Hindawi
Journal of Sensors
Volume 2019, Article ID 1787351, 14 pages
https://doi.org/10.1155/2019/1787351

Research Article

Sensor-Based Weightlifting Workout Assisting System Design and Its Practical Implementation

Marcel Luzar,[1] Andrzej Czajkowski,[1,2] Rafał Kasperowicz,[2,3] Maciej Rot,[2] Mateusz Semegen,[2] and Józef Korbicz[1]

[1]*Institute of Control and Computation Engineering, University of Zielona Góra, ul. Podgórna 50, 65-246 Zielona Góra, Poland*
[2]*Heavy Kinematic Machines Sp. z o.o., Komorowo 4, 64-200 Komorowo, Poland*
[3]*Poznań University of Economics and Business, al. Niepodległości 10, 61-875 Poznań, Poland*

Correspondence should be addressed to Marcel Luzar; m.luzar@issi.uz.zgora.pl

Received 15 October 2019; Revised 21 November 2019; Accepted 11 December 2019; Published 31 December 2019

Academic Editor: Jesús Lozano

The paper is devoted to the problem of counting repetitions and automatic weight stack detection in a weightlifting machine used for weight training. Some weightlifting machines include a weight stack that can be adjusted by a user. For example, the user can choose to increase or reduce the weight load using the weight stack, thus changing the difficulty level of a particular exercise. Users may want to perform a desired number of repetitions of an exercise or to perform an exercise with a desired range of motion when using such weightlifting machines. From a medical point of view, it is often required that an exact number of repetitions of a given load are performed in order to aid in a prompt recovery. This paper describes a complete design of the system and the algorithm that allows one to collect data from such a machine. Data can then be used for various purposes: in overviews and statistical analysis of a number of workouts. The approach presented in this paper is part of an application for European patent number 18461537.5-1126.

1. Introduction

Society is becoming increasingly health-conscious. A wide variety of exercises and workouts are now offered to encourage people to stay fit. Measuring many aspects of physical activity is also very popular [1, 2]. There are many devices on the market that allow one to, e.g., count the number of steps taken during a day or measure the heart rate while running (which is often combined with route-tracking capabilities). In most cases, these are wirelessly connected to mobile devices such as a smartphone, a smart armband, or a smartwatch [3]. Data gathered by such devices can be later used to calculate training statistics, which can be useful in determining what the subsequent stages of training should be, contributing to the achievement of one's personal development goals. Moreover, from the medical point of view, such data can provide valuable information to the clinicians and GPs, influencing the ways, in which the patient treatment is developed. Usually, in order to make such calculations, a dedicated mobile application needs to be developed.

In contrast to the number of devices monitoring aerobic exercises (such as running, walking, or climbing stairs), there are few devices used for measuring activity during strength training [4–8]. In general, a strength training plan incorporates information on the personal progress. In most cases, the type of exercise performed, the number of repetitions, or load is recorded manually using plain notebooks. Thus, there seems to be the need for a system that can collect data from gym equipment in an automatic way. In particular, such a system should count the repetitions and the weight lifted when using a machine (with a weight stack). This data may in turn be used for various summaries and statistical analyses of a number of workouts. Together with such a system, a suitable mobile application should be provided. Aimed at achieving the above, the weightlifting-assistant system is developed and presented in this paper.

Analyzing the state of the art in research literature, one notices that there are several solutions presented in [9]. The "Beast Sensor" is based on a wristband equipped with an accelerometer. The measurements taken in such an approach are very inaccurate. Another solution is based on the Microsoft Kinect sensor [10], which is also inaccurate and in addition cannot measure the weights lifted. There are also certain solutions proposed in the patent applications. The first one, entitled "Exercise monitoring system" [11], describes an exercise monitoring system consisting of a static-stack light transmitter, a static-stack light reflector, and a static-stack light receiver. The static-stack light transmitter should be positioned to emit a light beam towards the bottom of the weight stack where the static-stack light reflector is located. The beam goes through the holes in the weight stack with its length related to the height of the stack. The exercise monitoring system can also include an active-stack light transmitter, an active-stack light reflector, and an active-stack light receiver. The active-stack light transmitter should be positioned to transmit light to the active-stack light reflector located at the top of the weight stack. The active-stack reflector extends from the top of the active stack in such a manner so as to be in the path of the light emitted from the active-stack light transmitter in order to reflect light to the active-stack light receiver. The amount of reference light received by the active- or static-stack light receivers is to be used by an analyzer in order to output information regarding various features of the exercise being performed, such as the range of motion, the total amount of weights lifted, and the number of repetitions. The drawback of this solution is that due to the fact that the distance measurements have to be very precise (especially in the case of the static stack), the system requires very precise sensors. This in turn translates into higher costs of such equipment. The second patent application, entitled "Sensor arrays for exercise equipment and methods to operate the same" [12], describes a linear array of sensors. The array of sensors includes a number of sensors positioned adjacent and opposite to the resting position of each weight plate and at equally spaced locations above the example stack of weights up to the highest possible movement position of the top weight plate of the example stack of weights. Such a sensor array can be a built-in feature of the equipment or it can, in one way or another, be attached to it. The drawback of this solution is that it necessitates the use of a considerable number of sensors, which is in fact greater than the number of weight plates themselves. This is a result of the fact that the sensors must extend up to the highest possible movement position of the top weight plate. Therefore, due to the number of required sensors, this solution is also ineffective with relation to cost.

The analysis reveals that there is a need for a cost-effective solution where a low number of relatively low-complexity sensors would suffice to achieve the objective of counting repetitions and lifted weight in the above context. This paper presents such a solution. Moreover, the novelty is the fact that the data collected can be sent to an Internet cloud service, which is very popular nowadays (e.g., [13, 14]) or directly to a specified storage space (e.g., a smartphone using Bluetooth functionality). In this way, one can easily analyse the data using relevant computer software and, therefore, track the progress of the training. In addition, a physician or a physiotherapist prescribing a training regimen for their patient can monitor the progress (by, e.g., logging into a cloud service) of the patient and see whether the exercises were executed with the prescribed number of repetitions and with correct weights.

2. The Design of the Proposed System

The proposed system for assisting weightlifting workout involving a number of weight plates consists of

(i) a memory

(ii) a controller

(iii) at least three sensors each comprising an emitter/receiver pair managed by the controller

in which each weight plate contains a reflective marker designed to reflect a signal of the emitter and reflect it towards the receiver and the three sensors are located along the movement path of the stack as follows: the middle sensor being the first to detect the first weight plate, when the weight stack is in the resting position, taking into account the direction of the movement; the top sensor being the sensor following the middle sensor taking into account the direction of the movement; and the bottom sensor being the sensor preceding the middle sensor taking into account the direction of the movement. The system further comprises a proximity sensor configured to identify particular users operating the system. Each emitter is configured to emit a signal detectable by the receiver. The mentioned reflective markers occupy only a section of the side area of the corresponding weight plate, while the remaining side area of the corresponding weight plate reflects the emitted signal to a lesser extent than the reflective marker. Each reflective marker comprises two distinct elements: reflector A and reflector B, having different reflective properties. Reflector A reflects more light than reflector B, but both reflectors reflect light to such an extent that a receiver can differentiate between the reflectors themselves, a weight plate, or an empty space in front. Each sensor comprises an emitter as well as two receivers placed some distance from each other along the axis of the movement of the markers. The size of the respective type B reflectors and the type A reflectors is at least equal to the rectangle defined by vertices at the positions of the two receivers. There is either more than one top sensor present in the group of top sensors or more than one bottom sensor present in the group of bottom sensors. The system further comprises an external communication means configured to relay workout parameters and statistics to external devices.

