ICIC Express Letters ICIC International ©2018 ISSN 1881-803X
Volume **12**, Number **6**, June **2018** pp. **501–510**

USING HIDDEN MARKOV MODEL TO PREDICT HUMAN ACTIONS WITH HYBRID IMPROVED GRAVITATIONAL SWARM INTELLIGENCE SEARCH ALGORITHM

Zhicheng Lu and Yuk Ying Chung

School of Information Technologies
The University of Sydney
J12/1 Cleveland Street, Darlington NSW 2008, Australia
zhlu2106@uni.sydney.edu.au; vera.chung@sydney.edu.au

Received November 2017; accepted February 2018

Abstract. *In this paper, a new algorithm named improved gravitational particle swarm optimization (iGPSO) is proposed to predict the human actions using hidden Markov model (HMM). Two datasets in the UCI Machine Learning Repository – Occupancy Detection Data Set and Activity Recognition from Single Chest-Mounted Accelerometer Data Set – have been used to test our new algorithms. Compared to the conventional HMM, our proposed iGPSO-HMM produces 6% higher prediction accuracy, which is able to prove that our proposed method is more suitable for prediction of human actions.*
Keywords: Hidden Markov model, Particle swarm optimization, Gravitational search algorithm, Human activity prediction

1. **Introduction.** Hidden Markov model (HMM) was firstly proposed by Baum and his colleagues in the late 1960s and early 1970s [1, 2, 3, 4, 5]. HMM holds the dominant position in the area of speech recognition system [6] in the past and still holds the position currently [7]. HMM is also used in other areas related to pattern recognition such as handwriting recognition [8, 9] and gesture recognition [10, 11]. There are three essential parameters in the model of HMM, which are transition probability A, emission probability B and start probability π; together they are called the model parameters of the HMM, as $\lambda = (A, B, \pi)$. Model parameters λ are able to determine the output of the HMM and hence are the parameters that need to be optimized; however, optimization of HMM is considered to be difficult because there are exponential number of local optimum to search [12, 13].

It is well known that the traditional optimization algorithm, namely Baum-Welch method, is likely to be stuck in the local optimum in spite of its low computational cost. As a consequence, optimization algorithms are applied to HMM to optimizing the model parameters where particle swarm optimization (PSO) is one of the examples. PSO was firstly introduced by Kennedy et al. [14, 15], aiming to simulate social behavior such as fish schooling and bird flocking by swarm representation [16]. There are some algorithms proposed in the last decade to use PSO to train the HMM model parameters. Najkar et al. used PSO to train HMM model parameters for speech recognition system [17]. Hamdani et al. used multi models evolvement (MME) to improve the performance of PSO [18]. Li et al. further improved the performance of HMM-PSO [19]. Lu et al. proposed HMM-RPSO to improve the searching ability of the particles [20].

Gravitational search algorithm (GSA), an alternative optimization algorithm, is able to solve the optimization problem in a high-dimensional search space [21]. The algorithm, derived from the Newtonian gravity that describes the attraction force among two particles, simulates the environment that mass points move in the search space obeying Newton's law of universal gravitation [21]. In this paper, we propose a new method named improved gravitational particle swarm optimization (iGPSO). This proposed new iGPSO can prevent the solution stuck at local optima. In the testing experiments, the proposed HMM-iGPSO has been compared to the traditional HMM using Baum-Welch method and the HMM-PSO by predicting the human activities based on the data captured by sensors on human body. The predicting accuracy has been evaluated as the proportion that the predicted activities match the ground truth.

This paper is structured as follows. Section 2 gives the introduction of HMM, PSO and GSA algorithms mentioned in the article. Section 3 illustrates our proposed iGPSO and HMM-iGPSO methods. Section 4 shows all the experimental results using two different datasets. Section 5 gives the conclusion and the future work.

