Pervasive and Mobile Computing 16 (2015) 251–267

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Multi-scale Conditional Random Fields for first-person activity recognition on elders and disabled patients

Kai Zhan [a,*], Steven Faux [c,b], Fabio Ramos [a]

[a] School of Information Technologies, University of Sydney, Sydney, Australia
[b] Faculty of Medicine, University of New South Wales, Sydney, Australia
[c] Sacred Heart Rehabilitation Service, St Vincent's Hospital, Sydney, Australia

ARTICLE INFO

Article history:
Available online 5 December 2014

Keywords:
Activity recognition
Feature classification
Graphical model
First-person
Feature extraction
Computer vision

ABSTRACT

We propose a novel pervasive system to recognise human daily activities from a wearable device. The system is designed in a form of reading glasses, named 'Smart Glasses', integrating a 3-axis accelerometer and a first-person view camera. Our aim is to classify subject's activities of daily living (ADLs) based on their vision and head motion data. This ego-activity recognition system not only allows caretakers to track on a specific person (such as disabled patient or elderly people), but also has the potential to remind/warn people with cognitive impairments of hazardous situations. We present the following contributions: a feature extraction method from accelerometer and video; a classification algorithm integrating both locomotive (body motions) and stationary activities (without or with small motions); a novel multi-scale dynamic graphical model for structured classification over time. In this paper, we collect, train and validate our system on two large datasets: 20 h of elder ADLs datasets and 40 h of patient ADLs datasets, containing 12 and 14 different activities separately. The results show that our method efficiently improves the system performance (F-Measure) over conventional classification approaches by an average of 20%–40% up to 84.45%, with an overall accuracy of 90.04% for elders. Furthermore, we also validate our method on 30 patients with different disabilities, achieving an overall accuracy up to 77.07%.

1. Introduction

The challenges associated with an ageing population and pressures on carers and nursing services, are opening unprecedented opportunities for pervasive computing. Such systems can improve the quality of life of the aged particularly by making daily activities of living safer and easier to complete. However, most of the current approaches of activity tracking still rely on human, for which can be very costly and time-consuming. For example, at home, disabled subjects generally require supervision of caretakers. In a hospital, patients are often monitored and cared by health professionals. In other cases, subjects are asked to manually update and report their activities by themselves. Both cases have significant inefficiencies in terms of cost, accuracy, scope, coverage and privacy.

A central requirement of pervasive systems is to automatically recognise human activities over time, warning patients of hazardous situations, or reinforcing home-based activities and therapies. According to their activities, the system can be pre-programmed for assistance purposes. For example, patients with memory loss or cognitive disorders can be reminded

* Corresponding author.
E-mail addresses: kai.zhan@sydney.edu.au (K. Zhan), sfaux@stvincents.com.au (S. Faux), fabio.ramos@sydney.edu.au (F. Ramos).

http://dx.doi.org/10.1016/j.pmcj.2014.11.004
1574-1192/

Fig. 1. Senior patients wearing the Smart-Glasses prototype.

to take medicines after having meals or to pick up their walking stick when venturing outdoors. In addition, it is desirable that monitoring solutions also provide online interfaces for patients and carers to remotely access the patients' status, and to derive better treatments in a more effective manner.

Activity recognition is a relatively simple task for observers when watching the activities of other people. With years of social experience, the capacity of understanding and expressing the activities and intentions of others is notable. The recognition of activities by humans involves a series of tasks, from sensing to interpreting which can be complicated and challenging for an automatic system. For an instance, a man in a sitting-posture, on an observer perspective, might be watching TV, reading or just resting. Context generally help with this distinction. However, for a wearable device, it is a difficult challenge to tell what exactly the person is doing despite increasing amounts of sensory data. The main reason for this is that humans learn to interpret sensory data from past experiences, making new judgements significantly easier, in a sensing and learning process. The interpretation of sensory data is accomplished naturally using either the environment or other people to provide feedback. Ultimately, we would also prefer to have a machine with the same capabilities, i.e., automatically learning to interpret sensory data from past experiences and feedback received. The main objective of automatic activity recognition systems is to develop such algorithms combining elements of statistical learning with sensor technologies.

This paper focuses on automatic activity recognition and introduces a novel 'Smart Glasses' system to recognise complex daily living activities, which can be categorised into two major groups in terms of their motion magnitude: *Locomotive* and *Stationary*. A locomotive activity is defined as an activity involving high energy, with specific body movements, such as walking or climbing stairs. A stationary activity involves less or no motion, such as reading books or watching TV.

In this paper, we introduce an automatic activity recognition system integrating accelerometers and a first-person view camera embedded in conventional glasses. The current prototype consists of a smart phone (Android OS) attached on top of safety goggles as shown in Fig. 1. The device collects videos and 3-axis acceleration data. Both are synchronised and collected in parallel. With this data we develop our classification approach that includes the following contributions:

Feature extraction: We carefully select a number of activities following healthcare professionals' directives. Some of the activities have identical static postures. This makes features from the accelerometer less important. Therefore, in this paper, we use the video motion feature as a complementary element. This allows the system to track motion flow from consecutive egocentric images in order to improve the system performance. We design separate feature extraction algorithms for both accelerometer and video data, detailed in Sections 3.2 and 3.3.

Feature integration: From a series of experiments, we select the best classifier and settings for each set of features (from accelerometers and camera), and separate the local classification task into two categories, each associated with weighting parameters obtained during the training process. This allows the model to combine the best set of feature for the different activities.

Multi-scale graphical model: We develop a structured classification approach based on Conditional Random Fields (CRFs) to capture the multi-scale context in a sequence of activities. This model can help to predict user's activities at different temporal scales even when the local classification is significantly noisy or ambiguous. For example, when sitting, reading or drinking, there is a period with no motion features being detected from both accelerometer and camera features but the activity can be recognised from the context.

System validation on elders and patients: At the end of this study, we validate our system's performance on two groups: healthy elderly people and a mixed patient population with recent disabilities. The healthy elderly group includes 5 healthy subjects with an average age of 68 years old. The disabled population group of 30 people, ranged in age from 34 to 89 years old, each is suffering from a variety of disabilities which were loosely categorised as musculoskeletal, neurological and general weakness. Their conditions are detailed in Table 5. We separately evaluate and compare our system performance with conventional approaches.

The paper is organised as follows. We describe related work on automatic activity recognition from wearable and environment sensors in Section 2. A detailed description of our method including feature extraction, feature classification

and the proposed structured classification with a probabilistic graphical model is presented in Section 3. Individual performance assessments for each component of our system is discussed in Section 4. Experiments with real deployments of the system in humans, including elderly and disabled patients are described and discussed in Section 5. Finally, conclusions and suggestions for future work are presented in Section 6.

2. Related work

The complexity involved in recognising daily activities has promoted a diverse range of research in the area, relying mostly on motion sensing and visual clues. Wearable accelerometers are a popular motion sensor often used to classify basic activities of daily living (ADLs). The accelerometer placement on the body is essential because the signals vary across different body parts during the activities [1]. Generally, accelerometers are simply attached to the body parts where the movements are being monitored. Previous approaches use thigh or ankle for gait studies [2–4], or wrist and joints to monitor the progress of diseases [5,6]. Moreover, occasional events such as falls [7] or stumbles [8] have also been studied with accelerometers. In alternative approaches movements from the entire body have also been analysed [9,10]. During common activities, the acceleration signals increase in magnitude and frequency from head to ankle, where the frequency of the signals in the limbs, feet or hands increase to up to 60 Hz [11]. This suggests that preferable parts to mount the accelerometers are in the main body, such as pelvis [12], chest [13] or head [14]. Recently, such techniques have been integrated into mobile phones [15–17], or deployed with additional sensors to improve accuracy, such as gyroscopes [18], microphones [19], and floor sensors [20].

Despite major achievements in activity classification from accelerometers, classification of high-magnitude motion, stationary activities (especially in identical static postures), still remain challenging due to the similarity of the acceleration signals. For example, activities performed when sitting such as writing, reading a book or watching TV are difficult to be detected from accelerometer signals, however, such activities can be more easily identified with first-person vision. Therefore, in our research, we integrate wearable accelerometers and first-person vision into the designed glasses, which not only avoids the high motion-frequency regions, but also monitors the head motion in conjunction with a vision recognition system.

Computer vision techniques are emerging rapidly within the activity recognition community. Using single or multiple images containing parts of the human body, the subjects' postures and motions can be estimated to predict the ongoing activity. Notable approaches using external cameras are described in [21–23]. These methods are mostly applied to surveillance purposes, and are often constrained to a particular region of the field of view. Recently, first-person view methods were introduced [24–26]. In this approach, cameras are embedded into a wearable device, providing a similar field of view as the subject. Image features are then extracted from objects or motion flow information. Object recognition approaches rely on machine learning techniques to identify objects in the scene. They are particularly useful in classifying stationary activities [26,27]. However, these approaches have difficulties to recognise activities without foreground objects (e.g. walking). Alternatively, motion-based methods have reasonable performance in identifying activities using optical flow features such as those involved in sports [28].

Techniques for human motion recognition are growing in importance within academia and industry due to their vital social and technological importance. Several review articles have provided comprehensive analyses of the field [29–31]. Algorithmically methods for human activity recognition can be divided into three main streams: (1) classification techniques on independent spatial–temporal features [18,32]; (2) methods based on temporal logic representing critical sequential constraints [33]; (3) algorithms for structured classification, such as Hidden Markov Models (HMMs) [34–36]. HMMs were initially applied to human activities in [37], and have since been widely used for activity recognition in recent years [38–40] due to their simplicity and flexibility to reason about temporal patterns in sequential data. However, as a generative model, HMMs make fairly strong independence assumptions, which explicitly ignore possible long-term dependences and relationships between observed features. Recently, discriminative models such as Conditional Random Fields (CRFs) are becoming popular for activity recognition due to their flexibility to represent more complex observations [41–43], and dependence structures [44–46]. In this paper, we will introduce a novel systematic model with a multi-scale CRF method for human activity recognition to address the limitations of previous approaches.

