Research on HMM-based Exoskeleton Robot Falling Prediction Algorithm

Leilei Gao1,*, Keyong Deng1, a, Rui Zhang1, b, Meng Wang1, c and Hao Wang1, d
1School of Automation, Wuhan University of Technology, Wuhan 430070, China
*jzhli@whut.edu.cn; a dengkeyong@ctesi.com.cn; b 1025427063@qq.com;
c 1624078671@qq.com; d3297681786@qq.com

Abstract: Exoskeleton robot is essentially a wearable robot. In recent years, the research of exoskeleton robots has become a new hot spot, and has gradually been widely used in military, medical and civilian fields. However, the balance and safety issues of power-assisted exoskeleton robots need to be solved urgently. At present, although there are many researches on exoskeleton robots at home and abroad, the research on the falling problem of assisted lower limb exoskeleton robots is not in-depth. Existing studies on falls are mainly focused on the detection of falls, while the prediction of falls is relatively small, and these studies rely more on a single sensor, and its accuracy needs to be improved. In response to the status quo, this paper proposes a fall prediction algorithm based on HMM to study the fall behaviour of the exoskeleton human-machine system in advance, it provides a reference for the anti-fall measures of the exoskeleton robot and aims to effectively reduce the damage of the human-machine system.

1. Introduction
Since the 21st century, with the development of high-performance microprocessors, motors and a variety of high-precision sensors, the research on exoskeleton robots has gradually matured. As a wearable robot, exoskeleton robots have achieved unprecedented development in the military, medical and civilian field [1]. The term exoskeleton comes from biology, and with the development of technology, this concept has been applied to the field of robotics, and the integration of mechanical structure, information-coupling and automatic control technology has produced this kind of mechanical equipment worn outside the human body [2]. The main function of exoskeleton robots is to improve human exercise endurance and upper limit of weight, or to help the disabled and the elderly to walk [3].

Currently, exoskeleton robots have many applications in various fields, but there are still many problems to be solved on the road to commercialization. Among them, the balance and safety of exoskeleton robots are a difficult point. Based on this, this project mainly studies the fall methods and characteristics of the assisted lower extremity exoskeleton robot. By collecting data from multiple types of sensors on the exoskeleton robot, different fall behaviours are predicted and experimentally verified.

2. Mechanism analysis of man-machine system falling
Before analysing the fall process, the human-machine system fall needs to be defined. Here, the exoskeleton man-machine system fall is defined as: unintentionally falling on the ground or other lower positions due to terrain or the man-machine system itself [4]. The fall process of the human-machine system is generally divided into an initial state, an unbalanced state, an obstacle impact state, and a relatively stable state after falling to the ground. The various processes of falling behaviour are carried
out according to time. During the period, the more characteristic ones should be the changes in the orientation of the human-machine system and the drastic changes caused by collisions with low-power objects. When predicting the fall of the human-machine system, it is necessary to grasp the characteristics of the exoskeleton human-machine system before the first impact.

To study the kinematic characteristics of the fall behaviour of the exoskeleton human-machine system, the coordinates of each part need to be defined first [5]. The front of the exoskeleton robot is set to the positive direction of the x-axis, the left side is set to the positive direction of the y-axis, and the vertical upward direction is set to the positive direction of the z-axis, as shown in figure 1.

![Figure 1. Schematic diagram of the exoskeleton robot coordinate system](image)

The force changes during the movement of the human-machine system can be reflected by acceleration. Due to the different positions of its own motion or collision with surrounding objects, the force and acceleration changes of different body parts may be different. The force changes during the movement of the human-machine system can be reflected by acceleration. Due to the different positions of its own motion or collision with surrounding objects, the force and acceleration changes of different body parts may be different. Select the part that is more concentrated and stable in the impact force during different falls, then the change process of acceleration of this part can be used to describe the fall process. For example, select the acceleration of the waist and define the acceleration along the x axis as $a_x$. Similarly, we get the acceleration of the waist as $\vec{a} = (a_x, a_y, a_z)$, and the acceleration is:

$$a = \left( a_x^2 + a_y^2 + a_z^2 \right)^{1/2} \quad (1)$$

3. Design of man-machine system falling process

3.1. Hidden Markov model

Hidden Markov Model (HMM) is a statistical analysis model. Founded in the 1970s [6], after decades of development, it has been successfully used in speech recognition, behaviour recognition, text recognition and fault diagnosis.

Hidden Markov model describes the process of generating a random sequence of unobservable states randomly from a hidden Markov chain, and then generating an observable random sequence from each state. Any HMM can be described by a formula:

$$\lambda = (M, N, \pi, A, B) \quad (2)$$

Among them, $M$: the state number of the Markov process in HMM; $N$: the number of possible observations corresponding to the motion state; $\pi$: initial state probability vector; $A$: the state transition matrix of the hidden Markov process, $A = \{a_{ij}\}_{M \times N}$; $B$: observation probability matrix, $B = \{b_j(k)\}_{M \times N}$.