2. Background Knowledge.

2.1. Hidden Markov model.

2.1.1. *Model parameters of the HMM.* The model parameters of the HMM are $\lambda = (A, B, \pi)$. A, B and π are all probability distributions that could determine the HMM model, so in this section they will be introduced respectively. Suppose there are N hidden states (cannot be observed) $S = \{S_1, S_2, \ldots, S_N\}$, M observation symbols $V = \{V_1, V_2, \ldots, V_M\}$, a hidden state sequence $X = \{X_1, X_2, \ldots, X_T\}$ and an observed sequence $O = \{O_1, O_2, \ldots, O_T\}$, and then $\lambda = (A, B, \pi)$ can be explained as [22]:

1) Transition probability matrix A determines the hidden states of current frame depending on the states of the previous frame: $A = \{a_{ij}\}$ where $a_{ij} = Pr(X_t = S_j|X_{t-1} = S_i)$;
2) Emission probability matrix B determines the probability of a certain hidden state produces a certain output: $B = \{b_j(k)\}$ where $b_j(k) = Pr(O_t = V_k|X_t = S_i)$;
3) Start probability matrix π determines the probability of the state of the first instance: $\pi = \{\pi_i\}$ where $\pi_i = Pr(X_1 = S_i)$.

The optimization problem of HMM is then defined as: given the set of hidden states S, the set of observation symbols V and the set of observed sequence O, find the model parameters $\lambda = (A, B, \pi)$ to maximize the probability of $Pr(O|\lambda)$ [23].

2.1.2. *Baum-Welch method for HMM.* As there is no method to solve the optimization problem of the HMM in close form, Baum-Welch method is a popular method used to numerically optimize the HMM [24]. In the algorithm of Baum-Welch method, forward algorithm and backward algorithm [25] will be utilized. In each iteration of the Baum-Welch method, the model parameters $\lambda = (A, B, \pi)$ will be updated according to the forward and backward procedure to maximize the value of $Pr(O|\lambda)$ [25].

2.2. Particle swarm intelligence.

2.2.1. *PSO algorithm procedure.* PSO is an iterative algorithm for the optimization objective. In the PSO algorithm, particles are randomly initialized in the multi-dimensional search space. In each iteration, each particle will move towards the best position that particle has reached and the best position that all the particles have reached. More formally, the process of the PSO algorithm can be defined as [14, 26]:

1) Initialize n particles in the D-dimensional search space within the value domain, and each particle is initialized with 0 velocity;
2) Set the position of each particle as the $Pbest$ of the particle; compare and find the position of the particle with the highest fitness value, set the position as $Gbest$;
3) In each iteration t, for each dimension d of particle i, the velocity $v_i^d(t)$ will be updated as:

$$v_i^d(t+1) = w \times v_i^d(t) + r_1 \times \left(Pbest_i^d - x_i^d(t)\right) + r_2 \times \left(Gbest^d - x_i^d(t)\right) \quad (1)$$

The position $x_i^d(t)$ will be updated as:

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{2}$$

4) For each particle i, update its fitness and update its *Pbest* (if necessary);
5) Update the position of *Gbest* (if necessary);
6) The pre-defined number of iteration is reached, stop the algorithm; otherwise, go back to step 3.

2.2.2. *Applications of PSO.* PSO is useful when the optimal solution is less straightforward to calculate due to high dimensionality or wide-range value of real data. Although PSO may not return the optimal solution, the solution from PSO is sufficient for real-world applications since it is well-balanced in terms of the number of iterations (i.e., running speed) and the solution accuracy. The applications of PSO can be found in video tracking, communication networks [26] and transportation network design problems [27]. PSO can also be applied to optimizing HMM parameters [17, 18, 19] to solve some time-series problems.

2.3. **Gravitational search algorithm.**

2.3.1. *GSA algorithm procedure.* GSA was first introduced by Rashedi et al. to solve the optimization problem [28], which can be described as follows. Assume that in the D-dimensional search space, particle i is defined as:

$$X_i = \left(x_i^1, x_i^2, \ldots, x_i^D\right) \tag{3}$$

Then, the mass $M_i(t)$ of each particle i at time t can be calculated as [21, 28]:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{4}$$

$$M_i(t) = \frac{m_i(t)}{\sum_j m_j(t)} \tag{5}$$

where $fit_i(t)$ represents the fitness value of particle i at time t, and $worst(t)$ and $best(t)$ are the worst and best fitness value at time t respectively.