3. Methodology

3.1. System overview

Our algorithm can be described as a pipeline with three major steps: video and acceleration feature extraction, classification and structured prediction. As shown in Fig. 2, both video and accelerometer data are collected and processed in parallel, feeding features into separate classifiers. The classifier result is then transformed into a class probability vector. In the final step, these vectors are associated with a unary feature function which is then combined with pairwise functions in a graphical model to perform structured prediction. The final prediction is obtained after an inference procedure on the graphical model that takes temporal relationships into account, where the relative weights of unary and pairwise features are obtained through a learning procedure. We detail these steps below.

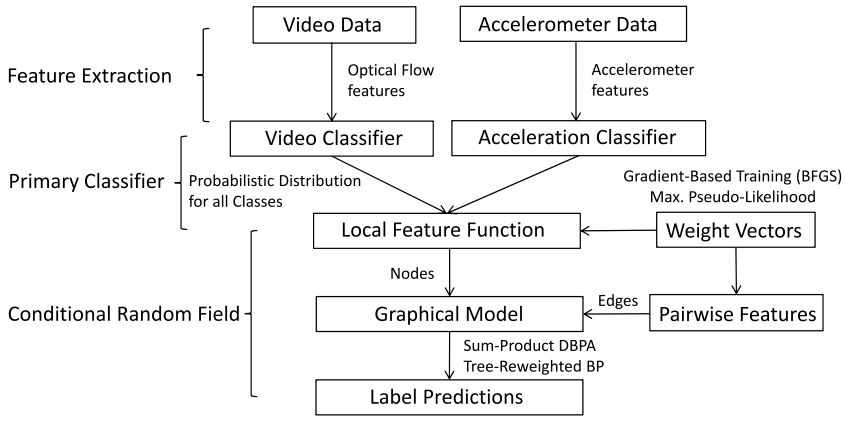

Fig. 2. Algorithm overview.

3.2. Acceleration features

Head acceleration is relatively complex because it is the result of a concatenation of motions of several body parts. Despite the complexity this type of signal is very valuable, providing information about the subject's body motion in addition to information about the head acceleration. For example, drinking water often results in head raising, hand-washing usually requires a head bowing. Here, we consider a wide range of activities covering various types of motions, including: locomotive or stationary, periodic or infrequent, body or hand activities. Generally, acceleration can be represented in both time and frequency domains. In this work we extract features from a sliding window containing a short period of time (the calculation of the window size is described in Section 4.1.1).

To cover the different types of activities, we conduct a comprehensive analysis of 13 time-domain and 20 frequency-domain features to be extracted from 3-axis accelerometer data. The *time-domain* features extracted directly from the 3-axis acceleration data, and fitted into the windows, the features include: mean, standard deviations, threshold crossing rate (of the mean value), energy of acceleration signals (sum of the square of the magnitude), correlation coefficient among 3 axes, number of maxima and minima (the peak values) and their mean values. The *frequency-domain* features are extracted with Fast Fourier Transform (FFT). We divide each feature window into 10 sub-bands. For each band, we obtain the magnitude and frequency of the peak value as our frequency feature. The cut-off threshold is set to 5 Hz since daily human activities are unlikely to require higher frequencies based on our ADLs database.

3.3. Video features

First-person video features can complement the body acceleration data. We extract motion features from egocentric images to monitor activities even when the subject is not moving. The video motion sensing focuses on the pixels flow from the entire image. This section briefly describes the approach in [39], also used in this work. It uses Lucas–Kanade based Dense Optical Flow [47] as the primitive features extraction method, obtained between pairs of consecutive images from the video. We downsize the image to 100×132 resolution in grey scale. For each frame, we partition the optical flow vectors into patches with i rows and j columns, then the flow vectors u (horizontal motion) and v (vertical motion) are the averaged flow magnitude over all patches. For each patch, the feature vector is described as

$$[\overline{u}, \overline{v}] = \frac{1}{M} \sum_{m=1}^{M} [u_m, v_m], \tag{1}$$

where $[\overline{u}, \overline{v}]$ is the averaged flow of one patch of frame and m represents the number of pixels from each patch. An example of "Drinking" and "Walking" is demonstrated in Fig. 3.

Following the dense optical flow, we conduct an average pooling process over n consecutive patches from all corresponding positions to obtain the feature vector with size $1 \times 2ij$. It contains horizontal and vertical motion information for the period covering n frames, written as

$$[\check{u}_{i,j}, \check{v}_{i,j}] = \frac{1}{N} \sum_{n=1}^{N} [\overline{u}^n_{i,j}, \overline{v}^n_{i,j}], \tag{2}$$

where $\overline{u}_{ij}$ and $\overline{v}_{ij}$ are the motion magnitudes for the patch i, j in both horizontal $\overline{u}_{ij}$ and vertical $\overline{v}_{ij}$ directions.

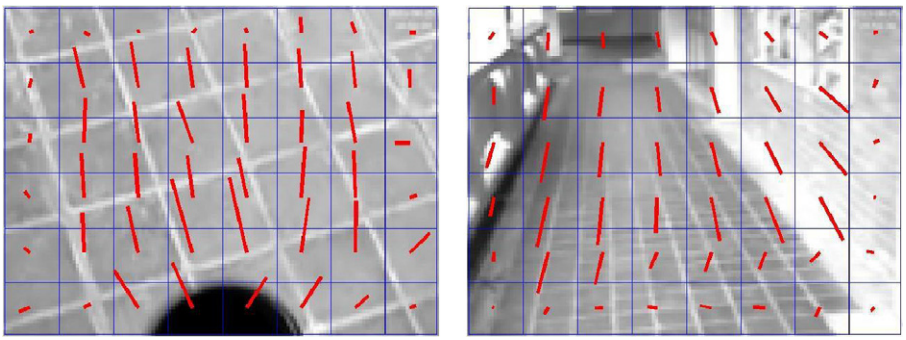

Fig. 3. The left image is a sample of Drinking, and the right image is Walking. The line patterns represent the motion flows in the video.

3.4. Classification

There are two levels of classification in our approach: local and structured. The former uses features directly extracted from raw sensor data and provides predictions for the current time window, independently of the past. The latter is based on the graph structure and takes into account temporal dependences. In this section, we focus on local classification. As shown in Fig. 2 and in the previous section, our system extracts local features from 3-axis head accelerometers and an egocentric camera. These features can differ drastically in terms of the magnitude and frequency of the measurements. Based on our previous work [39], we select the two most-reliable classifiers and compare them with the two sets of features described before. Our choice for the classifiers is based on their computational costs and scalability to handle high-dimensional data.

Support Vector Machine (SVM) is one of the most popular approaches for human activity recognition, especially from body acceleration [18,48–51]. SVMs are large-margin classifiers with strong generalisation properties, primarily designed for binary classification even though extensions to multi class exist [52,53]. We use a multi-class SVM based on an "One-Versus-One (OVO)" technique which fits binary sub-classifiers and obtain the final prediction through a voting process. We also estimate the class probabilities through a pairwise coupling method [54] for the structure classification stage. Three popular kernels were selected and compared: Linear, Polynomial and Gaussian Radial Basis Function (RBF).

Boosting was first introduced by Schapire et al. [55] in 1990. The algorithm produces an ensemble model by greedily adding weak learners trained on data points weighted by their classification error from previous rounds. Boosting significantly improves the accuracy of a base-level binary classifier (weak learner) and can learn complex nonlinear decision boundaries. Boosting has been widely and successfully applied to many fields [56–59]. Inspired by [60], we implement a LogitBoost algorithm which uses adaptive Newton steps to fit an adaptive symmetric logistic model with maximum likelihood. The LogitBoost algorithm is an improved version of the well known Adaboost [61], because it minimises the binomial deviance which causes misclassification over the minority of the observations [62]. In other words, LogitBoost assigns higher weights to misclassified data points which usually represent more important information. LogitBoost also provides a probability distribution of the class labels. In this work, we use a single-level decision tree (decision stump) as the weak learner.

3.5. Structured classification

3.5.1. Conditional random fields

One of the main drawbacks in conventional independent and identically distributed (i.i.d.) classification approaches is that context is not taken into account—classification is performed locally without considering any 'neighbours' in time or space. This can lead to mistakes that could be avoided otherwise. For example, a single sliding window might wrongly predict hand-washing due to strong image motion flow when the correct activity is walking. This could have been fixed had context been considered since a single hand-washing classification is unlikely to take place in a series of walking frames. We develop a structure classification approach for contextual activity recognition using conditional random fields (CRFs). CRFs are undirected graphical models designed for labelling sequences of data. It is a powerful tool in structured learning that allows us to model the correlations (through edges) between each defined pair of nodes in a graphical model, specifying a conditional probabilistic distribution over the query nodes given observed nodes [63].

For example, in the context of activity recognition, the nodes can be seen as containing local information of each time interval, while the edges are pairwise relations between consecutive intervals. Therefore, the order of the nodes can be explained as sequences of activities in time domain. A benefit of CRFs compared to a generative model, is that it models the conditional probability of hidden states directly. This provides more flexibility to define potential functions to capture more complex relationships. The CRF contains a normalising partition function that groups all potentials into a general format. In activity recognition, it allows us to integrate heterogeneous sensors into the graphical model seamlessly.