Another aim of the article is to present a method for automatic weight counting during an exercise session on a given weight machine. The method comprises the following steps:

(i) Activating all the sensors except for the top one

(ii) Awaiting for the middle sensor to detect a marker

(iii) Determining the lowest sensor that detected a marker

(iv) Activating the top sensor and determining the number of markers detected by the lowest-positioned sensor that detected at least one marker (in the case the top sensor detects the first weight plate)

(v) Calculating a number of markers using the following formula:

$$\text{sensors\_index} * \text{offset} + \text{count\_of\_step\_407}, \quad (1)$$

where the offset is the number of weight plates between the adjacent sensors and sensors_index is the number of sensors between the middle sensor and the lowest sensor that detected a marker

(vi) Summing the weights of all the weight plates by adding the number of plates counted by the last sensor to the number of plates calculated using the equation

In the case when the lowest sensor does not detect a marker, the number of markers detected by the middle sensor is taken into account after the top sensor detects the first weight plate.

The method used in calculating the number of repetitions consists of the following steps:

(i) Setting the "Ready to notify repetition" flag to false

(ii) Determining the direction of the movement of the weight stack

(iii) In the case the weight stack is moving downwards and the "Ready to notify repetition" flag is set to true, the repetition is recorded and the "Ready to notify repetition" flag is set to false. If the stack moves upwards, the "Ready to notify repetition" flag is set to true

3. A Description of the System Elements

A diagram of the described system is presented in Figure 1(a). The system consists of a set of modules aimed at monitoring workout activities on a weight stack system.

The electronic system can be realized using dedicated components or custom-made FPGA or ASIC circuits. The system comprises a system bus (*101*) connected to a memory (*104*). Additionally, other components of the system are connected to the system bus (*101*), and therefore, their operation can be altered by a controller (*105*). Obviously, instead of a system bus (*101*), separate electrical connections can also be used. An external communication means (*108*) can be made use of in order to update the instruction list in the controller (*105*) and to communicate workout parameters or statistics to external devices, e.g., via an Internet connection. Such an external communication means (*108*) can include, e.g., Bluetooth LE or Wi-Fi. A radiofrequency identification system, i.e., the RFID system proposed in [15], may be utilized in order to identify particular users operating the system. Such a user can be identified by the system by means of a smartphone device equipped with the RFID function or by means of a suitable piece of workout clothing, such as a glove, also equipped with the RFID function.

The system can also be designed using several modules located on the exercise machine, e.g., on a weight stack. Such modules comprise at least one reed switch (*106*) (or a similar contact/proximity sensor, such as a Hall effect sensor) providing operating current when two contacts are in proximity or directly connected to each other. Typically, a magnet will be used to activate the reed switch. The function of such a reed switch (*106*) is two-fold: it indicates a low power mode when such a switch has not been activated for a longer period of time (a predefined period of, e.g., 3 minutes), and it identifies the first weight plate of a given weight stack.

Moreover, the system comprises at least one emitter/-receiver pair (*102*, *103*). Preferably, at least one emitter is a light emitter (*102*) while the corresponding receiver (*103*) is configured to conditionally receive a signal from the corresponding emitter (*102*). The described conditional reception requires the presence of an appropriate reflective marker as will be explained shortly. Thus, in the described example configuration, each emitter (*102*) emits a signal in a given axis (for example, horizontally or vertically) depending on the positioning of the weight plates. Each emitter (*102*) is configured to emit a signal detectable by the receiver (*103*), such as a beam of visible light, although other signals, such as radio signals and infrared signals, may also be used. In the described example configuration, the emitter (*103*) is an infrared diode, while the receiver (*102*) is a phototransistor.

As will be evident from the material presented in the subsequent figures, at least one emitter/receiver pair (*102*, *103*) is positioned on one side of the exercise equipment. Therefore, each weight plate comprises a reflective marker configured to reflect the described detectable signal coming from the emitter (*102*) directing it towards the receiver (*103*).

As is evident from Figure 1(a), the system is fully operational with just a single emitter-receiver pair (*102*, *103*). However, a configuration using a number of the emitter-receiver pairs (*102*, *103*) is also possible as will be shown in the subsequent parts of the specification. The abovementioned reflective markers (*111*) are such that they occupy only a section of the side area of the corresponding weight plate (as shown in Figure 1(b)), while the remaining side area of the corresponding weight plate (*110*) is covered in matte paint (*112*) and reflects the emitted signal to a much lesser extent than the reflective marker.

When a weight plate, equipped with a reflective marker (*111*), passes by the phototransistor (receiver), it registers a change in properties of the received signal (light) from a given state to another state. Obviously, one is able to set different thresholds for the definitions of a high and a low state according to preferences.

The reflective marker (*111*) comprises two distinct elements (*111A*, *111B*). Their color is in principle of no significance: what is important is that their reflective properties are preserved. Reflectors A (*111A*) and B (*111B*) have different reflective properties. For example, type A reflectors (*111A*) reflect more light than type B reflectors (*111B*). Nevertheless,

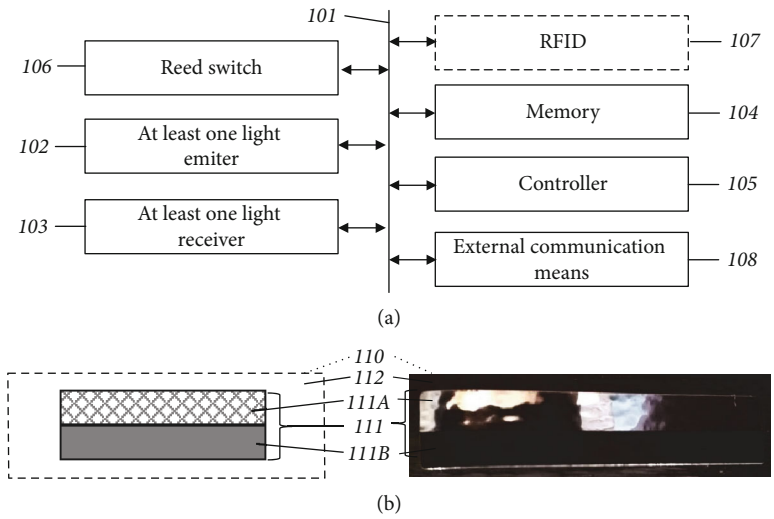


Figure 1: (a) System diagram. (b) A reflective marker—the scheme and a real-life example.

both types of reflectors (A and B) reflect light to such an extent that it is possible to differentiate between such reflectors and the weight plate as well as between situations where there is an empty space and between weight plates, in front of a receiver (*103*).

The reflective parameters of the markers (*111*) should fall in the range of values appropriate for an analog-to-digital converter (ADC) responsible for converting signals received from the respective sensors. In the case of a 12-bit ADC, the range is 0–4095. For the weight plate (when a typical dark matte paint, e.g., black, is used), the returned values are usually below 2000. The type B reflectors (*111B*) usually return values in the range of 1300 to 3500, and one should keep in mind that it ought to reflect twice as much light as the weight plate. In the case of ADC values above 3500, it is assumed that the type A reflectors (*111A*) are detected. The specific values and ranges as described above depend on the ADC resolution. The minimal value describes black while the maximal value describes white. For example, a 14-bit ADC will have a range of 0 to 16383 and the respective values of a 12-bit ADC will shift proportionally $\times 4$. Obviously, the respective reflective properties can be defined using different ranges as long as it is possible to clearly differentiate between the weight plate and the B reflectors (*111B*) and A reflectors (*111A*).

