User-adaptive Models for Recognizing Food Preparation Activities

Sebastian Stein and Stephen J. McKenna
CVIP, School of Computing
University of Dundee
Dundee, United Kingdom
{sstein,stephen}@computing.dundee.ac.uk

ABSTRACT

Recognizing complex activities is a challenging research problem, particularly in the presence of strong variability in the way activities are performed. Food preparation activities are prime examples, involving many different utensils and ingredients as well as high inter-person variability. Recognition models need to adapt to users in order to robustly account for differences between them. This paper presents three methods for user-adaptation: combining classifiers that were trained separately on generic and user-specific data, jointly training a single support vector machine from generic and user-specific data, and a weighted K-nearest-neighbor formulation with different probability mass assigned to generic and user-specific samples. The methods are evaluated on video and accelerometer data of people preparing mixed salads. A combination of generic and user-specific models considerably increased activity recognition accuracy and was shown to be particularly promising when data from only a limited number of training subjects was available.

Keywords

Activity recognition, sensor fusion, accelerometers, computer vision, user modeling

Categories and Subject Descriptors

I.5.5 [**Pattern Recognition**]: Applications; I.4.8 [**Scene Analysis**]: Sensor Fusion; I.2.10 [**Vision and Scene Understanding**]: Video Analysis; K.4.2 [**Social Issues**]: Assistive Technologies for Persons With Disabilities

General Terms

Algorithms, Experimentation, Measurement, Performance.

1. INTRODUCTION

Recognizing complex activities is a challenging research problem with a wide range of potential application areas including situational support, skill assessment, surveillance, content-based video retrieval and video summarization. Food preparation activities are particularly challenging to recognize as they involve complex interactions of a large number of entities and have a high intra-class variability. While vast amounts of training data may enable modelling such complex phenomena, training data are very limited in practice due to the considerable manual effort necessary to record and annotate such data. Therefore, it is important to investigate other approaches to increasing activity recognition accuracy. As food preparation activities are subject to large inter-person variability, we investigate how the adaptation of a recognition system to a particular user may increase recognition robustness in this context.

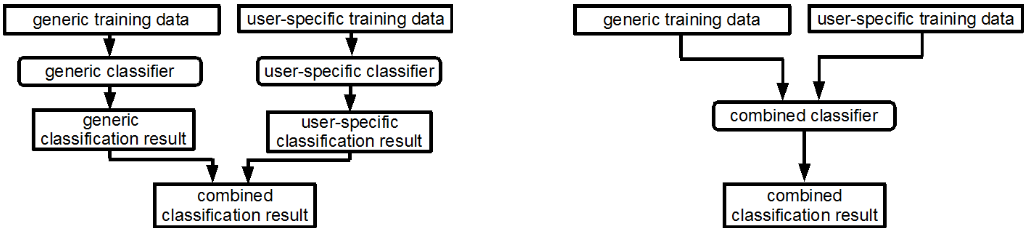
A particularly promising application for activity recognition with a strong potential for social impact is cognitive situational support. For example, an envisaged support system might guide people with dementia through activities of daily living and thereby enable them to live more independently of carers. As such a system would be deployed in a person's home and continuously gather data from the same person, adaptation to the user's idiosyncrasies is desirable.

In this paper we compare three methods for adapting generic, stereotypical activity models to specific individuals whose data were not included in the data used to train the generic models. Firstly, a user-adaptive discriminative recognition model is presented in which a generic and a user-specific classifier are trained independently and merged at test time by combining class posterior probabilities (see Figure 1a). This method does not require all generic training data to be available at the time the model is adapted to a particular subject, which is desirable for practical reasons of system deployment. Secondly, we investigate training a single support vector machine (SVM) jointly on generic and user-specific data (see Figure 1b). This approach requires the whole model to be retrained every time new user-specific data become available. Thirdly, we propose a K-nearest-neighbor classifier in which different probability mass is assigned to user-specific and generic training samples. While K-nearest-neighbor classification allows user-specific data to be added at any time with no cost for retraining, maintaining the entire set of generic data may be practically infeasible.