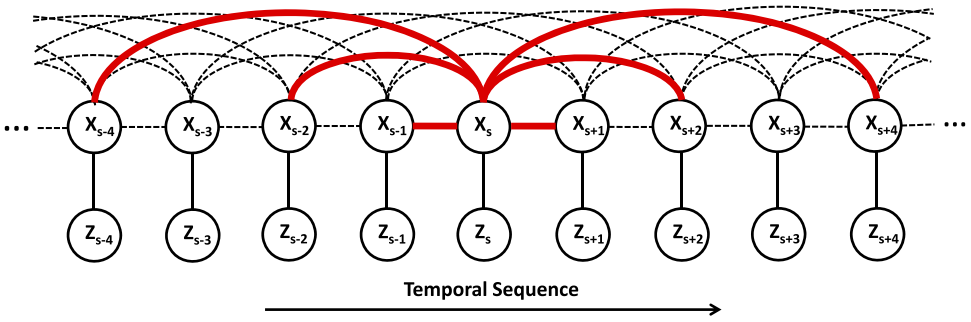

Fig. 4. Multi-scale CRF graph in temporal sequence. The bold edges represent the connection setting of '020305' for the node $\mathbf{x}_s$.

Theoretically, CRFs are a special case of Markov Random Fields (MRF). As described in [64,65], it models conditional distributions of the hidden nodes $\mathbf{x}$ given observations $\mathbf{z}$, written as $p(\mathbf{x}|\mathbf{z})$. Within the graph, the hidden nodes $\mathbf{x}$ are linked by edges following a predefined graph structure. Each fully connected subset of nodes (clique) $c \epsilon C$ is described by a non-negative clique potential function $\phi_c(x_c|z)$, which maps clique variables to a positive real number. A CRF distribution over the cliques can be written as,

$$p(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z})} \prod_{c \epsilon C} \phi_c(\mathbf{x}_c|\mathbf{z}), \tag{3}$$

where the partition function $Z(\mathbf{z})$ is expressed as

$$Z(\mathbf{z}) = \sum_{\mathbf{x}} \prod_{c \epsilon C} \phi_c(\mathbf{x}_c|\mathbf{z}). \tag{4}$$

The potentials $\phi_c(\mathbf{x}_c|\mathbf{z})$ are usually represented as log-linear combinations of feature functions $f_c(\mathbf{x}_c|\mathbf{z})$ with a weight factor w_c, expressed as:

$$\phi_c(\mathbf{x}_c|\mathbf{z}) = \exp(w_c^T f_c(\mathbf{x}_c|\mathbf{z})). \tag{5}$$

Combining Eqs. (3) and (5), the conditional distribution of the query nodes can be rewritten as

$$p(\mathbf{x}|\mathbf{z}) = \frac{1}{Z(\mathbf{z}, w)} \exp\left\{ \sum_{c \epsilon C} w_c^T f_c(\mathbf{x}_c|\mathbf{z}) \right\}. \tag{6}$$

3.5.2. Graph structure

Since ADLs are sequential events, we build a CRF model to capture temporal relationships. The graph structure is shown in Fig. 4. It contains sequences of observations $\mathbf{z}$ from sensor features, hidden nodes $\mathbf{x}$ of class probability assignments, and edges $\mathcal{E}$ representing pairwise relations. Each node in the sequence contains information from a short period, and connects to a number of nodes at different distances. For example, in Fig. 4, node x_s connects to 6 nodes in three different scales from shorter (1 unit) to a longer (4 units) distance. By having the connections, it computes the local activities with contextual information from the entire global network. The graph contains two types of potentials; local and pairwise potentials. The local potential captures local information within an interval, while the pairwise potential explains how the nodes relate to each other. Including both node and edge features, the overall clique potential can be written as:

$$\phi_c(\mathbf{x}_c|\mathbf{z}) = \exp\left(\sum_{s \in \mathcal{V}} w_s^T f_s(\mathbf{x}_s|\mathbf{z}) + \sum_{sd \in \mathcal{E}} w_{sd}^T f_{sd}(\mathbf{x}_s, \mathbf{x}_d|\mathbf{z}) \right) \tag{7}$$

where f_s is the local potential for a hidden node $\mathbf{x}_s$ and f_{sd} is a pairwise potential connecting a node $\mathbf{x}_s$ (source) and $\mathbf{x}_d$ (destination). Individual weights are also assigned to each of the functions. They encode the relative importance of each potential. We now describe in detail the local and pairwise potentials used.

Local potential—Represented by the observation nodes $\mathbf{z}$ in Fig. 4, they contain features extracted directly from data and encode local information. In our model, we have accelerometer and video features stored in two observation nodes $\mathbf{Z}_A$ and $\mathbf{Z}_V$. Each classifier uses the features to predict a class-probability vector $\mathbf{P}_A$ and $\mathbf{P}_V$ in the size of $\mathcal{M}$, where $\mathcal{M}$ is the number of activities in this study. Interestingly, acceleration and video features lead to very different performances on the classification of different activities (shown in the experiments section). Acceleration generally has good accuracy on locomotive activities, and video motion features are more accurate for stationary activities, especially the static activities. For this reason, the probabilistic vector P is divided into two class categories: α (acceleration) and β (video). α contains the activity classes

where the acceleration feature is better suited than video features. Conversely, β contains activities more suited for video features. To better represent these two sets of activities, the vectors with classification results are computed separately for each set and augmented with zeros. For example, assume there are two activities involved, let P_A be a normalised 1×2 vector from acceleration features, which has walking and sitting probabilities expressed as [0.35, 0.65]. If walking is registered as a dynamic activity, then $P_{A-\alpha}$ and $P_{A-\beta}$ become [0.35, 0] and [0, 0.65] respectively. In this way, we can assign a separate weight vector to different activity groups. More details are in Section 4.2.

Pairwise feature—These are potential functions defined over the edges connecting each pair of hidden nodes. They specify a transition probability from one state to another, using a matrix of size $\mathcal{M} \times \mathcal{M}$. In this research we use a point-to-point weight assignment method. This allows the model to define individual transition weights between different states. Therefore, a 4-activity model requires 4^2 (16) weights, which represent the likelihood of transitioning between activities, such as from sitting to drinking, walking to climbing stairs, or simply walking to walking. A traditional chain model refers to messages from the immediate neighbour node connected directly to the current node. Although past information is propagated from the previous node, it does not capture long-term relations directly.

Therefore, we introduce a distance-inference method into the model, or *multi-scale* temporal dependences. This allows the probability distribution of the current activity (node) to be correlated with multiple nodes from different distances (in temporal space). The example shown in Fig. 4 has a number of edges on the top, where the hidden node $\mathbf{x}_s$ connects the third $\mathbf{x}_{(s\pm2)}$ and fifth $\mathbf{x}_{(s\pm4)}$ neighbours. The transition matrices of $Edge_{(s,s\pm1)}$ and $Edge_{(s,s\pm4)}$ can be very different; the former refers to nodes from its immediate predecessor or successor, while the latter links to the ancestor or descendant. In this case, the node can be predicted using further contextual information from a larger group of neighbours to improve the system robustness.

3.6. Parameter learning

The overall goal of parameter learning is to determine the most suitable values for the weight vector w_s and w_{sd} in the feature functions. It involves an optimisation process to maximise the conditional likelihood of the training set. However, directly maximising the conditional likelihood can be extremely time consuming due to the need to perform inference in every step of the algorithm; the partition function Z needs to be computed in every iteration. In order to make learning tractable for these problems, we maximise the pseudo-likelihood of the training data [66]. This approximates the conditional likelihood by a product of distributions over all immediate neighbours (Markov Blanket) of x_s. Let N be the total number of nodes in our model, the pseudo-likelihood can be computed as,

$$p(\mathbf{x}) = \prod_{s=1}^{N} p(\mathbf{x}_s | MB(\mathbf{x}_s)). \tag{8}$$

For mathematical convenience, the optimisation on $p(\mathbf{x})$ can be achieved through a maximisation process in the log domain. To prevent getting huge weights during the optimisation process, the pseudo-log likelihood objective is typically regularised with a quadratic term,

$$\mathcal{L}(w) = \mathcal{L}(w|\mathcal{D}) - \frac{(w - \bar{w})^T (w - \bar{w})}{2\sigma^2} \tag{9}$$

where $\mathcal{D}$ is the training set expressed as $\mathcal{D} = (\mathcal{X}_s, \mathcal{Z}_s | s = 1, \ldots, N)$, and $\mathcal{L}(w|\mathcal{D})$ is the pseudo-log likelihood written as

$$\mathcal{L}(w|\mathcal{D}) = \sum_{s=1}^{N} \sum_{m=1}^{M} \left\{ w_s^T f_s(x_s|z_s) + \sum_{i=1}^{I} w_{sd(i)}^T f_{sd}(x_{s,m}|z_s) - \log\big(Z_m(z_s, MB(x_s), w_s, w_{sd})\big) \right\}, \tag{10}$$

where the local feature function $w_s^T f_s(x_s|z_s)$ is defined as

$$w_s^T f_s(x_s|z_s) = w_1^T f_{acc.\alpha} + w_2^T f_{acc.\beta} + w_3^T f_{vid.\alpha} + w_4^T f_{vid.\beta}. \tag{11}$$

As shown in Eq. (11), the 4 local potentials, as previous explained in Section 3.5.2, includes 2 acceleration and 2 video feature functions for two sensor sources, where $f_{acc.\alpha}$ represents a vector predicted from acceleration data for activities α, and $f_{vid.\beta}$ is the video vector for the group β.