Figure 2 presents an example of a combined emitter/receiver pair (*102*, *103*). In this particular configuration, an integrated sensor comprises an emitter (*203*), typically an infrared LED, as well as two receivers (*202*, *204*) positioned on a diagonal (e.g., top-left side and bottom-right side).

Typically, the two receivers (*202*, *204*) will be the same. A configuration with different receivers (*202*, *204*) is also possible; it is however a more complex situation to be managed by the controller (*105*). Regarding the mentioned diagonal placement, other arrangements are possible as long as there is a shift along the expected axis of movement of the markers positioned on the weight plates (typically vertically, along the y-axis). A shift between the two receivers (*202*, *204*) in the other axis (typically x) allows one to decrease the distance on the axis of movement, between the two

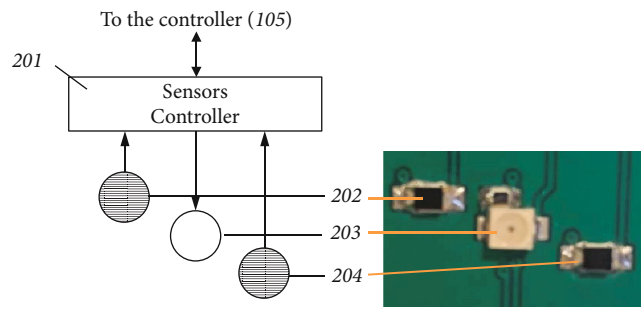


Figure 2: A combined emitter/receiver pair—the scheme and a real-life example.

receivers (*202*, *204*), and thus allows one to decrease the size of the markers. The receivers (*202*, *204*) are configured to register the reflected light emitted by the emitter (*203*). Additional electrical or digital components such as transistors and resistors required by the emitter (*203*) and the receivers (*202*, *204*) can be integrated in the form of a sensor controller (*201*) being controlled by and reporting to the main controller (*105*).

Now that the sensor arrangement has been presented, their method of operation is to be described in detail. The sensors, as well as their appropriate placement, allow for a proper detection of the weight plates themselves as well as of their direction of movement.

In principle, in order to detect the direction of movement, only one sensor with a single phototransistor should be sufficient—when the weight plate moves upwards, it is the A reflector that will be detected first (reflecting more light than the B reflector); in the case of the downward movement, the situation is reversed: the B reflector is first detected (it reflects less light than the A reflector). The authors, however, have found that in practice obtaining reliable reading is more involved. Data from phototransistor receivers (*103*, *202*, and *204*) are typically in the form of a stream of numerical values (after being converted by an analog-to-digital converter, which is preferably a part of the controller (*105*)). In order

to detect whether a given phototransistor receiver (*103*, *202*, and *204*) detects a relevant reflector (A or B), a range of values for the detection should be established experimentally for each of the markers. For example, for a given type A reflector, a range of 3500 to 4095 may be assumed, while for some other type B reflector, a range of 1300 (black color and colors in proximity to black are assumed to be below this value) to 3500 may be assumed. The provided values apply to a 12-bit analog-to-digital converter. Thus, when a marker is moved upwards on a weight plate, a type A reflector approaches a phototransistor receiver (*103*, *202*, and *204*) and thus the received numerical values gradually increase (more and more light is reflected in comparison to the dark surface of the weight plate) and cross the threshold between the type A and type B reflector ranges (as defined above), since finally the reading will reach a certain value, for example 4000, whereas the readings began at some other lower value, say 1300. As a result, one may obtain a false report of the B reflector to be the first even though, in the case of an upward movement, it does not precede the A reflector. This potential false detection is corrected with the use of the second phototransistor receiver (*202*, *204*). This has the consequence that the size of the respective type B (*111B*) and A type reflectors (*111A*) should be at least equal to the rectangle defined by the vertices located where the two receivers (*202*, *204*) are. In a different potential configuration, the size of the respective type B reflectors (*111B*) should be at least equal to the rectangle defined by the vertices at the positions of the two receivers (*202*, *204*), while the size of the respective type A reflector (*111A*) should be at least equal to 1/3 of that rectangle. It can potentially be smaller owing to higher reflective properties of the type A reflectors (*111A*), which may already be detected in proximity to the receiver. In practice however, it is advantageous to set the size of the reflectors as larger than the rectangle. Data from the two phototransistor receivers (*202*, *204*) are read simultaneously, and when both return values falling within a given range for a given reflector, it is determined that a given reflector (*111A*, *111B*) is detected. Otherwise, the data can be ignored. Since the movement of the weight stack is executed along the *y*-axis, the placement of the phototransistor receivers (*202*, *204*) must also occur along the same axis. An additional range that is taken into account while detecting a marker is the one defining the weight plate without a marker. Typically, this range falls between 0 and 650, since the weight plates are usually painted with a black or other dark-hued paint (obviously, other values for such a range are also possible).

With the aforementioned three ranges being defined, one can now define the states for phototransistor receivers (*202*, *204*) reading the following:

(i) UNKNOWN: data from the two phototransistor receivers (*202*, *204*) do not fall into the same range (e.g., the first phototransistor returns 2000 while the second phototransistor returns 400). Such a state is discarded

(ii) NONE: data from the two phototransistor receivers (*202*, *204*) fall into the first range, for example, 0–650 (i.e., a weight plate without a marker is detected—an assumption made for this range)

(iii) REFLECTOR_B: data from the two phototransistor receivers (*202*, *204*) fall into the second range 1300–3500 (i.e., a weight plate with reflector B is detected)

(iv) REFLECTOR_A: data from the two phototransistor receivers (*202*, *204*) fall into the third range 3500–4095 (i.e., a weight plate with reflector A is detected)

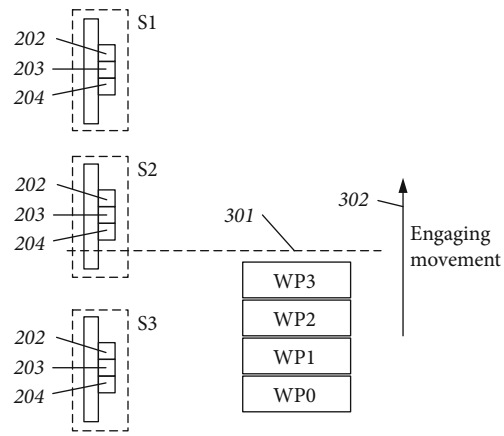


Figure 3: Sensor positioning on a weight stack.

Then, in two separate steps, the direction of a moving weight plate is detected. This is done by means of the following changes of the states (previous state→current state):

Upward movement:
NONE→REFLECTOR_A or
REFLECTOR_A→REFLECTOR_B or
REFLECTOR_B→NONE
Downward movement:
NONE→REFLECTOR_B or
REFLECTOR_B→REFLECTOR_A or
REFLECTOR_A→NONE

The presence of a weight plate is confirmed only when the following changes of states occur:
REFLECTOR_A→REFLECTOR_B or
REFLECTOR_B→REFLECTOR_A

This greatly simplifies the internal operations of the system in relation to the presented solution.

Having all the above information, the system is now able to detect (separately on each of the sensors, as described in Figure 2) how many weight plates have been moved in a given direction (each of the sensors may have its own counter of weight plates that are present above its location—it is a difference between the weight plate count detected during an upward movement and the weight plate count detected during a downward movement). This is crucial, since it means that there is some leeway in terms of the locations of the sensors.

The remainder of this section presents a general principle of weight counting and repetition counting.

