Motion Trajectory Sequence-Based Map Matching Assisted Indoor Autonomous Mobile Robot Positioning

Jianzhong Zhang[2] (✉), Wenping Yu[1], Jingdong Xu[2], and Yuwei Xu[2]

College of Computer and Control Engineering, Nankai University, Tianjin, China
[1]`{yuwenping}@mail.nankai.edu.cn`
[2]`{zhangjz,xujd,xuyw}@nankai.edu.cn`

Abstract. Position information is one of basic elements for context awareness of autonomous mobile robots. This paper studies the positioning algorithm of autonomous mobile robots suitable for search and rescue in dark building corridors and underground mine tunnels when an emergency occurs, and proposes a novel map matching aided positioning algorithm based on Hidden Markov Model. This algorithm does not rely on a camera, and only uses the inertial sensors installed in mobile robot and the indoor map to realize the fusion of dead reckoning and map matching. Firstly, it recognizes the position-related motion pattern during the motion of wheeled mobile robot, and then the motion trajectory is divided into a sub-trajectory sequence. By matching the sub-trajectory sequence with the indoor map, the proposed algorithm achieves accurate tracking and positioning of mobile robots. In order to verify the effectiveness of our proposed algorithm, this paper adopts four-wheel differentially driven robot to conduct experimental analysis in an actual indoor scenario. The experimental results show that compared with the traditional dead reckoning technology, this algorithm can distinctly reduce the average positioning error of mobile robot, and it is robust to heading angle and acceleration noises within a certain error range.

Keywords: Mobile Robot · Indoor Positioning · Hidden Markov Model · Motion Pattern Detection.

1 Introduction

With the advancement of artificial intelligence technology, network technology and sensor technology, the research and application of autonomous mobile robots have made remarkable progress in recent years. Indoor autonomous mobile robots are increasingly integrated into people's daily lives [1]. Autonomous mobile robots can be used not only in large-scale applications in modern intelligent warehouses, home services and many other aspects, but also in detection and search and rescue in corridors of complex buildings that are outages due to fire accidents, subway passages, and underground mines. . Therefore, the research of

indoor autonomous mobile robot technology has gradually become a hot topic, and many domestic research institutes such as Tsinghua University, Harbin Institute of Technology, Nankai University, South China University of Technology are committed to the research and development of indoor autonomous mobile robots [2-6]. The autonomous positioning of the mobile robot is a process in which the robot autonomously determines its position in the working environment, and is one of the most basic problems in improving the autonomous capabilities of the mobile robot.

In terms of outdoor positioning, the Global Positioning System (GPS) has become a widely used positioning technology for mobile robots. In terms of indoor positioning, due to the blocking and interference of GPS signals on the external walls of buildings and indoor complex electromagnetic environment, indoor mobile robots There is no universal solution to the positioning problem [7-8]. Currently, researchers have proposed a variety of positioning methods for indoor autonomous mobile robots, including navigation beacon-based positioning [9], computer vision positioning [10-11], dead reckoning positioning [12], map matching positioning [13-14] and synchronous positioning and map creation (SLAM) [15-16] and so on. Positioning based on navigation beacons relies on deployed feature signals in a range of environments to provide stable and accurate location information, but requires high deployment and maintenance costs. Dead reckoning positioning uses inertial sensors and encoders to provide relatively accurate phase positions over short distances, but there is cumulative error that gradually increases as the distance travels, and the robot's initial pose needs to be known in advance. Map matching positioning uses known indoor maps to construct topological maps, feature maps, and other abstract maps, and then the position of the mobile robot is obtained by matching the robot motion trajectory with the indoor path. From the map matching positioning realization principle, the real-time performance is relatively poor. Shortcomings. The SLAM technology has unique advantages in the face of unknown environments and can provide indoor planes or 3D maps while providing positioning [17]. However, this method requires mobile robots equipped with more complex sensor devices, such as infrared, ultrasonic radar, and RGBD vision systems. Therefore, it has higher implementation cost.

Complex corridors, subway station passages, or underground mines often have multiple passageways, similar to "mazes". In the event of a fire or other unexpected accident, the power supply is damaged, the communication infrastructure cannot be used, smoke, etc. It poses challenges for the positioning of autonomous mobile robots. Due to limitations in work environment or deployment conditions, it is difficult to establish visual or wireless navigation beacons in advance. Therefore, positioning technology based on navigation beacons is not suitable; the effects of high temperature and smoke on the indoor environment, cameras and other difficult to provide image information, computer The failure of visual positioning technology; the timeliness of SLAM technology can not meet the urgent need for time factors in the above scenarios. In response to these problems, the paper introduces a hidden Markov model (HMM) that does

not rely on a camera. It only uses inertial sensors (accelerometers, gyroscopes, and magnetometers) installed in autonomous mobile robots and known indoor maps to achieve dead reckoning. The combination of mapping technology and map matching technology.

The remainder of this paper is organized as follows: Section II reviews the related work in the literature. Section III gives an overview of the AiFiMatch system and its preprocessing components. Then, we detail the proposed HMM-based map matching algorithm in Section IV. Section V shows the experimental results and analysis of the AiFiMatch system. Finally, Section VI concludes this paper.

2 Related Work

PDR techniques relying on smartphones have attracted much attentions in the past few years, but we only position the literature with respect to our work in this section. We focus on techniques that utilize easily accessible infrastructure, such as indoor map information, inertial sensors embedded in smartphones and wireless access points in buildings. We divide these techniques into the following three categories: map information assisted PDR, activity detection based PDR and combination PDR with Wi-Fi Fingerprint.

2.1 Map Information Assisted PDR

The pedestrian motion trajectory is restricted by the indoor map, therefore, it is an opportunity to improving PDR using map information. Woodman and Harle propose a pedestrian positioning approach which combines a detailed building map model and a particle filter to provide absolute positioning [1]. An improved particle filter is also introduced into map matching in [2] and its computational complexity is less than traditional ones. Map matching algorithm is triggered only when a pedestrian is considered walking along a certain corridor. [3] leverages the topology of the indoor map to filter out infeasible locations over time with the particle filter. However, the computation time is still the major issues of particle filter based map matching algorithms, as a large number of particles are required to ensure good estimations. MapCraft proposed in [4] expresses the map matching based pedestrian positioning problem as a conditional random field (CRF), which is extremely computationally efficient and tracks well even when presented with very noisy sensor data. Both MapCraft and our work only requires a floor plan to track pedestrians, but a crowd-sourced Wi-Fi fingerprints database could be built by our work when wireless access points exist.