Eq. (10) can be explained in three components: local, pairwise and partition functions. The partition function Z_m is a local "committee" (compared to a global network), where Markov blanket dramatically reduces the computational cost from the original partition function. To elaborate it, we assume there are N nodes with M states each, in this case, Z_m sums M states from the local Markov blanket of $\mathbf{x}_s$. Therefore, computation is repeated $M \times N$ times for the entire space compared to Eq. (4), where $Z(\mathbf{z})$ needs to be evaluated M^N times. Since $\mathcal{L}(w|\mathcal{D})$ is a convex function, the local maximum can be achieved by a gradient descent algorithm. In this research we use the unconstrained L-BFGS method [67] on the negative version of $\mathcal{L}(w|\mathcal{D})$.

Table 1
A list of activity displays the average duration for each actions contained from our 40 dataset.

ID	Activity	Ave. duration (s)	ID	Activity	Ave. duration (s)
1	Walk	154.99	7	Sitting	46.08
2	Upstairs	59.05	8	Read	45.21
3	Downstairs	55.14	9	Watch program	253.58
4	Drink	15.72	10	Write	112.78
5	Stand-up	1.71	11	WaterTap	1.67
6	Sit-down	2.32	12	HandWash	10.39

Table 2
Patient activities.

ID	Activity	ID	Activity
1	Walk	8	Standing
2	Upstairs	9	Lying
3	Downstairs	10	Transfer
4	Drink	11	Open-door
5	Stand-up	12	Lie-down
6	Sit-down	13	Write
7	Sitting	14	Sit-up

3.7. Probabilistic inference

The inference procedure computes statistics for the hidden nodes **x** given the graph structure, and the observations **z**. There are two basic operations, the computation of *Marginal Distributions*—a joint distribution p for each of the variables **x**, and *Maximum A Posteriori Configuration*—the most likely assignment of **x**. Both the marginals and MAP configuration can be computed using belief propagation (BP) [68]. In particular, the sum–product version of BP performs marginalisation and the max-product version of BP computes the MAP configuration. Conventional BP when applied to tree graphs provides an exact answer. Messages are propagated from the leaves to the root node and back again. For arbitrary graphs, there is another popular variation called Loopy Belief Propagation (LBP) [69] which provides approximate answers. It updates messages in every iteration until converge (which is not guaranteed but typically happens). Another variation with stronger convergence properties is Tree-Reweighted BP (TRBP) [70]. It provides a guaranteed bound on the log partition function. It decomposes the original graph into a convex combination of tree-structured graphs allowing efficient computations, while the convex combination allows the computation of an upper bound on the optimal solution. For our model, as shown in Fig. 4, three methods are compared: BP on a chain model (CBP), LBP and TRBP to select the best approach for the problem.

4. Experiments

In this research, we focus on two population groups: healthy elders and disabled patients. The healthy elder group members were selected based on two criteria: being over 55 year-old and independent in activities of daily living (ADLs). ADLs include personal care such as grooming, domestic care such as cleaning and cooking and transferring one's body from a seated position into a standing or waiting position. From the health elder group, we collected 40 datasets of ADLs performances as our ADLs database, equally split between the 5 group members. Each dataset contains an average of 30-min sensors recording of realistic sequential ADLs in home or public environment. The list of activities is detailed in Table 1. Note that actions were repeated in each video to ensure reliability. To best represent realistic situations, the subjects were not asked to perform pre-defined sequences of activities or any detailed motions, but were simply asked to follow their ADLs sequences at their leisure.

In the disabled patient group, all individuals were diagnosed with at least one disability, and were receiving rehabilitation treatment at the local hospital. Therefore, due to differing mobility limitations between groups, we separately defined the activity list according to healthcare professionals' advices. The list of activities for the disabled patients are detailed in Table 2. Unlike the healthy elder group, the disabled patients were either admitted to hospital or were community dwelling outpatients receiving twice weekly treatments at the health department, therefore, we set up experiments within the rehabilitation building of a local hospital. We obtain additional 4 sets of ADLs datasets from each disabled subject. Each additional dataset contains 20-min of data records covering all activities based on patient's own preferences, as demonstrated in Fig. 5.

The first-person viewing angle is approximately 90°, and video is recorded at 15 Hz with 144×176 video resolution, synchronised with a 80 Hz 3-axis accelerometer using the android system timestamps. As a summary, we collect 1.1 million video frames plus 17.3 millions of acceleration samples over 5 subjects from elder group (equivalent of 20 h records, each dataset is around 30-min length, and 40 datasets in total from 5 elders, or 8 datasets for each person). For the disabled patient group, we obtained 2.3 millions of frames and 36.1 million samples (Equivalent to 40 h of records; each dataset is

Fig. 5. Experimental setup.

around 20-min length, and 120 datasets in total from 30 patients, with 4 datasets for each patient.) from the 30 disabled patients. As indicated earlier, each patient's dataset covers different activities according to the subject's preferred sequence of ADLs. We manually labelled all the data using a free standing camera, so that the acceleration and video data could be automatically annotated.

In this chapter, we firstly described the parametrisation, which included feature window size, classifier settings and multi-scale graph constructions. Secondly, we categorised and integrated the activities for both sensors. Thirdly, we compared and selected the optimal inference method for our graphical model. Finally, we conducted experiments on both groups, and discuss the results.

4.1. Parametrisation

For a better adaptation to general populations and various environmental settings, the parameters are selected based on the elder group's data since they represent the most typical set; healthy individuals with no locomotion and time limitations.

4.1.1. Window size

The window size defines how much information (duration) is required to classify an activity. Each window contains just sufficient data to describe the subject's current status. The amount of overlapping is another concern during data collection, it helps the classifier distinguish the contents between consecutive windows with a transition period. It also helps to recognise the feature from a wider context in order to compensate the possible errors from the window itself. We run cross validation over 10 ADLs videos to determine a suitable windows size and overlapping portion chosen from 9 different settings: 2, 3, 5 s of window length with 25%, 50% and 75% overlapping. After a number of cross validation tests from our independent training database, we estimate the 3-s window with 50% overlap has the best performance over all settings.

4.1.2. Local classifiers

There are two levels of classification in this system: local (i.i.d) and structured. The former uses features directly extracted from raw sensor data and provides predictions independently of time. The latter depends on the graph structure and takes into account temporal dependences. As stated in Section 3.4, we experiment with two local classifiers in the first-person context: LogitBoost and SVMs. The LogitBoost algorithm requires a proper number of Weak Learners (WLs) for both features (acceleration and vision). We take 5 independent video segments for each activity, each segment is 5-min long. We first extract video and acceleration features into windows and re-arrange them in a random order. Then we conduct a 10-fold cross validation on both features, and test LogitBoost with 10 different numbers of weak learners, from 50 to 500. Our results show that 150-WLs is the most reliable for the video features and 50-WLs for the acceleration features. On the other hand, selecting a suitable kernel for SVM can be difficult. We run the same validation process on three popular kernels: Linear, Polynomial and Gaussian Radial Basis Function (RBF). The averaged precision and recall results for both classifiers are displayed in Table 3.

As it can be seen, SVMs with linear kernel has the best performance for the acceleration features. LogitBoost does better on video optical flow features. Note that all of our datasets are quite realistic including various ADLs; the activities include both dynamic and static actions. Some of the static activities, such as watching or sitting might not contain any acceleration signals. Conversely, motions such as walking and climbing stairs would create lot of noise to the egocentric vision. Therefore, the local feature classifier in isolation does not achieve very high accuracy.

Table 3
Classifier accuracy on acceleration and video features.

Classifier	LogitBoost	SVM		
Acceleration	50-WLs	Linear	Polynomial	RBF
Averaged precision	68.99%	**69.39%**	43.6%	37.01%
Average recall	61.01%	**62.96%**	34.98%	34.89%
Overall accuracy	77.40%	**78.07%**	60.51%	66.92%
Video	150-WLs	Linear	Polynomial	RBF
Averaged precision	**53.12%**	37.58%	35.19%	43.86%
Average recall	**44.76%**	30.11%	26.39%	42.78%
Overall accuracy	**68.91%**	56.39%	44.63%	61.68%

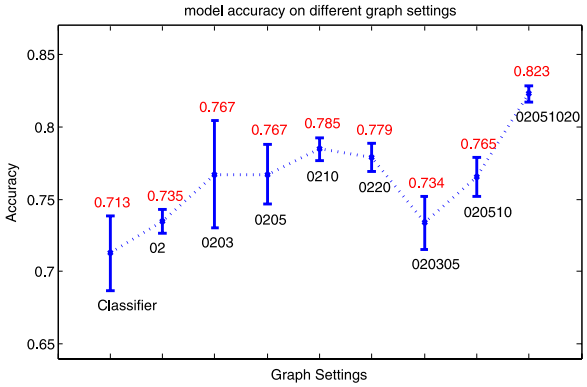

Fig. 6. Overview of model accuracy when using different graph settings.

4.1.3. Graph structure

In Section 3.5.2, we introduce the multi-scale graphical model. Each node connects to multiple neighbours to create a more powerful inference network. In this section, we validate 8 different connection settings from 5 different scales: 2nd neighbour (chain model), 3rd neighbour, 5th neighbour, 10th neighbour and 20th neighbour. The number indicates the distance before/after the chosen node in a temporal scale. It represents time between the pair of connected nodes. In the last section, we selected the time window interval as 3 s with 50% overlapping. This means a 5th neighbour is $4 \times 1.5 = 6$ s away from the current node, 10th neighbours is 13.5 s away and 20th is 28.5 s apart.

