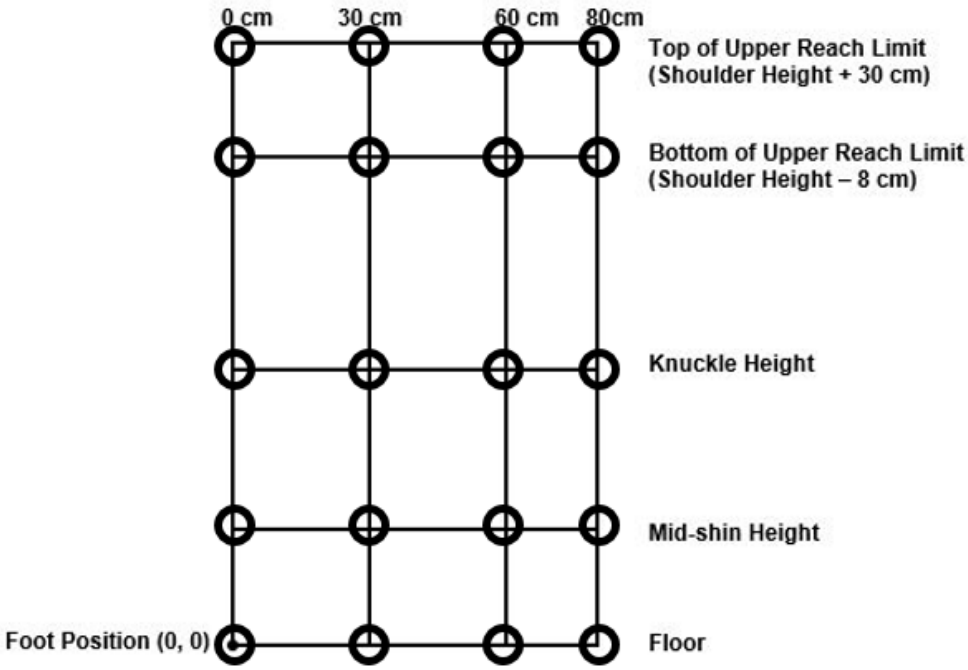


Figure 6.5: Representation of intercepts of 3 Horizontal and 4 Vertical Zones.

The lifting postures were generated by inputting the hand locations according the mannequin's anthropometry and applying the posture prediction feature of 3DSSPP (Hoffman, Reed and Chaffin, 2007; Chaffin, 2008). The bounding box dimensions of captured images in pixels were measured using University of Wisconsin MVTA software (Yen and Radwin, 1995). To determine the lifting state, the angles of the torso and upper leg of the subject were read from the 'Body Segment Angles' window in 3DSSPP. The goal was to record measurements for attainable lifting tasks with hand positions in each of the 12 TLV zones, and to consider each of the three different lifting postures within each zone for different anthropometries to demonstrate variability of lifting posture states between models of varying sizes and gender. This involved utilization of subjects from six anthropometry classifications available in 3DSSPP, including 5th, 50th, or 95th percentile height males and females. To account for subjects of varying anthropometries, the mannequin standing height was used for normalizing the bounding box H and W. The respective dimensions H and W were ratios of the bounding box height to standing height, and bounding box width, in pixels to standing height, in pixels.

After excluding those locations that were unreachable due to anthropometry, bounding box data was generated for 105 distinct lifting postures generated by 3DSSPP across six different anthropometries and classified into the three lifting states. Of these, 43 were classified as squats, 13 as stoops, and 49 as stands. These locations reflect the variation of parameters across differing lifting states and demonstrate how parameters vary for a lift based on the lifting posture definitions used in this study. Since in the workplace it is not always possible to locate a camera angle perpendicular to the subject, the mannequin for each of the 105 tasks were rotated by 30° with respect to the sagittal plane and bounding boxes were drawn and measured, creating 105 additional conditions.

An example of how the bounding box varies when the camera angle changes from a perpendicular view of the sagittal plane to a view rotated by ±30° degrees is shown in Figure 6. Since the width of the bounding box is the projection of the horizontal distance from the farthest left to right point on the mannequin image in the video plane, the width when rotated 30°, W_{30} = cos (30°) W = .866 W (Figure 7). Since the bounding box dimensions are identical when the camera is rotated 30° clockwise or counterclockwise, measurements were only taken when the camera was rotated clockwise in pixels using the MVTA software. The standing height in pixels was also measured for each anthropometry, and the bounding box dimensions of each additional condition were

normalized as ratios to the standing height of their associated anthropometry. Summary data for the bounding box dimensions of the 210 cases is provided in Table 1.

A B C

Figure6. 6: Examples of mannequin performing the same lifting posture with the camera rotated counterclockwise by 30° (A), perpendicular to sagittal plane (B), and rotated clockwise by 30° (C), and their associated bounding boxes.

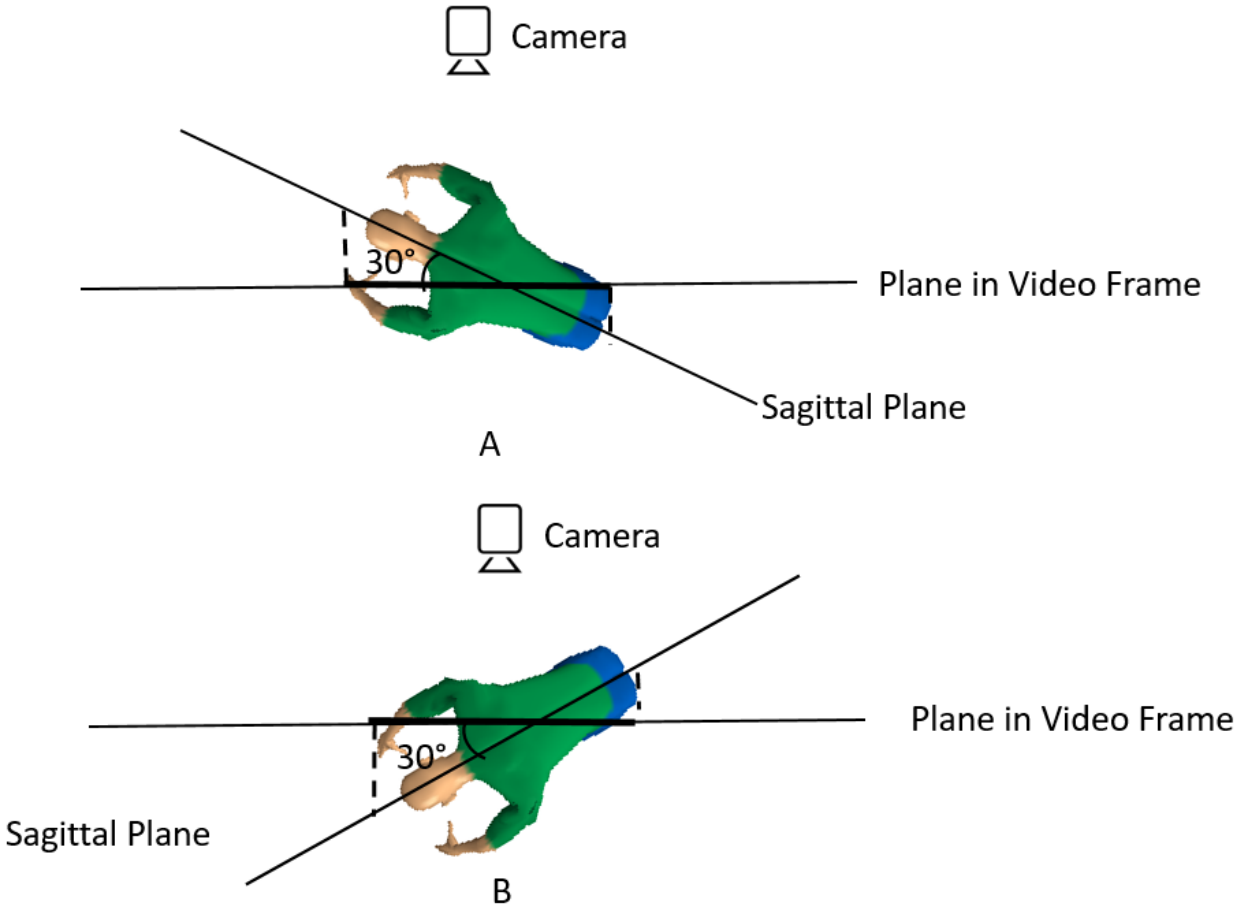


Figure 6.7: Demonstration of bounding box width as the projection of the mannequin's sagittal plane length on the plane on video frame, and the bounding box widths are identical for when the mannequin is rotated counterclockwise (A) and clockwise (B).

Table 6.1: Average ± Standard Deviation of Box Dimensions, Normalized to Subject Standing Height

		Box Height (H)		Box Width (W)	
Lifting Posture	Cases	Sagittal Plane	30º Rotation	Sagittal Plane	30º Rotation
Squat	86	0.54 ± 0.17	0.50 ± 0.08	0.58 ± 0.09	0.49 ± 0.11
Stoop	26	0.78 ± 0.07	0.81 ± 0.07	0.65 ± 0.06	0.61 ± 0.06
Stand	98	1.02 ± 0.06	1.02 ± 0.07	0.43 ± 0.09	0.36 ± 0.09

Because the 3DSSPP posture prediction algorithm prefers squat over stoop for the majority of entered hand locations, an imbalanced dataset resulted in which the number of stoop lifts were significantly less than the other classes. Random oversampling with replacement was applied in order to avoid biasing the algorithm towards the more represented classes. An additional 66 data points were resampled from the stoop class in both the sagittal plane and rotated 30°, resulting in 92 stoop lifts, which was the average number of cases in squat and standing lifts. In total, there were 276 data points in the training-set used to develop the decision tree. After randomly reordering the dataset, a classification and regression tree (CART) algorithm (Breiman et al., 1984) was used to generate a tree model that uses the bounding box H and W to classify the data into the lifting posture categories (Mathworks, 2017). The accuracy of the tree model was evaluated using a 10-fold cross validation method (Breiman et al., 1984).

Algorithm Validation

A total of 76 static images of lifting postures encompassing a variety of manual material handling tasks were randomly selected from videos recorded in industrial settings to test the algorithm's ability to classify lifting postures (Bao et al., 2016; Lu et al., 2013; safetyvideopreviews, 2012; University of Michigan Center for Ergonomics, 2014, 2017). These videos were recorded in various manufacturing plants and warehouses. Some examples of the tasks included lifting and stacking metal panels, lifting metal cylinders onto hooks on a conveyor, lifting and moving stacks of metal frames, lifting crates from a stack, and lifting a large, heavy stack of paper. The inclusion criteria for video frames are a) the full body of the subject is visible at the origin and destination, without twisting, b) the load is not occluded by other objects, and c) the camera is perpendicular to the subject's sagittal plane. Since only ten squat lifts were found, to attain equal number of lifting postures each class in the test dataset, ten lifting postures were randomly selected from the stooping and standing respectively, resulting in a dataset of 30 lifting postures.