2. RELATED WORK

There is a large body of work on activity recognition in the computer vision and ubiquitous computing communities. For an overview we refer the reader to recent reviews by Aggarwal and Ryoo [1] and Figo et al. [4]. State-of-

To appear in the Proceedings of CEA'13: 5th International Workshop on Multimedia for Cooking & Eating Activities in conjunction with the 21st ACM International Conference on Multimedia, October 21, 2013, Barcelona, Spain.


generic training data
user-specific training data
generic classifier
user-specific classifier
generic classification result
user-specific classification result
combined classification result
generic training data
user-specific training data
combined classifier
combined classification result

(a) Classifier combination

(b) Joint classifier training

Figure 1: User-adaptive recognition. (a) Classifier combination in which the classification results of generic and user-specific discriminative classifiers are combined at test time. (b) A single classifier is trained on both generic and user-specific data.

the-art approaches to visual activity recognition use bags-of-words over local features such as spatio-temporal interest points [5], point trajectories, and HoG, HoF or MBH descriptors [13]. For recognizing food preparation activities based on accelerometers, Pham et al. [8] proposed statistical features extracted from the temporal domain. Recently, Plötz et al. [9] introduced a method for learning features from accelerometer data. As accelerometers and video data provide complementary information we follow a multi-modal approach to activity recognition, using both accelerometers attached to kitchen objects and an RGBD-video camera [11]. In contrast to body-worn accelerometers [3], accelerometers attached to objects identify the object being moved and can be used for visually localizing and tracking objects without relying on their visual appearance [10].

User-adaptation of recognition systems has been studied in a wide range of application areas including hand-written text recognition [7], speech recognition [12] and recognition of activities from accelerometer data [2]. Nosary et al. [7] proposed an unsupervised method for online-adaptation of a handwritten text-recognition system exploiting lexical context. Tang et al. [12] represent speaker idiosyncrasies by a vector in a low-dimensional space representing variation across speakers, which is adapted using maximum-likelihood estimation on user-specific data. Bao et al. [2] compared classifiers for activity recognition trained exclusively on data from the target user to classifiers trained on data from other users. In this paper we investigate three methods for combining generic and user-specific data from multiple sensor modalities in the context of recognizing food-preparation activities.

3. FEATURES

Accelerometers attached to objects provide information about *which* objects are being moved and capture translational acceleration describing *how* these objects move relative to a local frame of reference. Video data obtained from a stationary camera allow features to be extracted that represent motion relative to a global reference frame and *spatial relationships* of multiple moving objects. As these cues complement each other, we investigate a multi-modal approach to activity recognition using these types of sensors (see Fig. 2). While creating a sensor-rich environment in some cases might seem impractical, the benefit of highly accurate activity recognition systems outweighs the cost of creating a sensor-rich environment in many specialized applications such as cognitive situational support, rehabilitation, skill assessment, and automatic supervision of assembly tasks.

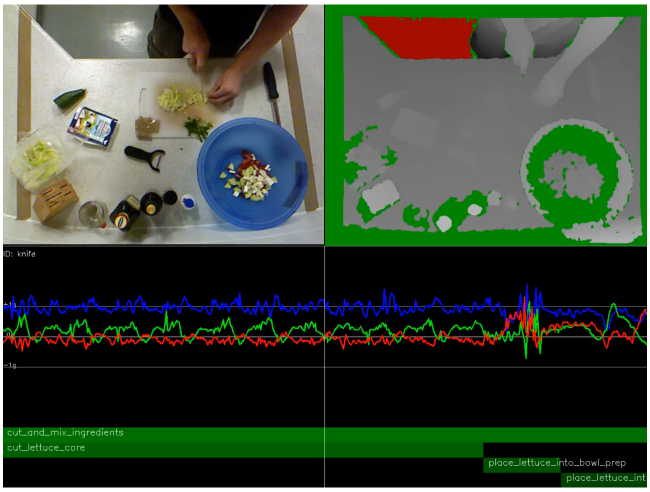


Figure 2: Illustration of the *50 Salads* dataset: RGB-D video data was captured by a camera with top-down view onto the work-surface. Wireless accelerometers were embedded into the handles of kitchen utensils and attached to other kitchen objects. At the time the image above was captured only one out of seven accelerometers was moving.

