IMPROVING SUPERVISED CLASSIFICATION OF DAILY ACTIVITIES LIVING USING NEW COST SENSITIVE CRITERION FOR *C*-SVM

M'hamed Bilal Abidine, Belkacem Fergani

Speech Communication & Signal Processing Laboratory.
Faculty of Electronics and Computer Sciences
USTHB, Algiers, Algeria
abidineb@hotmail.com, bfergani@gmail.com

ABSTRACT

The growing population of elders in the society calls for a new approach in care giving. By inferring what activities elderly are performing in their houses it is possible to determine their physical and cognitive capabilities. In this paper we show the potential of important discriminative classifiers namely the Soft-Support Vector Machines (C-SVM), Conditional Random Fields (CRF) and k-Nearest Neighbors (k-NN) for recognizing activities from sensor patterns in a smart home environment. We address also the class imbalance problem in activity recognition field which has been known to hinder the learning performance of classifiers. Cost sensitive learning is attractive under most imbalanced circumstances, but it is difficult to determine the precise misclassification costs in practice. We introduce a new criterion for selecting the suitable cost parameter C of the C-SVM method. Through our evaluation on four real world imbalanced activity datasets, we demonstrate that C-SVM based on our proposed criterion outperforms the state-of-the-art discriminative methods in activity recognition.

KEYWORDS

Activity Recognition, C-*SVM, Wireless Sensor Networks, Machine Learning, Imbalanced Data*

1. INTRODUCTION

In 2030, nearly one out of two households will include someone who needs help performing basic Activities of Daily Living (ADL) [1] such as cooking, brushing, dressing, toileting, bathing and so on. For their comfort and because the healthcare infrastructure will not be able to handle this growth, it is suggested to assist sick or elderly people at home. Sensor based technologies in the home is the key of this problem. Sensor data collected often needs to be analysed using data mining and machine learning techniques to build activity models and perform further means of pattern recognition [2, 3]. The learning of such models is usually done in a supervised manner (human labelling) and requires a large annotated datasets recorded in different settings. Recognizing a predefined set of activities is a classification task: features are extracted from signals gathered by the sensors within a time window and then used to infer the activity. The classification algorithm has to be trained using a set of samples representing the activities that have to be recognized.

Sundarapandian et al. (Eds) : ICAITA, SAI, SEAS, CDKP, CMCA-2013
pp. 415–424, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3834

State of the Art methods used for recognizing activities can be divided in two main categories: the so called generative models and discriminative models [5-8]. The generative methods perform well but require data modelling, marred by generic optimization criteria and are generally time consuming. Discriminative ones received the most attention in literature for its simplicity-model and good performance. Therefore, we have studied in this paper, different discriminative classification methods.

However, activity recognition datasets are generally imbalanced, meaning certain activities occur more frequently than others (e.g. sleeping is generally done once a day, while toileting is done several times a day). However, the learning system may have difficulties to learn the concept related to the minority class, and therefore, not incorporating this class imbalance results in an evaluation that may lead to disastrous consequences for elderly person. Recently, the class imbalance problem has been recognized as a crucial problem in machine learning [9-12]. Most classifiers assume a balanced distribution of classes and equal misclassification costs for each class and therefore, they perform poorly in predicting the minority class for imbalanced data [13]. They optimize the overall classification accuracy and hence sacrifice the prediction performance on the minority classes. Compared with other standard classifiers, SVM is more accurate on moderately imbalanced data. The reason is that only Support Vectors are used for classification and many majority samples far from the decision boundary can be removed without affecting classification [3]. However, It has been identified that the separating hyperplane of an SVM model developed with an imbalanced dataset can be skewed towards the minority class [14], and this skewness can degrade the performance of that model with respect to the minority class.
Previous research that aims to improve the effectiveness of SVM on imbalanced classification [14-16], and some good results have been reported [10]. Approaches for addressing the imbalanced training-data problem can be categorized into two main divisions: the data processing approach and the algorithmic approach. At the data level, these solutions can be divided into : oversampling [14] (in which new samples are created for the minority class), undersampling [14] (where, the samples are eliminated for the majority class) or some combination of the two is deployed. Vilarino et al. used Synthetic Minority Oversampling TEchnique (SMOTE) [17] oversampling. At the algorithmic level, the solutions include adjusting the costs associated with misclassification so as to improve performance [18, 19], adjusting the probabilistic estimate at the tree leaf (when working with decision trees), adjusting the decision threshold, and recognition-based (i.e., learning from one class) rather than discrimination-based (two class) learning [14]. Akbani et al. proposed the SMOTE with Different Costs algorithm (SDC) [14]. SDC conducts SMOTE oversampling on the minority class with different error costs. Wu et al. proposed the Kernel Boundary Alignment algorithm (KBA) that adjusts the boundary toward the majority class by modifying the kernel matrix [15]. In addition to the naturally occurring class imbalance problem, the imbalanced data situation may also occur in one-against-rest schema in multiclass classification. Therefore, even though the training data is balanced, issues related to the class imbalance problem can frequently surface.