For each video frame containing the lifting postures in the validation dataset, the bounding box height and width of the subject were measured using MVTA software (Yen and Radwin, 1995) and then normalized to ratios relative to the subject's standing height. The knee angle and trunk angle were also measured in MVTA to determine the classification of each lifting posture according to the definitions previously used.

This research complied with the American Psychological Association Code of Ethics and was approved by the Institutional Review Board (IRB) at the University of Wisconsin-Madison. Informed consent was obtained from each participant in accordance with the individual collaborating IRB.

6.3 Results

Decision Tree Model

The resulting decision tree representation of the bounding box height and width for each state is shown in Figure 8. The spatial relationships for the decision tree demonstrate how the algorithm would classify postures based on the H and W of a bounding box are shown in Figure 9.

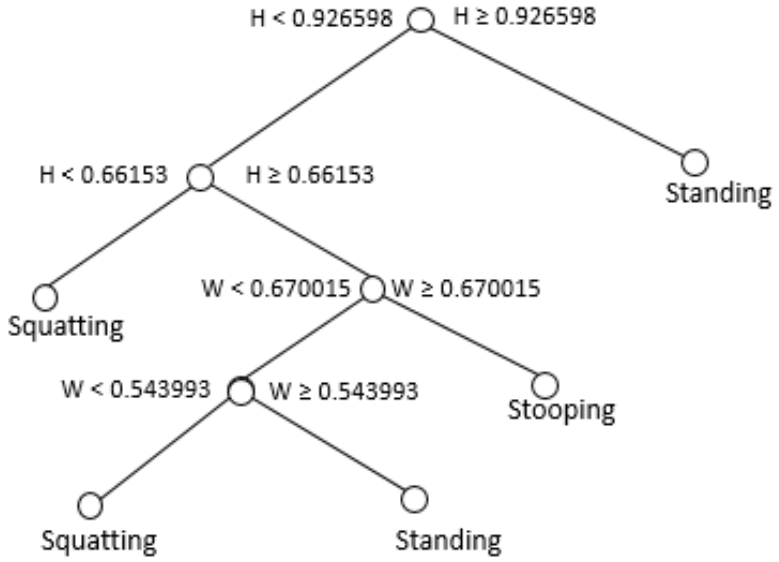


Figure 6.8: Decision tree generated from training data set, where H is the standing height-normalized height and W is the standing height-normalized width of the bounding box.

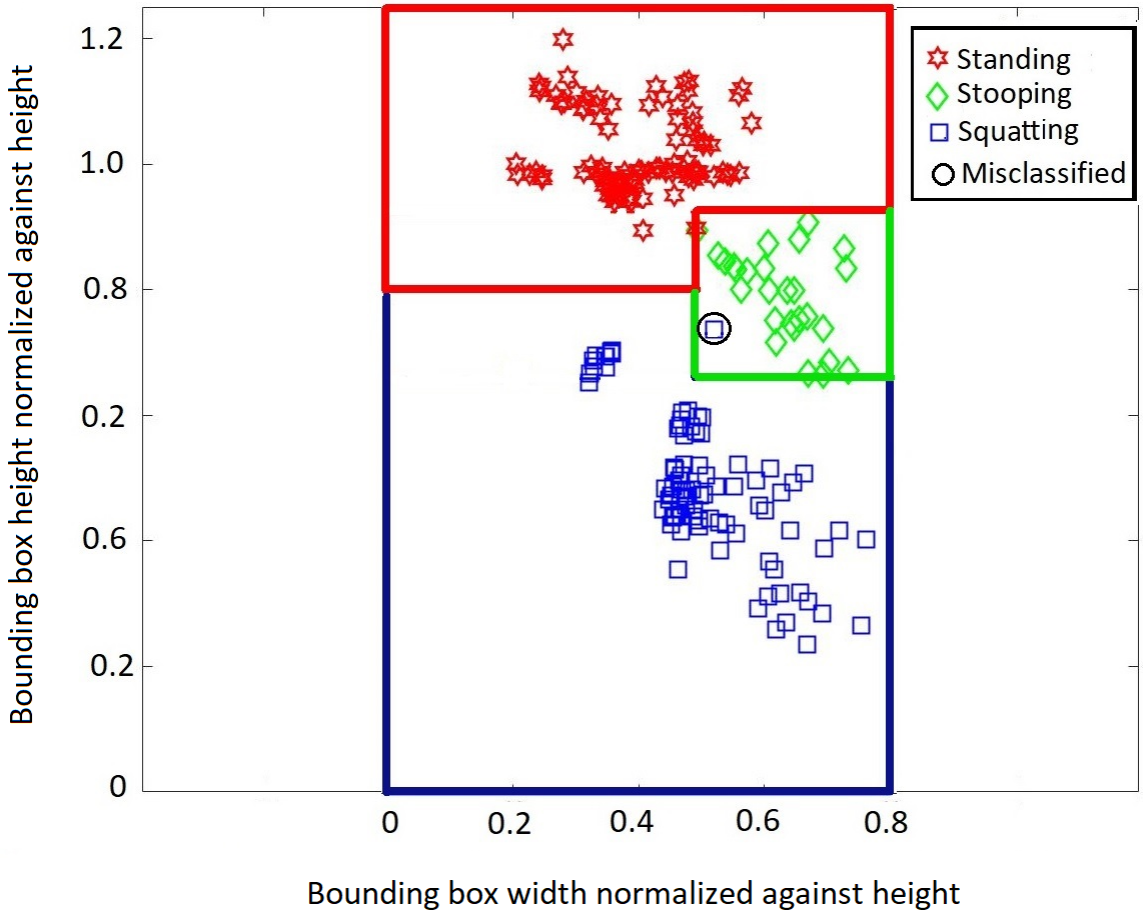


Figure 6.9: Spatial representation of decision tree classification results for the training-set based on the bounding box height (H) and width (W), each normalized by the subject's stature.

The re-substitution error obtained from applying the algorithm back against the training data set was 0.36%. The error rate of the tree model was 0.0108 using the 10-fold cross validation method.

A confusion matrix was created containing the number of lifting posture states correctly and incorrectly classified (Table 2). There was only one error when testing the algorithm against the training data, misclassifying the lifting posture as stooping while the lifting posture was squatting. The sensitivity and specificity of squat, stoop, and standing, respectively were 98.8% and 100.0%, 100.0% and 99.5%, and 100.0% and 99.5%.

Table 6.2: Confusion Matrix

	Algorithm Classification		
Ground Truth Classification	Squatting	Stooping	Standing
Squatting	85	1	0
Stooping	0	92	0
Standing	0	0	98

Algorithm Validation

The validation misclassified 3.33% of the test-set cases, in which 0 of 10 squats, 1 of 10 stoops, and 0 of 10 stands were misclassified. In the misclassified case the stoop lifting posture was classified as standing. The sensitivity and specificity were 100.0% and 100.0% for squat, 90.0% and 100.0% for stoop, and 100.0% and 95.0% for standing, respectively. The distribution of normalized bounding box heights and widths of the test-set and classification results using the decision tree algorithm generated with the training-set are presented in Figure 10.

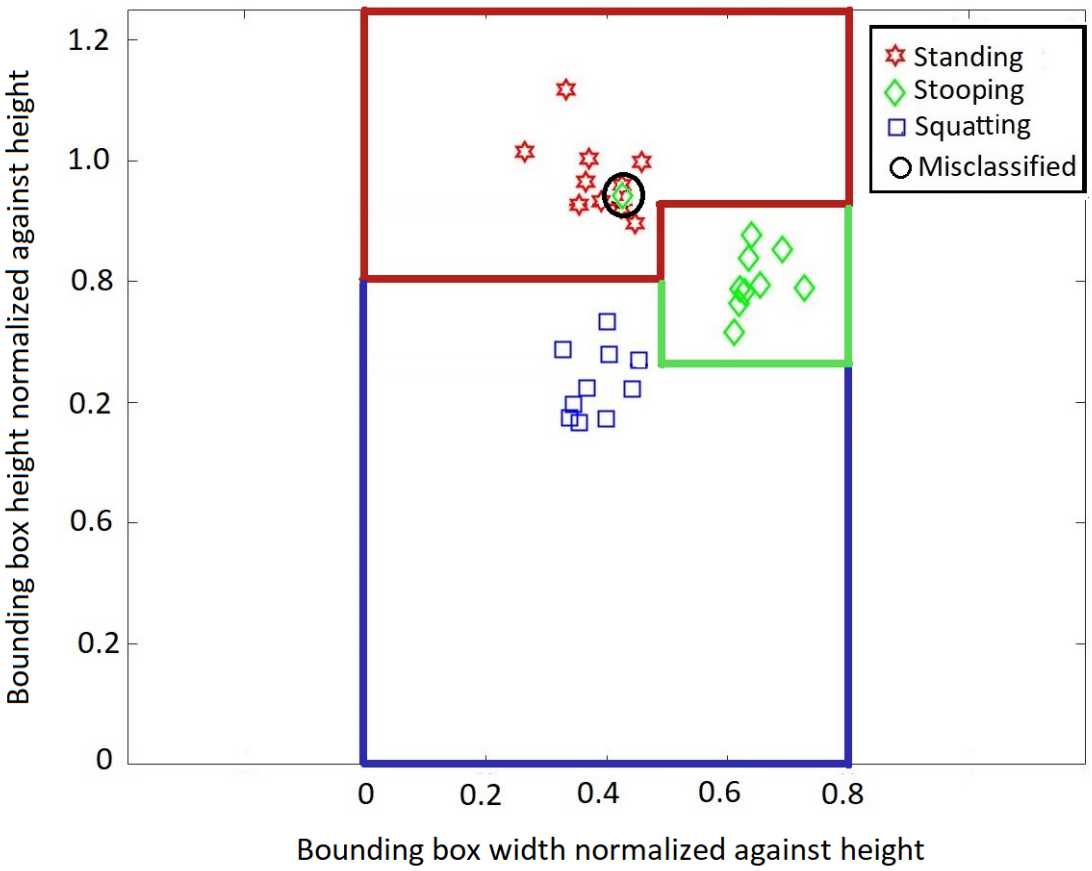


Figure 6.10: Spatial representation of decision tree classification results of the test-set based on the bounding box height (H) and width (W), each normalized by the subject's stature.

6.4 DISCUSSION

This research demonstrated the feasibility of classifying lifting postures based on only dimensions of bounding boxes tightly drawn around the subject, which are associated with the position of the hands and feet relative to the hip and knee angles, without direct angle measurements, instrumentation, markers, or fitting the image to a skeletal model. The results from the analysis demonstrate a relatively high rate of success for the algorithm correctly classifying a lifting posture. The re-substitution error of 0.36% obtained through applying the decision tree algorithm back to the original training-set appears to be extremely small. This is because re-substitution error is a very optimistic measure of error rate, and we have provided other measures of error to have a more comprehensive idea about the performance of the algorithm. The ten-fold cross-validation error of the decision tree algorithm was 1.08%. To further understand the algorithm's ability to classify cases, a set of 30 actual lifting tasks were entered to test the algorithm, and only one case was misclassified. Because the algorithm contained low training and validation errors for the wide range of samples used, there is high confidence that this algorithm could be applied to identify the lifting posture state for arbitrary lifts.