3 Robot Motion Model and Positioning Method

In the field of indoor autonomous mobile robot positioning, dead reckoning technology and map matching technology have a good complementarity. This paper

proposes a map matching-assisted positioning method based on moving trajectory sequences of mobile robots to realize the fusion of the above two technologies. The positioning method uses stairs, corridor corners, etc. in the indoor environment as virtual landmarks. When the mobile robot passes through these landmark locations, the inertial sensor data will show a specific pattern. Therefore, in the text, the above landmark location is called a posture correlation. Position, when the robot's movement distance is short, the dead reckoning technology can give the real-time position of the robot. In the robot's movement distance is long, the robot's motion trajectory can be divided into multiple trajectory fragments according to the landmark position, continuous trajectory fragment The track sequence is composed of hidden Markov models. The above trajectory sequence can be matched to the corresponding path in the indoor map, and then the position estimation of the mobile robot is given. Further, when the robot's motion trajectory is long enough, the robot's When the initial position is unknown, the absolute position of the mobile robot can still be estimated.

This section first introduces the relevant modules and simplified motion models of the four-wheel differential-driven autonomous mobile robot used in subsequent experiments, and then derives the dead reckoning technology based on the motion model. Then the overall architecture of the fusion positioning method proposed in this paper is introduced. And the indoor map abstraction module, the final description of the mobile robot motion gesture recognition and use of the four-wheel differential drive robot designed in this paper to conduct a related pre-experiment. The trajectory matching technology based on Hidden Markov Models is the core module of the positioning method, which will be discussed in detail in the next section.

### 3.1 Robot Motion Model and Dead Reackoning Principle

In this paper, a four-wheel differentially driven wheeled mobile robot is used to study the positioning problem of autonomous mobile robots in indoor environment. The driving motor is a DC motor. The two driving motors on one side are connected in reverse parallel and use the L298N motor driving module. Control the DC motor, and the mobile robot adopts the Raspberry Pi second-generation B-board (B.V1.2) as the main control chip. According to the driving mode of the mobile robot, the movement models of the two wheels on each side of the wheeled robot are the same. Therefore, the movement model of the mobile robot can be simplified to a left and right two-wheel differential driving mode. Figure 1(a) shows the mobile robot. The simplified model, where (x, y) is the position coordinate of the mobile robot in the global coordinate system, which is the included angle between the mobile robot and the true north direction, ie the heading angle.

Dead reckoning technology is widely used in the positioning of indoor autonomous mobile robots, and is particularly suitable for short-range positioning. The autonomous mobile robot used in this article has a built-in digital compass, three-axis accelerometer and gyroscope. The digital compass gives the initial attitude of the mobile robot. The accelerometer and the gyroscope can measure the

movement acceleration and rotation angular velocity of the mobile robot, and then pass it. Integral operation can calculate the distance and heading change of the mobile robot, and then analyze the latest position and attitude of the mobile robot.

3.2 Architecture Overview

The overall architecture of the AiFiMatch system is shown in Fig. 1. The input to the system is a indoor floor plan and time-stamped sensor data including motion data and Wi-Fi fingerprint. AiFiMatch utilizes embedded sensors of a smartphone to simultaneously collect motion data and Wi-Fi RSS values while the indoor floor plan can be manually entered or generated by crowdsourcing based indoor map construction algorithm.
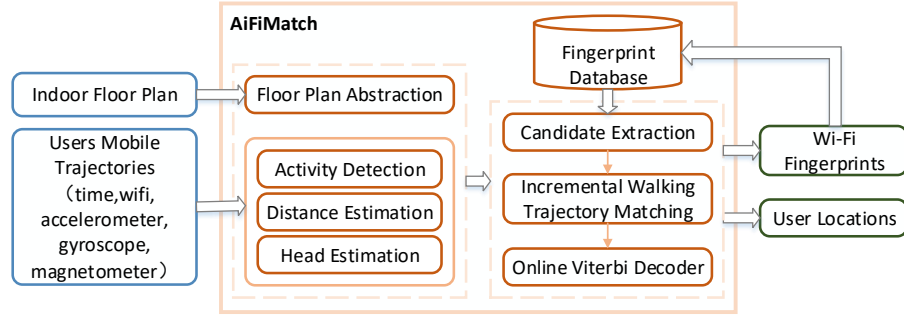
Fig. 1. AiFiMatch System Architecture.

Firstly, AiFiMatch starts by indoor floor plan abstraction to create a directed graph to facilitate the map matching model establishment. Secondly, the system estimates the walking distance and heading direction with data of motion sensors. Finally, location-related activities are detected with the pre-trained decision tree, and then the pedestrian's trajectory data including motion data and Wi-Fi RSS values are divided into trajectory subsets sequence.

The directed graph of indoor floor plan and walking trajectory subsets sequence are then passed to the HMM based map matching module whose outputs include the pedestrian position estimations and Wi-Fi fingerprints. These Wi-Fi fingerprints can be used to update indoor radio map. Our HMM-based map matching module contains three sub-modules: Candidate Extraction, Incremental Walking Trajectory Matching and an Online Viterbi Decoder. The Candidate Extraction module determines the candidate road segments from the directed graph of indoor floor plan that satisfy the spatial and Wi-Fi signal conditions. The module takes into account the errors of pedestrian heading direction estimation and the previous bound Wi-Fi fingerprint. The Incremental Walking Trajectory Matching module integrates a number of modifications to the standard HMM based map matching algorithm to take walking trajectory subsets

sequence into account as well as the detected activities to enhance the accuracy of the estimated indoor road paths. Finally, the Online Viterbi Decoder uses dynamic programming to efficiently determine the most probable indoor road segments sequence.

3.3 Preprocessing

In the balance of this section, we give the details of three preprocessing components and leave the details of the map matching and Wi-Fi enhancement algorithm to the next section.