Our objective is to deal the class imbalance problem to perform automatic recognition of activities from binary sensor patterns in a smart home. The main contribution of our work is twofold. Firstly, we propose a new criterion to select the cost parameter C for the discriminative method Soft-Support Vector Machines (C-SVM) [3, 7] to appropriately tackle the problem of class imbalance caused by imbalanced activity datasets. Secondly, this method is compared with Conditional Random Fields (CRF) [5], The k-Nearest Neighbors k-NN [2] and the traditional SVM utilized as reference methods. Especially, CRF is a generative probabilistic model have been mainly used as a reference methods which recently gained popularity and work well in recognition activity field [5].

The remainder of this paper is organized as follows, Section 2 describes the different discriminative methods and the weighted C-SVM method combined with our proposed criterion for parameter C setting. Then, Section 3 presents the setup and discusses the results acquired through a series of experiments using different datasets. Finally, we conclude in Section 4.

2. DISCRIMINATIVE METHODS FOR ACTIVITY RECOGNITION

2.1. Conditional Random Fields (CRF)

Conditional Random Fields (CRF) have an exponential model for the conditional probability (1) of the entire sequence of labels Y given an input observation sequence X. CRF is defined by a weighted sum of K feature functions f_i that will return a 0 or 1 depending on the values of the input variables and therefore determine whether a potential should be included in the calculation. Each feature function carries a weight λ_i that gives its strength to the proposed label. These weights are the parameters we want to find when learning the model. CRF model parameters can be learned using an iterative gradient method by maximizing the conditional probability distribution defined as

$$P(Y \mid X) = \frac{1}{Z(X)} \exp \sum_{t=1}^{T} \left(\sum_{i=1}^{K} \lambda_i f_i(y_t, y_{t-1}, x_t) \right) \quad (1)$$

$$\text{With } Z(X) = \sum_{y} \left(\exp \sum_{t=1}^{T} \left(\sum_{i=1}^{K} \lambda_i f_i(y_t, y_{t-1}, x_t) \right) \right) \quad (2)$$

One of the main consequences of this choice is that while learning the parameters of a CRF we avoid modelling the distribution of the observations, p(x). As a result, we can only use CRF to perform inference (and not to generate data), which is a characteristic of the discriminative models. To find the label y for new observed features, we take the maximum of the conditional probability.

$$\hat{y}(x) = \operatorname{argmax}_y p(y \mid x) \quad (3)$$

2.2. *k*-Nearest Neighbors (*k*-NN)

The k-Nearest Neighbors (k-NN) algorithm is amongst the simplest of all machine learning algorithms [2], and therefore easy to implement. The m training instances $x \in R^n$ are vectors in an n-dimensional feature space, each with a class label. The result of a new query is classified based on the majority of the k-NN categories. The classifiers do not use any model for fitting and are only based on memory to store the feature vectors and class labels. They work based on the minimum distance from an unlabelled vector (a test point) to the training instances to determine the k-NN. The positive integer k is a user-defined constant. Usually Euclidean distance is used as the distance metric.