Given that the full decision tree generated multiple permutations, the branches obtained for the final tree had shown up in most iterations. When compared with the alternative decision trees generated using the three parameters in addition to the bounding box height to width ratio, the training errors remained practically unchanged. It was determined that the height of the bounding box was effective as a distinguishing parameter, as it was used at the first level of branching for all decision trees generated.

In the test-set, one stoop lift was classified as standing. The trunk flexion angle of the subject did not significantly exceed the threshold for stoop lifts ($\alpha_T = 41.67°$), and the legs were straight ($\alpha_k = 180°$). Consequently, the height of the resulting bounding box exceeded the threshold for standing, misclassifying it as a stand. Although this misclassified a lower-risk lifting posture (standing) as a higher-risk lifting posture (stooping), when trunk flexion angle is small, trunk flexion angle contributes less to the moment on the low back.

In addition to measuring bounding boxes when the camera's viewing angle was perpendicular to the mannequin's sagittal plane, bounding box dimensions were measured when the camera was rotated ±30° horizontally, for a 60° range off of the sagittal plane to account for when the camera is not perfectly aligned. The algorithm performed well despite a camera offset. It might be further improved by incorporating more camera angles, and the exact range of tolerance before lifting postures start to confound should be further studied to ensure appropriate limits. Future work will investigate methods of accommodating videos taken from various views so that the analysis is more tolerable of variations that may be encountered in field video recordings.

This research utilized the 12 TLV lifting zones and 3 anthropometries to account for the variability of lifting postures in relation to hand position for a given subject size and normalizing the bounding box dimensions as a ratio relative to standing height. The benefit of using 3DSSPP to train the algorithm is that a great number of lifting postures were easily generated using different TLV lifting zone boundary intersection locations. Since the lifting posture prediction feature of 3DSSPP automatically positions the subject according to an inverse kinematics algorithm based on the location of the hands in relation to the feet (University of Michigan Center for Ergonomics, 2017), the resulting lifting postures in this research are reproducible. The lifting postures generated by 3DSSPP, however, are not actual lifts and therefore may not be representative of actual lifting postures due to variations in individual characteristics, workers' preferences, training, and environmental factors including the nature of the object, obstructions, traction, and the workers apparel (University of Michigan Center for Ergonomics, 2017). Although the bounding box dimensions obtained from the test-set in the current study show similar spatial distribution as those of the 3DSSPP-generated lifting postures and the test of the algorithm on industrial tasks produced satisfactory results, the algorithm still needs to be tested on a wide variety of lifting tasks. The current study does not consider the effect of load on posture.

Another future improvement is to identify additional easily recordable parameters that could help further distinguish lifting states and assess their risk to the lower back. One parameter that might be important is the position of the feet relative to the hands and the horizontal corners of the bounding box. There is potential that this parameter would correlate with the degree of hip and knee flexion, though it would require an adjustment of the hand position measurements, since 3DSSPP determines those relative to the feet being the origin of the coordinate plane. To accomplish this, we are currently working on methods of detecting the hand and feet at the origin and destination of the lift from videos using computer vision.

This research demonstrates that it is possible to classify lifting postures as squat, stoop, or stand based on the bounding box dimensions of a subject with a significant degree of accuracy, which has several potential applications in assessing risks of manual lifting tasks. The findings from this analysis can be used to guide development of future computer vision algorithms that obtain data from continuous video recordings of actual occupational lifting tasks, with the overall goal of improving recommendations for occupational lifting safety.

We have developed computer vision software based on video background subtraction that continuously tracks the subject's body, applies the described bounding box and identifies origins and destinations of lifts. This program can perform these analyses automatically with reasonable accuracy. We intend to incorporate the posture classification algorithm described in this paper into this computer vision program, to automatically classify dynamic working postures for a series of consecutive frames in a video. In addition to standing, squatting, and stooping lift postures, the algorithm can easily identify other related states during work including walking (the bounding box moves without changing dimensions), lifting overhead (the bounding box height grows larger than the stature) and standing still (the bounding box does not change in dimensions or move).

Separating the human body from other elements in a video image based on the different patterns of movement, our computer vision based background subtraction algorithm can distinguish the human body regardless of variations in shape and size of the object being lifted. Additionally, since the computer vision algorithm draws the bounding box by detecting the silhouette of the subject and then drawing the box tightly around the silhouette, even if the object being handled blocks part of the body, the shape of the silhouette detected does not change and therefore the bounding box is not affected either. This feature is useful for providing reliability for applications in the field, where variations of objects being handled are common, and may obstruct the body linkages.

The method discussed in the current paper offers a practical solution to identifying working postures. Research has associated additional important risk factors in lifting in addition to working postures. For example, Marras et al. (1993) reported association between trunk dynamics and risks of low-back disorders. Lavender et al. (2003) found lifting speed as a significant factor for spine loading in L5/S1. Additionally, the weight of the object being handled also directly affects L5/S1 moment (Lavender et al., 2003), and is related to low-back disorders (e.g. Eriksen et al., 1999; Kerr et al., 2001). We expect to incorporate these additional factors with our computer vision-based approach to provide a comprehensive lifting analysis. The computer vision background subtraction algorithm and the posture classification method described in the current paper may be useful for continuously calculating the frequency of lifting tasks and the duration that each lifting posture is assumed during work, and this can greatly facilitate the analysis and evaluation of lifting tasks. We are using the same bounding box methodology for estimating parameters utilized in calculating the NIOSH lifting equation for evaluating the risks involved with lifting tasks, including horizontal, vertical and distance measures. Together these continuous evaluations may prove useful for offering additional information for continuously evaluating the risks associated with lifting, rather than just the NIOSH equation alone.

We anticipate that given the simplicity of the resulting algorithm, it could be executed with minimal processing time, requiring low processor power. With the advantages of tolerance of noise and low computing requirements, we anticipate that such methods can be implemented on a hand-held device such as a smart phone, making it readily accessible to ergonomic practitioners in the industry.

7. The Accuracy of a 2D Video-Based Lifting Monitor

7.1. Introduction

Overexertion during manual lifting ranks first among the leading causes of disabling workplace injuries in the US, and in 2017 cost businesses $13.79 billion in direct costs and accounted for 23 percent of the overall national burden (2017 Liberty Mutual Workplace Safety Index). In 2016, injuries and illness to the back resulted in higher rates of cases with both days away from work and days of job transfer or restricted work, compared to the head and hands (Bureau of Labor Statistics, 2016).

The revised NIOSH lifting equation (RNLE) (Waters, Putz-Anderson, Garg & Fine, 1993) is the most widely used tool to assess the risk of low back pain associated with lifting and lowering tasks in the workplace (Lowe, Dempsey, Jones & NIOSH, 2018). The recommended weight limit (RWL) for occupational lifting, calculated by the RNLE, is defined for a specific set of task conditions as the weight of the load that nearly all healthy workers could perform over a substantial period without an increased risk of developing lifting-related low back pain. The RWL is implemented in the International Organization for Standardization (ISO) standard 11228 Part 1 Manual Lifting and Carrying, and routinely used by practitioners for prevention of lifting-related low back pain. The RWL is calculated as:

$$RWL = LC \times HM \times VM \times DM \times AM \times FM \times CM \quad (1)$$

where LC is a load constant equal to 23 kg, *HM*, *VM*, and *DM* are a function of the distance between the location of the hands and feet at the origin and destination of the lift, AM is a function of the angle of torso rotation in relation to the feet, *FM* is a function of the frequency of lifting, and CM is a function of the quality of the hand-to-object coupling.

Today's manual materials handling jobs have evolved from single tasks decades ago to multiple and varying tasks, specifically in the manufacturing and transportation sectors, such as warehousing, distribution centers, package delivery trucks, baggage handling, lean or just-in-time manufacturing, kitting, palletizing and shipping. For prevention of work-related low back pain, it has therefore become increasingly important to frequently or continuously monitor workers' exposure to physical demands for varying job tasks.

Manually measuring the dimensions needed to calculate the RNLE is challenging, particularly in situations where lifting occurs in numerous locations involving varying body postures throughout the workday (Dempsey, 2002). Direct measurement and instrument-based methods have been tested that employ sensors or markers attached to the participant for measuring certain variables (Li & Buckle 1999; Juul-Kristensen, Hansson, Fallentin, Andersen & Ekdahl, 2001; Kim & Nussbaum 2013). However, these methods suffer from invasiveness of the measurement, the space needed for equipment, and high cost (Patrizi, Pennestri & Valentini 2016).

With the advancement of video technologies, cameras have become a prevalent, low-cost, highly efficient, and non-intrusive option for monitoring ergonomic aspects of job tasks. Both 2D (RGB) and depth (RGB-D) cameras have attracted researchers' attention for developing direct-reading technologies for ergonomic risk assessments. In recent years, the development of depth cameras, e.g. Kinect (Microsoft, Redmond, WA), has stimulated a new trend in measuring human body dimensions. These approaches rely on matching the recognized human body parts against a pre-trained skeletal model, which consists of the position of joints and body linkages (Gabel, Gilad-Bachrach, Renshaw & Schuster, 2012). Previous researchers used conventional cameras for creating a skeletal model from a large image data set (Bogo et al., 2016; Howe, Leventon & Freeman, 2000; Meherizi, et al., 2017, 2018a, 2018b).

Although 2D cameras are more prevalent than depth cameras, 2D cameras propose a more challenging partial body recognition problem. With one more dimension of information, RGB-D camera-based methods have been shown to provide more accuracy than RGB cameras (Patrizi et al., 2016; Spector, Lieblich, Bao, McQuade & Hughes, 2014; Delpresto et al., 2013; Plantard, Shum, Pierres & Multon, 2017). Two studies (Patrizi et al.,

2016; Spector et al., 2014) used the Kinect to measure RNLE parameters. However, these Kinect-based approaches were limited when there was occlusion; namely, when the Kinect did not capture a frontal view of the participant. Since the location of the hands and ankles for calculating the RNLE was dependent on the accumulation of intermediate predictions of each skeletal model linkage position, there were numerous distance errors.

In this paper, we describe a straightforward and practical video-based approach to automatically extract spatial and temporal factors necessary for applying the RNLE using a single 2D camera commonly available in hand-held mobile devices. It leverages motion information to directly detect the hand and ankle locations during lifting. It avoids the challenging partial body recognition problem by not depending on a skeletal model. The proposed method can be used to evaluate a work task with fewer computations and sufficiently high accuracy when compared with 3D camera approaches.