To obtain an optimal graph setting, we take 10 datasets from our entire ADLs database, train on 8 datasets, and test on 2 using a sum–product TRBP inference method. For each dataset, we set 8 edge configurations: '02' (chain graph), '0203', '0205', '0210', '0220', '020305', '020510' and '02051020', and use LogitBoost as our feature classifier on acceleration features. The settings, for example, '020305' means each node connects to 2nd nodes (immediate nodes), 3rd nodes and 5th nodes. An example is shown in Fig. 4. In the example X_s is with the '020305' setting. Note that the edge distance impacts on the updating frequency and the contextual information. A longer-distance graph generally requires more memory and results in a lower updating frequency. e.g. 20th neighbour requires to store at least 28.5 s of data, and 100th neighbour would need 142.5 s of data. Therefore, by balancing these factors, we choose up to 20th neighbour as our maximum inference distance. To compare these edge settings, we use the averaged precision value as our evaluation preference. The results of selected configurations are illustrated in Fig. 6, plotted in the form of error-bars. The first term is the precision of the local classification result before applying the graphical model.

In Fig. 6, the graph setting '02051020' shows the highest precision, outperforming the local feature classifier by 11% achieving 82.3%, followed by '0210' at 78.5% and '0220' at 77.9%, where '020305' and '02' have the worst performance with accuracy of 73.4% and 73.5% respectively, but still better than the local classifiers. Therefore, we hypothesise the graph '02051020' as the best setting, thus to be applied in the next set of experiments.

4.2. Activity categorisation

In Section 3.5.2, we introduce the local features integration method, which requires to separate the activities into two categories. However, our ultimate goal is to let the model determine which feature is better for each activity, such that the CRF is able to train and assign appropriate weights for both vision and acceleration features. In order to achieve this, a direct approach is used to inspect their current classification performance, and to find the prediction accuracy of each activity using both features. In Section 4.1.2 we learned that SVMs (linear kernel) have the best performance on acceleration

Table 4
Results from acceleration and video features with activities following the order in Table 1.

ID	Acceleration		Video	
	Pre.	Rec.	Pre.	Rec.
1	**0.936**	**0.961**	0.422	0.384
2	**0.878**	**0.879**	0.305	0.305
3	**0.868**	**0.868**	0.212	0.200
4	**0.670**	**0.584**	0.523	0.375
5	**0.754**	**0.679**	0.238	0.227
6	**0.771**	**0.521**	0.207	0.229
7	0.250	**0.418**	**0.350**	0.333
8	0.503	0.624	**0.682**	**0.749**
9	0.581	0.505	**0.722**	**0.684**
10	0.762	**0.759**	**0.828**	0.753
11	0.429	0.375	**0.438**	**0.475**
12	0.375	0.136	**0.577**	**0.482**
Ave.	**0.648**	**0.609**	**0.459**	**0.433**

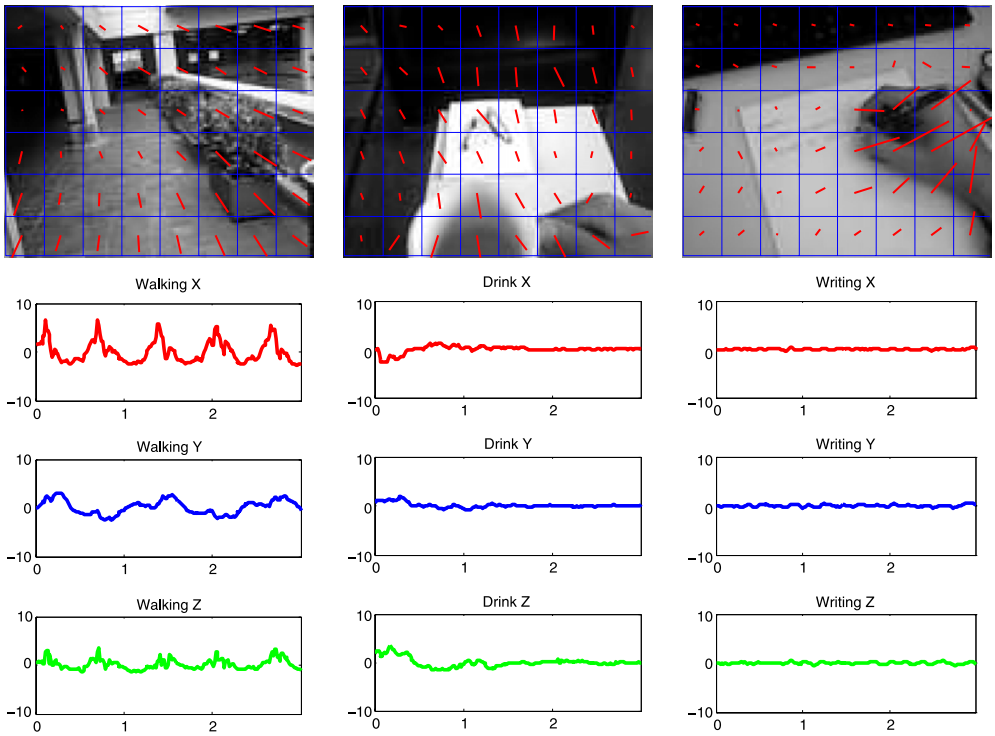

Fig. 7. Examples of optical flow features (1st row) and corresponding 3-axis acceleration window for three typical activities: Walking, Drinking and Writing.

features, and LogitBoost is good on video feature classification. Therefore, we run a 10-fold cross validation over 10 datasets (from ADLs database) using both classifiers. All activities are evaluated using precision and recall [71]. The classification results are shown in Table 4.

As shown in this table, it is interesting to note that classification from acceleration features has good precision on most of the dynamic motion against the statics ones, while the video features have better prediction on stationary activities (Sitting, Reading, Watching TV/monitor, Writing, Switching Water-Tap and Hand-Washing) than locomotive activities (Walking, Going Upstairs, Going Downstairs, Drinking, Stand Up and Sit Down). The reason that may explain this phenomenon is: locomotive activity usually involves a number of periodic motions or infrequent motion with high magnitude, which can be easily captured from an accelerometer. On the other hand, activities involving less mobility such as sitting or reading can be better classified with optical flow features because the user is in a stationary position, but the motions can be captured from the egocentric video. Fig. 7 demonstrates three examples for both acceleration and vision features: *Walk*, *Drink* and *Write*. Note that in the *Drink* window, the signal change at the beginning represents a head-lifting motion. Writing involves only a small motion compared to the others.

Table 5
List of 30 patients' age, sex and diagnosis.

Age	Sex	Diagnosis	Age	Sex	Diagnosis
34	M	Lost legs	54	M	Stroke
57	M	Lung transplants	81	M	Stroke
82	M	Parkinson's disease	56	F	Parkinson's disease
44	F	Multiple sclerosis	71	F	Hip replacement
46	M	Decreased mobility	63	M	Accident: multi-fractures
68	M	Injuries to legs	60	M	Fall: fractured spines
68	F	Lung disease	65	M	Fall: fractured pelvic bones
86	M	Deconditioned	69	M	Heart failure and stroke
52	M	Hip surgery	61	M	spinal injury after fall
75	M	Deconditioned	58	M	Hip replacement
62	M	Foot surgery	72	F	Knee replacement
75	F	Knee replacement	64	F	Spinal surgery
83	F	Fall-head injury	89	F	Fall: fractured hip
47	M	Lung transplants	62	F	Accident: fractured collar bone
67	F	Stroke	75	M	Heart surgery

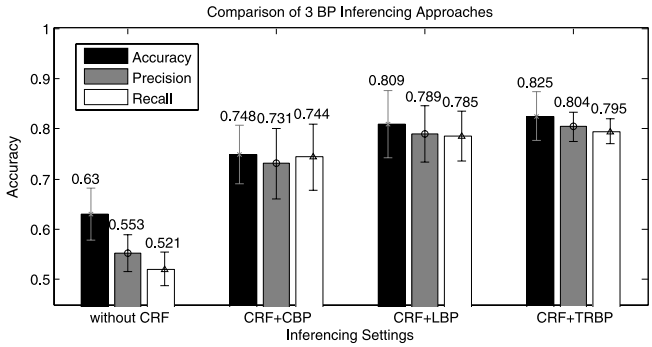

Fig. 8. Comparison of different inferences.

4.3. Inference approaches

In this section, we compare 3 BP inference approaches with leave-one-out cross validation, using the same training datasets from Section 4.2. We train and test them on both vision and acceleration features, and evaluate the model performance in terms of the classification accuracy and processing speed. The average performance is detailed in Fig. 8.

The chart displays the comparisons of 3 BP approaches. TRBP has the best performance with an averaged precision of 80.4%, followed by LBP at 78.9% and CBP at 73.1%. We also test the processing speed by evaluating all three approaches on one 30-min dataset. The average running times are: *CBP-15.19* s, *LBP-30.20* s, *TRBP-5.17 s*. The result shows that both TRBP and LBP have a higher accuracy compared to CBP, but TRBP is faster than the others. We conclude, the TRBP is the most appropriate inference algorithm from our validation.