2.3. *C*-Support Vector Machines (*C*-SVM)

SVM classifies data by determining a hyperplane into a higher dimensional space (feature space) [3]. For a two class problem, we assume that we have a training set $\{(x_i, y_i)\}_{i=1}^{m}$ where $x_i \in R^n$ are the observations and y_i are class labels either 1 or -1. The primal formulation of the soft-margin in SVM maximizes margin $2/K(w,w)$ between two classes and minimizes the amount of total misclassications (training errors) ξ_i simultaneously by solving the following optimization problem :

$$\min_{w,b,\xi} \quad 1/2.K(w,w) + C\sum_{i=1}^{m}\xi_i$$
$$\text{subject to} \quad y_i(w^T\phi(x_i)+b) \geq 1-\xi_i, \xi_i \geq 0, i=1,..,m \tag{4}$$

where w is normal to the hyperplane, b is the translation factor of the hyperplane to the origin and $\phi(.)$ is a non-linear function which maps the input space into a feature space defined by $K(x_i,x_j)=\phi(x_i)^T\phi(x_j)$ that is kernel matrix of the input space.

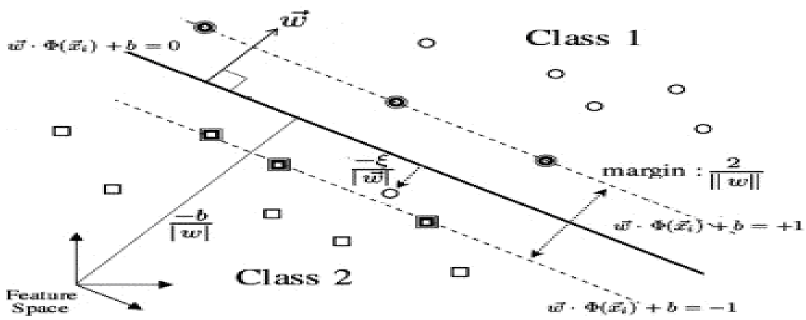

Figure1. *C*-SVM classification problem: The classes are linearly separated in a feature space

We choose the popular Radial Basis Function (RBF kernel): $K(x,y)=exp\left(-\left\|x_i-x_j\right\|^2/2\sigma^2\right)$ where σ is the width parameter. It is a reasonable first choice for the classification of the nonlinear datasets, as it has fewer parameters. The construction of such functions is described by the Mercer conditions [20]. The regularization parameter C is used to control the trade-off between maximization of the margin width and minimizing the number of training error of nonseparable samples in order to avoid the problem of overfitting [2]. A small value for C will increase the number of training errors, while a large C will lead to a behavior similar to that of a hard-margin SVM. In practice the parameters (σ and C) are varied through a wide range of values and the optimal performance assessed using a cross-validation technique to verify performance using only training set [20].

The dual formulation of the soft margin SVM can be solved by representing it as a Lagrangian optimization problem as follows [3] :

$$\max_{\alpha_i} \quad \sum_{i=1}^{m}\alpha_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\alpha_i\alpha_j y_i y_j K(x_i,x_j)$$
$$\text{Subject to } \sum_{i=1}^{m}\alpha_i y_i = 0 \ \ and \ \ 0 \leq \alpha_i \leq C\,, \tag{5}$$

Solving (5) for α gives a decision function in the original space for classifying a test point $x \in R^n$ [3] is presented by the following formula

$$f(x) = sgn\left(\sum_{i=1}^{m_{sv}}\alpha_i y_i K(x,x_i) + b\right) \tag{6}$$

where m_{sv} is the number of support vectors $x_i \in R^n$. $\alpha_i > 0$ are Lagrange multipliers. The training samples where $\alpha_i > 0$ are called support vectors.