7.2. Methods

Lifting Monitor Algorithm

The lifting monitor algorithm takes advantage of motion information to distinguish the moving figure from the static background. Based on the spatial and temporal features of the video scene, a ghost effect is exploited to detect the lifting instance and hand locations. A rectangular bounding box drawn around the human figure is used to locate the feet for each video frame.

The algorithm contains three steps: (1) moving target detection, (2) action feature extraction, and (3) feature recognition, as shown in Figure 1. Each step is illustrated in Figure 2.

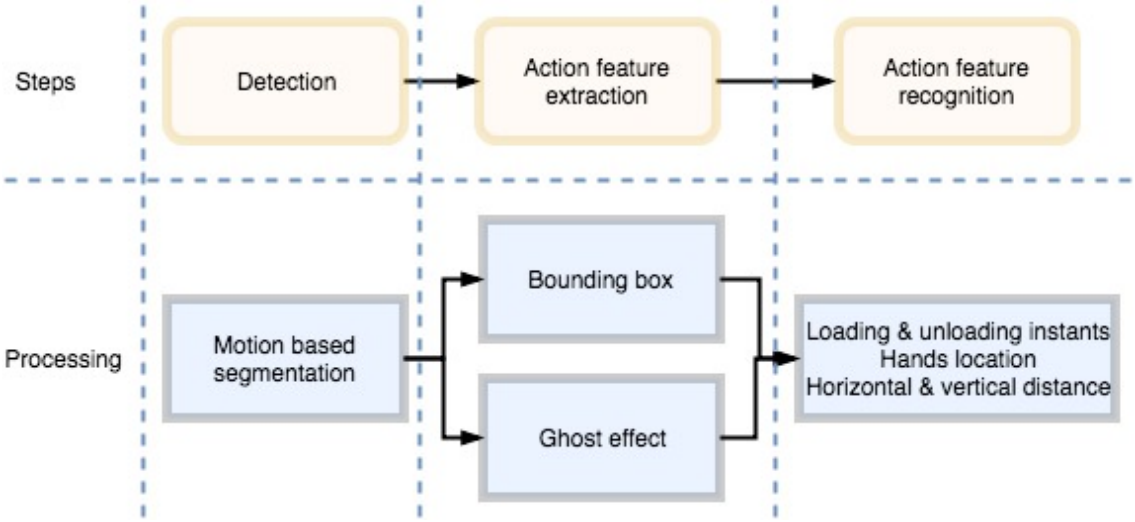


Figure 7.1. Flowchart of lifting monitoring algorithm

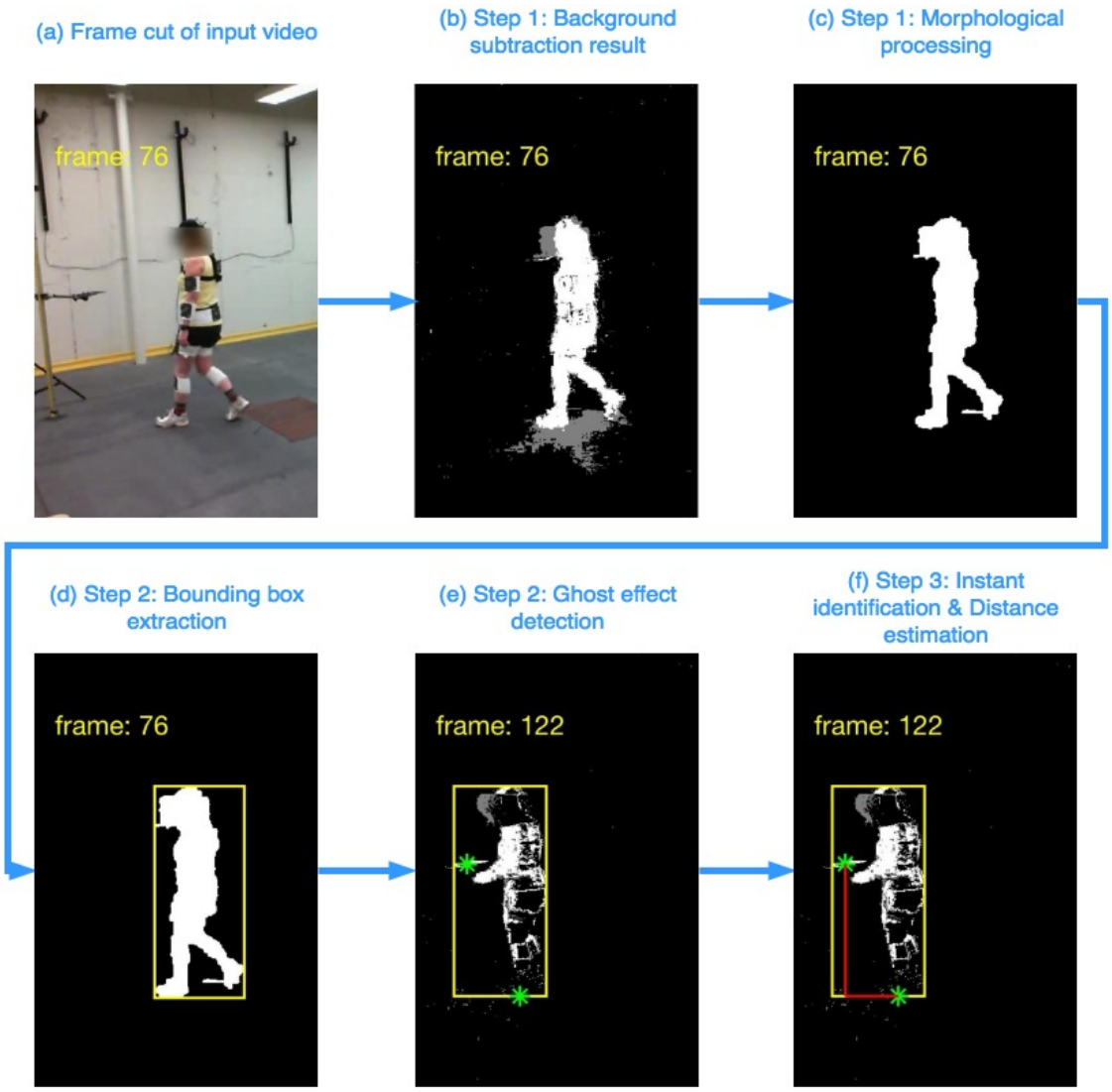


Figure 7.2. A demonstration of the video processing methodology for each step of the algorithm. Images (a), (b), (c) and (d) show the sequence of procedures for the 76th frame of a video where the participant walks to the lift location in the 122nd frame. Images (e) and (f) demonstrate hand and ankle detection at the 122nd frame of the video where the participant is lifting the object. A rectangular bounding box encloses the worker. The stars show the detected location of the hands and ankles. The lines connecting the stars indicate the horizontal distance from hands to the ankles and the vertical distance from the hands to the ground, respectively.

The moving target detection step leverages the fact that the figure moves during lifting while most of the surrounding environment is static, and thus a motion-based technique could segment out the moving items. In this algorithm, a mixture-of-Gaussian background subtraction algorithm (Zivkovic, 2004; Zivkovic & Van Der Heijden 2006), known as MOG2, is applied. The background subtraction algorithm builds a background model for each pixel using a mixture of Gaussian distributions and updates the weights of the texture to represent the time proportions that the pixel colors remain in the scene. The result is an $M \times N$ matrix map, where *M* and *N* are the height and width of the image in the unit of pixels, labeling each pixel as foreground (moving), background (static), or shadow, with respective colors white, black and gray. Thus, the detected foreground can be identified pixel-wise by the white-colored foreground mask.

Each of the connected clusters of white pixels is referred to as a “blob.” In a working environment where only the operator moves, the resulting map would capture the human figure perfectly with a foreground mask identifying the human figure's silhouette. But several moving objects and light reflections could introduce noise (false positives), and the foreground mask of the human figure might be incomplete and distorted (false negatives), as shown in Figure 2 (b). These false detections are due to a false match to the background model. False detections are eliminated by morphological operations based on their size relative to the larger size of the subject's silhouette, as shown in Figure 2 (c).

The action feature extraction step utilizes information from the first step. The height and width of the human figure's silhouette are extracted, represented by a rectangular bounding box tightly covering the human figure's foreground mask (Figure 2 (d)). Another feature utilized is a condition of the background subtraction, named the ghost effect, which is a set of connected points detected as in-motion, but not corresponding to any real moving object. This phenomenon appears when an object is moved away or set to a location, changing the appearance of an area (Figure 2 (e)). It persists for a short period, depending on how fast the background subtraction restores the original appearance of that area. The ghost effect is detected in accordance to the following four properties:

(1) Consistency in time. A blob of the ghost effect area should show up in the same location in subsequent *N* frames.

(2) Gradual vanishing. The size of the blob should be no larger than the size of the lifted object and should gradually get smaller.

(3) Proximity. When the ghost effect shows up, the human figure's silhouette (large-sized blob) should be close to it. In a non-ideal environment where there is more than one moving object, the ghost effect might occur in multiple places. Proximity is therefore used to focus the ghost effect caused by the human figure.

(4) Frame number. The frame number *N* is related to the duration that the ghost blob persists when the background model updates the new appearance of the ghost effect area into the background model.

The feature recognition step exploits the fact that the ghost effect occurs when the appearance in a location changes when a stationary object is moved by the participant, and, therefore, the loading instance can be identified from the start of a ghost effect. At these detected lifting instances, the respective location of each ghost effect indicates where the hands were, and the bottom blobs of the silhouette show where the feet were during the initiation of the lift. If the participant is recorded in the view from the sagittal plane, and the hands and feet are moving symmetrically about the sagittal plane, the ghost blob center can be used as an approximation of hands center during the initiation of the lift. Similarly, when an object stops moving, such as when setting down an object at the termination of a lift, the ghost effect occurs and persists indicating the instance of release.

The geometric center of the bottom portion of the silhouette blob (i.e. the lower 10% region of the bounding box while the participant is standing) can be used to approximate the horizontal location of the midpoint of the participant's ankles. Thus, the horizontal distance H from hands center to ankles (for the HM calculation) and the vertical distance V from hands to the ground (bottom edge of the bounding box for VM and DM calculations) can be estimated (Figure 2 (f)). The loading and unloading instances are important for H and V distance estimation because they indicate when to detect the hand and ankle locations. The algorithm considers the hands as the center by the ghost blob, which does not move in time. Distances measured in pixels were calibrated against the participant’s standing height.

