Algorithms **2009**, *2*, 282-300; doi:10.3390/a2010282

OPEN ACCESS

algorithms

ISSN 1999-4893

www.mdpi.com/journal/algorithms

Article

Recognizing Human Activities from Sensors Using Hidden Markov Models Constructed by Feature Selection Techniques

Rodrigo Cilla *, Miguel A. Patricio, Jesús García, Antonio Berlanga and Jose M. Molina

Computer Science Department, Universidad Carlos III de Madrid, Avda. de la Universidad Carlos III, 22, Colmenarejo, Spain

E-mails: mpatrici@inf.uc3m.es, jgherrer@inf.uc3m.es, aberlan@ia.uc3m.es, molina@ia.uc3m.es

* Author to whom correspondence should be addressed; E-mail: rcilla@inf.uc3m.es

Received: 28 November 2008; in revised form: 2 February 2009 / Accepted: 16 February 2009 / Published: 21 February 2009

Abstract: In this paper a method for selecting features for Human Activity Recognition from sensors is presented. Using a large feature set that contains features that may describe the activities to recognize, Best First Search and Genetic Algorithms are employed to select the feature subset that maximizes the accuracy of a Hidden Markov Model generated from the subset. A comparative of the proposed techniques is presented to demonstrate their performance building Hidden Markov Models to classify different human activities using video sensors

Keywords: Computer vision; Human Activity Recognition; Feature Selection; Hidden Markov Models

1. Introduction

Sensors allow computers to perceive the world that surrounds them. By the use of sensors, like thermostats or photocells, computers can meassure the temperature or lighting conditions of a room. In the last years,the deployment of multisensor networks has become popular due to the cut down on sensor prizes. This networks can include different kind of sensors, maybe more complex, like cameras, indoor location systems (ILS), microphones, etc ... By the use of sensor networks computer systems can take more accurate decissions due to the richer information about the environment that they have.

A common task in sensor processing is the recognition of situations in the environment perceived. If

the problem can be tackled as a classification task. Then, the problem to solve is to find a model relating the data from sensor readings with situation labels. Some sensor data may be too noisy, and other not provide any information for label prediction. Relevant sensor data has to be identified to build the model from them, not including irrelevant data.

An application where the use of sensor networks can be useful is Human Activity Recognition, i.e., the understanding by the computer of what humans are doing. This field has received increasing attention in the last years, due to their promising applications that have in surveillance, human computer interaction or ambient intelligence, and the interests that governments and commercial organizations have placed in the area. Human Activity Recognition systems can be integrated with existing systems as the proposed by Corchado *et al.* [1] to monitor alzehimer patients. If a patient falls to the floor, the system can alert a nurse to attend the accident. Human Activity Recognition systems can also be integrated with the system proposed by Pavón *et al.* [2], detecting forbidden activities being performed and alerting security staff.

Human activity recognition may be considered as a classification task, and human activity recognizers can be created using supervised learning. A set of activities to recognized has to be defined a priori. Different observations about the activities to recognize are extracted using different sensors. Then, the problem to solve is to find the function that best relates observations to activity labels. Data from sensors is not free from noise, so relevant attributes have to be identified before training the activity recognizer. Better results are expected to be obtained when this previous step is performed.

The sensors that provides the most information for Human Activity Recognition are video cameras. Works in activity recognition from video could be divided in two groups [3]: (1) those that are centered in small duration activities (i.e. walking, running,...); and (2) those that deal with large duration activities (i.e. leaving place, going to living room,...). The former are centered in choosing good features for activity recognition, whereas the latter usually tackle the temporal structure in the classifier.

Regarding small duration activities, Neural networks, Neuro-fuzzy systems and C4.5 have been succesfully employed [4]. Also, time delay neural networks have been used to recognize the case where the activities are hand gestures [5]. In [6] is shown how to perform feature extraction, feature selection and classification using a simple bayesian classifier to solve the small duration activity recognition problem. In their approach, they use ground truth data from CAVIAR* dataset of people bounding boxes instead of a blob tracker output. Perez *et al.* [7] use different time averaged speed measures to classify the activities present in the CAVIAR dataset using HMM.