In this study, a software package LIBSVM [21] was used to implement the multiclass classifier algorithm. It uses the One-vs-One method [3]. Although SVM often produce effective solutions for balanced datasets, they are sensitive to imbalanced training datasets and produces sub-optimal models because the constraint in (4) imposes equal total influence from the positive and negative support vectors. To cope the imbalanced samples set, we choose the weighted *C*-SVM formulation [3] and we propose a new criterion for tuning the parameter *C*.

2.3.1. Weighted SVM

In this method, the SVM soft margin objective function is modified to assign two different penalty constraints C^+ and C^- for the minority and majority classes respectively, as given in the quadratic optimization below

$$\begin{aligned} &\min_{w,b,\xi} \; 1/2.K(w,w)+C^+\sum_{y_i=1}\xi_i+C^-\sum_{y_i=-1}\xi_i \\ &\text{subject to} \quad y_i(w^T\phi(x_i)+b)\geq 1-\xi_i, \xi_i\geq 0, i=1,\dots,m \end{aligned} \tag{7}$$

The SVM dual formulation gives the same Lagrangian as in the original soft-margin SVM in (5), but with different constraints on α_i as follows:

$$\begin{aligned} &\max_{\alpha_i} \quad \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m \alpha_i\alpha_j y_i y_j K(x_i, x_j) \\ &\text{Subject to} \quad 0\leq\alpha_i\leq C_+, \quad \text{if } y_i=+1, \quad \textit{and} \\ &\qquad\qquad\qquad 0\leq\alpha_i\leq C_-, \quad \text{if } y_i=-1 \end{aligned} \tag{8}$$

In the construction of cost sensitive SVM, the cost parameter plays an indispensable role. For the cost information, some authors [18, 19] have proposed adjusting different penalty parameters for different classes of data which effectively improves the low classification accuracy caused by imbalanced samples. For example, it is highly possible to achieve the high classification accuracy by simply classifying all samples as the class with majority samples (positive observations), therefore the minority class (negative observations) is the error training. Veropoulos *et al*. in [19] propose to increase the tradeoff associated with the minority class (i.e., $C^- > C^+$) to eliminate the imbalance effect. Veropoulos et *al*. have not suggested any guidelines for deciding what the relative ratios of the positive to negative cost factors should be.

2.3.1.1. Proposed Criterion

Our proposed criterion advocates analytic parameter selection of C_i regularization parameter in *N*-multi class problem for each class *i* directly from the training data, on the basis of the proportion of class data. This criterion respects the reasoning of Veropoulos that is to say that the tradeoff C^- associated with the smallest class is large in order to improve the low classification accuracy caused by imbalanced samples. It allows the user to set individual weights for individual training examples, which are then used in *C*-SVM training. We give the main cost value C_i in function of m_+ the number of majority class and m_i the number of other classes samples, it is given by:

$$C_i = \left[m_+/m_i\right] \tag{9}$$

[ ] is integer function and $C_i \in \{1,\dots,m_+/m_i\}$, $i=1,\dots,N$

For the two-class training problem, the primal optimization problem of the soft-margin in SVM can be constructed via this criterion and become:

$$\begin{aligned} &\min_{w,b,\xi} \ 1/2K(w,w)+\sum_{y_i=1}\xi_i+[m_+/m_-].\sum_{y_i=-1}\xi_i \\ &\text{subject to} \quad y_i(w^T\phi(x_i)+b)\geq 1-\xi_i, \xi_i \geq 0, i=1,\ldots,m \end{aligned} \tag{10}$$

The SVM dual formulation gives the same Lagrangian as in the soft margin SVM in (8) with $C_+ = 1$ and $C_- = m_+/m_-$.

3. Experimental Results And Discussion

3.1. Datasets

Experiments were performed using a datasets gathered from three houses having different layouts and different number of sensors [5, 22]. Each sensor is attached to a wireless sensor network node. The activities performed with a single man occupant at each house are different from each other. Data are collected using binary sensors such as reed switches to determine open-close states of doors and cupboards; pressure mats to identify sitting on a couch or lying in bed; mercury contacts to detect the movements of objects like drawers; passive infrared (PIR) sensors to detect motion in a specific area; float sensors to measure the toilet being flushed. Time slices for which no annotation is available are collected in a separate activity labelled 'Idle'. The data were collected by a Base-Station and labelled using a Wireless Bluetooth headset combined with speech recognition software or a handwritten diary for the house C.