PDR Implementation Based on Smartphone: PDR is a positioning scheme that derives the current position by adding the estimated displacement to the previous one based on motion sensors carried by smartphones. In PDR scheme, the pedestrian's position is usually updated by the step and the heading direction during one step is supposed to be unchanged. The pedestrian's position after $k-1$ steps can be denoted as (x_{k-1}, y_{k-1}). The k-th step length and heading direction are denoted as sl_k and θ_k, respectively. Then, the pedestrian's position after k steps can be obtained by the following equation:

$$\begin{pmatrix} x_k \\ y_k \end{pmatrix} = \begin{pmatrix} x_{k-1} + sl_k \cdot \sin\theta_k \\ y_{k-1} + sl_k \cdot \cos\theta_k \end{pmatrix} \tag{1}$$

The heading direction is measured by the fusion of gyroscope and magnetometer. The step length and step count estimate approach is out of scope of this paper and can be found in [12].

Activity Detection: In this paper, AiFiMatch considers four types of pedestrian activities: stationary, normal walking, turning at a corner (left or right turn) and turning around (U-turn), which would occur on the flat ground. The decision tree for activity detection and signal features of each activity is shown in Fig. 2a. The top level separates walking and stationary based on the variance of the accelerometer. With the help of gyroscope, the second level uses the rotation rate to separate the normal walking and turns while the third level separates the U-turn case from the left or right turn case based on the rotation angle during the turning. Three participants with different brands of smartphones were asked to complete three activities (U-turn, left and right turn) in our experimental environment. The sample size of each activity was 60 traces. Fig. 2b shows an example for turning activity detection and the activity detection result is summarized in Table 1.

Table 1. Confusion Matrix of Activity Detection

Activity Type	Left Turn	Right Turn	U-turn	No Type
Left Turn	58	0	0	2
Right Turn	0	59	0	1
U-turn	0	0	60	0

(a) *Decision Tree for Activity Detection*


(b) *Turning Activity Detected Example*
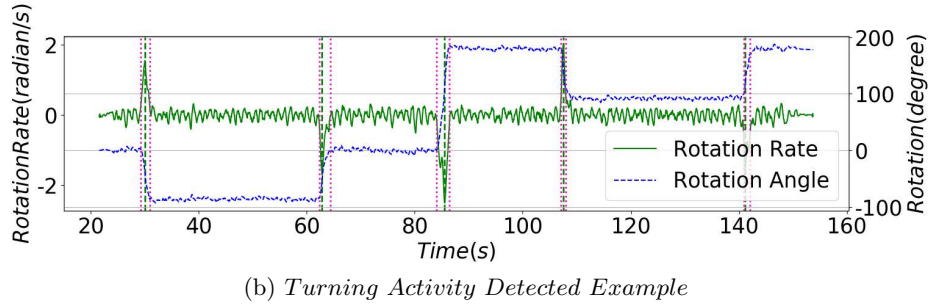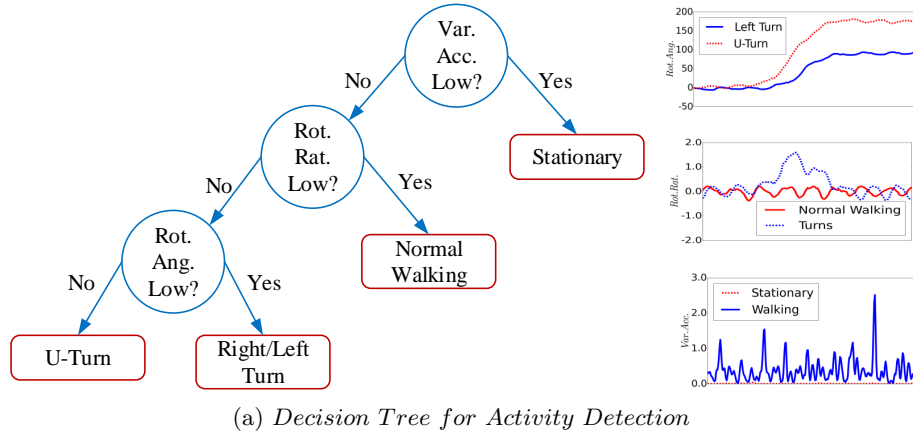
Fig. 2. Decision Tree for Activity Detection and An Example

Indoor Floor Plan Abstraction: In the indoor environment, there are many activity-relative locations such as corridor corners and entrances of rooms where pedestrians may perform different activities other than normal walking. These special locations divide the indoor roads into segments. With road segments as nodes in the form of *(coordinate of first endpoint (x_1, y_1), accessible direction of first endpoint (φ_1), coordinate of second endpoint (x_2, y_2), accessible direction of second endpoint (φ_2))*, the activity type from one road segment to another as directional edges in the form of *(Activity Type (AT))*, the indoor floor plan would be represented as a direction graph. Fig. 3 shows an example of indoor floor abstraction.

4 HMM based Map Matching Algorithm

Through this section, we detail the HMM-based map matching module of AiFi-Match system. We first start by providing the novel HMM model and the details of its components. Then, we describe the Wi-Fi enhancement algorithm for our proposed HMM model.
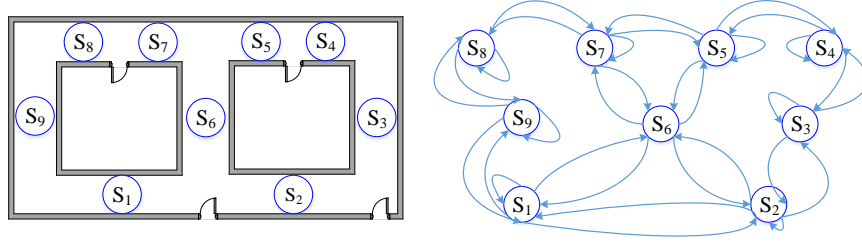
Fig. 3. Floor Plan Abstraction Example.