The preferred configuration, as shown in Figure 3, comprises three sensors (S1, S2, and S3) presented in Figure 2.

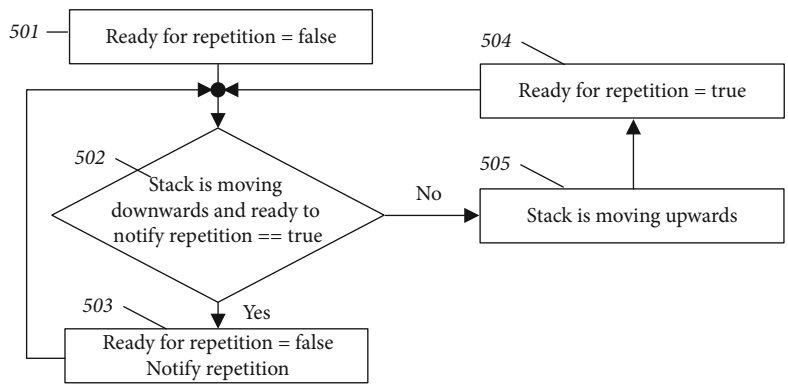


Figure 4: The process diagram of repetition counting.

Each sensor (S1, S2, and S3) is initially positioned so that it reports the NONE state.

The distances between the sensors (S1, S2, and S3) are set during the mounting and setup and are fixed for a given weight stack device. Nevertheless, these distances are preferably a multiple of the height of a single weight plate (WP0–WP3). For example, four weight plates' distance equals 10 cm for a weight plate having the height of 2.5 cm. In principle, the bottom sensor (S3) is responsible for counting weights, while the top sensor (S1) is responsible for counting repetitions. The middle sensor (S2) is positioned just above (level *301*) the top weight plate of the weight stack at rest.

In order to simplify the explanation, we have chosen to focus on a description of a weightlifting machine, where the weight stack travels vertically. Obviously, the presented solution can also be applied in the cases of machines that move weights in different directions, for example, horizontally (weight pulling/pushing). In such cases, the top sensor becomes the left/right sensor and the bottom sensor will be positioned on the opposite side.

In view of the above, one needs to differentiate between the engaging movement (vector (*302*) in Figure 3), which requires the application of an external force (e.g., weightlifting by the user), and a return movement, which typically does not require the external force and is affected by means of gravity. The following presents an alternative naming scheme for the sensors:

(i) The middle sensor (S2): the first sensor to detect the front (i.e., first) weight plate, in the initial state, when considering the direction of the engaging movement, e.g., the top weight plate in the case of vertical movement (WP3 in Figure 3), the leftmost weight plate in the case of horizontal movement towards the left side, or the rightmost weight plate in the case of horizontal movement towards the right side

(ii) The top sensor (S1): the sensor(s) following the middle sensor (S2) when considering the direction of the engaging movement

(iii) The bottom sensor (S3): the sensor(s) preceding the middle sensor (S2) when considering the direction of the engaging movement

As will be evident from the following specification, the system is functional with only one top sensor (S1), only one middle sensor (S2), and at least one bottom sensor (S3) present in the group of bottom sensors.

Repetition counting is executed by having a global state denoting the movement of the weight stack. Such a state can be updated by each sensor except for the bottom sensor (S3). In practice however, only the top sensor (S1) updates the state, when it has detected at least one marker on a weight plate. When the top sensor (S1) has detected a change to the UPWARDS state and subsequently detects a change to the DOWNWARD state, the system (i.e., the controller (*105*)) signals a single repetition. The above applies to a simple configuration involving 3 sensors, and in the case of more sensors, the highest placed sensor that was reached by a marker present on the top weight plate will trigger an update of the repetition counter. The details of repetition counting, in view of the general principles above, is presented in Figure 4.

The weight is calculated using various methods depending on the number of the weight plates. First, in order to properly detect the number of the lifted weight plates (WP0, WP3), the system requires information from two sensors. In the case when less than four weight plates are lifted (i.e., with reference to the distance multiplication factor of 4 referred to earlier), the total weight is detected based on the data from the sensors (S1, S2). In this particular scenario, it must be that the bottom sensor (S3) has not detected any marker, the middle sensor (S2) has detected fewer than 5 markers, and the top sensor (S1) has detected at least one marker.

In the above case, the number of the markers detected by the middle sensor (S2) during the UPWARD movement is assumed to represent the weight of the series (after multiplication by a weight of a single weight plate) when the weight stack returns to its initial position (i.e., when the number of weight plates above the middle sensor (S2) is equal to zero) for a predefined minimum time threshold. The predefined minimum time threshold (postrepetition rest time) can be set during the setup phase and be assigned to a specific weight stack machine, or it can be set by the user in an accompanying software application (the latter preferably takes precedence). The value of the time threshold can be set to, e.g., 500 ms or 1000 ms.

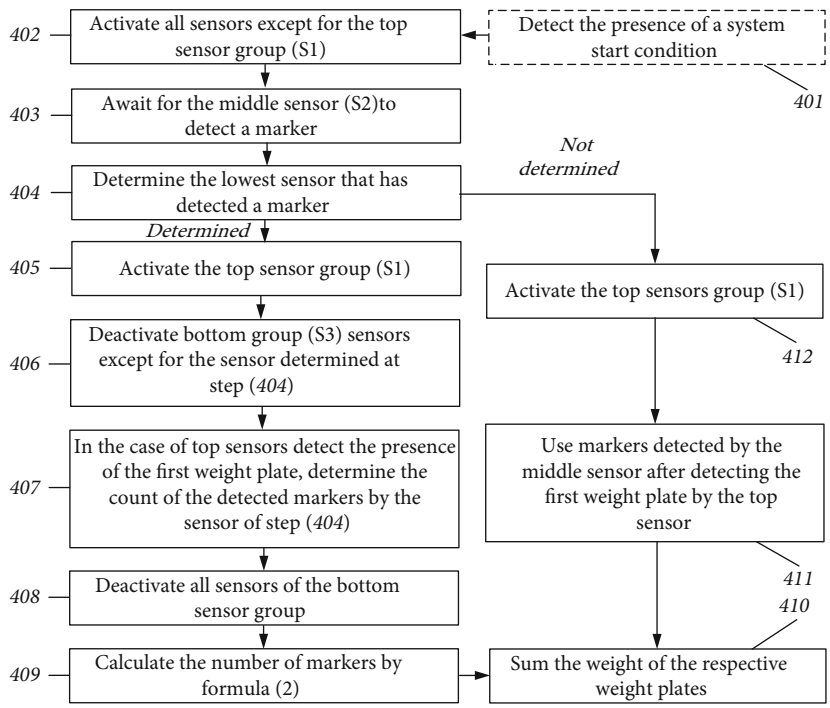


Figure 5: The process diagram of weight counting.

For example, in the case where more than four weight plates are lifted, the bottom sensor (S3) also reports state changes and hence the count of the markers detected by the bottom sensor (S3) can be established. In this case, in order to correctly detect the number of weight plates in a given repetition, a further condition must be met: the middle sensor (S2) must detect at least four markers during the UPWARD movement. In such a case, it must be that the number of markers detected by the bottom sensor (S3) does not change. In that scenario, the total weight is calculated using the number of markers detected by the bottom sensor (S3) together with the four weight plates that are above it, which is multiplied by the weight of a single weight plate.

A detailed schema describing the algorithm for weight counting that uses the above-described methods is presented in Figure 5.