Table 1. Overview of activities and the number of observations for each house [5, 22].

House $A_{(1)}$	House $A_{(2)}$	House B	House C
Idle$_{(4627)}$	Idle$_{(6031)}$	Idle$_{(5598)}$	Idle$_{(2732)}$
Leaving$_{(22617)}$	Leaving$_{(16856)}$	Leaving$_{(10835)}$	Leaving$_{(11993)}$
Toileting$_{(380)}$	Toileting$_{(382)}$	Toileting$_{(75)}$	Eating$_{(376)}$
Showering$_{(265)}$	Showering$_{(264)}$	Showering$_{(112)}$	Toileting$_{(243)}$
Sleeping$_{(11601)}$	Brush teeth$_{(39)}$	Brush teeth$_{(41)}$	Showering$_{(191)}$
Breakfast$_{(109)}$	Sleeping$_{(11592)}$	Sleeping$_{(6057)}$	Brush teeth$_{(102)}$
Dinner$_{(348)}$	Breakfast$_{(93)}$	Dressing$_{(46)}$	Shaving$_{(67)}$
Drink$_{(59)}$	Dinner$_{(330)}$	Prep.Breakfast$_{(81)}$	Sleeping$_{(7738)}$
	Snack$_{(47)}$	Prep.Dinner$_{(90)}$	Dressing$_{(112)}$
	Drink$_{(53)}$	Drink$_{(12)}$	Medication$_{(16)}$
		Dishes$_{(34)}$	Breakfast$_{(73)}$
		Eat Dinner$_{(54)}$	Lunch$_{(62)}$
		Eat Breakfast$_{(143)}$	Dinner$_{(291)}$
		Play piano$_{(492)}$	Snack$_{(24)}$
			Drink$_{(34)}$
			Relax$_{(2435)}$

3.2. Setup and Performance Measures

We separate the data into a test and training set using a "leave one day out cross validation" approach. Sensors outputs are binary and represented in a feature space which is used by the model to recognize the activities performed. We do not use the raw sensor data representation as observations; instead we use the "*Change point*" and "*Last*" representation which have been shown to give much better results in activity recognition [5]. The raw sensor representation gives a 1 when the sensor is firing and a 0 otherwise. The "change point" representation gives a 1 when the sensor reading changes. While the last sensor representation continues to assign a 1 to the last sensor that changed state until a new sensor changes state.

As the activity instances were imbalanced between classes, we evaluate the performance of our models by two measures, the accuracy and the class accuracy. The accuracy shows the percentage of correctly classified instances which is highly affected by the sample distribution across activity classes, the class accuracy taking into account the class imbalance shows the average percentage of correctly classified instances per classes

$$Accuracy = \frac{\sum_{i=1}^{m}[inferred(i)=true(i)]}{m} \tag{11}$$

$$Class = \frac{1}{N}\sum_{c=1}^{N}\left[\frac{\sum_{i=1}^{m_c}\left[inferred_c(i)=true_c(i)\right]}{m_c}\right] \tag{12}$$

in which [a = b] is a binary indicator giving 1 when true and 0 when false. m is the total number of samples, N is the number of classes and m_c the total number of samples for class c.

3.3. Results

We compared the performance of the CRF, k-NN and C-SVM on the imbalanced dataset of the house $A_{(1)}$ in which minority class are all classes that appear at most 1% of the time, while others are the majority classes that typically, have a longer duration (e.g. leaving and sleeping). These algorithms are tested under MATLAB environment and the SVM algorithm is tested with implementation LibSVM [21].