Validation of the Algorithm

Laboratory Data

Data utilized in the current study were from a previous study conducted by researchers at the National Institute for Occupational Safety and Health (NIOSH). The aim of the original study was to record symmetrical lifting

tasks using the combination of two lifting task variables (horizontal and vertical distances) defined by the American Conference of Governmental Industrial Hygienists (ACGIH) Threshold Limit Values (TLV) for lifting (ACGIH, 2007). The TLV for lifting classifies 12 risk zones using the two task variables for symmetrical lifting in the sagittal plane. The horizontal distance (H) is defined as the projected distance on the transverse plane from the center of two ankles to the center of two hands. The vertical distance (V) is defined as the distance from the center of two hands to the ground. The midpoints of risk zones were used as the positions for starting the lifting tasks, except for Zones 1-3, 4, 7, and 10. The alternative starting locations of the lift tasks for the exceptional zones were chosen for realistic lifting motion within each participant's reach envelope. The origins of the 12 lifting tasks in relation to the participant's neutral body position are shown in Figure 3. Each task was repeated three times for a total of 36 lifting trials for each participant. These trials were assigned to each participant in a random order.

A 0.45 kg wire grid measuring 36 cm × 12 cm, with two cut-out handles, was used to simulate lifting a tote box during the trials. The grid was designed to help participants create realistic lifting motions while minimizing obstructions for motion tracking. The grid was set on a small platform (12 cm × 12 cm) for setting the initial lifting height. The platform was connected to a movable clamp on a metal pole for adjusting the lifting height. The small platform provided clearance for participants to lift the grid in a natural motion. Participants aligned their toes against one of three marked lines in the lifting area to create three horizontal distances for three grouped risk zones (Group 1: Risk Zones 1,4,7 and 10; Group 2: Risk Zones 2,5,8 and 11; Group 3: Risk Zones 3,6,9,12). These lines were 25.4 cm, 45.7 cm and 71.1 cm from the center of the grid (i.e., the center of two hands) to the center of their ankles.

Participants were asked to walk from a marked line (i.e. initial position) to the lifting area and line up their toes against one of the marked lines for the lifting task. The vertical height of the grid for the lifting task was set up prior to each trial. The distance between the initial position and the starting point of the lift task typically required participants to take 3-4 steps prior to lifting the grid. After lifting the grid, participants were asked to turn around and continue to carry the grid and set it down on a shelf in a fixed height of 77.5 cm. After setting down the grid, they were asked to turn around and walk to a marked finish line for completing each trial. The distance between the shelf and the finish line typically required participants to take three to four steps to complete the trial. Participants were instructed to walk and lift/carry the grid with two hands at their own pace and in their preference of direction for the two turnarounds. They were also instructed to carry the grid in front of their body to minimize trunk asymmetry. A trial was completed continuously in about 15 seconds. Participants practiced a few times until they familiarized themselves with the entire experimental procedure.

The video data were recorded by a web camera (Microsoft 1080p LifeCam) in synchronization with whole body motion capture system (Optritrack 12 IR camera system, model Flex 13 with the MotionMonitor data acquisition program, Innovative Sports, Inc., Chicago, USA). The web camera was set up in a fixed location at the typical eye level (165 cm from ground) and approximately 3.92 m in distance from the beginning of each trial. The video camera viewing angle was 88° measured between lines drawn from the camera to the participants' initial standing point and drawn perpendicular to the participant's path, and was near perpendicular to the direction of the lift. The camera viewing angle was offset by 31.7° from perpendicular to the sagittal plane at the lifting location. The resolution of the video recordings was set at 640 pixels × 480 pixels at a 30 fps rate to meet the hardware synchronization requirements for the motion capture system.

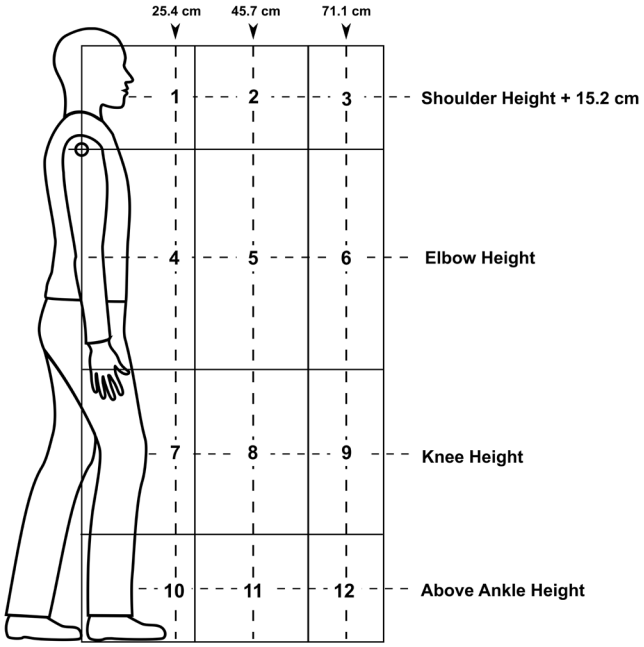


Figure 7.3. The starting point of 12 different lifting tasks was designed using the 12 risk zones of the ACGIH TLV for Lifting. The intersections of the dotted lines are the origins of the tasks and most were in the center of the risk zones. The vertical heights for Tasks 1-3 were adjusted to 15.2 cm above the participants' shoulder height. The horizontal distances for tasks 1, 4, 7 and 10 were adjusted to 15.4 cm from the center of the two ankles. These adjustments were made to create a realistic lifting motion. Adapted from ACGIH_TLV lifting risk zone system.

Motion tracking marker clusters were attached to 13 body segments for tracking whole-body motion by the MotionMonitor program using 12 IR cameras. The body locations for marker clusters and five inertial moment unit (IMU) wearable sensors (Kinetic Inc.) are shown in Figure 4. The (IMU sensors were for a previous study and not used in the current study). The motion capture system was calibrated according to the standard operating procedure of the OptiTrack company to achieve an average 0.7 mm accuracy across ten test sessions for motion measurements in three-dimensional space.

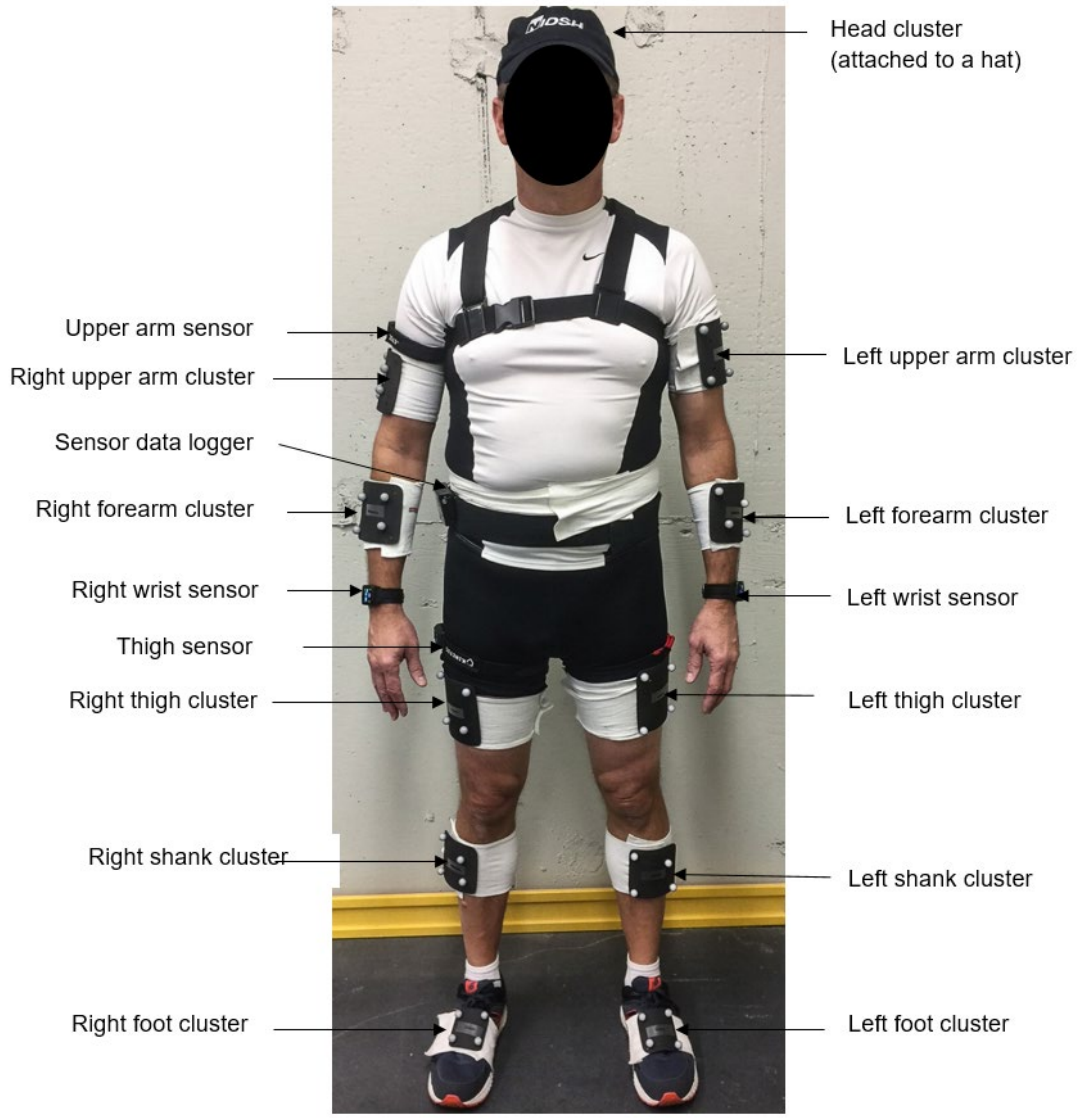


Figure 7.4. Demonstration of wearable sensor and marker cluster attachments to the body landmarks for the motion capture system. Each cluster has four small retro reflective Styrofoam spheres geometrically configured to be different from one another for motion measurements. Two clusters (upper and lower back) and one wearable sensor attached to the back of the chest Velcro assist harness are not visible in this picture. White elastic Velcro straps are used for the clusters, and thinner black elastic Velcro straps are used for the upper arm and thigh wearable sensors. Two wrist sensors are attached by adjustable rubber bands.

Data acquired and derived from the motion capture system were used as ground truth for evaluating the accuracy of the video lifting monitoring algorithm. Four variables were used for validation: the time instance at the beginning of the lift (BOL), the time instance when the plate was set down at the end of the lift (EOL) as well as H and V for the BOL. Two NIOSH researchers reviewed video recordings of the trials in the MotionMonitor program and manually recorded the video frame numbers to establish the BOL and EOL. The criterion for determining the frame numbers was based on the moment when the grid started to move for the BOL with two hands. The two NIOSH researchers independently reviewed half of the trials, but when in doubt, they discussed questionable frame numbers to reach agreement on the final value. Once the video frames for the BOL were determined, the data were used to identify the H and V variables calculated with the motion capture data. A similar procedure was used to determine the EOL when the grid was initially set down.