The process of detecting the weight for a greater number of weight plates relies on the same principles. To this end, the arrangement of the sensors (S1, S2, and S3) in a sequence is fixed and the sensors are all spaced equally from one another. Nevertheless, for specific weight stack machines, the distances among groups of sensors can differ, for example, in the case when a given exercise requires lifting a set of weight plates to a predefined initial level and continuing with the exercise with the weights there.

In another possible configuration, at least one top sensor (S1) must be present in a group of top sensors. This allows a more precise tracing of a weight stack. Naturally, this configuration can also be combined with the configuration involving a group of bottom sensors (S3). For example, assuming the spacing of 4 weight plates between sensors, a weight stack of 12 weight plates will require 5 sensors (3 sensors in the bottom sensor group) and a weight stack of 20 weight plates will typically require 7 sensors (5 sensors in the bottom sensor group). Similarly, assuming the spacing of 5 weight plates between sensors, a weight stack of 10 weight plates will require 4 sensors (2 sensors in the bottom sensor group) and a weight stack of 20 weight plates will require 6 sensors (4 sensors in the bottom sensor group). The spacing of the plates influences the way, in which the minimal movement required for the repetition to be valid is detected. Therefore, in principle, the more sensors the greater the possible range of movement (a greater number of sensors in the top group (S1)). Furthermore, an increased number of sensors allows for the detection of more weight plates or greater accuracy of repetition detection owing to the decreased spacing between the successive sensors.

Figure 5 presents the process diagram of weight counting. The first, optional, step (*401*) is to detect the presence of a system start condition, for example, using the reed switch (*106*).

Subsequently, at step (*402*), the system powers on (activates in general) all sensors except for the top sensor (S1). Next, at step (*403*), the system awaits for the middle sensor (S2) to detect a marker (*111*). After that, the system can identify all the sensors that have detected a marker. It is also determined which of these is positioned the lowest (*404*). This can be used to deactivate all the bottom group (S3) sensors (*406*) except for the sensor determined at step (*404*). This step is aimed at saving energy use and is the presented configuration. Nevertheless, the deactivation of the sensors may also be omitted.

Subsequently, after activating the top sensor group (*405*), in the case a top sensor (S1) detects the presence of the first (top) weight plate (WP3), the number (*407*) of markers (*111*) detected is determined by the sensor identified in step

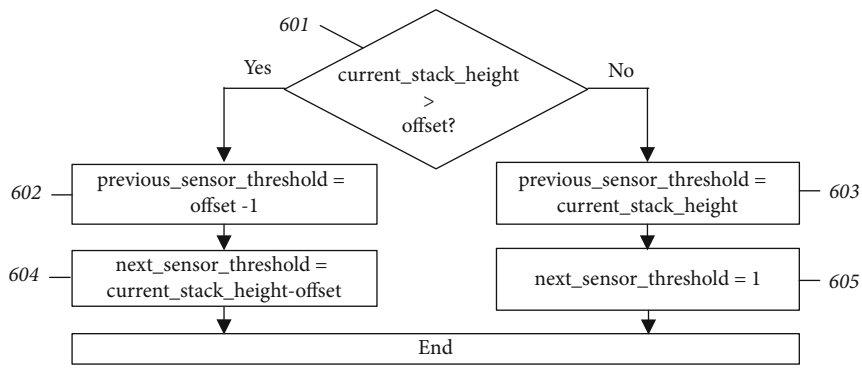


Figure 6: The process diagram of setting threshold values responsible for determining which sensors to switch on or off.

(*404*). The presence of the plate is detected when the pass from the REFLECTOR_A state to the REFLECTOR_B state is realized, as described above.

Next, at step (*408*), all sensors of the bottom sensor group (S3) are deactivated. Subsequently, at step (*409*), a number of markers (weight plates) are calculated using the following formula:

$$\text{sensors\_index} * \text{offset} + \text{count\_of\_step\_407}, \quad (2)$$

where offset is the number of weight plates between directly neighboring sensors (four in the example of Figure 3) and sensors_index is the number of sensors between the middle sensor (S2) and the sensor of step (*404*). This value needs to be multiplied by the weight of a single weight plate (*410*), or in more sophisticated systems, where weight plates have different weights, it may be a sum of the weight of all the weight plates starting from the back/last (WP0) up to the weight plate indicated by the result of step (*409*). The calculated weight is kept until the end of the series (i.e., when the middle sensor (S2) detects as many upward moving markers as the downward moving ones and the predefined postrepetition rest timeout elapses). In the case step (*404*) returns an empty result, the system can take into account the number determined in step (*411*), i.e., the number of markers detected by the middle sensor (S2) after detecting the first weight plate (WP3) by the top sensor (S1). The bottom sensors (S3) can be deactivated in such a case.

Figure 4 shows the process diagram of repetition counting. The default value of the "Ready to notify repetition" flag is set to false (*501*). Its purpose is to notify that a change from an upward movement to a downward movement is detected. Next, at step (*502*), the direction of the movement of the weight stack (as previously defined with respect to the upward movement and the downward movement) is determined.

The existence and use of the "Ready to notify repetition" flag are required in the case of systems having a number of repetition detection sensors (i.e., the top sensor group (S1) within which a repetition detection sensor is selectively identified). A repetition is detected on a sensor defined as a repetition_detection_sensor; the selection of which is described in the following paragraphs. This process is preferably executed as a separate processing thread, wherein for each sensor, there is data regarding the direction of the movement of the weight stack. If this direction differs from the currently reported one, a notification is triggered, which is processed according to the rules set out in the method described in Figure 4.

Figure 6 shows the process diagram of setting threshold values responsible for determining which sensors to switch on or off.

The top sensor (S1) or a group of top sensors in general is used to detect repetitions. During this process, there are sensors switched on and off depending on their use in the repetition counting. Alternatively, all sensors may be permanently active but at the expense of higher energy use, which is not beneficial in the case of battery-powered systems. In general, when the vertical weight stack moves upwards, a higher sensor is switched on while when the vertical weight stack moves downwards, a lower sensor is switched on. A decision on which sensor (*102*, *103*) to switch on is made based on the number of weight plates that have been counted by the active sensor. In order to make that decision, two thresholds are used: previous_sensor_threshold and next_sensor_threshold.

In addition to these thresholds, there are the following variables defined:

(i) current_stack_height: representing a current stack height determined during weight counting and expressed preferably in a number of weight plates (e.g., 5 weight plates)

(ii) offset: the distance between neighbouring sensors, expressed in a number of weight plates (e.g., 4 weight plates)

(iii) repetition_detection_sensor: an indicator of the sensor currently detecting a change of the direction of the weight stack movement (it is the currently active sensor, which initially is the lowest sensor of the top sensor group (S1). Typically, the sensors are numbered with increasing indicator values depending on the height at which a given sensor is located: for example, 0, 1, 2, and 3 or -2, -1, 0, 1, and 2 in order to clearly indicate the middle sensor as 0)

The two thresholds, previous_sensor_threshold and next_sensor_threshold, defining when to switch on a sensor

below (preceding) the repetition_detection_sensor (next_sensor_threshold) or above (successive) the sensor indicated by the repetition_detection_sensor (previous_sensor_threshold) are calculated as defined in Figures 7(a) and 7(b). The process described in Figure 6 is executed only once: after the weight is determined. The process starts at step (*601*) by verifying whether the current_stack_height variable is greater than the offset variable. If it is, at step (*602*), the process sets the previous_sensor_threshold to the offset-1 value, which is followed by setting the next_sensor_threshold to current_stack_height-offset (*604*). Otherwise, when the current_stack_height variable is equal or lower than the offset variable, at step (*603*), the previous_sensor_threshold is set to current_stack_height and at step (*605*), the threshold next_sensor_threshold is set to 1.