In the next step, according to the gravitational force [29, 30] and Newton's Second Law [31], we can compute the acceleration $a_i^d(t)$ of dimension d in particle i by:

$$F_i^d(t) = \sum_{j, j\neq i} rand_j \times G(t)\frac{M_j(t)M_i(t)}{R_{ij}(t)+\epsilon}\left(x_j^d(t) - x_i^d(t)\right) \tag{6}$$

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} \tag{7}$$

where $rand_j$ is a random number between 0 and 1, $G(t)$ is the gravitational constant, $R_{ij}(t)$ is the Euclidean distance between particle i and particle j at time t, and ϵ is a small constant value.

The positions of the particles are then updated with equations:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{8}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{9}$$

2.3.2. *Comparison between PSO and GSA.* Although PSO and GSA are both optimization algorithms, they have differences in the following aspects [21, 28].

1) PSO simulates the behavior of animals especially bird, while GSA uses physical law.
2) The movement of the particles is determined by *Gbest* and $Pbest_i$ in PSO, while the movement is affected by all the other particles in GSA.
3) PSO memorizes the value of *Gbest* and *Pbest*, while in GSA only the current position of the particles will be considered.

3. The Proposed Method.

3.1. Improved gravitational particle swarm optimization (iGPSO).

As Section 2.3.2 states, PSO and GSA have differences, but both of them try to simulate some certain environment for the particles in the search space. As a result, they can be merged together to become gravitational particle swarm optimization (GPSO) when computing the velocity $v_i^d(t+1)$:

$$a_1 = r_1 \times \left(Pbest_i^d - x_i^d(t)\right) + r_2 \times \left(Gbest^d - x_i^d(t)\right) \tag{10}$$

$$a_2 = \frac{\sum_{j,j\neq i} rand_j \times G(t)\frac{M_j(t)M_i(t)}{R_{ij}(t)+\epsilon}\left(x_j^d(t) - x_i^d(t)\right)}{M_i(t)} \tag{11}$$

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_1 + s \times a_2 \tag{12}$$

where $rand_i$ is random value between 0 to 1, and s is the scale factor depending on the data set.

Even though PSO and GSA seem less likely to converge pre-maturely compared to traditional Baum-Welch method, the particles may still stick at a local maximum for the real-world data because of the wide range of the data. Therefore, improved versions of the GPSO (iGPSO) are proposed in Equation (13). More specifically, when the algorithm is implemented, it might be found that the training accuracy converges or stagnates at a certain stage, says iteration n. Therefore, our improved GPSO (iGPSO) has been proposed to re-intialize the position of $x_i^d(t+1)$ if the position of *Gbest* stays for a pre-defined k iterations as below:

$$x_i^d(t+1) = \begin{cases} x & Gbest \text{ stays for more than } k \text{ iterations} \\ x_i^d(t) + v_i^d(t+1) & \text{otherwise} \end{cases} \tag{13}$$

where x is a new random position for dimension d of particle i within the range of the search space, and $v_i^d(t+1)$ is computed via Equation (12). Theoretically, this modification will enable GPSO algorithm to search more space and thus improve the performance of the algorithm.

3.2. HMM-iGPSO.

In order to use iGPSO to train the model parameters of the HMM, the number of dimensions of the HMM needs to be determined. Assume the HMM has N hidden states and M potential observations. Then the total dimension of the HMM is given by:

$$D = N^2 + 2MN + N \tag{14}$$

because transition probability matrix has N^2 dimensions, emission probability matrix has $2MN$ dimensions for continuous HMM, and start probability matrix has N dimensions.

Then the iGPSO algorithm is to optimize this D dimensional space as Algorithm 1 HMM-iGPSO shows below.