5. Results and discussions

5.1. Results from Elders

For the experiment with the elder group, we apply leave-one-out cross validation on 40 independent datasets using the settings determined in the previous sections. These experiments are used to compare several approaches in terms of averaged precision, recall and overall accuracy. Our evaluation focuses on the performance of structured classification, as well as the benefits of integrating local-pairwise features. In this section, we observe the system performance for 7 settings: (1) VID: video feature (classified LogitBoost) only. (2) ACC: acceleration feature (SVM) only. (3) LBAV: LogitBoost Classifier on one combined feature vector of acceleration and vision. (4) CRFV: Apply CRF MODEL on top of setting 1. (5) CRFA: CRF on setting 2. (6) CRFLB: CRF on setting 3. (7) CRFAV: Integrates both feature classifiers with a CRF, as explained in Eqs. (10) and (11). We plot the error-bar for all configurations in Fig. 9. This includes the average Precision, Recall and F-Measure [71] of all 12 activities. We also plot the overall accuracy over the entire datasets. Note that the overall accuracy is generally higher than others. This is due to the unbalanced distribution of different activities. e.g. video contains more 'Walking' than 'Hand-Washing'.

Fig. 8 shows the benefits of integrating local classification with the CRF model. Settings 1–3 use local classifier only, and 4–7 are with an additional CRF structured classification. The system overall accuracy is improved by an average of 10.5% and the F-Measure by 7.6%. Notably, the CRFAV algorithm achieves the best result with an accuracy of 90.38% and F-Measure

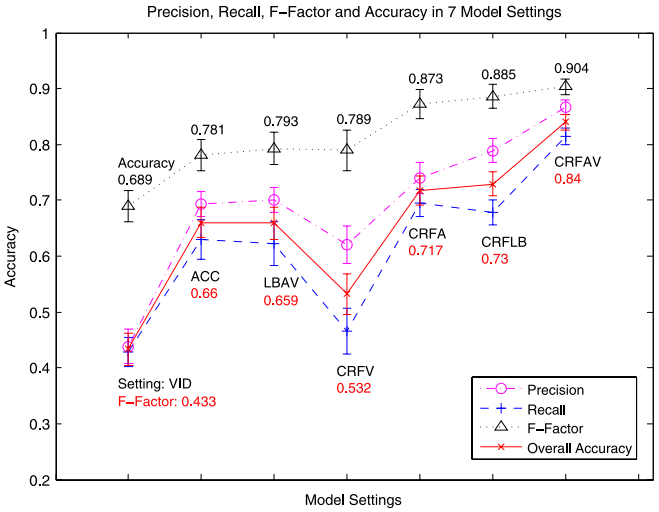

Fig. 9. System performance from 7 model configurations.

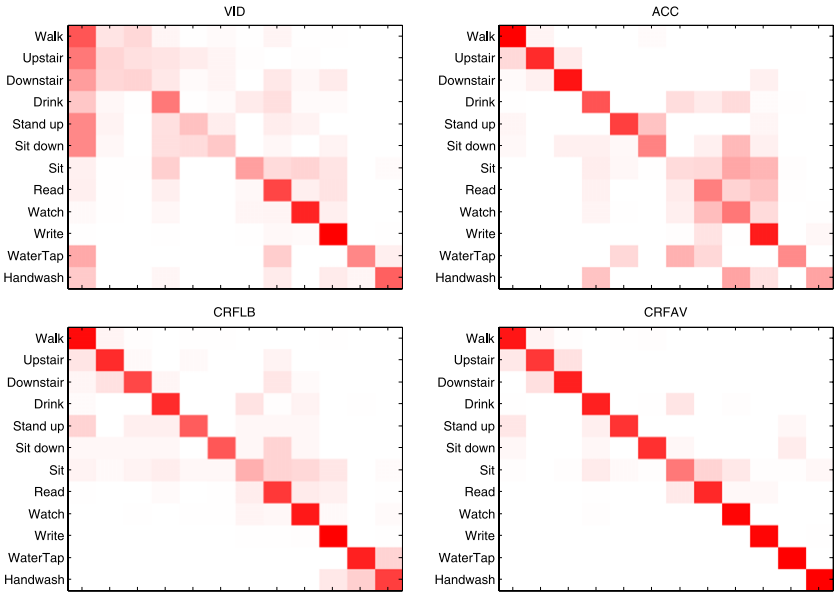

Fig. 10. Comparison of confusion matrices for four configurations: 'VID', 'ACC', 'CRFLB' and 'CRFAV'. See the text for details.

of 84.45%. It also indicates that video features are not as good as acceleration features when used on ADLs. This might be caused by the unbalanced distribution between locomotive motions and stationary activities from our ADLs database. To be more specific, we include 4 confusion matrices in Fig. 10.

The top two plots in Fig. 10 are the feature classifier results. The locomotive activities are mistaken from video feature, including climbing stairs, sit-down and stand-up. The acceleration feature classifier has relatively poor results in hand activities. Mistakes occur among sitting, watching and reading. The 'CRFLB' matrix removes most of the incorrect predictions, has fewer outliers between stationary activities, as well as between switching water-tap and hand-washing. Overall, 'CRFAV' is the best option. It removes most of the outliers, and it only has minor errors between up and downstairs, sitting and reading.

5.2. Results from patients

In the last section, we observe the benefits of different model settings in the healthy elder group experiment. In this section, we introduce another popular graphical model: Hidden Markov Model (HMM) for comparison purposes to the

Table 6
Comparisons of approach accuracies of patient group.

	SVMAV	SVM + HMM	SVM + CRF
Same patient	53.80%	72.61%	73.91%
Cross patient	60.76%	69.67%	77.07%
Cat.1 musculoskeletal	51.76%	66.59%	72.71%
Cat.2 neurological	56.11%	80.2%	82.84%
Cat.3 general weakness	34.90%	53.65%	55.21%
	LBAV	LB + HMM	LB + CRF
Same patient	48.91%	62.50%	63.15%
Cross patient	57.72%	73.26%	75.43%
Cat.1 musculoskeletal	44.94%	80.00%	84.00%
Cat.2 neurological	52.48%	67.00%	71.95%
Cat.3 general weakness	37.50%	61.98%	69.79%
Total average	49.89%	68.75%	72.61%

existing multi-scale CRF model. HMMs predict the most likely states from current observations exploiting nearby temporal relationships. This idea is widely employed in speech [72] and activity classifications [38] due to the efficiency of learning and inference algorithms. In this work, we use a HMM to improve the activity predictions for independent classifiers. This can be done by estimating a prior transition and an observation matrix based on human knowledge among 14 activities. Then these matrices are calculated by the training activity sequences using Baum–Welch algorithm [73].

We evaluate and compare the system performance on disabled patient group using three-stage approach: conventional classification (LB and SVM), Hidden Markov Model (HMM) and multi-scale CRF model. The patient group contains 30 individuals, all of them are receiving treatments from the rehabilitation unit at St Vincent's hospital. Due to the complexity of patients' conditions (as displayed in Table 5), we validate the approaches from three aspects to accurately interpret the system performance:

Same patient validation: 4-fold cross validation process from the same subject. As mentioned earlier, we collect 4 datasets for every patient, each contains all the activities listed in Table 2. In this setting, we use 3 datasets for training, and test on the 4th dataset from the same person.

Cross-patients validation: 30-fold cross validation over all subjects. We conduct a leave-one-out process, training the system using 29 patients' datasets and test on the other patient.

Cross-patient validation in three categories: Following the instructions of health professionals, we split the patients into 3 groups according to their diagnosis from Table 5: Musculoskeletal (14 subjects), Neurological (9 subjects) and General Weakness (7 subjects).

In order to make three approaches comparable, we use LBAV and SVMAV as the basic classification methods and apply the classifier on one combined feature vector of acceleration and vision. Then we apply HMM and multi-scale CRF on the top of the classifiers to obtain the evaluation results. Table 6 details the overall system accuracies over all validation categories using three approaches.

Observing the columns, the local classifiers (LB and SVM) have an average accuracy of 49.89%, while the structured classification approaches dramatically outperform the conventional approaches. HMM increases the system accuracy by an average of 19% to approximately 68.75%, while the multi-scale CRF further improves the percentages by an average of 4% at 72.61%. Observing the rows, it is interesting to note that the cross-patient validation approach performs better than the same patient results. For the cross-patient of same categories, the general weakness group has a local classification accuracy of 34.9% and 37.5%, which are most difficult to be predicted, then the Musculoskeletal on 44.94% and 51.76%, followed by the Neurological Patients of 52.48% and 56.11%. A reason may explain the differences between cross-patient and same-patient that the cross-patient approach contains a much larger training database (from 29 subjects) compared to the same patient validation, where in general, more datasets allow for training a more sophisticated classifier. Confusion matrices for the three classification methods between cross and same patient validation approaches are presented in Fig. 11.

Comparing cross and same-patient results for the first column in Fig. 11, the local classifiers have the similar results in between up- and down-stairs possibly due to similarities of the patterns. This is also noticeable in between stand-up and sit-down, and for sitting and standing. Unlike the elder group, the disabled patient group is asked to conduct activities at specific locations, and the subjects are asked to remain sitting or standing without specific purposes. Therefore, the status of "standing/sitting still" is unlikely to have any hand or other motions that could be recognised by the video features. This may be the reason the error prediction for the subject's static postures. For the second and third columns in Fig. 11, we apply and compare HMMs and multi-scale CRFs against the local classifier predictions. We verify that the application of structured classification effectively removes the error predictions using the contextual information. Note that the cross patient approach misclassifies in between walk and stairs, while the same-patient validation does not. This may be due to the differences of individual habits during walking and going up stairs.