Typically, features are extracted from accelerometers and video in isolation. In this paper we additionally combine information obtained from accelerometers and video data via accelerometer localization [10] as proposed in [11]. The following subsections briefly describe the types of features we extract from accelerometer and video data. For details we refer the reader to the original publication [11]. The way in which these features are combined for activity recognition depends on the classifier and will be described in Section 4.

3.1 Acceleration Statistics

From each accelerometer we extract statistical features from temporal windows of 256 samples and estimate its orientation. In the temporal domain, the mean, standard deviation, energy and entropy are extracted from acceleration data along each of the three axes separately. Pitch and roll are estimated from four temporal subwindows of 32 samples evenly spaced within a temporal window.

3.2 Accelerometer Localization

In order to establish correspondences between accelerometer and video data we localize accelerometers in the camera's field of view [10]. Each accelerometer is localized by matching acceleration data from the device to acceleration estimated along visual point trajectories. Point trajectories are initialized on a regular grid in the image and their locations are updated in every frame based on displacement vectors in a dense optical flow field. The similarity between acceleration measured by a device $\mathcal{A}_{dev} : (\mathbf{a}^{(0)}_{dev}, \ldots, \mathbf{a}^{(t)}_{dev})$ and acceleration estimated along a point trajectory $\mathcal{A}_{vis} : (\mathbf{a}^{(0)}_{vis}, \ldots, \mathbf{a}^{(t)}_{vis})$ is computed incrementally with a temporal decay α:

$$S_t(\mathcal{A}_{dev}, \mathcal{A}_{vis}) = 1\left[|\mathbf{a}^{(t)}_{vis}| \geq T_{loc} \wedge |\mathbf{a}^{(t)}_{dev}| \geq T_{loc}\right] + \alpha \cdot S_{t-1}(\mathcal{A}_{dev}, \mathcal{A}_{vis}) \quad (1)$$

The location of an accelerometer is estimated as the location of the most similar point trajectory in the last frame.

3.3 Visual Displacement Statistics

The localization algorithm can be used to extract an accelerometer's motion relative to the camera reference frame. From fixed length sequences of visual displacements of each accelerometer we extract the mean, standard deviation, energy and entropy, similarly to Subsection 3.1. Fixed length point trajectories (tracklets) are encoded as a histogram over codebook tracklets following the standard bag-of-words approach [13]. In addition to this histogram of absolute tracklets we construct histograms of tracklets relative to accelerometer trajectories, one for each accelerometer-equipped object.

4. USER-ADAPTIVE RECOGNITION

4.1 Adaptation by Classifier Combination

One approach to adapting a *generic* classifier to a target user involves training a separate classifier from *user-specific* training data and then combining the classification results obtained by the generic and user-specific classifiers. Based on classifiers that estimate posterior probabilities of activity-classes given observations, we can combine these distributions by taking their weighted sum,

$$p(c|\mathbf{o})_{comb} = w_g p(c|\mathbf{o})_g + (1 - w_g) p(c|\mathbf{o})_s, \quad (2)$$

where $w_g \in [0, 1]$ is the weight of the contribution the generic classifier makes to the combined classification result, c denotes the class and $\mathbf{o}$ the observed data. The activity class with maximum $p(c|\mathbf{o})_{comb}$ is considered to be the recognized activity.