Algorithm 1 HMM-iGPSO

```
while stopping criteria have not been met do
    for each particle p do
        for each dimension d do
            update the position according to Equation (13)
            normalize the probabilities so that the sum is 1
        end for
        if fitness(p, Q, O) > fitness(pbest[p], Q, O) then
            pbest[p] ← p
        end if
        if fitness(p, Q, O) > global then
            global ← fitness(p, Q, O)
            gbest ← p
        end if
    end for
    if gbest stays for k iterations then
        update the position of all the particles
    end if
end while
```

4. **Experiments and Results.**

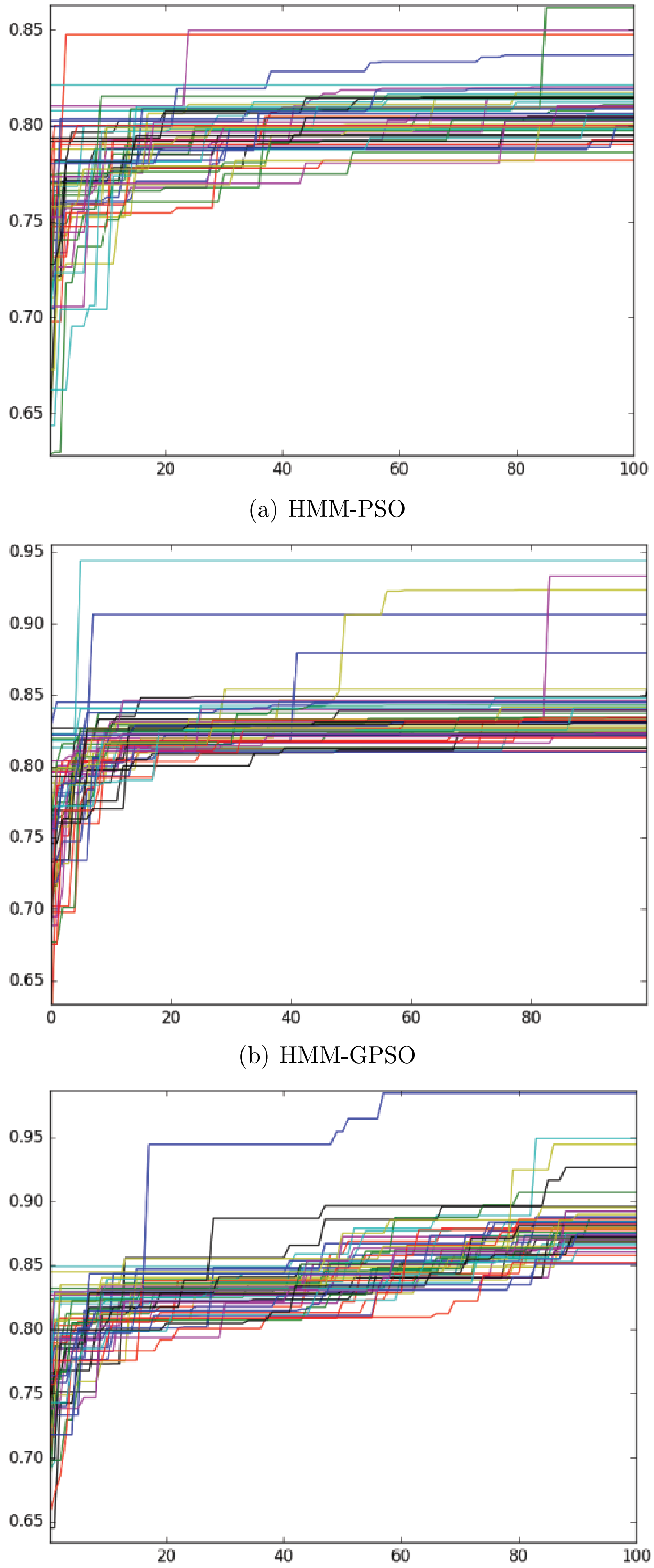
4.1. **Occupancy Detection Data Set.**

Occupancy Detection Data Set is an experimental data set for binary classification [32]. The data set consists of 4 attributes which are temperature, humidity, light and CO_2, and one binary class (occupied or not occupied).

For our experiments, there have 20 particles and 100 iterations for PSO-based algorithms, 2000 iterations for Baum-Welch method. The scale vector has been set to 100, and the particles have been re-initialized if the *Gbest* has been kept the same for 10 iterations. Since random values affect the performance of the PSO significantly, we have run all the PSO-based algorithms 50 times to get the average accuracy.