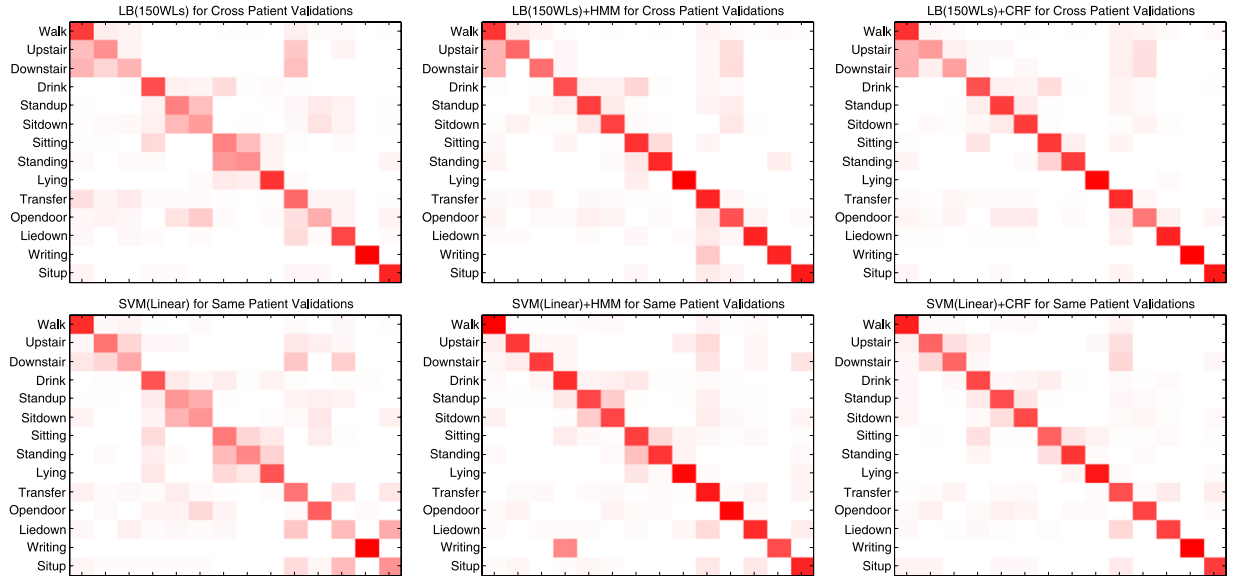

Fig. 11. Comparison of confusion matrices from cross-patient (top row) and same-patient (bottom row) validations using LogitBoost (150 WLs) and Support Vector Machines (Linear).

5.3. Discussion

Activity recognition is the basis for the development of an interactive system to assist healthy elder and disabled patients to maintain a good quality of life, remain in their homes and working, thereby avoiding institutional care. We believe that this is one of the first studies to investigate the validity and reliability of activity recognition systems in the disabled population.

In this study, we explore contextual learning for recognition of activities of daily living. In real life, recognising varied and realistic human activities is a challenge in terms of reliability. We address the problem for ADLs recognition, and our experimental results confirm the benefits of multi-scale CRFs compared to traditional local and HMM approaches. This involves the temporal inference method in a graphical model in conjunction with a feature integration method. Multi-scale context improves the performance by learning the subject's ADLs sequence with temporal correlations. The feature integration method allows the system to learn the relative weights among different sources, as well as linking local features into a global network. The framework improves the local classification by an average of 10%–20%, and improves accuracy to over 80%.

Our model can now be implemented in the Smart-Glasses prototype with a 1 GHz CPU processor and 512 MB RAM. The tasks, such as 'video/sensor data collection', 'video/sensor features extraction' and 'local feature classifications' can be executed almost instantaneously (exclude the training process, which had been done offline). The TRBP inference procedure generally requires an average of 2–3 s from a 30-s updating block, which is equivalent to a 20-window sliding block. Moreover, the current platform is a temporary prototype consisting of an android cellphone mounted on safety goggles. We tested the approach on over 30 patients and 5 elders. The feedback on the comfort level has been positive, however, it cannot be worn over an hour due to its weight of current design.

6. Conclusion and future work

This work presented a novel activity recognition approach from first person perspective. The algorithm is developed using a multi-scale CRF model, which exploits two important factors for ADLs recognition, feature interpretation and contextual structure, thus to cover a wide range of human activities. We utilise the head acceleration and egocentric vision as our primary sources of information, which have great potential to capture general body motion and hand activities. Embedding the sensors into glasses makes the system simple and of easy adoption for elder and disabled patient populations. In this research, we annotate, train, and validate our method using a large realistic ADL dataset covering several motion types, different environmental settings and various locations. Results demonstrate that the model outperforms a number of existing methods, and the system is tested and proved reliable in both indoor and outdoor environments.

In the next stage, we will conduct more validation studies and investigate the robustness of our model in more challenging activities on the targeting population. Further, the development, utilisation and patient acceptability of accurate activity recognition devices needs to be promoted. In this line, the system can be deployed as an app for Google Glasses potentially reaching a large population that will result in large datasets. In such cases, manual labelling for ground truth can be quite challenging. Therefore, a semi-supervised learning algorithm is the next target of research. This includes the development of a semi-supervised classification system for both first-person and free standing sensors.

References

[1] M.J. Mathie, A.C. Coster, N.H. Lovell, B.G. Celler, Accelerometry: providing an integrated, practical method for long-term, ambulatory monitoring of human movement, Physiol. Meas. 25 (2) (2004) R1.
[2] J. Bussmann, W. Martens, J. Tulen, F. Schasfoort, H. Van Den Berg-Emons, H. Stam, Measuring daily behavior using ambulatory accelerometry: the activity monitor, Behav. Res. Methods Instrum. Comput. 33 (3) (2001) 349–356.
[3] M.A. Lafortune, Three-dimensional acceleration of the tibia during walking and running, J. Biomech. 24 (10) (1991) 877–886.
[4] K. Aminian, P. Robert, E. Buchser, B. Rutschmann, D. Hayoz, M. Depairon, Physical activity monitoring based on accelerometry: validation and comparison with video observation, Med. Biol. Eng. Comput. 37 (3) (1999) 304–308.
[5] P.H. Veltink, E.O. Engberink, B.J. Van Hilten, R. Dunnewold, C. Jacobi, Towards a new method for kinematic quantification of bradykinesia in patients with Parkinson's disease using triaxial accelerometry, in: IEEE 17th Annual Conference Engineering in Medicine and Biology Society, 1995. Vol. 2, IEEE, 1999, pp. 1303–1304.
[6] K. Duk-Jin, B. Prabhakaran, Motion fault detection and isolation in body sensor networks, in: 2011 IEEE International Conference on Pervasive Computing and Communications, PerCom, 2011, pp. 147–155.
[7] A.K. Bourke, G.M. Lyons, A threshold-based fall-detection algorithm using a bi-axial gyroscope sensor, Med. Eng. Phys. 30 (1) (2008) 84–90.
[8] N.H. Chehade, P. Ozisik, J. Gomez, F. Ramos, G. Pottie, Detecting stumbles with a single accelerometer, in: 2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC, 2012, pp. 6681–6686.
[9] M. Ermes, J. Parkka, J. Mantyjarvi, I. Korhonen, Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions, IEEE Trans. Inf. Technol. Biomed. 12 (1) (2008) 20–26.
[10] G. Plasqui, A. Bonomi, K. Westerterp, Daily physical activity assessment with accelerometers: new insights and validation studies, Obes. Rev. 14 (6) (2013) 451–462.
[11] A.M. Khan, Human activity recognition using a single tri-axial accelerometer (Ph.D. thesis), 2011.
[12] F.R. Allen, E. Ambikairajah, N.H. Lovell, B.G. Celler, Classification of a known sequence of motions and postures from accelerometry data using adapted Gaussian mixture models, Physiol. Meas. 27 (10) (2006) 935.
[13] A. Godfrey, A. Bourke, G. Olaighin, P. Van De Ven, J. Nelson, Activity classification using a single chest mounted tri-axial accelerometer, Med. Eng. Phys. 33 (9) (2011) 1127–1135.
[14] S.M. Duma, S.J. Manoogian, W.R. Bussone, P.G. Brolinson, M.W. Goforth, J.J. Donnenwerth, R.M. Greenwald, J.J. Chu, J.J. Crisco, Analysis of real-time head accelerations in collegiate football players, Clin. J. Sport Med. 15 (1) (2005) 3–8.
[15] T. Brezmes, J.-L. Gorricho, J. Cotrina, Activity recognition from accelerometer data on a mobile phone, in: Distributed Computing Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living, in: Lecture Notes in Computer Science, vol. 5518, Springer, Berlin, Heidelberg, 2009, pp. 796–799.
[16] J.R. Kwapisz, G.M. Weiss, S.A. Moore, Activity recognition using cell phone accelerometers, ACM SIGKDD Explor. Newslett. 12 (2) (2011) 74–82.
[17] J. Leppanen, A. Eronen, Accelerometer-based activity recognition on a mobile phone using cepstral features and quantized GMMs, in: Acoustics, Speech and Signal Processing, ICASSP, 2013 IEEE International Conference on, IEEE, 2013, pp. 3487–3491.
[18] D. Anguita, A. Ghio, L. Oneto, X. Parra, J.L. Reyes-Ortiz, Human Activity Recognition on Smartphones Using a Multiclass Hardware-Friendly Support Vector Machine, Springer, 2012, pp. 216–223.
[19] J. Ward, P. Lukowicz, G. Tröster, T. Starner, Activity recognition of assembly tasks using body-worn microphones and accelerometers, IEEE Trans. Pattern Anal. Mach. Intell. (2006) 1553–1567.
[20] M. Sousa, A. Techmer, A. Steinhage, C. Lauterbach, P. Lukowicz, Human tracking and identification using a sensitive floor and wearable accelerometers, in: Proceedings of the 11th IEEE International Conference on Pervasive Computing and Communications, PERCOM, 2013, pp. 166–171.
[21] L. Wang, L. Cheng, T.H. Thi, J. Zhang, Human action recognition from boosted pose estimation, in: Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications, DICTA'10, IEEE Computer Society, Washington, DC, USA, 2010, pp. 308–313.
[22] B. Yao, A. Khosla, L. Fei-Fei, Classifying actions and measuring action similarity by modeling the mutual context of objects and human poses, in: International Conference on Machine Learning, ICML, 2011.
[23] W. Shandong, O. Oreifej, M. Shah, Action recognition in videos acquired by a moving camera using motion decomposition of Lagrangian particle trajectories, in: 2011 IEEE International Conference on Computer Vision, ICCV, 2011, pp. 1419–1426.
[24] L. Yong Jae, J. Ghosh, K. Grauman, Discovering important people and objects for egocentric video summarization, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2012, pp. 1346–1353.
[25] A. Fathi, J.K. Hodgins, J.M. Rehg, Social interactions: a first-person perspective, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2012, pp. 1226–1233.
[26] H. Pirsiavash, D. Ramanan, Detecting activities of daily living in first-person camera views, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2012, pp. 2847–2854.
[27] A. Fathi, R. Xiaofeng, J.M. Rehg, Learning to recognize objects in egocentric activities, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2011, pp. 3281–3288.
[28] K.M. Kitani, T. Okabe, Y. Sato, A. Sugimoto, Fast unsupervised ego-action learning for first-person sports videos, in: 2011 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2011, pp. 3241–3248.
[29] D.M. Gavrila, The visual analysis of human movement: a survey, Comput. Vis. Image Underst. 73 (1) (1999) 82–98.
[30] W. Hu, T. Tan, L. Wang, S. Maybank, A survey on visual surveillance of object motion and behaviors, IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. 34 (3) (2004) 334–352.
[31] J. Aggarwal, M.S. Ryoo, Human activity analysis: a review, ACM Comput. Surv. (CSUR) 43 (3) (2011) 16.
[32] Z. Zhao, Y. Chen, J. Liu, Z. Shen, M. Liu, Cross-people mobile-phone based activity recognition, in: IJCAI 2011, 2011, pp. 2545–2550.
[33] S. Hongeng, R. Nevatia, F. Bremond, Video-based event recognition: activity representation and probabilistic recognition methods, Comput. Vis. Image Underst. 96 (2) (2004) 129–162.
[34] N.M. Oliver, B. Rosario, A.P. Pentland, A Bayesian computer vision system for modeling human interactions, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 831–843.
[35] W. Jinjun, X. Changsheng, C. Engsiong, Automatic sports video genre classification using pseudo-2D-HMM, in: 18th International Conference on Pattern Recognition, 2006, ICPR 2006, vol. 4, 2006, pp. 778–781.
[36] K. Doki, K. Hashimoto, S. Doki, S. Okuma, A. Torii, Estimation of next human behavior and its timing for human behavior support, in: 2010 11th International Conference on Control Automation Robotics and Vision, ICARCV, 2010, pp. 952–957.
[37] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using hidden Markov model, in: 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92, IEEE, pp. 379–385.
[38] K. Eunju, H. Sumi, D. Cook, Human activity recognition and pattern discovery, IEEE Pervasive Comput. 9 (1) (2010) 48–53.
[39] K. Zhan, F. Ramos, S. Faux, Activity recognition from a wearable camera, in: 2012 12th International Conference on Control Automation Robotics and Vision, ICARCV, 2012, pp. 365–370.
[40] P. Natarajan, R. Nevatia, Hierarchical multi-channel hidden semi Markov graphical models for activity recognition, Comput. Vis. Image Underst. 117 (10) (2013) 1329–1344.
[41] L. Liao, D. Fox, H. Kautz, Hierarchical Conditional Random Fields for GPS-Based Activity Recognition Robotics Research, in: Springer Tracts in Advanced Robotics, vol. 28, Springer, Berlin, Heidelberg, 2007, pp. 487–506.
[42] E. Zhang, Y. Zhao, A multi-scale conditional random field model for human action recognition, in: Image and Signal Processing, CISP, 2012 5th International Congress on, IEEE, 2012, pp. 77–81.