In our experiments, the C-SVM hyper-parameters (σ, C) have been optimized in the range (0.1-2) and (0.1-10000) respectively to maximize the class accuracy of leave-one-day-out cross validation technique. The best pair parameters (σ_{opt}, C_{opt}) = (1, 5) are used, see table 2. Then, we tried to find the penalty parameters $C_{adaptatif}$ (class) adapted for different classes by using our criterion, see table 3.

Table 2. Selection of parameter C_{opt} with the cross validation for C-SVM.

C_{opt}	0.1	**5**	50	500	1000	5000	10000
Class (%)	51.7	**61**	61	61	61	61	61

Our empirical results in table 2 suggest that the value of regularization parameter C has negligible effect on the generalization performance (as long as C is larger than a certain threshold analytically determined from the training data (C =5)).

Table 3. Selection of parameter C_{opt} adapted for each class with our criterion for C-SVM.

ADL	Id	Le	To	Sh	Sl	Br	Di	Dr
C_{opt}	5	1	59	85	2	207	65	383

We see in table 3 that the minority class requires a large value of C compared with the majority class. This fact induces a classifier's bias in order to give more importance to the minority ones. The summary of the accuracy and the class accuracy obtained with the concatenation matrix of "Changepoint+Last" for CRF, k-NN, C-SVM using cross validation research and wighted C-SVM using our criterion are presented in Table 4. This table shows that C-SVM+our criterion performs better in terms of class accuracy, while others methods performs better in terms of accuracy.

Table 4. Accuracy and class accuracy for CRF, *k*-NN, *C*-SVM+cross validation search and *C*-SVM+our criterion.

Methods	Feature representation	Accuracy	Class
CRF [5]	Changepoint+Last	**95.6**%	70.8%
k-NN	Changepoint+Last	94.4%	67.9%
C-SVM+CV	Changepoint+Last	95.4%	61%%
C-SVM +Our criterion	Changepoint+Last	92.5%	**72.4 %**

We report in figure 2 the classification results in terms of accuracy measure for each class with CRF, *k*-NN, *C*-SVM+CV and *C*-SVM+our criterion methods. CRF, *k*-NN and *C*-SVM+CV perform better for the majority activities, while *C*-SVM+our criterion performs better for minority activities (other classes).

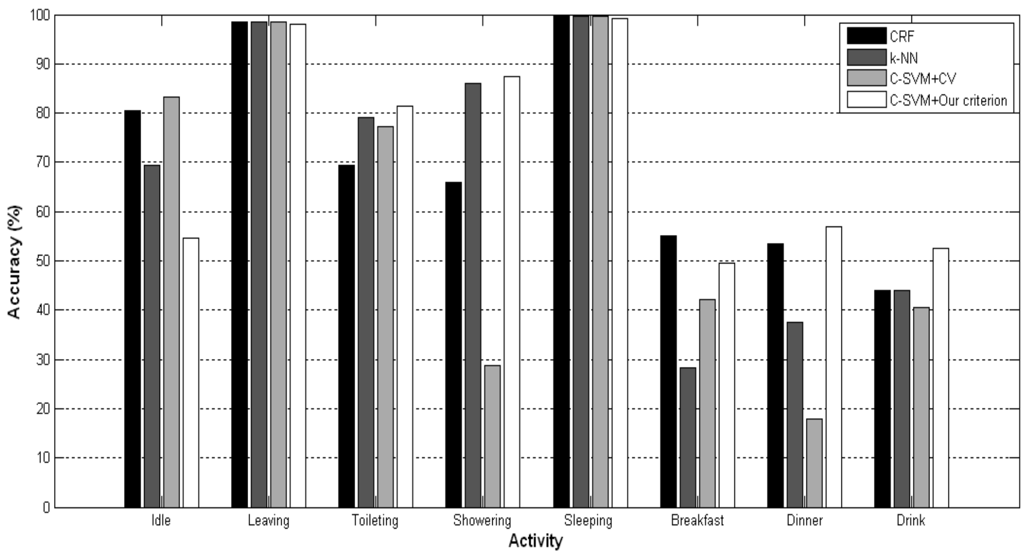

Figure 2. Comparison of accuracy of classification between CRF, *k*-NN, *C*-SVM+CV and *C*-SVM+our criterion for different activities