4.1 Hidden Markov Model

Taking computing power and energy limit of a smartphone into account, our proposed map matching algorithm select HMM to match the pedestrian's trajectory subsets sequence to the directed graph of indoor floor plan. A HMM can be represented as $\lambda = (S, V, A, B, \pi)$, where:

1) $S = \{s_1, s_2, s_3, \ldots, s_N\}$ is the set of possible states and $N = |S|$. In our case, each state represents an indoor road segment, that is, a node of the directed graph. Therefore, a state s is represented by the ordered tuple in the form of $(id, x_1, y_1, \varphi_1, x_2, y_2, \varphi_2)$, where id is the identification of road segment, x_1, y_1, φ_1, x_2, y_2 and φ_2 are different attributes of node of the directed graph, respectively. $s.leng$ denotes the length of road segment and can be obtained from the coordinates of two endpoints of this road segment. *Note that two or more road segments connected by a straight line can form new states (Fig. 4).*

2) $V = \{v_1, v_2, v_3, \ldots, v_M\}$ is the set of observations from the model and $M = |V|$. For each walking trajectory of a pedestrian, PDR technique gives displacement and heading direction estimations. However, due to the interference of many kinds of metal materials to the magnetic field in buildings, the heading direction estimation has a large error. Therefore, in our case, each observation is a displacement estimation and is represented by *(dist)*.

3) $A = \{a_{ij}\}$ is the state transition probability distribution, where $a_{ij} = p\{q_{t+1} = s_j | q_t = s_i\}, i, j \le N$, where q_t denotes the state at time t. In our case, the transition probability a_{ij} is the probability of walking to the next road segment $s_{j,t}$ given the current road segment is $s_{i,t-1}$. Intuitively, for probable transition between two road segments, the pedestrian's activity should match the activity type between the same two segments. Therefore, given the detected activity of a pedestrian $Ped_{AT}(t)$ at time t and the activity type between two segments $Seg_{AT}{}^{ij}$, AiFiMatch models this intuition by the equation 2, where $p(Ped_{AT}(t)|Seg_{AT}{}^{ij})$ can be found in confusion matrix.

$$p(s_{j,t}|s_{i,t-1}) = p(s_j|s_i, Ped_{AT}(t)) = p(Ped_{AT}(t)|Seg_{AT}{}^{ij}) \qquad (2)$$

4) $B = \{b_i(k)\}$ is the observation probability distribution in state i, where $b_i(k) = p\{o_t = v_k | q_t = s_i\}, 1 \le i \le N, 1 \le k \le M$ and o_t, q_t are the observation and state at time t, respectively. Observation probabilities also called emission

probabilities represent the likelihood that a measurement resulted from a given state. In our case, given a displacement observation $v_k.dist(t)$, there is an emission probability $p(v_k.dist(t)|s_i.leng)$ for each candidate road segment s_i, where $s_i.leng$ denotes the length of road segment s_i which can be obtained from the coordinates of two endpoints of this road segment. If a pedestrian has passed the complete road segment s_i, AiFiMatch models the emission probability as a Gaussian Distribution:

$$f_1(v_k.dist(t), s_i.leng) = \frac{1}{\sqrt{2\pi}\sigma_d} e^{-\frac{(v_k.dist(t)-s_i.leng)^2}{2\sigma_d^2}} \tag{3}$$

where σ_d is the standard deviation of the measured displacement. Based on the distance calculation method of PDR, the displacement is in direction proportion to step length. Therefore, σ_d can be obtained based on the standard deviation of step length σ_s. Considering the situation that a pedestrian is walking on the road segment, all long enough candidate road segments should have the same probability, AiFiMatch models the situation using the equation as:

$$f_2(v_k.dist(t), s_i.leng) = \frac{1}{\sqrt{2\pi}\sigma_d} e^{-4.5}, v_k.dist(t) + 3\sigma_d \le s_i.leng \tag{4}$$

Hence, the final emission probability, $p(v_k|s_i)$, is modeled as:

$$p(v_k|s_i) = f(v_k.dist(t), s_i.leng) = \begin{cases} \frac{1}{\sqrt{2\pi}\sigma_d} e^{-4.5}, v_k.dist(t) + 3\sigma_d \le s_i.leng \\ \frac{1}{\sqrt{2\pi}\sigma_d} e^{-\frac{(v_k.dist(t)-s_i.leng)^2}{2\sigma_d^2}}, otherwise \end{cases} \tag{5}$$

5) $\pi = \{\pi_i\}$ is the initial state distribution, where $\pi_i = p\{q_1 = S_i\}, 1 \le i \le N$. If the starting point or the first road segment is known, the initial state distribution is 1 since the first road segment is the only candidate; otherwise, all candidate road segments are selected by the *Candidate Extraction* module and the initial state distribution is uniform in all candidates. Let S_c denote the set of all candidates, and the initial state distribution is re-estimated after each step by the equation 6 until the first location-related activity is detected.

$$\pi_{i,t} = \pi_{i,t-1} \cdot f(v_k.dist(t), s_i.leng), s_i \in S_c \tag{6}$$

4.2 Candidate Extraction

AiFiMatch uses the heading direction of a pedestrian and Wi-Fi fingerprint to select the candidate states. Here, we describe the extraction algorithm by heading direction and leave the Wi-Fi fingerprint extraction algorithm to the *Wi-Fi Enhancement* Section.

Intuitively, the heading direction of a pedestrian should match the direction of road segment. Therefore, AiFiMatch models the direction difference by the following function to select the candidates:

$$g_1 = g_1(Ped_{dir}, s) = \begin{cases} 1, if \ |Ped_{dir} - s.\varphi_j| < H_{TH}, j = 1, 2 \\ 0, otherwise \end{cases} \tag{7}$$

where Ped_{dir} denotes the heading direction of a pedestrian, s is a state, H_{TH} is the threshold for candidate extraction, which is set to *55 (degree)* based on the experiments.

4.3 Wi-Fi Enhancement

Map matching offers an important source of information and an excellent way to improve the position estimations, especially in narrow corridors, but it has some limitations when the indoor environment presents symmetries that generates multiple hypotheses. Supposing that a pedestrian walks along this segments sequence (s_3, s_4, s_5, s_6), as shown in Fig. 4, for the given observable states, it is directly to be seen that the segments sequence (s_6, s_7, s_8, s_9) may have almost the same matching probability as the actual one. In this situation, traditional HMM based map matching algorithm fails [7].
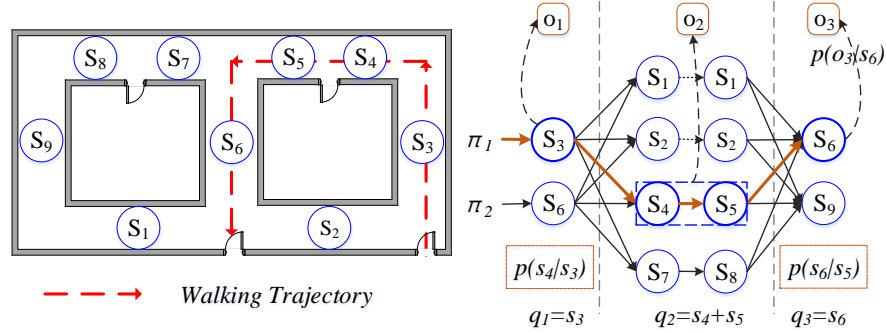

Fig. 4. Illustration of the Viterbi Decoding.