Next, how to apply HMM to the fall prediction of the human-machine system? First, we need to build a fall process model of the man-machine system. Secondly, the data collected by the human-machine system acceleration sensor is fused and extracted as the observation sequence of the model, and the model parameters are estimated through the learning problem. Finally, through model matching,
the acquired acceleration time series belong to the model probability of the fall process. The following will introduce the method of solving the two problems of parameter estimation and probability estimation in the prediction problem.

3.2. Forward algorithm
The forward algorithm will first find the formula of the local state recursion during calculation, and then gradually advances the optimal solution of the sub-problem to the optimal solution of the entire problem, which is essentially a dynamic programming algorithm [7].

First, define the hidden state at time \( t \) as \( q_t \) (\( i_t = q_t \)), and the probability that the sequence of the observed state is \( O = \{ o_1, o_2, \ldots, o_t \} \) is the forward probability. Recorded as:

\[
\alpha_t(i) = P(o_1, o_2, \ldots, o_t, i_t = q_t | \lambda) \tag{3}
\]

Figure 2 illustrates the recursive principle of the forward algorithm: \( \alpha_t(j) a_{ji} \) is the observed sequence \( O = \{ o_1, o_2, \ldots, o_t \} \), and when the hidden state at time \( t \) is \( q_j \), the hidden state at time \( t+1 \) is the probability of \( q_i \). So \( 2^M \times \alpha_{t+1}(i) b_{lj} \) is the observed sequence \( O = \{ o_1, o_2, \ldots, o_t \} \), the probability that the hidden state at time \( t+1 \) is \( q_i \). According to the forward probability definition, \( \alpha_{t+1}(i) \) is that the hidden state at \( t+1 \) is \( q_i \), and the sequence of the observed state is \( O = \{ o_1, o_2, \ldots, o_{t+1} \} \). The probability that observation sequence \( O = \{ o_2, \ldots, o_t \} \) has been observed at time \( t \) and the observation state at time \( t+1 \) is \( o_{t+1} \). Then we can deduce \( \alpha_{t+1}(i) = [ \sum_{j=1}^{M} \alpha_t(j) a_{ji} ] b_{lj}(o_{t+1}) \), \( i = 1, 2, \ldots, M \).

![Schematic diagram of the recursive principle of forward algorithm](image)

Figure 2. Schematic diagram of the recursive principle of forward algorithm

Suppose the length of the observation sequence is \( L \), and the dynamic programming starts at time 1 and ends at time \( L \). The steps of the forward algorithm are as follows:

1) Initialize, calculate the forward probability of each hidden state at time 1:

\[
\alpha_1(i) = \pi_i b_1(o_1), \quad i = 1, 2, \ldots, M \tag{4}
\]

2) Recursion, the forward probability of recursion to time \( L \):

\[
\alpha_{t+1}(i) = [ \sum_{j=1}^{M} \alpha_t(j) a_{ji} ] b_{lj}(o_{t+1}), \quad i = 1, 2, \ldots, M \tag{5}
\]

3) Terminate, and finally get the result:

\[
P(O|\lambda) = \sum_{t=1}^{M} q_t(i) \tag{6}
\]

From this, the time complexity of the forward algorithm is \( O(LM^2) \).

Therefore, the total probability formula is:

\[
P(O|\lambda) = \sum_{t=1}^{M} \alpha_t(i) \beta_t(i) \tag{7}
\]
3.3. Fall prediction algorithm based on HMM

The falling process of the human-machine system is a movement state that can be combined in chronological order. However, the current state cannot be judged by simply setting the threshold value directly through the acceleration signal. It is necessary to analyse the characteristics of the fall process through the change of acceleration [8]. In this paper, the HMM method is used to establish a probabilistic mathematical model for the fall process to realize the prediction of the fall process of the man-machine system.

(1) Data sample division

By repeatedly collecting data at each stage of the fall process of the human-machine system, the samples collected during the fall process are divided into training sample sets, statistical sample sets, and test sample sets. See table 1 for specific division and composition.

| Sample set          | Fall | Walk | Squat up | Bend over | Total |
|---------------------|------|------|----------|-----------|-------|
| Training samples $\Phi_T$ | 80   | 0    | 0        | 0         | 80    |
| Statistical sample $\Phi_S$ | 40   | 5    | 15       | 10        | 70    |
| Test sample $\Phi_T$       | 40   | 5    | 10       | 10        | 65    |
| Total                | 160  | 10   | 25       | 20        | 215   |