Finally, we presented a way of compactly presenting all results in a single table 5, allowing a quick comparison between CRF, *k*-NN, *C*-SVM+CV and *C*-SVM+our criterion performed using three real world datasets recorded in three different houses $A_{(2)}$, B, C. We utilize the leave-one-day-out cross validation technique for the selection of width parameter. We found σ_{opt}=1, σ_{opt}=1 and σ_{opt}=2 for these datasets respectively. Our results give us early experimental evidence that our method *C*-SVM combined with our proposed criterion works better for model classification; it consistently outperforms the other methods in terms of the class accuracy for all datasets.

Table 5. Accuracy and class accuracy for CRF, *k*-NN, *C*-SVM+cross validation search and *C*-SVM+our criterion with three houses datasets.

Houses	Models	Class(%)	Accuracy(%)
$A_{(2)}$	CRF [22]	57	91
	$k\text{-}NN_{k=7}$	55.9	90.5
	C-SVM + $CV_{C=5}$	50.3	**92.1**
	C-SVM+our criterion	**62**	88
B	CRF [22]	46	**92**
	$k\text{-}NN_{k=9}$	31.3	67.7
	C-SVM+$CV_{C=5}$	39.3	85.5
	C-SVM+our criterion	**46.4**	62.7
C	CRF [22]	30	78
	$k\text{-}NN_{k=1}$	35.7	78.4
	C-SVM +$CV_{C=500}$	35.6	**80.7**
	C-SVM +our criterion	**37.2**	76.8

3.4. Discussion

Using experiments on three large real world datasets, we showed the class accuracy obtained with house (C) is lower compared to others houses for all recognition methods. We suspect that the use of a hand written diary for annotation (used in house C) results in less accurate annotation than using the bluetooth headset method (used in houses A and B).

In the rest of section, we explain the difference in terms of performance between CRF, *k*-NN, *C*-SVM+CV and *C*-SVM+our criterion for the house $A_{(1)}$. The CRF model does not model each action class individually, but use a single model for all classes. As a result classes that are dominantly present in the data have a bigger weight in the CRF optimisation. This is why CRF performs better for the majority activities ('Idle', 'Leaving' and 'Sleeping'). In *k*-NN method, the class with more frequent samples tends to neighbourhood of a test instance despite of distance measurements, which leads to suboptimal classification performance on the minority class. A multiclass *C*-SVM+CV trains several binary classifiers to differentiate the classes according to the class labels and optimise a single parameter *C* for all class. When not considering the weights in *C*-SVM formulation, this affect the classifiers performances and favorites the classification of majority class. *C*-SVM+our criterion including the individual setting of parameter *C* for each class separately shows that *C*-SVM becomes more robust for classifying the rare activities.

The recognition of the three kitchen activities 'Breakfast' 'Dinner' and 'Drink' is lower compared to the others activities for all methods. In particular, the 'Idle' is one of the most frequent activities in all datasets but is usually not a very important activity to recognize. It might therefore be useful to less weigh this activity. The kitchen activities are food related tasks, they are worst recognized for all methods because most of the instances of these activities were performed in the same location (kitchen) using the same set of sensors. For example, 'Toileting' and 'Showering' are more separable because they are in two different rooms, which make the information from the door sensors enough to separate the two activities. Therefore the location of the sensors is of great importance for the performance of the recognition system.

4. CONCLUSION

This paper introduces a simple criterion that have the power to effectively control the cost of the *C*-SVM learning machine by dealing imbalanced activity recognition datasets. We demonstrate that our proposed strategy is effective to classify multiclass sensory data over common techniques such as CRF, *k*-NN and *C*-SVM using an equal misclassification cost. Usual method for choosing classifiers's parameters, based on grid search using cross validation become intractable as soon as the number of parameters exceeds two. Our criterion using different penalty parameters in the weighted *C*-SVM formulation improves the low classification accuracy caused by imbalanced activity recognition datasets.