AiFiMatch introduces Wi-Fi dissimilarity to distinguish multiple hypotheses due to the symmetry of building structure. After AiFiMatch determines a pedestrian's walking trajectory by the Online Viterbi Algorithm [13], the pedestrian's positions can be derived by PDR using the two endpoints of determined segments as the starting point. Therefore, Wi-Fi fingerprints collected synchronously during the pedestrian walking can be bound to the corresponding positions by time alignment and then update the Wi-Fi fingerprint database of indoor environment where the pedestrian is located. In order to improve the HMM based map matching algorithm described above, the paper not only binds Wi-Fi fingerprints to physical positions but also road segments. All Wi-Fi fingerprints of one road segment are sorted in the road segment's direction. For two fingerprints f_x, f_y, and MAC address set of their Access Points X, Y, define dissimilarity J_δ between f_x, f_y as follows:

$$J_\delta(f_x, f_y) = 1 - J(X, Y) = \frac{|X \cup Y| - |X \cap Y|}{|X \cup Y|} \qquad (8)$$

where equation 8 is known as Jaccard distance. With this dissimilarity of two fingerprints, we define the similarity function $L(S_a, S_b)$ between two fingerprint

sequences $S_a = (f_1^a, f_2^a, \cdots, f_w^a)$, $S_b = (f_1^b, f_2^b, \cdots, f_k^b)$:

$$L(S_a, S_b) = \min(\frac{1}{w} \sum_{t=1}^{w} J_\delta(f_t{}^a, f_{i+t}{}^b), 0 \le i \le k - w,$$
$$\frac{1}{w} \sum_{t=1}^{w} J_\delta(f_t{}^a, f_{j-t}{}^b), w + 1 \le j \le k + 1) \tag{9}$$

where $w = |S_a|$, $k = |S_b|$ and $2 < w < k$. Given segment candidate set G and the set R of all road segments bound with Wi-Fi fingerprints sequence, Finally, AiFiMatch models the dissimilarity of two Wi-Fi fingerprint sequences and the different stages of fingerprint database by the following function to select the candidates:

$$g_2 = g_2(S_{ped}, S_i) = \begin{cases} 0, if\, i \neq \arg\min_{j}\{L(S_{ped}, S_j), S_j \in G, G \subseteq R\} \\ 0, if\, \min\{L(S_{ped}, S_i),\ S_i \in G \cap R\} > d \\ 1, otherwise \end{cases} \tag{10}$$

where d is the threshold to distinguish two Wi-Fi fingerprint sequences.

A pilot study is conducted to evaluate the dissimilarities between two fingerprint sequences of same and different road segments. A prototype installed in three smartphones (360 N5, OPPO R827T and Xiaomi 3W) is used to collect Wi-Fi RSS values at 1Hz. Each participant was asked to walk along two symmetric road segments in two directions. The participant with 360 N5 smartphone was asked to repeat 12 times while the others were asked to repeat 6 times. Among them, 6 Wi-Fi data collected by 360 N5 smartphone were used as Wi-Fi fingerprint database, while other Wi-Fi data were used as online Wi-Fi fingerprint sequences. The maximum and minimum dissimilarities between fingerprint sequences of same and different road segments are shown in Fig. 5. From Fig. 5, with increases in the length of Wi-Fi fingerprint sequence w, the dissimilarity between two sequences of same segments gradually decreases, while the dissimilarity between two sequences of different segments gradually increases. Considering the device diversity, in order to avoid missing the correct alternative segments, the minimum value of the threshold d is 0.773.

4.4 Optimal State Sequence Estimation

Once the HMM parameters are estimated, we can user the Viterbi algorithm to get the most probable hidden states sequence $Q = (q_1, q_2, ..., q_k)$ for a given observation sequence $O = (o_1, o_2, ..., o_k)$. A Viterbi variable $\delta_t(i)$ is defined by equation 11 to represent that, at time t, the HMM model reaches the hidden state s_i along a certain path and outputs the maximum probability.

$$\delta_t(i) = max\{p(q_1, q_2, ..., q_t = s_i, o_1, o_2, ..., o_t | \lambda)\} \tag{11}$$

At time $t + 1$, the maximum probability reaching the hidden state s_j can be recursively derived from the Viterbi variable at time t by the following equation.

$$\delta_{t+1}(j) = [\max_i\{\delta_t(i) \cdot p(q_{t+1} = s_j | q_t = s_i)\}] \cdot p(o_{t+1} | q_{t+1}), 1 \le t \le k \tag{12}$$
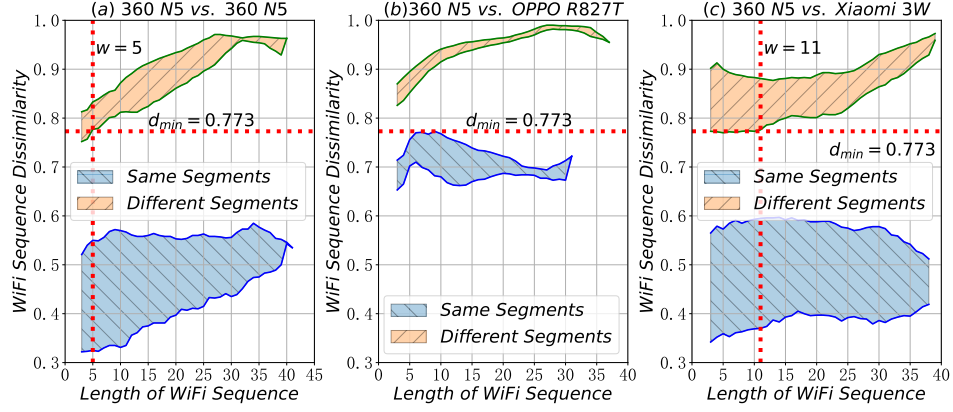
Fig. 5. Dissimilarity between Two Wi-Fi Fingerprint Sequences from Multiple Devices.