Participants

Six participants were recruited among employees in the division of the Applied Research and Technology office of NIOSH in Cincinnati, Ohio. Prior to data collection, written consent was obtained from the participants using the NIOSH-approved IRB study protocol. The participant inclusion criteria were individuals that were capable of (1) lifting a 1.4 kg mass in a location combining different horizontal distances from their body to the load within their reach and vertical distances from the shin to shoulder height, (2) lifting and carrying the 1.4 kg mass 3 m, and (3) repeating 12 different tasks (described previously) 3 times for a total of 36 lifting trials. Exclusion criteria were individuals with musculoskeletal disorders or any pain at the time of recruitment or in the past three months, individuals under age 18 years, and being pregnant. Demographic and measured anthropometric data relevant to the study are provided in Table 1.

Table 7.1. Demographics and anthropometric properties (Mean ± SD) of study participants (Male: N=3; Female: N=3).

Gender	Mass (kg)	Height (cm)	Age (years)	Forearm length (cm)	Upper arm length (cm)	Thigh length (cm)
M	101.8 ± 14.7	176.3 ± 1.4	55 ± 1.9	27.7 ± 1.4	32.3 ± 1.5	44.6 ± 3.5
F	69.7 ± 7.4	163.6 ± 4.6	48 ± 13.6	26.4 ± 1.7	30.7 ± 1.7	44.3 ± 4.5
Average	82.5	169.3	51.3	27.1	31.5	43.3
SD	18.9	9.0	11.5	1.9	2.0	3.9

The lifting monitoring algorithm was applied using the validation dataset (6 participants with 36 clips each). Given a video clip as input, the algorithm automatically outputs the bounding box of the human figure for each frame, the detected loading and unloading instance, hands location and feet location at the loading and unloading instance, and the RWL.

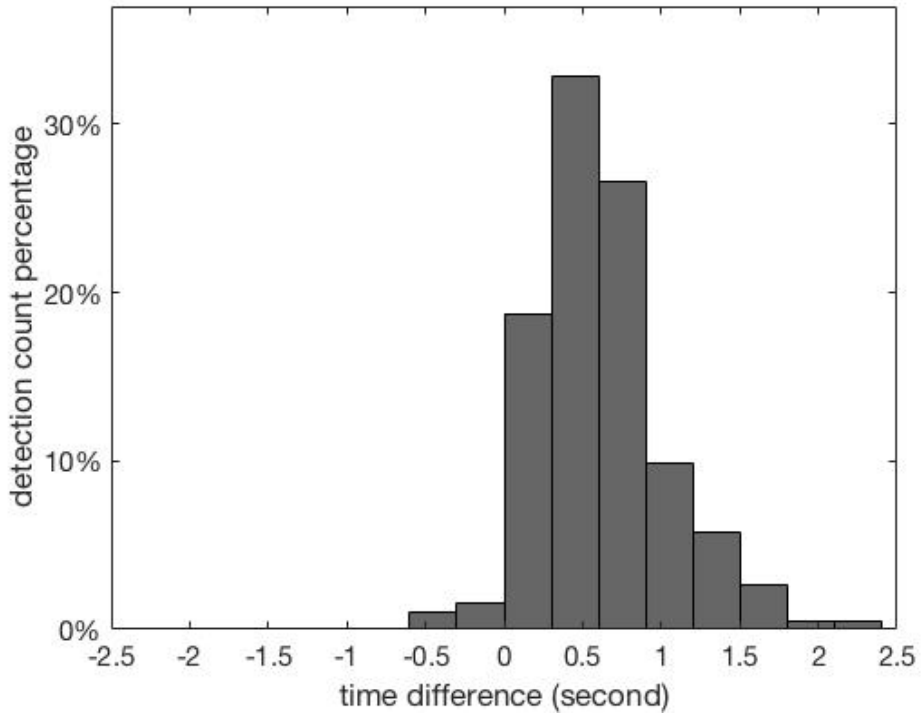
The NIOSH study was designed to focus on the origin of the lift in the 12 lift zones. The RWL for the loading instance at the BOL was calculated, including the horizontal distance from the hands center to the ankles center (H), and the vertical distance from the hands midpoint to the ground (V).

7.3. Results

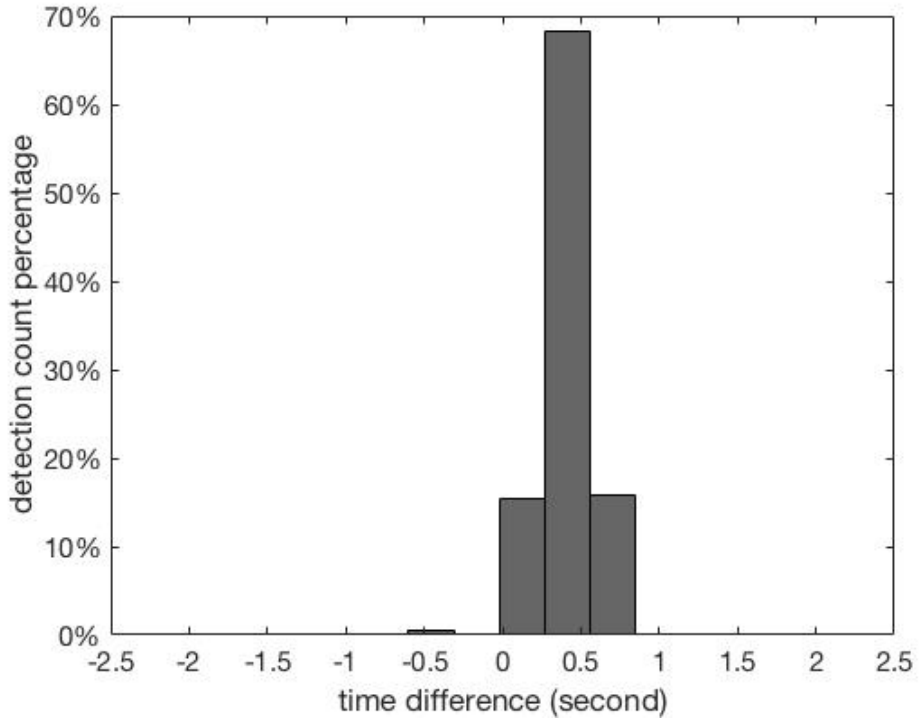
Lift and release instance detection

Since the ankles during loading remained steady for a short period, when the predicted lifting instance and the ground truth instance coincided with this period, the predicted lifting instance was considered successful. In this validation experiment, the lifting instance estimation was successful for all clips. The predicted lifting instance timestamp was compared with the manually observed instance (ground truth). The time difference between prediction and ground truth was calculated.

A histogram of the time difference for the BOL is plotted in Figure 5 (A). The mean time difference was 0.624 s (SD = 0.42 s). Based on this distribution, 99.5 percent of the measured lifting instance detections were within [-0.636, 1.884] s of the ground truth measure. A similar time difference for the EOL histogram for the time difference is plotted in Figure 5 (B). The mean time difference was 0.144 s (SD=1.52 s). According to this distribution 99.5% of the measured unloading instance detections were within [0.125, 0.833] s of the ground truth measure.



A



B

Figure 7.5. Histogram for the time difference between ground truth and (A) the beginning of lift instance detection time, and (B) the end of lift detection time.

H detection at the loading instance

The horizontal distance from the hands to the ankles (H) is a key factor in calculating the RWL. We reference the H data from the 3D motion capture system as ground truth and calculate the H difference between the video estimated H and ground truth H. A histogram for the H difference is plotted in Figure 6. The mean of the H difference was -1.16 cm (SD = 4.72 cm). Based on this distribution, 68% of the measured lifting instance detections were within ± 5 cm of the ground truth measure.

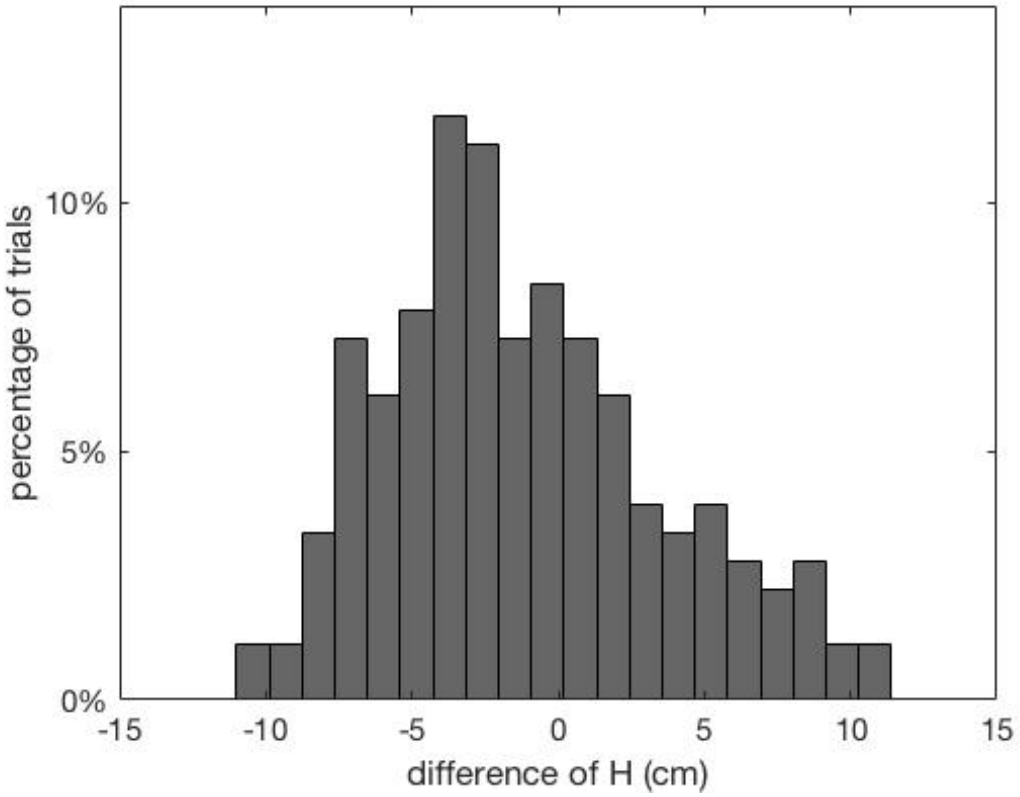


Figure 7.6. Histogram of the difference of H between motion capture and video detection at the loading instance.

V detection at the loading instance

The vertical distance from the hands to the floor (V) is another key factor in calculating the RWL. We reference the V data from the 3D motion capture system as ground truth and calculate the V difference between estimated and ground truth V. A histogram of the V difference is plotted in Figure 7. The mean of the V difference was -0.36 cm (SD = 3.22 cm). Based on this distribution, 88% of the measured lifting instance detections were within ± 5 cm of the ground truth measure.