We use support vector machines (SVMs) as base classifiers. Class posterior probabilities are estimated as proposed by Wu et al. [14]. In order to combine feature types, $f : 1, \ldots, F$, we estimate the mean of radial basis function kernels,

$$K(\mathbf{x}, \mathbf{x}^{(i)}) = \frac{1}{F} \sum_f exp(-\gamma_f D_f(\mathbf{x}_f, \mathbf{x}^{(i)}_f)), \quad (3)$$

where γ_f is a feature-specific scaling parameter. The distance function, D_f, is the Euclidean distance for *Acceleration Statistics* and *Visual Displacement Statistics*, and the χ^2-distance for *Relative Tracklets*. The scaling parameters, γ_f, for *Acceleration Statistics* and *Visual Displacement Statistics* are estimated by cross-validation. For *Relative Tracklets*, γ_f is set to $\frac{1}{A}$, where A is the mean distance between training samples [15]. We use one-vs-one multi-class SVM as provided by the LibSVM library[1].

4.2 Adaptation by Joint Classifier Training

Adapting a recognition system to a target user by combining generic and user-specific classifiers does not require the full set of generic training data to be available at the time the system is adapted to a target user. The late combination of recognition results may, however, yield suboptimal recognition performance and jointly training a single classifier from both generic and user-specific training data may prove to be superior. In this section we introduce two models for joint classifier training and show how the relative contribution of generic and user-specific data may be modified.

4.2.1 Joint SVM Training

We investigate jointly training SVMs from generic and user-specific training data. The contribution of the set of generic training data and the set of user-specific training data to the objective function can be controlled by the relative number of samples in these subsets. If the number of user-specific samples is greater than the number of samples from generic training data the relative cost of misclassifying user-specific data is greater than the cost of misclassifying generic data. If the number of training samples are equal, generic training samples are taken from a large number of different activity-sequences, and user-specific training data is only available for, e.g., a single sequence, then the cost of misclassifying samples from the user-specific sequence is proportionally higher than the cost of misclassifying samples from each sequence in the generic training data set.

4.2.2 Weighted K-Nearest-Neighbour

A simple yet effective [6] discriminative classification algorithm is K-nearest-neighbour. This algorithm does not involve any learning but needs all training data to be available at test time. The class-posterior distribution given an observation is estimated by identifying the set of K training samples that are closest to the observation in feature space given some distance metric. Let K_c be the number of samples $\mathbf{x}^{(i)}_c$ from class c in this set. The class posterior distribution is defined as the proportion of samples from class c in the set, i.e.,

$$p(c|\mathbf{o}) = \frac{K_c}{K}. \quad (4)$$

This treats all training samples as being equally important. A generalization of Eq. (4) assigns a different probability mass to each training sample, i.e.,

$$p(c|\mathbf{o}) = \frac{\sum_{\mathbf{x}^{(i)}_c \in NN(\mathbf{o})} m^{(i)}}{\sum_{\mathbf{x}^{(j)} \in NN(\mathbf{o})} m^{(j)}}, \quad (5)$$

where $NN(\mathbf{o})$ is the set of K nearest neighbours of $\mathbf{o}$. This is equivalent to Eq. (4) if $m^{(i)} = 1/N$ for all training samples. The relative contributions of generic and user-specific data to the estimated class-posterior probability can be adjusted using Eq. (5) by assigning probability mass m_g to all generic samples and probability mass m_s to all user-specific samples. In the case where m_s is greater than m_g the class-posterior probability $p(c|\mathbf{o})$ is higher if there is a user-specific

[1] LibSVM: `www.csie.ntu.edu.tw/~cjlin/libsvm/`

training sample of class c in the local neighborhood of the observation.

For fusing different feature types we combine the distance metrics defined over the individual feature spaces:

$$D(\mathbf{x}, \mathbf{x}^{(i)})_{comb} = \sum_f \gamma_f D_f(\mathbf{x}_f, \mathbf{x}_f^{(i)}), \qquad (6)$$

where γ_f and D_f are defined as described in Sec. 4.1.