Robertson *et al.* [3] uses trajectory and velocity concatenated data for five frames, and blob optical flow, to decide what is the current small duration activity. This small duration activity is then introduced in a Hidden Markov Model (HMM) to decide which is the small duration activity performed.

To classify large duration activities, Brand *et al.* [8] have used HMMs for modelling activities performed in an office. They use different spatial meassures taken from the blob bounding ellipse.The HMM approach has been extended in [9] to model activities involving two agents, using a Coupled-HMM. They use different velocity meassures as features, being all of them invariant with respect to camera rotations and translations, providing camera independent recognizers.

HMMs are recognized as one effective technique for activity classification, because they offer dynamic time warping, have clear bayesian semantics and well-understood training algorithms. In this paper a method to adapt them, selecting the best features in the observation space, is defined, trying to

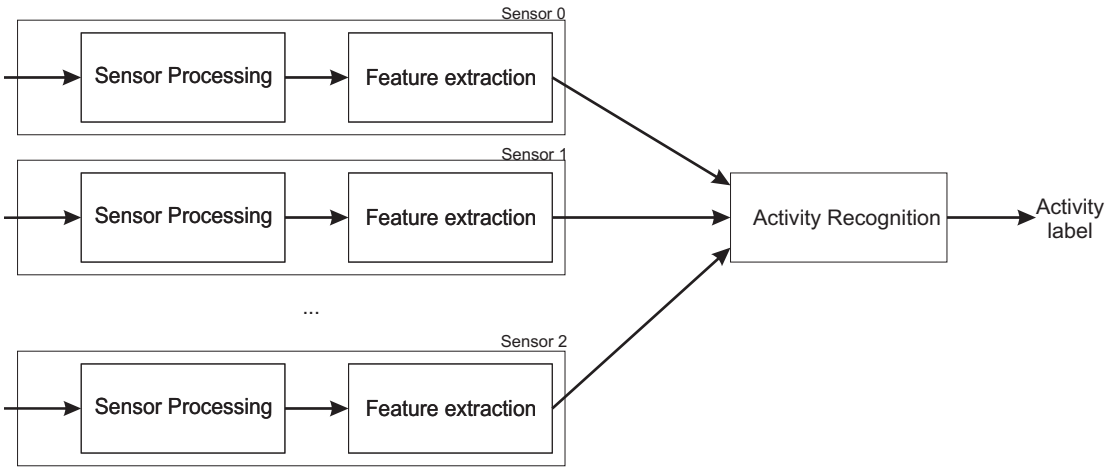
maximize its accuracy for classifying short durative activities. HMMs will be incrementally built using both heuristic search and genetic algorithms. A comparative of the performance obtained using these algorithm for HMM construction will be shown. Our approach differs from the proposed by Ribeiro *et al.* in some aspects: (1) blob tracker output is used instead of ground truth data; (2) the foreground mask of the blob tracker is used to extract features; (3) the classifier used for the recognition of activities is an HMM; (4) we use different temporal window sizes; and (4) we use different search methods for feature selection.

This paper is organized as follows: on section 2, an overview of the activity recognition system where the feature selection is going to be performed is given; on section 3, the feature selection algorithms that are used to build HMMs are presented; on section 4, experiments selecting good features for human activity recognition are performed and results are discussed; finally, on section 5 conclusions of this work are presented and future lines are discused.

2. Definition of Activity Recognition problem

2.1. Functional description

Figure 1. Overview of the activity recognition process from multiple sensors



In the proposed human activity recognition architecture (see Figure 1), the objective is to select from a set of activity labels $A = \{a_1, \ldots, a_N\}$ the one that is the most likely according to the signal $p(t)$ retrieved by the sensors at each time step. Sensors could be video cameras, thermal cameras or indoor localization systems, depending of the available resources and the conditions of the environment. In case of imaging sensors, they get a frame on each time step, $p(t) \equiv I(t, x, y)$. Using $p(t)$, the system needs to extract the position and size $b_i(t)$ of each human i in the environment. When using imaging sensors, $b_i(t)$ may be obtained using a blob tracker [10]. Blob tracker takes an image sequence and provides the location of each object i moving in the scene, mantaining a temporal coherence of objects between frames. Human tracking is a hard task and, despite of the advances in the last years, tracking methods still provide a noisy output, caused in part by scene illumination changes or by the complexity of object