4.1.1. *Curve of training accuracies.* PSO is an algorithm that tries to reach better local optimum at each iteration, Figure 1(a) shows the training accuracies during the training process of HMM-PSO, Figure 1(b) illustrates that of HMM-GPSO and Figure 1(c) is for HMM-iGPSO. We can easily find that HMM-PSO algorithm seems to converge in the early stage and stay for the rest of the process. HMM-GPSO has the better performance. However, HMM-iGPSO continuously increases in training accuracy because of the re-initialization step in the algorithm. Besides, it can also be found that HMM-iGPSO results in higher training accuracies than the other two, resulting in higher prediction accuracies.

4.1.2. *Accuracy tests.* The experiment is conducted as mentioned in this section, the original test data will be split into 11 sets according to their corresponding date, and the testing results can be found in Table 1. It can be found that the traditional Baum-Welch method gives 80.76% in accuracy, and HMM-PSO and HMM-GPSO are slightly better, yielding accuracies of 81.72% and 82.05% respectively. Our proposed method HMM-iGPSO performs the best with an accuracy of 86.41%, a significant boost of more than 4%. As a result, our proposed algorithm could help HMM to predict the human activities more accurately.



(a) HMM-PSO

(b) HMM-GPSO

(c) HMM-iGPSO

Figure 1. Occupancy training accuracies

Table 1. Occupancy accuracy test

Test Set	Conventional HMM	HMM-PSO	HMM-GPSO	HMM-iGPSO
1	79.32%	81.97%	83.51%	85.19%
2	81.25%	81.10%	81.43%	87.76%
3	80.17%	83.25%	82.12%	86.65%
4	80.01%	82.15%	81.36%	87.31%
5	83.10%	79.22%	80.05%	88.53%
6	81.87%	79.58%	83.36%	85.38%
7	79.50%	81.09%	82.58%	84.54%
8	81.91%	83.56%	81.97%	87.64%
9	78.62%	83.40%	82.00%	82.01%
10	80.08%	82.34%	81.26%	87.10%
11	82.48%	81.34%	82.94%	86.49%
average	80.76%	81.72%	82.05%	86.41%

4.2. **Activity Recognition from Single Chest-Mounted Accelerometer Data Set.** The Activity Recognition from Single Chest-Mounted Accelerometer Data Set was taken by Casale and his colleagues in 2012 [33]. Data are captured from 15 volunteers performing 7 different activities (Working at Computer; Standing Up, Walking and Going updown stairs; Standing; Walking; Going updown stairs; Walking and Talking with Someone; and Talking while Standing) in time-series. In each frame of each volunteer, x acceleration, y acceleration, z acceleration and the activity the person performs will be recorded.

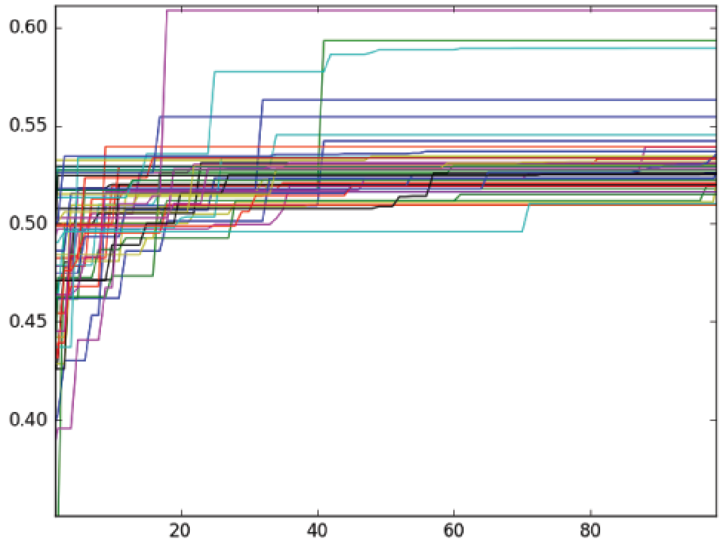
For our experiment, we will use n-fold cross validation: the data from each person will be used as the test set once while the rest will work as the training set. In this case, the fitness value in Algorithm 1 will be set as the average prediction accuracies of 14 training sets.