5. EVALUATION

5.1 Dataset & Evaluation Protocol

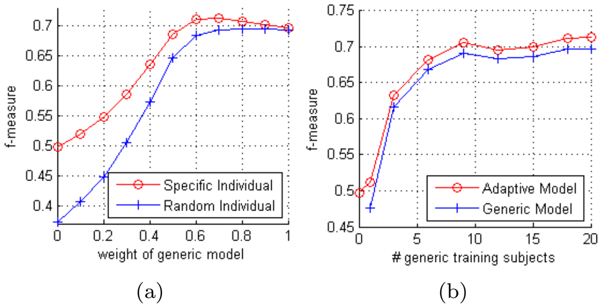
The proposed methods are evaluated on the *50 Salads* dataset [11]. The dataset contains RGBD-video data and data from accelerometers attached to kitchen objects acquired while 25 people prepared two mixed salads each (see Fig. 2). The subjects cover a wide range of age, gender, ethnicity and food-preparation experience. Preparing the salad involved mixing a dressing, cutting ingredients into pieces, mixing the ingredients, serving the salad onto a plate and adding the dressing to the salad. The order in which steps in the recipe were executed was randomly sampled from a statistical recipe model. Accelerometers were attached to a knife, a peeler, a large spoon, a small spoon, a dressing glass, a pepper dispenser and an oil bottle.

The recognition task was to classify temporal sliding windows into one out of $|\mathcal{C}| = 10$ activity classes: *give pepper, add oil, mix dressing, peel cucumber, cut ingredient, place ingredient into bowl, mix ingredients, serve salad,* and *NULL*, where *NULL* indicates that none of the other activities currently occurs. The dataset was split into 5 cross-validation partitions containing two sequences of each of 20 subjects for training and two sequences of each of 5 subjects for testing. No subject was part of both training and test set in any partition. For generic classifier training a stratified sample of $N_g = 5000$ data-points was used. In order to evaluate user-adaptation, models were additionally trained on samples from one user-specific sequence for each subject in the test-set. Two models were trained per subject using one sequence for training and the other one for testing, and vice versa. As some activities only last a few frames and some might not occur at all in any particular sequence, the user-specific training data was highly imbalanced. An approximately-stratified sample was taken from the user-specific training sequence, including all samples from activities for which not more than $N_s/|\mathcal{C}|$ samples were available, and a stratified sample from all other classes. For SVM training the regularization parameter was set differently for each class based on the number of samples from that class in the training set: $C_c = \frac{N_s}{|\mathcal{C}| N_c}$. The scaling parameters, γ_f, were determined by 5-fold cross-validation and set to $\gamma_{AS} = 0.03125$ for *Acceleration Statistics* and $\gamma_{VDS} = 2$ for *Visual Displacement Statistics* in all experiments. Activity recognition performance was evaluated using average precision and average recall over activity classes, and their harmonic mean (f-measure).

5.2 Empirical Results

5.2.1 Adaptation by Classifier Combination

Adding more training data to a learning algorithm may improve recognition accuracy regardless of whether these additional training data are from the user the system is evaluated on or not. To investigate whether our models actually learn user-specificities we performed a randomized control trial, in which the adaptive model trained on the same subject was compared with an adaptive model trained on a randomly selected other subject from the test set. We refer to these cases as *Specific Individual* and *Random Individual*, respectively. Recognition performance as f-measure with varying weight w_g for mixing the generic with the individual model is plotted in Figure 3a. Combining the generic model with the specific individual model improved performance by 0.017 to 0.714 with a peak at $w_g = 0.7$, whereas the combination with a model trained on a random individual did not increase recognition performance. Note that this improvement is based on a single sequence and is expected to increase with additional user-specific data.



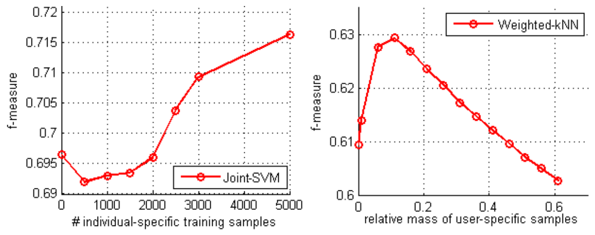
(a) (b)

Figure 3: Recognition performance of user adaptation via classifier combination with (a) varied mixing weight w_g, and (b) varied number of generic training subjects using $w_g = 0.7$.