[43] D.L. Vail, M.M. Veloso, J.D. Lafferty, Conditional random fields for activity recognition, in: Proceedings of the 6th International Joint Conference on Autonomous Agents and Multiagent Systems, ACM, Honolulu, Hawaii, 2007, pp. 1–8.
[44] M. Shimosaka, T. Mori, T. Sato, Robust action recognition and segmentation with multi-task conditional random fields.
[45] A. Quattoni, S. Wang, L.-P. Morency, M. Collins, T. Darrell, Hidden conditional random fields, IEEE Trans. Pattern Anal. Mach. Intell. 29 (10) (2007) 1848.
[46] J. Zhang, S. Gong, Action categorization with modified hidden conditional random field, Pattern Recognit. 43 (1) (2010) 197–203.
[47] B. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision (DARPA), in: Proceedings of the 1981 DARPA Image Understanding Workshop, 1981, pp. 121–130.
[48] Y. Nam, J.W. Park, Child activity recognition based on cooperative fusion model of a triaxial accelerometer and a barometric pressure sensor, IEEE J. Biomed. Health Inform. 17 (2) (2013) 420–426.
[49] H. Zhenyu, Activity recognition from accelerometer signals based on wavelet-AR model, in: 2010 IEEE International Conference on Progress in Informatics and Computing, PIC, vol. 1, 2010, pp. 499–502.
[50] J.-K. Min, S.-B. Cho, Activity recognition based on wearable sensors using selection/fusion hybrid ensemble, in: 2011 IEEE International Conference on Systems, Man, and Cybernetics, SMC, 2011, pp. 1319–1324.
[51] S. Liu, R.X. Gao, D. John, J.W. Staudenmayer, P.S. Freedson, Multisensor data fusion for physical activity assessment, IEEE Trans. Biomed. Eng. 59 (3) (2012) 687–696.
[52] B.E. Boser, I.M. Guyon, V.N. Vapnik, A training algorithm for optimal margin classifiers, in: Proceedings of the Fifth Annual Workshop on Computational Learning Theory, ACM, 1992, pp. 144–152.
[53] C. Cortes, V. Vapnik, Support vector networks, Mach. Learn. 20 (1995) 273–297.
[54] T.-F. Wu, C.-J. Lin, R.C. Weng, Probability estimates for multi-class classification by pairwise coupling, J. Mach. Learn. Res. 5 (2004) 975–1005.
[55] R. Schapire, The strength of weak learnability, Mach. Learn. 5 (2) (1990) 197–227.
[56] S.B. Kotsiantis, Bagging and boosting variants for handling classifications problems: a survey, Knowl. Eng. Rev. First View (2013) 1–23.
[57] P. Casale, O. Pujol, P. Radeva, Human activity recognition from accelerometer data using a wearable device, in: Proceedings of the 5th Iberian Conference on Pattern Recognition and Image Analysis, Springer-Verlag, Las Palmas de Gran Canaria, Spain, 2011, pp. 289–296.
[58] Y. Jongmin, K. Daijin, Frontal face classifier using adaboost with MCT features, in: 2010 11th International Conference on Control Automation Robotics and Vision, ICARCV, 2010, pp. 2084–2087.
[59] Y. Cai, K. Feng, W. Lu, K. Chou, Using logitboost classifier to predict protein structural classes, J. Theoret. Biol. 238 (1) (2006) 172–176.
[60] J. Friedman, T. Hastie, R. Tibshirani, Special invited paper. Additive logistic regression: a statistical view of boosting, Ann. Statist. 28 (2) (2000) 337–374.
[61] Y. Freund, R.E. Schapire, A desicion-theoretic generalization of on-line learning and an application to boosting, in: Computational Learning Theory, Springer, 1994, pp. 23–37.
[62] B. Verma, A. Rahman, Cluster-oriented ensemble classifier: impact of multicluster characterization on ensemble classifier learning, IEEE Trans. Knowl. Data Eng. 24 (4) (2012) 605–618.
[63] C. Sutton, A. McCallum, An introduction to conditional random fields, arXiv Preprint arXiv:1011.4088.
[64] S. Lauritzen, Graphical Models, Oxford University Press, 1996.
[65] D. Koller, N. Friedman, Probabilistic Graphical Models: Principles and Techniques, MIT Press, 2009.
[66] J. Besag, Statistical analysis of non-lattice data, Statistician (1975) 179–195.
[67] D.C. Liu, J. Nocedal, On the limited memory BFGS method for large scale optimization, Math. Program. 45 (1–3) (1989) 503–528.
[68] J. Pearl, Reverend Bayes on inference engines: a distributed hierarchical approach, in: Proceedings of the American Association of Artificial Intelligence National Conference on AI, 1982, pp. 133–136.
[69] K.P. Murphy, Y. Weiss, M.I. Jordan, Loopy belief propagation for approximate inference: an empirical study, in: Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc., 1999, pp. 467–475.
[70] V. Kolmogorov, Convergent tree-reweighted message passing for energy minimization, IEEE Trans. Pattern Anal. Mach. Intell. 28 (10) (2006) 1568–1583.
[71] C. van Rijsbergen, Information Retrieval, 2nd ed., Butterworths, London, 1979.
[72] M. Gales, S. Young, The application of hidden Markov models in speech recognition, Found. Trends Signal Process. 1 (2007) 195–304.
[73] L. Rabiner, A tutorial on hidden Markov models and selected applications in speech recognition, Proc. IEEE 77 (2) (1989) 257–286.