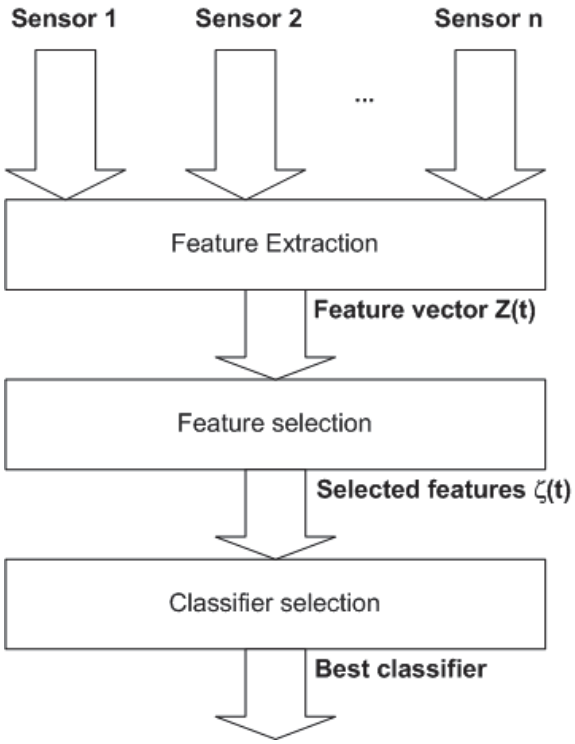
able to differenciate between the activities to be recognized. Finally, using $\zeta_i(t)$, a function $\Lambda(\zeta_i(t))$, $\Lambda : \zeta \rightarrow A$ is used to decide wich is the activity being performed.

Activity recognition solves the correspondence:

$$q_{it} = \arg\max_{a_k} P(\zeta_i(t), \zeta_i(t-1) \ldots, \zeta_i(1) \mid a_k) \quad (1)$$

where $q_i t$ is the activity that maximizes the probability of the observations $\zeta_i(t)$,$\zeta_i(t-1)$,...,$\zeta_i(0)$ at instant t by individual i.

Figure 2. The three steps to create an activity recognizer



To build an activity recognition system, three processes have to be performed (see Figure 2). First, an extensive feature set $Z_i(t)$ has to be extracted from $b_i(t)$ and $I(t, x, y)$. On a second step, the subset $\zeta_i(t)$ has to be selected from $Z(t)$, selecting the subset with the most predictive power for A. The last problem to solve is to select the best classifier architecture for activity recognition.

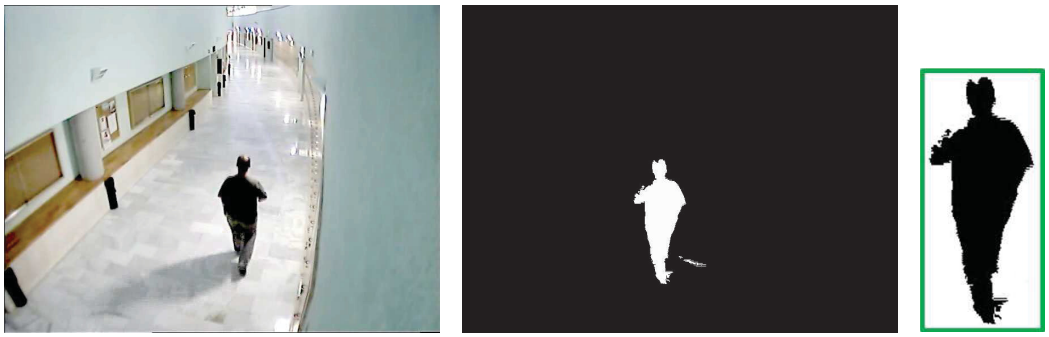
2.2. *Feature extraction for activity representation using video sensors*

The features that can be used for activity representation depend of the type of sensor inputs to be processed. The most common sensor used for Human Activity recognition are video cameras, so all the features presented here are computed from image sequences.