For AiFiMatch to operate in real-time, it cannot wait until the whole sequence is available. Hence, The online Viterbi algorithm [14] is applied to compute the maximum likelihood sequence of states using dynamic programming. AiFiMatch users Viterbi algorithm in an incremental manner. Every time a new step of a pedestrian is detected, the HMM parameters are calculated for the new introduced sensors data and the associated candidate states. Fig. 4 illustrates the proposed HMM model Viterbi decoding. The decoded sequence is colored in blue.

During the beginning process of the algorithm, if the number of states is too small, the most probable states sequence is not always the correct one, especially when the starting point of pedestrian is unknown. Therefore, we give a criteria by the equation $c = p_{fir}/p_{\text{sec}}$ to determine the status of this algorithm, where p_{fir} is the highest probability of the states sequence and p_{sec} is the second highest probability of the states sequence. We set a threshold, if c is greater than or equal to the threshold, we choose the Viterbi decoding result. The selected hidden states sequence represents the pedestrian's passed indoor road segments.

AiFiMatch algorithm is summarized with the pseudocode in Algorithm 1.

Algorithm 1 AiFiMatch online map matching algorithm

Input: Sensor data up to current time t: $data_{1:t}$
Input: Abstract Indoor Floor Plan: fp
Input: Hidden State Sequence and Its Length up to last time $t-1$: $S_k(t-1)$, k
Input: Array of Viterbi variable and backward pointer: $VtbArr_k(t-1)$
Output: Pedestrian's position estimation at current time t: $Ped_{loc}(t)$
Output: Optimal state sequence estimation at current time t: $Q(t)$
 1: $Ped_{AT}(t), at \leftarrow activity\_detect(data_{1:t})$
 2: **if** $Ped_{AT}(t) \mathrel{!=} None$ **then**
 3:     $st \leftarrow at$
 4:     // Calculate the transition probability
 5:     $S(t) \leftarrow [\,]$
 6:     **for** $s_i \in S_k(t-1)$ **do**
 7:         // Select next segments according to fp
 8:         $S^i(t) \leftarrow next(fp, s_i)$
 9:         // Extract segments according to Wi-Fi data
10:         $S^{i,wifi}(t) \leftarrow extract(data_{st:t}^{wifi}, S^i(t))$

---

5 Evaluation

5.1 Environment Setup

In this section, we show our evaluation for the performance of AiFiMatch in real-world environment. We conduct experiments in the fifth floor of a teaching hall on campus. Approximately, we cover a $78.95m \times 56.40m$ floor plan, as shown in Fig. 9a. A prototype is implemented and installed on a 360 N5 Android version 6.0.1 smartphone, a OPPO R827T Android version 4.4.0 smartphone and a Xiaomi 3W Android version 6.0.1 smartphone. The prototype samples the accelerometer, gyroscope and magnetometer sensors at 50Hz and Wi-Fi at 1Hz. Three participants (two males and one female) were asked to complete the experiments. The participants were asked to walking along Trajectory $No.1$ (T_1), Trajectory $No.2$ (T_2) and Trajectory $No.3$ (T_3) in a constant speed. Each trajectory was repeated six times by all participants. In order to record the walking trajectories, all the participants' shoes were painted with colored powder. This offered the ground truth.

5.2 Performance of Map Matching

To evaluate AiFiMatch map matching performance, we start by showing the online positioning performance of AiFiMatch and basic PDR technique. After that, we analyze the effect of step length and heading errors on positioning accuracy. Then, we discuss the convergence performance as compared to an activity sequence-based map matching algorithm proposed by *Zhou et al* [7]. Finally, we show the offline positioning results when using AiFiMatch.

```
11:        for $s_j \in S^{i,wifi}(t)$ do
12:            $tr[i][j] \leftarrow p(s_j|s_i, Ped_{AT}(t))$
13:        end for
14:        $S(t).extend(S^{i,wifi}(t))$
15:    end for
16:    // Calculate the observation probability
17:    $z_t \leftarrow dead\_reckon(data_{st:t})$
18:    for $s_i \in S(t)$ do
19:        $ob[i] \leftarrow p(z_t|s_i)$
20:    end for
21:    $VtbArr_{k+1}(t) \leftarrow viterbi(VtbArr_k(t-1), ob[\,], tr[\,][\,])$
22:    $k \leftarrow k + 1$
23:    $S_k(t) \leftarrow S(t)$
24: else// Not a location-related activity
25:    $ob \leftarrow [\,], S_k(t) \leftarrow S_k(t-1)$
26:    // Update the observation probability
27:    $z_t = dead\_reckon(data_{st:t})$
28:    for $s_i \in S_k(t)$ do
29:        $ob[i] \leftarrow p(z_t|s_i)$
30:    end for
31:    $VtbArr_k(t) \leftarrow viterbi(VtbArr_k(t-1), ob[\,])$
32: end if
33: // Optimal State Sequence
34: $Q(t) \leftarrow None$
35: if $check\_status(VtbArr^k(t)) = Convergence$ then
36:    $Q(t) \leftarrow viterbi\_decode(VtbArr^k(t))$
37: end if
38: // Update the location estimation according to $Q(t)$
39: $Ped_{loc}(t) \leftarrow update\_location(data_{st:t}, Q(t))$
```

Online Positioning Performance: Euclidean distance between the estimated position and the ground truth is introduced to indicate positioning accuracy. Fig. 6 shows the online positioning results of all trajectories for basic PDR algorithm with known initial point and AiFiMatch without known initial point. Here, the mean position of all segment candidates' starting point is taken as initial point of AiFiMatch algorithm. Generally, for one trajectory, the greater the traveled distance, the larger the positioning error of PDR due to cumulative errors, Fig. 6 shows the trend. However, after passing a number of steps, the traveled segment sequence determined by AiFiMatch even without known the initial point and the cumulative errors are eliminated successfully.