Moreover, in this experiment, there will be 20 particles and 100 iterations for PSO-based algorithms, 2000 iterations for Baum-Welch method; the scale vector will be set to 100 as well. Since random value affects the performance of the PSO significantly, PSO-based algorithms will run 50 times to get the average accuracy.

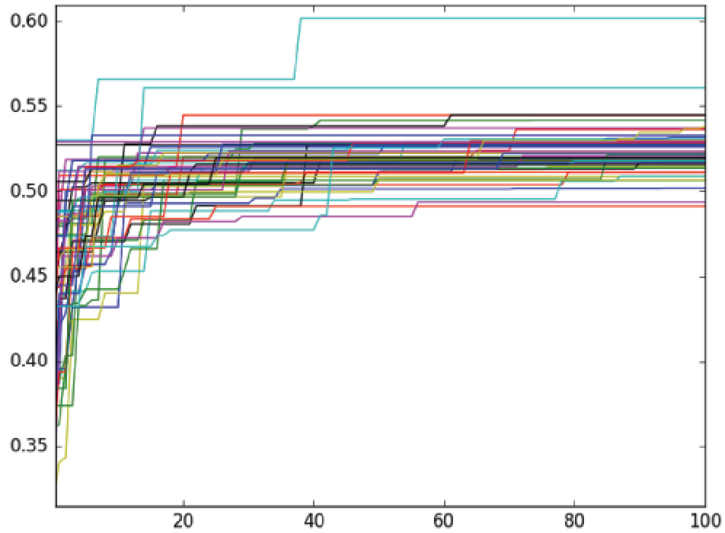
4.2.1. *Curve of training accuracies.* Similar to Section 4.1.1, we will use the dataset 1 to dataset 14 as the training dataset, and then the increase in training accuracies is recorded in Figure 2. By comparing the graph in Figure 2(a), Figure 2(b) and Figure 2(c), it can be found that HMM-iGPSO could result in a continuous increase and higher level convergence in training accuracies. Although higher training accuracies do not mean better model, it suggests that our proposed model is able to search more space than the traditional algorithm.

4.2.2. *Accuracy tests.* Results are obtained by using the n-fold cross validation and are presented in Table 2, where we have listed the testing accuracies for all the test validations and four different models. It can be found that the average accuracy of conventional HMM is 55.23%, and HMM-PSO is slightly better with an average accuracy of 56.61%. HMM-GPSO is slightly better than the HMM-PSO with 56.95%. With the further modification and improvement, our proposed HMM-iGPSO is much better than the other algorithms, exceeds conventional HMM by around 6%, and achieves approximately 5% more accuracy than HMM-PSO and HMM-GPSO.

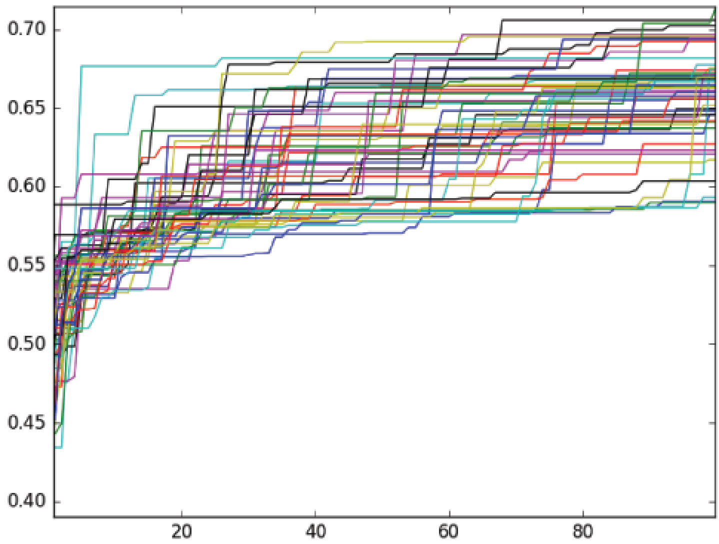
5. **Conclusion.** In this article, we have proposed novel algorithms iGPSO and HMM-iGPSO to predict human actions. The experiments in Section 4 illustrate that the proposed HMM-iGPSO can perform better than the conventional HMM and HMM-PSO, and