(2) Feature extraction of acceleration signal

First, perform feature extraction on the back three-axis acceleration signal in the training sample set [9]. According to formula (1), the three-axis acceleration signal is converted into a combined acceleration signal and the obtained acceleration signal is pre-processed, and then the data of each $m$ sampling point of the combined acceleration time series is merged into a group, and the $m$ sampling points The mean value of is used to characterize the resultant acceleration data in this $m \times T$ time period. Here, because the value of $m$ is 3, the algorithm calculation process can be simple and time-sensitive. Therefore, the expression of the resultant acceleration $a_c$ can be obtained:

$$\left(a_c\right)_i = \text{mean}\left(\sum_{j=1}^{m} a_j\right), \quad i = 1, 2, \ldots, n$$

(8)

Through experimental analysis, when the human body is standing or walking normally, the combined speed is generally about 9.8m / $s^2$. When the human body falls, the combined acceleration can reach 8 times the acceleration of gravity. Therefore, it is necessary to use a piecewise function to express the total acceleration value in sections and convert the acceleration value of each section into a label.

The value range of $a_c$ is $[0, +\infty)$, and the number of labels is an integer $K$. Based on the above analysis, $[9,11] m/s^2$ can be used as the first segment. The instantaneous acceleration will far exceed the normal acceleration, so $[20, +\infty) m/s^2$ is used as the last segment. According to multiple experiments, take $K = 8$, and set $[0,9)$. And $[11,20)$ are divided into six segments. Finally, the piecewise function of the characteristic time series $c_i$ is:

$$c_i = \begin{cases} 
   j + 1 , & (a_c)_i \in [3j, 3j + 3) \\
   4 , & (a_c)_i \in [9,11) \\
   k + 5 , & (a_c)_i \in [3k + 11,3k + 14) \\
   8 , & (a_c)_i \in [20, +\infty) 
\end{cases}, \quad i = 1, 2, \ldots, n$$

(9)

among them $j = 0,1,2, \quad k = 0,1,2$.

(3) Establishment of a fall model

After many experimental tests, the parameters in the fall model HMM can be selected. The values are as follows:

1) $M$: The number of motion states of the human-machine system before the collision impact. After many experiments, take $M = 3$.

2) $N$: The number of observations, that is, the number of tags above, $N = K = 8$. Suppose the set of observations is: $V = \{v_1, v_2, \ldots, v_9\}$.

3) $\pi$: Initial state probability vector, $\pi = \{\pi_1, \pi_2, \ldots, \pi_M\}$. Generally, the human-machine system
is in a stable state at the beginning, that is, the implicit state is $q_1$ at the beginning, then $\pi=\{1,0,0\}$ can be set.

4) $A$: The state transition matrix of the human-machine system before the fall, $A = \{a_{ij}\}_{M \times N}$, $a_{ij} = 1/M$.

5) $B$: The observation probability matrix of the human-machine system in the early stage of falling, $B = \{b_j(k)\}_{M \times N}$, $b_j(k) = 1/N$.

(4) Fall model training

According to the fall model $\lambda = (M, N, \pi, A, B)$, designed in the previous step, the acceleration characteristic time sequence $\{c_n\}$ in the early stage of the fall in the training sample set $\Phi_T$ is substituted for multiple times for training. The training uses MATLAB’s HMM tool library, which encapsulates algorithms such as B-W and Viterbi.

For B-W algorithm, the results are often too idealized if the model values are simply uniformly distributed. Therefore, four representative acceleration time series are selected in the experiment, and feature extraction is performed. Finally, the acceleration characteristic time series and the motion state sequence are put into the hmmestimate function in the HMM tool library for calculation. Get four fall models $\lambda_i = (M, N, \pi, A_i, B_i)$, $i = 1, 2, 3, 4$, and average the results, that is, $A = \sum_{i=1}^{4} A_i / 4$, $B = \sum_{i=1}^{4} B_i / 4$. Finally, the initial values of $A$ and $B$ can be obtained:

$$A = \begin{bmatrix} 0.8 & 0.2 & 0 \\ 0 & 0.8333 & 0.1667 \\ 0 & 0 & 1 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.5 & 0.3333 & 0.1667 & 0 & 0 & 0 & 0 & 0 \\ 0.3333 & 0.1667 & 0.1667 & 0 & 0 & 0 & 0 & 0.1667 \\ 0 & 0.3333 & 0.1667 & 0.1667 & 0 & 0.1667 & 0.1667 & 0.1667 \end{bmatrix}$$

According to the above method, the 76 samples in the training sample set $\Phi_T$ are trained to obtain the average value of 76 models $\lambda_T$, namely:

$$\lambda_p = (\pi, A, B) = \frac{1}{76} \sum_{t=1}^{76}(\pi_t, A_t, B_t)$$

(5) Falling behaviour prediction algorithm

![Figure 3](image-url)

**Figure 3.** The training process and matching process of the fall model

The fall model is a hidden Markov model established for the stage before the human-machine system falls and impacts. The resultant velocity time series is used as the observation value sequence $O$ of the new model, and the model $\lambda_p$ is calculated by the forward and backward algorithm. The occurrence probability $P(O|\lambda_p)$ of the sequence $O$ can be predicted by the judgment of the output probability $P(O|\lambda_p)$ of the model during the recognition process. The training process is shown in figure 3.