Influence of Step Length and Heading direction Errors: The major errors of PDR techniques are mainly from step length and heading direction errors. Therefore, we analyze the influence of step length and heading direction errors to the average online positioning errors. Both of them are supposed to follow the Gaussian distributions with a mean zero. σ_s denotes the standard deviation of step length estimation and σ_φ denotes the standard deviation of heading direction estimation. When σ_s changes from 0.1 m to 0.5 m, σ_φ is set
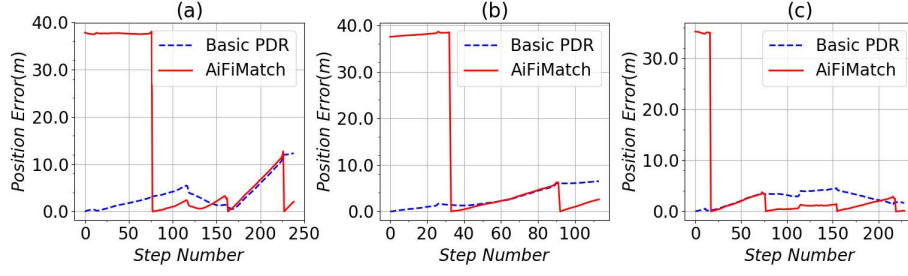
Fig. 6. Online Positioning Results for Each Trajectory. (a) T_1. (b) T_2. (c) T_3

to 10 degree. When σ_φ changes from 10 degree to 50 degree, σ_s is set to 0.1 m. Fig. 7 shows the results of all trajectories with different errors. The performance of AiFiMatch in all error cases is superior to the basic PDR technique. From Fig. 7 *(a)-(c)*, with increasing step length error, the average online positioning errors increases gradually. The same trend is shown in Fig. 7 *(d)-(f)*, reflecting the influence of heading direction errors on the average online positioning errors. T_1 and T_3 have five location-related activities while T_2 has only two. As a result, given the same heading direction errors, the online positioning performance of AiFiMatch is more robust to T_1 and T_3 than T_2, which can also be seen from Fig. 7 *(d)-(f)*.
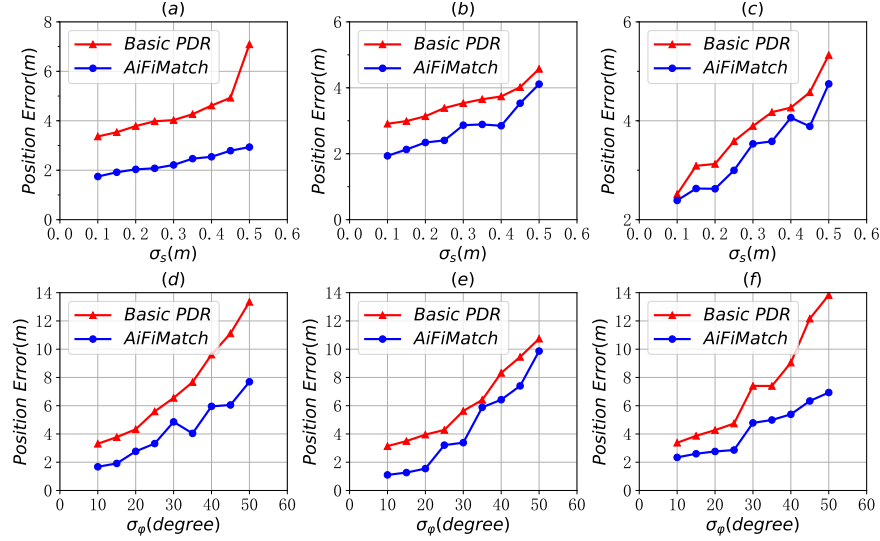
Fig. 7. Online Positioning Errors for Each Trajectory with Given Standard Deviations of Step Length and Heading Direction. (a) T_1. (b) T_2. (c) T_3. (d) T_1. (e) T_2. (f) T_3.

Convergence Speed: We compare the performance of AiFiMatch algorithm in terms of convergence to map matching algorithm proposed by *Zhou et al.*
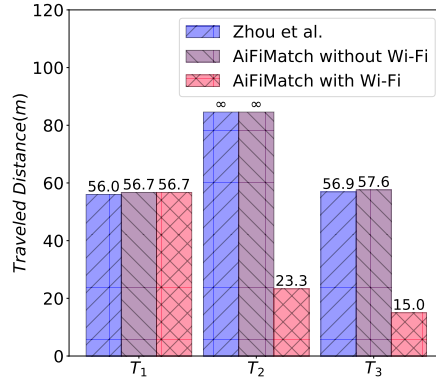
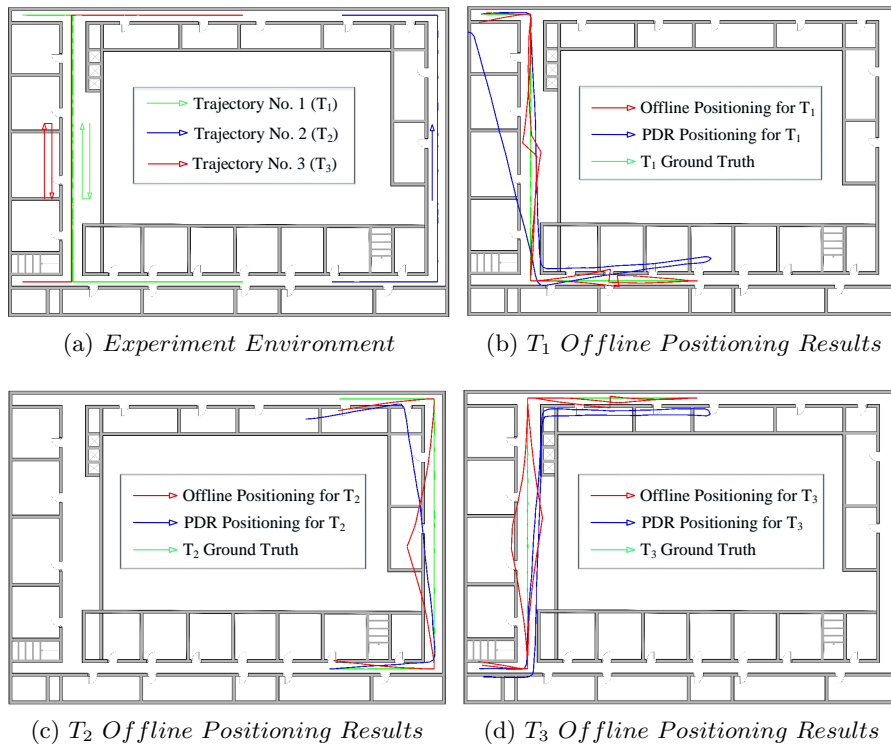**Fig. 8.** Distance traveled before convergence for each trajectory.