The objective of this section is to present a large set of features to characterize the activities to classify as complete as possible. The input to the feature extraction process includes the original frame $I(t, x, y)$, its foreground mask $Fg(t, x, y)$, and the blob bounding box $b(t)$ given by a blob tracker, that represents

Most of the features presented here have been proposed in [6]. The first group of extracted features comes from the blob bounding box properties and its time evolution, taking first order derivatives and time averaged derivatives, using a temporal window of T frames, as shown on Table 1

Figure 3. Inputs for the feature extraction process



(a) Original frame (b) Foreground mask (c) Bounding box

Table 1. First feature group: blob bounding box properties

#	Feature name	Definition
		Bounding box properties at time t
1	Position	$p(t) = (x(t), y(t))$
2	Size	$size(t) = (w(t), h(t))$
3	Velocity	$s(t) = \frac{\partial p(t)}{\partial t} = \left(\frac{\partial x(t)}{\partial t}, \frac{\partial y(t)}{\partial t}\right)$
4	Size derivative	$\frac{\partial size(t)}{\partial t} = \frac{\partial w(t)}{\partial t}, \frac{\partial h(t)}{\partial t}$
5	Speed	$s(t) = \sqrt{\left(\frac{\partial x(t)}{\partial t}\right)^2 + \left(\frac{\partial y(t)}{\partial t}\right)^2}$
6	Area	$s(t) = w(t) * h(t)$
		Properties averaged over T frames
7	Mean speed	$\bar{s}_T(t) = \frac{1}{T} \sum_{i=t-T+1}^{T} s(i)$
-	Mean velocity norm	$\|\bar{\vec{s}}_T(t)\|$ 3 different methods:
8	Averaging vectors	$\|\bar{\vec{s}}_T(t)\|_1 = \left\| \frac{1}{T} \sum_{i=t-T}^{t-1} \frac{\vec{p}(t) - \vec{p}(i)}{t-i} \right\|$
9	Mean vectors	$\|\bar{\vec{s}}_T(t)\|_2 = \left\| \frac{p(t) - p(t-T+1)}{T} \right\|$
10	Linear fitting	$\|\bar{\vec{s}}_T(t)\|_3 =$ Linear Least Squares Fitting
11..13	speed/velocity ratio	$R_{\bar{s}_T,i}(t) = \frac{s_T(t)}{\|\vec{s}_T(t)\|_i} \quad i = 1,2,3$
		Second order moments
14	speed	$\sigma^2_{v,T}(t)_1 = \frac{1}{T-1} \sum_{i=t-T+1}^{t} s^2(i)$
15..17	speed (centered)	$\sigma^2_{v,T}(t)_{1+j} = \frac{1}{T-1} \sum_{i=t-T+1}^{t} (s(i) - \bar{s}_T(t))_j$ j=1,2,3
-	velocity	$\Sigma_{\vec{s},T}(t)_1 = \frac{1}{T-1} \sum_{i=t-T+1}^{t} \vec{s}(i) \vec{s}(i)'$
-	velocity centered	$\Sigma_{\vec{s},T}(t)_{1+j} = \frac{1}{T-1} \sum_{i=t-T+1}^{t} \left(\vec{s}(i) - \bar{\vec{s}}_T(t)_j\right) \left(\vec{s}(i) - \bar{\vec{s}}_T(t)_j\right)'$ j=1,2,3
18..21	trace	$tr_{\Sigma_{\vec{s},T}}(t)_i = trace\left(\Sigma_{\vec{v},T}(t)\right)$ i=1,2,3,4
22..25	eigenvalues ratio	$R_{\Sigma_{\vec{s},T}}(t)_i = \frac{\lambda_{min}}{\lambda_{max}}\left(\Sigma_{\vec{v},T}(t)\right)$ i=1,2,3,4

Second feature group consist of the seven Hu invariant moments [12] $\{hu_1, hu_2, hu_3, hu_4, hu_5, hu_6, hu_7\}$ of the foreground mask enclosed by the blob bounding box. These seven moments are shape descriptors invariant under translation, rotation and scaling. Each Hu moment is numbered from 26 to 32.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.