**Table 2.** Maximum value of model matching probability under each behaviour mode

| Sports behaviour | Walk | Bend over | Squat | Fall forward | Fall backward | Fall sideways | Stumble |
|------------------|------|-----------|-------|--------------|---------------|--------------|---------|
|                  |      |           |       |              |               |              |         |
Maximum probability (%) 0.4418 0.1782 0.6462 0.9829 0.9261 0.9802 0.8373
Minimum probability (%) 0 0 0 0.1786 0.1734 0.1660 0

After many experiments on the statistical sample set, Table 2 shows the maximum value of the model λ matching probability under each behaviour mode. After many adjustments, a pair of relatively suitable thresholds is obtained: \( P_1 = 0.41 \), \( P_2 = 0.67 \). When \( P(O|\lambda_p) \in [0, P_1) \), it can be judged that the current human-machine system is in a normal state of motion; when \( P(O|\lambda_p) \in [P_1, P_2) \), further judgment is required, which is recorded as state 1; \( P(O|\lambda_p) \in [P_2, 1] \), it can be judged that the current human-machine system is in the state of falling weightless, which is recorded as state 2.

![Figure 4. Prediction results of forward fall](image)

Select a forward fall sample of the sample set for intuitive verification, and the output results are shown in Figure 4. At the beginning, the resultant acceleration is about 9.8 m/s\(^2\), and the prediction algorithm judges that it is in state 0, that is, the normal standing stage. When the falling behaviour is just beginning, the combined acceleration has a slower downward trend, and the prediction algorithm judges that it is in state 1, that is, a relatively unstable state. When the combined acceleration drops to about 5 m/s\(^2\), the prediction algorithm judges that it is in state 2, that is, the state of falling weightless. When the acceleration increases sharply, the prediction algorithm judges that the state returns to 0. In summary, the model based on the HMM pattern recognition method before the impact of the fall and collision can better identify the early actions of the fall and achieve certain prediction purposes.

4. Conclusions
This article is a research on the fall prediction algorithm of the assisted lower limb exoskeleton robot. In the research process, the fall behaviour and characteristics of the assisted lower extremity exoskeleton human-machine system are analysed first, and the falls are classified and the stability margin changes during the fall are discussed; The improved algorithm based on HMM is used to predict different fall behaviours and perform experimental verification. The main idea of the improved algorithm is to first analyse the acceleration time series with the HMM-based fall prediction model, analysing the risk of falling, then the values selected for further verification and identification of the type fall prediction algorithm according to whether a fall risk of falling. The improved fall prediction algorithm has a relatively high accuracy rate, and it can predict the occurrence of falling behaviour in a relatively short time. This laid the foundation for the subsequent study of fall protection measures.
Acknowledgement
This work was supported by the National Key R&D Program of China under Grant NO.2017YFB1300502.

References
[1] Guizzo E, Goldstein H 2005 The rise of the body bots[J]. 42(10):42-48.
[2] Tao, Yu, Zhang 2012 A control system based on MCU for wearable power assist legs[C]. IEEE International Conference on Robotics & Biomimetics. IEEE, 2193-2198.
[3] Zhu Y, Cui J and Zhao J 2013 Biomimetic design and biomechanical simulation of a 15-DOF lower extremity exoskeleton[M]. 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE, 1119-1124.
[4] Gardner A D, Potgieter J 2015 A review of commercially available exoskeletons' capabilities[C]. 2017 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP). 1-5.
[5] Feng Chen, Yong Yu, et al 2007 A PAWL for Enhancing Strength and Endurance during Walking Using Interaction Force and Dynamical Information[C]. IEEE International Conference on Robotics & Biomimetics. IEEE, 654-659.
[6] Rabiner L R 1989 A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition[J]. IEEE, 77.
[7] Talaty M, Esquenazi A, et al 2013 Differentiating ability in users of the ReWalk powered exoskeleton: An analysis of walking kinematics[C]. 2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR). IEEE, 1-5.
[8] Kazerooni, H 2006 Hybrid Control of the Berkeley Lower Extremity Exoskeleton (BLEEX)[J]. The International Journal of Robotics Research, 25(5-6):561-573.
[9] Noury N, Fleury A, et al 2007 Fall detection--principles and methods[J]. 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 1663-1666.