Distance traveled before converging to a unique states sequence reflects the convergence speed. The greater the traveled distance, the slower is the convergence speed. Fig. 8 shows the traveled distance before convergence for *Zhou et al*, AiFiMatch without Wi-Fi enhancement and AiFiMatch with Wi-Fi enhancement. At the initial stage, fingerprint database is empty (T_1), AiFiMatch and *Zhou et al.* both successfully converge even without known the initial point. AiFiMatch without Wi-Fi enhancement and map matching algorithm proposed by *Zhou et al.* fail to converge (∞ means T_2 cannot be converged.) due to the symmetry of building structure. However, with the help of Wi-Fi signal, T_2 reaches convergence quickly. Mostly, with Wi-Fi enhancement, the traveled distance is much shorter than without Wi-Fi enhancement(T_2, T_3).

Offline Positioning Performance: The offline positioning results are derived retrospectively after matching the walking trajectory of a pedestrian to a unique road segments sequence by AiFiMatch. Fig. 9 shows the tracking trajectories and some details are summarized in Table 2. AiFiMatch system tracked pedestrians' trajectories accurately in the experiment environments and the mean error of the offline positioning is about 1.24 m.

Table 2. Evaluation Results

Trajectory No.	Length (m)	Mean Error (m)	Step Number	Detected Step	Activity Number	Detected Activity
1	176.1	1.09	238	**240**	5	5
2	84.6	1.65	114	114	3	3
3	175.9	1.19	228	**227**	5	5

(a) *Experiment Environment*

(b) *T_1 Offline Positioning Results*

(c) *T_2 Offline Positioning Results*

(d) *T_3 Offline Positioning Results*

Fig. 9. Environment setup and offline positioning results for each trajectory

6   Conclusion

In order to solve the difficult problem of positioning of autonomous mobile robots in dark complex building corridors, subway station corridors, or underground mines after a sudden accident such as a fire, this paper proposes an indoor autonomous mobile robot tracking based on hidden Markov model for map matching assistance. And positioning methods. In the structured indoor environment, this method uses the recognition of position-related gestures to match the motion trajectory of the mobile robot to the indoor map. Compared with the traditional dead reckoning technology, the method can significantly reduce the influence of cumulative error on the positioning accuracy, and for certain The degree of heading angle and acceleration errors are robust. This method does not rely on cameras. It uses only accelerometers, gyroscopes, magnetometers, and known indoor maps installed in autonomous mobile robots to achieve fusion positioning of dead reckoning and map matching techniques, even when the starting point is unknown. In addition, the effective tracking and positioning of mobile robots can also be achieved, with features such as simple deployment, low manufacturing cost, and easy operation.

References

1. Woodman, O., Harle, R.: Pedestrian localisation for indoor environments. In: Proceedings of the 10th international conference on Ubiquitous computing, ACM (2008) 114–123
2. Bao, H., Wong, W.C.: An indoor dead-reckoning algorithm with map matching. In: Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International, IEEE (2013) 1534–1539
3. Rai, A., Chintalapudi, K.K., Padmanabhan, V.N., Sen, R.: Zee: Zero-effort crowdsourcing for indoor localization. In: Proceedings of the 18th annual international conference on Mobile computing and networking, ACM (2012) 293–304
4. Xiao, Z., Wen, H., Markham, A., Trigoni, N.: Lightweight map matching for indoor localisation using conditional random fields. In: Proceedings of the 13th international symposium on Information processing in sensor networks, IEEE Press (2014) 131–142
5. Bobkov, D., Grimm, F., Steinbach, E., Hilsenbeck, S., Schroth, G.: Activity recognition on handheld devices for pedestrian indoor navigation. In: 2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE (2015) 1–10
6. Guo, S., Xiong, H., Zheng, X., Zhou, Y.: Activity recognition and semantic description for indoor mobile localization. Sensors **17**(3) (2017) 649
7. Zhou, B., Li, Q., Mao, Q., Tu, W., Zhang, X.: Activity sequence-based indoor pedestrian localization using smartphones. IEEE Transactions on Human-Machine Systems **45**(5) (2015) 562–574
8. Abdelnasser, H., Mohamed, R., Elgohary, A., Alzantot, M.F., Wang, H., Sen, S., Choudhury, R.R., Youssef, M.: Semanticslam: Using environment landmarks for unsupervised indoor localization. IEEE Transactions on Mobile Computing **15**(7) (2016) 1770–1782

9. Zhou, B., Li, Q., Mao, Q., Tu, W., Zhang, X., Chen, L.: Alimc: Activity landmark-based indoor mapping via crowdsourcing. IEEE Transactions on Intelligent Transportation Systems **16**(5) (2015) 2774–2785

10. Hoang, M.K., Schmalenstroeer, J., Drueke, C., et al.: A hidden markov model for indoor user tracking based on wifi fingerprinting and step detection. In: 2013 Proceedings of the 21st European Signal Processing Conference (EUSIPCO), IEEE (2013) 1–5

11. Radu, V., Marina, M.K.: Himloc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced wifi fingerprinting. In: 2013 International Conference on Indoor Positioning and Indoor Navigation (IPIN), IEEE (2013) 1–10

12. Wang, H., Sen, S., Elgohary, A., Farid, M., Youssef, M., Choudhury, R.R.: No need to war-drive: Unsupervised indoor localization. In: Proceedings of the 10th international conference on Mobile systems, applications, and services, ACM (2012) 197–210

13. Aly, H., Youssef, M.: semmatch: Road semantics-based accurate map matching for challenging positioning data. In: Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems, ACM (2015) 5

14. Bloit, J., Rodet, X.: Short-time viterbi for online hmm decoding: Evaluation on a real-time phone recognition task. In: ICASSP, IEEE (2008) 2121–2124