The process of switching the sensors (during the movement of the weight stack) is shown in Figures 7(a) and 7(b).

Figure 7(a) shows the sensor-switching process when the current_stack_height > 1. At step (*701*), an update of the repetition detection sensor takes place, ensuring that all the information is up to date and is taken from the repetition detection sensor that is currently active.

Next, at step (*702*), the system checks whether the number of weight plates counted by the repetition detection sensor is greater than or equal to previous_sensor_threshold. If that is the case, data are obtained from the successive (above) sensor (*703*). Subsequently, if the weight plates counted by the sensor above are not equal to zero (*704*), at step (*705*), the indicator of the current sensor is increased: repetition_detection_sensor = repetition_detection_sensor + 1. This means that the upper (successive) sensor will be switched on (the weight stack is moving upwards). The current sensor can be switched off depending on the respective threshold values.

If the condition of step (*702*) is not met, at step (*706*), the system verifies whether the number of weight plates, as counted by the repetition detection sensor, is less than or equal to next_sensor_threshold. In that case, at step (*707*), data are obtained from the preceding sensor. Next, if the weight plate count, for plates detected by the current repetition detection sensor, is equal to zero (*708*), the indicator of the current sensor is decreased: repetition_detection_sensor = repetition_detection_sensor − 1 (*709*). This means that the lower (preceding) sensor will be switched on (the weight stack is moving downwards). The current sensor may remain switched on (*705*), depending on the respective threshold values, in the case of a rapid return of the weight stack.

If the current_stack_height is equal to 1, the process shown in Figure 7(b) is executed. At step (*710*), an update on the repetition detection sensor takes place, ensuring that all the information is up to date and is taken from the repetition detection sensor that is currently active. Next, at step (*711*), the direction of the stack is detected. If the stack is moving upwards, data are taken from the successive (above) sensor (*712*). Next, if the weight plate count, as detected by the successive sensor, is greater than zero (*713*), at step (*714*), the counter related to the current sensor is increased: repetition_detection_sensor = repetition_detection_sensor + 1. If the condition of step (*711*) is not met, at step (*715*), the system checks whether the weight stack is moving downwards. In that case, data are obtained from the preceding sensor, at step (*716*). Next, if the weight plate count, as detected by the preceding sensor, is equal to zero (*717*), the indicator of the current sensor is decreased: repetition_detection_sensor = repetition_detection_sensor − 1 (*718*).

The requirement of having two thresholds, i.e., the next_sensor_threshold and the previous_sensor_threshold, is related to the fact that when counting the weight plates using a sensor, the number of weight plates increases (0, 1, 2, and 3) during the upward movement and decreases (3, 2, and 1) during the downward movement. Therefore, in order to maintain symmetry between the switch-on and switch-off phases for the respective sensors, one requires, both during an upward and during a downward movement, two counters. Now, the sensor determined at step (*404*) in the weight detection algorithm can be switched on. When the middle sensor (S2) and the step (*404*) sensor reach the value of the weight plate count equal to 0 (the number of weight plates counted by the given sensor decreases during the downward movement), the system assumes its initial state. Alternatively, the switching on of the (*404*) sensor may be omitted, conserving energy.

The mentioned switching on and off of the respective sensors during the repetition counting operate independently of the switching on and off of the sensors during the weight counting. These processes are separated in the sense that the weight counting is executed first (having its own rules of switching on and off the sensors as indicated in Figure 5) and when the weight is determined, the repetition counting algorithm is initiated (cf. Figure 4) (the algorithm introduces its own rules of switching on and off the respective sensors—as indicated in Figures 7(a) and 7(b)).

On the basis of the above, two additional functions can also be implemented, i.e., calculating the time of a repetition as well as the height of a weight lift. The time of a repetition may be registered by calculating the time elapsed from the moment of detection of the first/top weight plate by the middle sensor (S2) until the middle sensor's weight plate count (S2) value equals 0. In practice, this can be considered the time between steps (*504*) and (*502*). The height of a weight lift can be determined using the known height of the highest sensor to detect a weight plate together with the height of the weight plates detected using that particular sensor (the height of the weight plates is known as a configuration parameter and typically all the weight plates have the same height even though they can have different weights).

Figures 8(a)–8(f) show the system states during the movement of a weight stack.

Figure 8(a) presents 5 sensors, with S2 being the middle sensor, S1-2 and S1-1 belonging to the top sensor group (S1), and S3-1 and S3-2 belonging to the bottom sensor group (S3). The sensors are located in a distance of 3 weight plates from each other as specified by the offset variable. The other variables can remain in their default (unspecified) states, such as null. There are 8 weight plates (WP0–WP7) wherein in its initial state, the system activates sensors S3-2, S3-1, and S2.

In Figure 8(b), the weight stack moved upwards by 4 weight plates, which results in the S1-1 sensor being switched

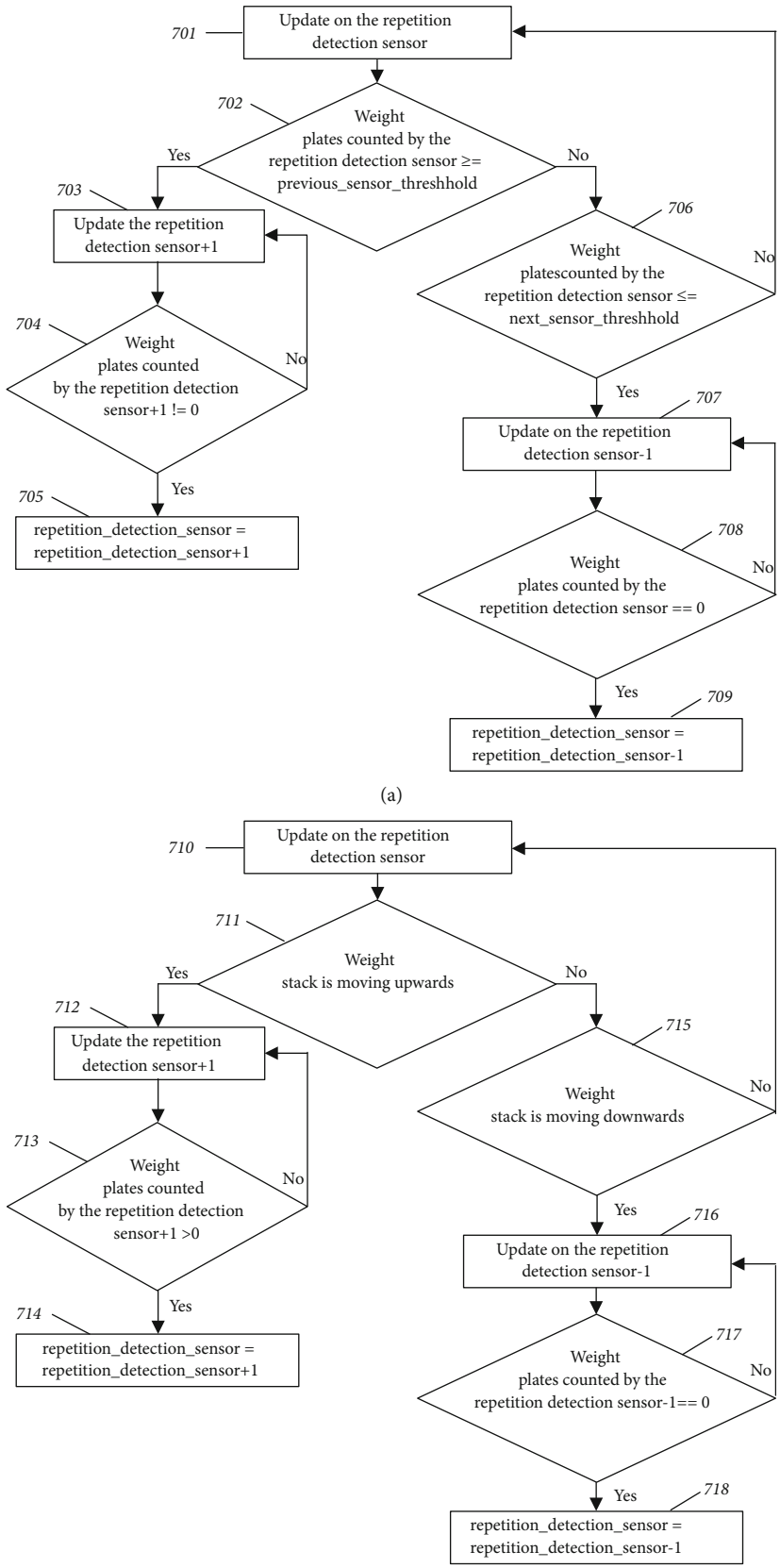
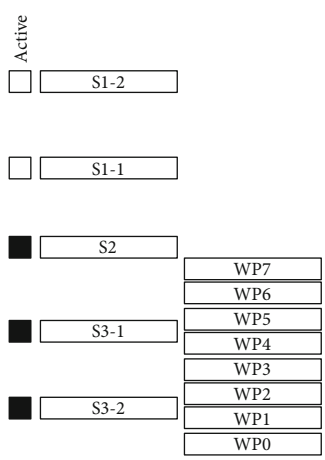
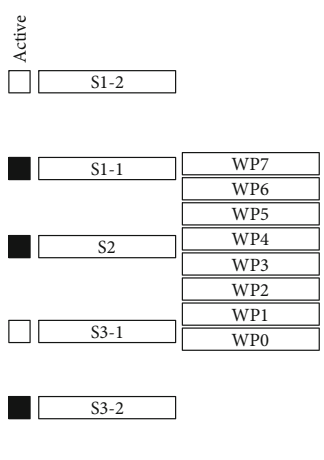