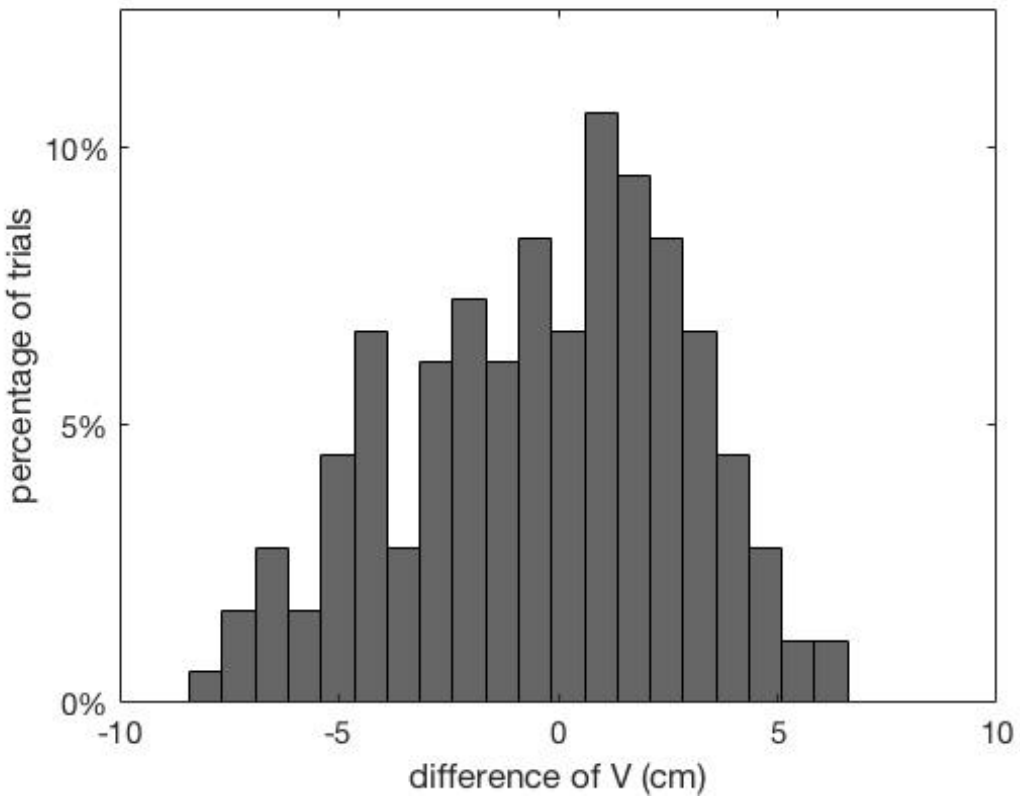
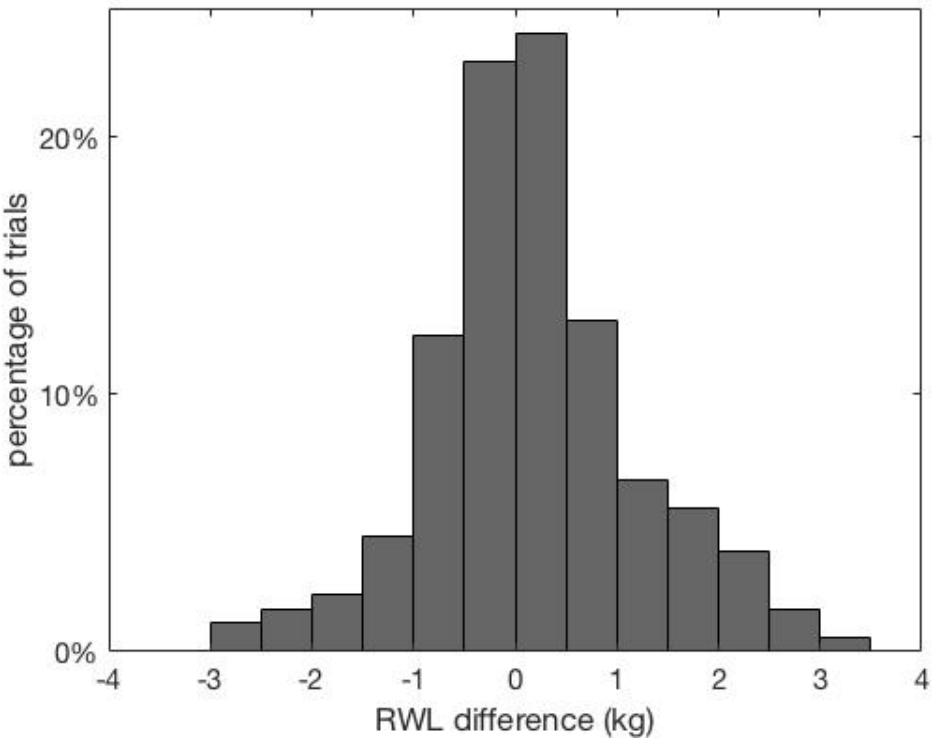


Figure 7.7. Histogram of the difference of V at the loading instance.

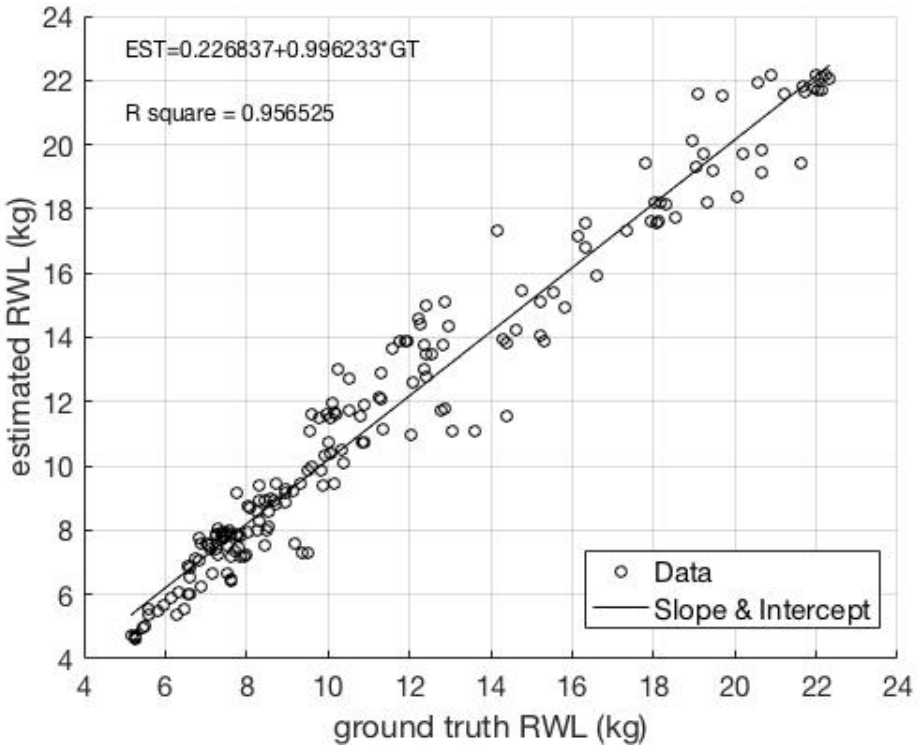
RWL prediction at the loading instance

Comparing the multipliers between the computer vision estimations and motion tracking ground truth, the 95% confidence interval of the VM difference was 0.005 ± 0.016, the DM difference was 0.002 ± 0.011, and the HM difference was 0.012 ± 0.111. In calculating the RWL, the LC was 23 kg. AM was set equal to one because the participants were asked to perform the lifting trials while keeping their torso upright without twisting. CM was also set equal to one because the effectiveness of hand-to-object coupling was considered good for all lifting trials. Since each lifting task was performed only once and was not repetitive, we calculated the Frequency Independent Recommended Weight Limit (FIRWL) and set FM = 1. The variables V and H were the values at the loading instance. D was the vertical distance from the origin of the object to the instance before the object is loaded at the destination. VM, HM, and DM were calculated according to the multiplier transformation function required in the RNLE, respectively.

The RWL difference between estimation and ground truth was calculated and shown in Figure 8 (A) where the histogram is plotted. The mean RWL difference was 0.18 kg (SD = 1.03 kg). Based on this distribution, 91% of the measured lifting instance detections were within ± 2 kg of the ground truth measure. The linear regression between the video estimated RWL and ground truth RWL ($R^2 = 0.96$) is shown in Figure 8 (B). A Bland-Altman analysis for the limits of agreement (mean difference $\pm$ 1.96 SD of differences) for the estimation and the motion capture data were between -1.84 kg and 2.21 kg. This demonstrates that the estimation sufficiently agreed with the ground truth measure.



A



B

Figure 7.8. (A) Histogram of the difference of RWL at loading instance. (B) Linear regression for ground truth motion capture and video estimated RWL.

7.5. Discussion

Accuracy of Distances

The calculation of RWL in the validation experiment relies on the variables VM, HM, and DM, which are functions of V, H, and D, respectively. The greatest bias of RWL came from the HM because its error range was on the order of 10^{-1}, while the error range of VM and DM was on the order of 10^{-2}. The relatively larger error in H can be explained by the camera viewing angle in this study.

The algorithm is built on the assumption that the camera viewing direction is perpendicular to the sagittal plane and thus the hands and feet should overlap because of symmetry. Therefore, the detection of the lifted object can be ascertained from either hand. However, in this validation experiment, the camera viewing angle was offset by 31.7° from the perpendicular at the lift location, where ideally there would be no offset. Consequently, the geometric center of the un-overlapped extremities would likely bias the location measured. Thus, this validation data were for the worst-case scenario. Since the data were from a previous study, it was not possible to change the video camera angles.

The computer vision algorithm measured distance in units of pixels, while the motion capture ground truth measurements were provided in centimeters. The conversion between these two was therefore dependent on the video camera set-up. The angle between the sagittal plane at the lifting point and the camera viewing plane was 6.1°. After converting the projection of the H distance from the video camera viewing plane back to the sagittal plane, the distance estimation assumed that all the dimensions of the participant share the same depth from the camera. Without knowing the actual focal length of the camera, we used the participant's standing height as a reference for calibration. However, in actual video recordings, the calibration is distorted at the video frame edge and the variant depths are not consistent with uniform calibration.

As detection of center of the hands and feet directly comes from the center of the corresponding blobs, the precision of the algorithm relies on the precision of the blob. If the blob is incomplete because of dilution or enlarged by noisy blobs, the center of the blob will be biased, and thus the location estimation will be biased. If the resolution of the video was greater so that the calibration ratio was less, then the error caused by non-uniform shaped blobs would be reduced.

False negative detections

If the algorithm cannot identify a ghost blob, or if the location of the detected ghost blob does not comply with human body structure, the algorithm would fail. Using this criterion, there were 17.13% false negative detections in this dataset.

In order to be detected by the algorithm, the ghost blob should be complete and not diluted for a consecutive sequence of N frames. To avoid noisy blobs, an arbitrary N was set as 20 frames. If the isolated ghost blob lasted less than N frames after detaching from the participant's silhouette, the algorithm would not identify the ghost blob. A long-duration overlap of the ghost blob and the participant's silhouette mostly depends on the participant’s moving speed. In this validation dataset of six participants, one participant moved significantly slower than the others. This made the isolated blob persist for a shorter time resulting in ten false negative detections for this participant, while for other participants, there were no more than three false negative detections. Because the algorithm depends on motion, a very slow-motion lift could potentially impair accuracy. This effect will be studied in future investigations.