(a) HMM-PSO



(b) HMM-GPSO



(c) HMM-iGPSO

FIGURE 2. Chest training accuracies

TABLE 2. Chest accuracy test

Testing Validation	Conventional HMM	HMM-PSO	HMM-GPSO	HMM-iGPSO
1	55.77%	57.11%	56.80%	62.22%
2	56.32%	56.86%	57.57%	60.75%
3	54.76%	57.57%	54.36%	60.99%
4	55.11%	56.21%	61.25%	61.59%
5	55.08%	56.94%	55.63%	61.21%
6	54.89%	56.83%	58.93%	61.71%
7	54.36%	54.00%	54.55%	60.59%
8	53.97%	55.29%	53.54%	60.12%
9	55.29%	56.46%	58.08%	58.68%
10	55.91%	58.12%	59.11%	60.58%
11	54.82%	54.65%	55.59%	61.10%
12	56.02%	56.78%	54.70%	62.45%
13	55.71%	57.05%	56.82%	61.87%
14	54.91%	59.22%	58.29%	60.42%
15	55.56%	56.07%	59.13%	60.78%
Average	55.23%	56.61%	56.95%	61.01%

to be more specific, the proposed HMM-iGPSO has about 5% higher average prediction accuracy than HMM-PSO, and at least 6% higher average prediction accuracy than the conventional HMM.

Future works can be done by further modifying the HMM-iGPSO algorithm to optimize the HMM model parameters and studying the influence of the scale factor and the number of iterations leading to re-initialization.

REFERENCES

[1] L. E. Baum and T. Petrie, Statistical inference for probabilistic functions of finite state Markov chains, *The Annals of Mathematical Statistics*, vol.37, no.6, pp.1554-1563, 1966.

[2] L. E. Baum and J. A. Eagon, An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology, *Bulletin of the American Mathematical Society*, vol.73, no.3, p.360, 1967.

[3] L. E. Baum and G. R. Sell, Growth transformations for functions on manifolds, *Pacific Journal of Mathematics*, vol.27, no.2, pp.211-227, 1968.

[4] L. E. Baum, T. Petrie, G. Soules and N. Weiss, A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, *The Annals of Mathematical Statistics*, vol.41, no.1, pp.164-171, 1970.

[5] L. E. Baum, An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process, *Inequalities*, vol.3, pp.1-8, 1972.

[6] T. Starner and A. Pentland, Real-time American sign language recognition from video using hidden Markov models, *Motion-Based Recognition*, pp.227-243, 1997.

[7] C. Champion and S. M. Houghton, Application of continuous state hidden Markov models to a classical problem in speech recognition, *Computer Speech & Language*, vol.36, pp.347-364, 2016.

[8] H. Cao, R. Prasad and P. Natarajan, OCR-driven writer identification and adaption in an HMM handwriting recognition system, *International Conference on Document Analysis and Recognition (ICDAR)*, pp.739-743, 2011.

[9] V. Ghods and M. K. Sohrabi, Online Farsi handwriting character recognition using hidden Markov model, *JCP*, vol.11, no.2, pp.169-175, 2016.

[10] N. Liu, B. C. Lovell, P. J. Kootsookos and R. I. Davis, Model structure selection & training algorithms for an HMM gesture recognition system, *The 9th International Workshop on Frontiers in Handwriting Recognition*, pp.100-105, 2004.

[11] J. Ding and C. W. Chang, An adaptive hidden Markov model-based gesture recognition approach using Kinect to simplify large-scale video data processing for humanoid robot imitation, *Multimedia Tools and Applications*, vol.75, no.23, pp.15537-15551, 2016.