Figure 7: (a) The process diagram of switching the sensors during movement of the weight stack when the current_stack_height > 1. (b) A process diagram of switching the sensors during movement of the weight stack when the current_stack_height = = 1.



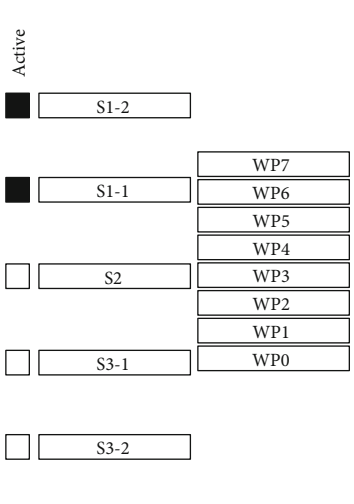
Process variable	Value
Step (404) sensor	NA
(sensors_index)*offset+count_of_step_407	NA
current_stack_height	NA
offset	3
repetition_detection_sensor	NA
next_sensor_threshold	NA
previous_sensor_threshold	NA

(a)



Process variable	Value
Step *(404)* sensor	S3-2
(sensors_index)*offset+count_of_step_407	8
current_stack_height	8
offset	3
repetition_detection_sensor	S1-1
next_sensor_threshold	5
previous_sensor_threshold	2

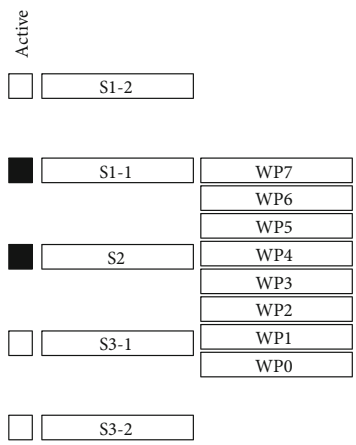
(b)



Process variable	Value
Step (*404*) sensor	S3-2
(sensors_index)*offset+count_of_step_407	8
current_stack_height	8
offset	3
repetition_detection_sensor	S1-1
next_sensor_threshold	5
previous_sensor_threshold	2

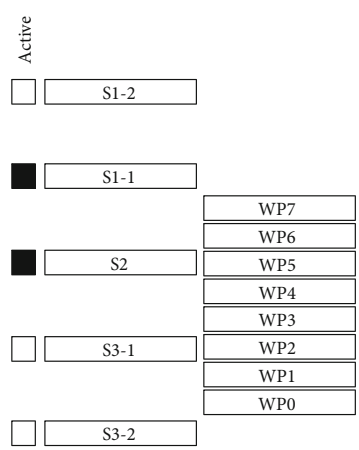
(c)

Figure 8: Continued.



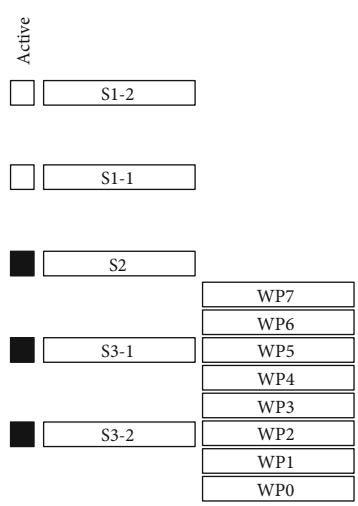
Process variable	Value
Step (*404*) sensor	S3-2
(sensors_index)*offset+count_of_step_407	8
current_stack_height	8
offset	3
repetition_detection_sensor	S2
next_sensor_threshold	5
previous_sensor_threshold	2

(d)



Process variable	Value
Step (*404*) sensor	S3-2
(sensors_index)*offset+count_of_step_407	8
current_stack_height	8
offset	3
repetition_detection_sensor	S2
next_sensor_threshold	5
previous_sensor_threshold	2

(e)



Process variable	Value
Step (*404*) sensor	NA
(sensors_index)*offset+count_of_step_407	NA
current_stack_height	NA
offset	3
repetition_detection_sensor	NA
next_sensor_threshold	NA
previous_sensor_threshold	NA

(f)

Figure 8: (a) The first state during the movement of a weight stack. (b) The second state during the movement of a weight stack. (c) The third state during the movement of a weight stack. (d) The fourth state during the movement of the weight stack. (e) The fifth state during the movement of a weight stack. (f) The sixth state during the movement of a weight stack.