A weak appearance of an object can lead to a weaker blob. In this validation experiment, the lifted object was a thin grid, and thus its ghost blob was often weak and seen as a shadow or was diluted. Additionally, this study used a relatively low definition video (640 pixels $\times$ 480 pixels) due to the limitation of the 3D motion capture system. Consequently, the plate appeared as a line segment when viewing in the plane of the plate surface. The background subtraction for this plate was represented as a 2 pixels $\times$ 20 pixels blob. This caused the ghost blob to rapidly diminish and was considered as noise with high probability. This coarse representation

undoubtedly provided less information for the algorithm, while a higher definition video should significantly improve the resolution of the blobs and thus the algorithm accuracy. A solid object would certainly form a much stronger blob.

The current study was conducted in the laboratory under controlled conditions where the background was stationary and there were no extraneous motions. In an industrial setting it is anticipated that the background might contain other workers, vehicles, or machinery that can introduce multiple ghosting images from their movements. Proximity was used to identify the ghost that was caused by the participant. This methodology will be refined in future work in order to reject noise introduced by extraneous motions.

Since the algorithm relies on motion information to detect a moving target, if the appearance (colour and tone) of the foreground and the background were close, the motion would be hard to differentiate, and the algorithm would be challenged, introducing error. However, the algorithm does not detect targets according to colour features, so the markers worn by participants for the motion capture system did not have any noticeable effect on the outcome.

Compared with previous research aimed at estimating body position using 2D or 3D videos (Bogo et al., 2016; Howe, Leventon & Freeman, 2000; Meherizi, et al., 2017, 2018a, 2018b), the current algorithm directly detects the hand and feet locations while avoiding the previous intermediate steps of recognizing specific body segments or fitting skeletal models. This approach not only reduces computational complexity but is not sensitive to possible errors cause by miscalculation of any single body segment location, which can offset the entire measurement. Future research will study how this feature will help prevent errors potentially introduced in the industrial setting where obstacles, poor illumination or occlusion may interfere with obtaining accurate measurements.

Spector et al. (2014) collected a large dataset comprising six participants performing lifting tasks in a laboratory setting. The RNLE parameters were provided using a Kinect, while the ground truth was provided by a 3D optical motion capture system. A manual inspection was performed to exclude the obvious outliers before data processing. However, in the current study, no outliers were excluded, and all data were used regardless of lifting style or anomalies encountered. Comparing with the modelled data, our H estimation error was more concentrated around 0 with its 25^{th} to 75^{th} percentiles ranging within [-0.04, 0.02] m and median being -0.02 m. For the modelled data in Spector et al. (2014), the 25^{th} to 75^{th} percentiles ranged from [0.06, 0.13] m, and the median was 0.09 m. The V estimation error in the current study had 25^{th} to 75^{th} percentiles ranging within [-0.03, 0.02] m and a median of 0 m, while the 25^{th} to 75^{th} percentiles in Spector et al. (2014), ranged within [-0.02, 0.04] m and median was -0.02 m. This shows that the V estimation errors were similar for both studies. The cited values from Spector are the average of approximated values observed from the boxplot figures published in Spector et al. (2014). The current study estimation of H and V had no outliers removed, compared with the many outliers in Spector et al. (2014).

The accuracy of location prediction for the current algorithm is dependent on the detection and accuracy of blobs. Shadows, shading and highlights caused by illumination, blur the boundary between the foreground and the background, and are important factors affecting the performance of segmentation accuracy. Chondagar et al. (2015) acknowledge that although multiple solutions for these problems have been proposed, they have not yet been solved. As discussed previously, there were 17.13% false negative detections in this dataset (i.e. 82.9% correct detections). The-state-of-art semantic segmentation approach by Long et al. (2015), which uses a fully convolutional network, presents pixel segmentation accuracy only as high as 90.3%. Consequently, good illumination is necessary for accuracy using the current approach. The current approach combines the ghost appearance feature and motion information to improve the accuracy of blob detection. Considering its computation complexity, the current approach has achieved good performance in blob segmentation for a laboratory setting.

Recommendations for improvement

An effective improvement in error could be achieved by providing a quality assurance measurement. Some factors influencing the algorithm accuracy include, but are not limited to, how close the participant's hands and shoes are similar to the background in colour as well as whether there was proper illumination to make the significant objects clear without shadows.

A higher definition video could significantly help to detect the ghost blobs. For example, the thin plate which was represented by a $2 \text{ pixels } \times 20$ pixels blob in the current $640 \text{ pixels} \times 480$ pixel resolution video, would appear as a $9 \text{ pixels} \times 120$ pixels blob in a 4K ($3840 \text{ pixels} \times 2160$ pixels) resolution video. Thus, it is less probable it would be eliminated as a noise blob. The videos in the current study were limited to $640 \text{ pixels} \times 480$ pixels resolution in order to synchronize the video with the 3D motion capture data, but we anticipate that a more dense pixel resolution will provide more distinct blobs with greater contrast and details. Also, an object with more visual density than the current transparent thin plate would certainly improve detection. Furthermore, increased pixel resolution would provide greater precision in the distance measurements.

This validation dataset was conducted under a worst-case viewing angle (31.7°) and would certainly achieve a better distance estimation if the camera were set perpendicular to the sagittal plane. Future studies will examine the amount of error that was contributed by camera viewing angle.

The camera technology of today’s devices has better resolution, the algorithms offer suitable computational simplicity, and the method has suitable stability for a hand-held camera. Additionally, an anti-shake algorithm will be explored. In the natural work setting, the most challenging problem for this algorithm is multiple moving objects. As the final design of the video lifting monitor is implemented in a hand-held device, most of the noisy moving objects could be avoided in the image by manually choosing the recording angle.

This investigation was a laboratory study, involving controlled conditions, limited to symmetrical lifting and included relatively few participants (six). Future field studies will further test the algorithm performance, including actual lifting conditions and a larger number of participants to truly validate the method; the current investigation however was important because it enabled comparison of conventional video against ground truth motion capture data, something very difficult to control in the field. Future demonstration studies will be conducted in industrial facilities involving actual manual materials handling tasks in production, warehousing and supply chain operations. It is acknowledged that generalizability of this study is limited, given the number of participants, controlled lifting parameters, and restrictions imposed by laboratory conditions, however the results demonstrate that this novel approach has promise for future field applications.

In this paper, a 2D-video based lifting analysis was presented. The system provides a robust, non-intrusive method to automatically extract the spatial and temporal factors necessary for applying the RNLE using a single video camera in the view from the sagittal plane. Compared with 3D motion capture, our approach provides sufficiently high accuracy of the RWL predictions. It also provided automatic detection of lifting instances. Efficiency in computation load and non-intrusive properties will make it possible to be implemented on a hand-held device and accessible for a wide variety of applications.

8. Publications from this Research:

Li, Y., Greene, R. L., Mu, F., Hu, Y. H. and Radwin, R. G., Towards Video-Based Automatic Lifting Load Prediction, *Proceedings of the Human Factors and Ergonomics Society 64th International Annual Meeting*, 2020.

Greene, R. L., Lu, M. L., Hu, Y. H., and Radwin, R. G., Relationship Between Computer Vision Estimated Trunk Kinematics and Work-Related Low-Back Pain, *Proceedings of the Human Factors and Ergonomics Society 64th International Annual Meeting*, 2020.

Greene, R. L., Lu, M. L., Barim, M. S., Wang, X., Hayden, M., Hu, Y. H. and Radwin, R. G., Estimating Trunk Angle Kinematics During Lifting Using a Computationally Efficient Computer Vision Method, *Human Factors*, 2020.

Greene, R. L, Lu, M-L., Barim, M. S., Wang, X., Hayden, M., Hu, Y. H., and. Radwin, R. G., Estimating trunk angles during lifting using computer vision bounding boxes, *Proceedings of the Human Factors and Ergonomics Society 2019 Annual Meeting*, Seattle, WA., 2019.

Akkas, O., Harris-Adamson, C., Bao, S., Lin, J-H., Meyers, A. R., Rempel, D., and Radwin, R. G., Defining repetitive hand exertions for exposure assessment, *Tenth International Conference on the Prevention of Work-Related Musculoskeletal Disorders*, Bologna, Italy, 2019.

Harris, C., Akkas, O., Bao, S., Lin, J-H., Meyers, A. R., Rempel, D., and Radwin, R. G., Biomechanical risk factors for distal upper extremity tendinosis, *Tenth International Conference on the Prevention of Work-Related Musculoskeletal Disorders*, Bologna, Italy, 2019.

Wang X., Hu, Y. H., Lu, M. L., and Radwin, R. G., The accuracy of a 2D video-based lifting monitor, *Ergonomics*, 68(8), 1043-1054, 2019.

Wang, Xuan, Efficient Methods for Video-based Human Activity Analysis, PhD Thesis, University of Wisconsin-Madison, 2019.

Greene, R., Hu, Y. H., Difranco, N., Wang, X., Lu, M. L., Bao, S., Lin, J. H., and Radwin, R. G., Predicting sagittal plane lifting postures from worker image bounding box dimensions, *Human Factors*, 61 (1), 64-77, 2019.

Radwin, R. G., Hu, Y. H., Wang, X., Difranco, N., “Movement Monitoring System,” U.S. Patent No. 10,482,613, November 19, 2019.

Radwin, R. G., Hu, Y. H., Wang, X., “Movement Monitoring System,” U.S. Patent Application No. 16/038,664, Filed June 22, 2018, Filed July 18, 2018.

Akkas, O., Bao, S., Harris-Adamson, C., Hu, Y. H., Lin, J-H., Meyers, A. R., Rempel, D., and Radwin, R. G., The Speed Calculated Hand Activity Level (HAL) Matches Observer Estimates Better than the Frequency Calculated HAL, *20th Congress of International Ergonomics Association - IEA 2018*, Florence, Italy, 2018.

Greene, R., Hu, Y. H., Difranco, N., Wang, X., Lu, M-L., Bao, S., Lin, J. H., and Robert G. Radwin, Automated Video Lifting Posture Classification Using Bounding Box Dimensions, *20th Congress of International Ergonomics Association - IEA 2018*, Florence, Italy, 2018.

Akkas, O., Hu, Y. H., Lee, C-H., Harris-Adamson, C., Rempel, D., Bao, S., Lin, J-H., Meyers, A. R., and Radwin, R. G., How Do Computer Vision Upper Extremity Exposure Measures Compare Against Manual Measures?, *Human Factors and Ergonomics Society International Annual Meeting*, Philadelphia, PA 2018.

Akkas, O., Lee, C-H., Hu, Y. H., Harris Adamson, C., Rempel, D., and Radwin, R. G., Measuring exertion time, duty cycle and hand activity level for industrial tasks using computer vision, *Ergonomics*, 60(12), 1730-1738, 2017.