[12] R. Granat, A method of hidden Markov model optimization for use with geophysical data sets, *Proc. of the 2003 International Conference on Computational Science: Part III*, pp.892-901, 2003.
[13] D. Roman, G. Mitra and N. Spagnolo, Hidden Markov models for financial optimization problems, *IMA Journal of Management Mathematics*, vol.21, no.2, pp.111-129, 2009.
[14] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proc. of IEEE International Conference on Neural Networks*, pp.1942-1948, 1995.
[15] Y. Shi and R. Eberhart, A modified particle swarm optimizer, *Proc. of IEE International Conference on Evolutionary Computation*, pp.69-73, 1998.
[16] J. Kennedy, The particle swarm: Social adaptation of knowledge, *Proc. of IEE International Conference on Evolutionary Computation*, pp.303-308, 1997.
[17] N. Najkar, F. Razzazi and H. Sameti, A novel approach to HMM-based speech recognition systems using particle swarm optimization, *Mathematical and Computer Modelling*, vol.52, no.11, pp.1910-1920, 2010.
[18] M. Hamdani, H. El Abed, T. M. Hamdani, V. Margner and A. M. Alimi, Improving a HMM-based off-line handwriting recognition system using MME-PSO optimization, *DRR*, 2011.
[19] H. Li, Y. Yi, X. Li and Z. Guo, Human activity recognition based on HMM by improved PSO and event probability sequence, *Journal of Systems Engineering and Electronics*, vol.24, no.3, pp.545-554, 2013.
[20] Z. Lu, Y. Y. Chung, H. W. F. Yeung, S. M. Zandavi, W. Zhi and W. C. Yeh, Using hidden Markov model to predict human actions with swarm intelligence, *International Conference on Neural Information Processing*, Springer, Cham, pp.21-30, 2017.
[21] E. Rashedi, H. Nezamabadi-Pour and S. Saryazdi, GSA: A gravitational search algorithm, *Information Science*, vol.179, no.13, pp.2232-2248, 2009.
[22] K. Stratos, M. Collins and D. Hsu, Unsupervised part-of-speech tagging with anchor hidden Markov models, *Transactions of the Association for Computational Linguistics*, vol.4, pp.245-257, 2016.
[23] C. Kohlschein, An introduction to hidden Markov models, *Probability and Randomization in Computer Science Seminar in Winter Semester*, vol.2007, 2006.
[24] J. Kabudian, M. R. Meybodi and M. M. Homayounpour, Applying continuous action reinforcement learning automata (CARLA) to global training of hidden Markov models, *Proc. of International Conference on Information Technology: Coding and Computing*, vol.2, pp.638-642, 2004.
[25] A. Paris, S. Arbaoui, N. Cislo, A. El-Amraoui and N. Ramdani, Using hidden semi-Markov model for learning behavior in smarthomes, *IEEE International Conference on Automation Science and Engineering (CASE)*, pp.752-757, 2015.
[26] R. Poli, J. Kennedy and T. Blackwell, Particle swarm optimization, *Swarm Intelligence*, vol.1, no.1, pp.33-57, 2007.
[27] A. Babazahed, H. Poorzahedy and S. Nikoosokhan, Applications of particle swarm optimization to transportation network design problem, *Journal of King Saud University Science*, vol.23, no.3, pp.293-300, 2011.
[28] E. Rashedi, H. Nezamabadi-Pour and S. Saryazdi, Filter modelling using gravitational search algorithm, *Engineering Applications of Artificial Intelligence*, vol.24, no.1, pp.117-122, 2011.
[29] B. Schutz, *Gravity from the Ground Up: An Introductory Guide to Gravity and General Relativity*, Cambridge University Press, 2003.
[30] D. Holliday, R. Resnick and J. Walker, *Fundamentals of Physics*, 1993.
[31] J. H. Gundlach, S. Schlamminger, C. C. Spitzer, K. Y. Choi, B. A. Woodahl, J. J. Coy and E. Fischbach, Laboratory test of Newtons Second Law for small accelerations, *Physical Review Letters*, vol.98, no.15, 2007.
[32] L. M. Candanedo and V. Feldheim, Accurate occupancy detection of an office room from light, temperature, humidity and CO_2 measurements using statistical learning models, *Energy and Buildings*, vol.112, pp.28-39, 2016.
[33] P. Casale, O. Pujol and P. Radeva, Personalization and user verification in wearable systems using biometric walking patterns, *Personal and Ubiquitous Computing*, vol.16, no.5, pp.563-580, 2012.