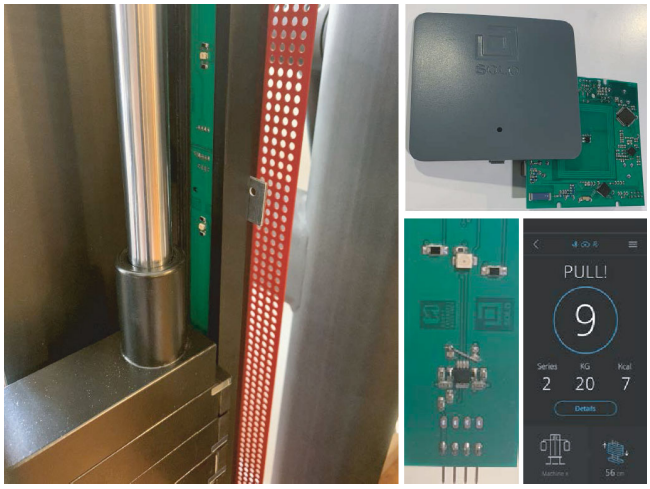


Figure 9: A working example of the proposed system.

on (because it was the next sensor to detect an upward moving weight stack and it is additionally the sensor responsible for weight detection (the top sensor of step (*407*))) and the sensor S3-1 being switched off. The S2 sensor detected movement, and S1-1 awaits a marker. The remaining variables are set as follows:

(i) The step (*404*) sensor = S3-2

(ii) (sensors_index) ∗ offset + count_of_step_406 = 8

(iii) current_stack_height = 8

(iv) repetition_detection_sensor = S1 − 1

(v) next_sensor threshold = 3

(vi) previous_sensor threshold = 2

In Figure 8(c), the weight stack moved upwards by 1 weight plate, which results in the S3-2 sensor being switched off with no changes to the listed variables. The lifted weight was detected and the (*702*) condition is met, resulting in the switching on of the S1-2 sensor (*703*).

Figure 8(d) corresponds to Figure 8(b), but since now the downward movement was detected, the sensor S3-2 is not switched (compare Figure 8(b)). The repetition_detection_sensor is now S2, as the condition of step (*702*) is not met for S1-1 and steps (*706*) and (*708*) are met.

In Figure 8(e), the repetition_detection_sensor is set to S2, since it is the highest sensor detecting weight plates in this downward movement. For S2, the condition of step (*702*) is met; therefore, S1-1 remains active.

Lastly, in Figure 8(f), the system returns to its initial state; i.e., the number of weight plates counted by the middle sensor S2 equals zero.

4. Experimental Tests

In order to show the effectiveness of the developed approach, the prototype of the system has been constructed and added to a weightlifting machine that was already in use rather than a brand new one. The electronic components of the system were attached to the machine frame and reflective markers to the weight plates in the way presented in Figure 9. The system was connected to the smartphone equipped with a mobile application in order to observe whether repetition was correctly detected and what was the weight of the stack. The user performed repetitions with random acceleration, raising the stack to different heights, changing the weight every few repetitions. By comparing the real observations with the data displayed on the smartphone screen, it was found that the system works with 100% efficiency (all repetitions and weights were detected correctly). Moreover, the same prototype was presented at the world's biggest trade show for fitness (FIBO'19) and was thoroughly tested by many visitors [16]. In the authors' opinion, the system is ready for production.

5. Summary

The article described a system assisting in counting both the number of repetitions and the weight of the weight plates lifted during an exercise session using a weight machine. It has been observed by the authors that most users of such machines usually write down the repetitions and weights lifted in their training logs. The authors proposed a system consisting of a number of sensors and controllers to aid in the automation of this task.

The above-described system consists of two main methods: one counting the repetitions and the other determining the weight of the lifted plates. Each method is related to a bespoke algorithm used for determining the relevant parameters. The described methods have been experimentally verified using a working example of such a system, as described in Figure 9. Both were shown to operate as expected.

Currently, the authors are working on a similar solution that will be used in dumbbells and barbells. Developing such a system is much more difficult, because exercising with dumbbells may be performed in a variety of ways. Therefore, the neural network (i.e., implemented in the toolbox [17, 18]) will be used in order to determine the type of motion.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The work was supported by the National Centre for Research and Development in Poland under grant number POIR.01.01.01-00-1324/17.

References

[1] K. Monisha and M. Rajasekhara Babu, "A novel framework for healthcare monitoring system through cyber-physical system,"

in *Internet of Things and Personalized Healthcare Systems*, pp. 21–36, Springer, 2019.

[2] T. T. Tran, J. W. Choi, C. Van Dang, G. SuPark, J. Y. Baek, and J. W. Kim, "Recommender system with artificial intelligence for fitness assistance system," in *2018 15th International Conference on Ubiquitous Robots (UR)*, pp. 489–492, Honolulu, HI, USA, June 2018.

[3] C. Shen, B. J. Ho, and M. Srivastava, "Milift: efficient smartwatch-based workout tracking using automatic segmentation," *IEEE Transactions on Mobile Computing*, vol. 17, no. 7, pp. 1609–1622, 2018.

[4] F. Caputo, E. D'Amato, A. Greco, I. Notaro, and S. Spada, "Human posture tracking system for industrial process design and assessment," in *International Conference on Intelligent Human Systems Integration*, pp. 450–455, Springer, 2018.

[5] M. Kranz, A. Möller, N. Hammerla et al., "The mobile fitness coach: towards individualized skill assessment using personalized mobile devices," *Pervasive and Mobile Computing*, vol. 9, no. 2, pp. 203–215, 2013.

[6] K. Takata, J. Ma, B. O. Apduhan, R. Huang, and Q. Jin, "Modeling and analyzing individual's daily activities using lifelog," in *2008 International Conference on Embedded Software and Systems*, pp. 503–510, Sichuan, China, July 2008.

[7] K. Takata, M. Tanaka, J. Ma, R. Huang, B. O. Apduhan, and N. Shiratori, "A wearable system for outdoor running workout state recognition and course provision," in *International Conference on Autonomic and Trusted Computing*, pp. 385–394, Springer, 2007.

[8] F. Wiehr, F. Kosmalla, F. Daiber, and A. Krüger, "Footstriker: an ems-based assistance system for real-time running style correction," in *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '17*, p. 56, Vienna, Austria, September 2017.

[9] C. Balsalobre-Fernández, D. Marchante, E. Baz-Valle, I. Alonso-Molero, S. L. Jiménez, and M. Muñóz-López, "Analysis of wearable and smartphone-based technologies for the measurement of barbell velocity in different resistance training exercises," *Frontiers in Physiology*, vol. 8, p. 649, 2017.

[10] A. Chatzitofis, N. Vretos, D. Zarpalas, and P. Daras, "Three-dimensional monitoring of weightlifting for computer assisted training," in *Proceedings of the Virtual Reality International Conference on Laval Virtual - VRIC '13*, Laval, France, March 2013.

[11] J. Somers, "Excercise monitoring system, 2009," US Patent 8062182B2.

[12] A. Menektchiev, R. Kohan, and E. Minzenberger, "Sensor arrays for exercise equipment and methods to operate the same, 2006," US Patent 20070213183A1.

[13] J. J. Chou, C. S. Shih, W. D. Wang, and K. C. Huang, "IoT sensing networks for gait velocity measurement," *International Journal of Applied Mathematics and Computer Science*, vol. 29, no. 2, pp. 245–259, 2019.

[14] Y. Ngoko, C. Cerin, and D. Trystram, "Solving SAT in a distributed cloud: a portfolio approach," *International Journal of Applied Mathematics and Computer Science*, vol. 29, no. 2, pp. 261–274, 2019.

[15] P. Jankowski-Mihułowicz, M. Węglarski, G. Pitera, D. Kawalec, and W. Lichoń, "Development board of the autonomous semi-passive RFID transponder," *Bulletin of the Polish Academy of Sciences: Technical Sciences*, vol. 64, no. 3, pp. 647–654, 2016.

[16] B. Penkert, "FIBO 2019: these sports trends will soon conquer the gyms," in Sports TechX, 2019, https://tinyurl.com/y3uduonu.

[17] M. Luzar and A. Czajkowski, "SSNN toolbox for non-linear system identification," *Journal of Physics Conference Series*, vol. 659, 2015.

[18] M. Luzar and A. Czajkowski, "LPV system modeling with SSNN toolbox," in *2016 American Control Conference (ACC)*, pp. 3952–3957, Boston, MA, USA, July 2016.