REFERENCES

[1] M.Wallace "Best practices in nursing care to older adults" in Try This, issue no. 2, Hartford Institute for Geriatric Nursing, 2007.

[2] C. Bishop, Pattern Recognition and Machine Learning, Springer. New York, ISBN: 978-0-387-31073-2, 2006.

[3] V.N. Vapnik. The Nature of Statistical Learning Theory. (Statistics for Engineering and Information Science). Springer Verlag, Second Aufl., 2000.

[5] T. van Kasteren, A. Noulas, G. Englebienne, and B. Krose, "Accurate activity recognition in a home setting", in UbiComp '08. New York, NY, USA: ACM, 2008, pp. 1-9.

[6] A. Fleury, M. Vacher, N. Noury, "SVM-Based Multi-Modal Classification of Activities of Daily Living in Health Smart Homes : Sensors, Algorithms and First Experimental Results," IEEE Transactions on Information Technology in Biomedicine, Vol. 14(2), pp. 274-283, March 2010.

[7] M.B. Abidine, B. Fergani. Evaluating C-SVM, CRF and LDA classification for daily activity recognition. In Proc. of IEEE Int. Conf. on Multimedia Computing and Systems (ICMCS), pp. 272–277, Tangier-Morocco, May 10-12, 2012.

[8] M.B. Abidine and B. Fergani. Evaluating a new classification method using PCA to human activity recognition. In Proc. of IEEE Int. Conf. on Computer Medical Applications (ICCMA), pages 1-4, Sousse, Tunisia, January 20-22, 2013.

[9] N. Chawla. Data mining for imbalanced datasets: An overview. Data Mining and Knowledge Discovery Handbook, pages 875-886, 2010.

[10] N. V. Chawla, N. Japkowicz, and A. Kotcz, "Editorial: special issue on learning from imbalanced data sets," SIGKDD Explorations, vol. 6, no. 1, pp. 1–6, 2004.

[11] N. Chawla, N. Japkowicz, and A. Kolcz. editors 2003. Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Data Sets.

[12] G. M. Weiss, "Mining with rarity: a unifying framework," SIGKDD Explorations, vol. 6, no. 1, pp. 7–19, 2004.

[13] G.M. Weiss and F. Provost, Learning when training data are costly: the effect of class distribution on tree induction, Journal of Artificial Intelligence Research 19 :315-354, 2003.

[14] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in Proc. of the 15th European Conference on Machine Learning (ECML 2004), pp. 39–50, 2004.

[15] G. Wu and E. Y. Chang, "KBA: Kernel boundary alignment considering imbalanced data distribution," IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 6, pp. 786–795, 2005.

[16] X. Chen, B. Gerlach, and D. Casasent. Pruning support vectors for imbalanced data classification. In Proc. of International Joint Conference on Neural Networks, 1883-88, 2005.

[17] N.V. Chawla, K.W. Bowyer, L.O. Hall, and W.P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research, 16, 321-357, 2002.

[18] Thai-Nghe N. : Cost-Sensitive Learning Methods for Imbalanced Data, Intl. Joint Conf. on Neural Networks, 2010.

[19] K. Veropoulos, C. Campbell and N. Cristianini, "Controlling the sensitivity of support vector machines", Proceedings of the International Joint Conference on AI, 1999, pp. 55-60.

[20] J. Shawe-Taylor and N. Cristianini. "Kernel Methods for Pattern Analysis", Cambridge University Press, p220, 2004.

[21] C. C. Chang and C. J. Lin, LIBSVM: [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin-/libsvm/

[22] T.L.M. van Kasteren, H. Alemdar and C. Ersoy. Effective Performance Metrics for Evaluating Activity Recognition Methods. ARCS 2011 Workshop on Context-Systems Design, Evaluation and Optimisation, Italy, 2011