When training data are scarce, e.g., only a single training sequence is available, it seems particularly valuable for this data to be obtained from the target user. When $w_g = 0$ in Figure 3a, which represents training a single classifier from a single training sequence, we observe 0.125 absolute and 0.335 relative performance increase switching from training on a random subject to training on the target user. We verified this hypothesis by training a classifier from data of a varied number of random subjects. The generic classifier was combined with an individual-specific classifier using weight $w_g = 0.7$, whereby the ratio of training sequences from random subjects to training sequences from the target user was varied. Results in Figure 3b show that the performance gain from user-adaptation reached its minimum when training the generic classifier on data from three subjects, and that the improvement remained approximately constant even after adding data from a further 17 subjects. This experiment confirmed that high gains from user-adaptation can be expected when training data are only available from a few subjects. Furthermore, the maintained performance gain indicates that user-adaptation is beneficial even if data are gathered from a large number of subjects.

5.2.2 Adaptation by Jointly Training SVM

User adaptation by training a single SVM on data from randomly selected subjects and on the target user was evaluated by varying the number of user-specific training samples. Results in Figure 4a show that the performance initially dropped but then continuously rose with added user-specific data to 0.716. The initial drop is due to our method for es-



(a) Joint SVM (b) Weighted kNN

Figure 4: Recognition performance of user adaptation via joint classifier training (a) SVM with varied number of training samples from the target user, and (b) weighted K-Nearest-Neighbor with varied probability mass m_s of user-specific samples.

timating class-dependent regularization parameters C_c. In the presence of imbalanced data, which only occurred when user-specific data were added to the training set, the regularization parameters only provide valid *relative* weights between classes. The *absolute* regularization for a given binary optimization problem can however be greater or smaller than with a stratified sample. The highest performance gain (0.020) was observed when 5000 user-specific training samples were added to the generic training data. While absolute performance after user-adaptation by jointly training a single SVM was higher compared to combining classifiers, retraining with generic data considerably increased the training time.

5.2.3 Adaptation with Weighted K-Nearest-Neighbour

For weighted K-nearest-neighbour classification we added 5000 user-specific samples to the generic training data and the number of samples in the local neighbourhood of a test sample was set to $K = 64$ which showed good recognition performance in preliminary evaluation. Evaluation results with varied relative probability mass for user-specific training samples are shown in Figure 4b. Although the highest observed performance increase (0.020) after adding user-specific data with $m_s = 0.11 \cdot m_g$ was comparable to SVM joint training, the absolute recognition performance of 0.63 was not competitive. As there was considerable overlap between temporal windows sampled from user-specific data, the i.i.d. assumption was strongly violated. This explains why the optimal probability mass m_s was below m_g.

5.2.4 Variation across Individuals and Activities

The previous experiments showed that the proposed methods for user-adaptation are well suited to capturing idiosyncrasies. The benefit of adapting a recognition model to a particular subject intuitively depends on the difference of their task-execution *style* from the norm, and on their *consistency*. We expect the recognition performance after user-adaptation to increase if the execution style is particular to the user, and to decrease after training on one sequence if the target user is inconsistent. Figure 5 illustrates the variation in performance gained after user-adaptation across individuals. It provides evidence confirming the intuition, showing that user-adaptation is beneficial for most users, particularly beneficial for some, and has adverse effects for others.

Similarly, the performance gain from user-adaptation varies across activities. Some activities may be performed in many different ways potentially involving different utensils. Additionally, as the amount of available user-specific data may vary across activities, higher gains are expected for activities for which more user-specific data are available. Figure 6 shows performance gain per activity obtained with the proposed methods. Particularly high gains were observed for activities that can be executed with a wide range of strategies. Placing ingredients into the bowl, for example, can be performed by grasping the cut ingredients by hand and placing them into the bowl, picking up the chopping board and scraping the ingredients into the bowl by hand, or scraping the ingredients off the chopping board using a knife. In contrast, the gain from user-adaptation for *cutting into pieces* was below average, indicating only minor variation in strategy across users; more than 40% of the time during which data were acquired consisted of this activity.

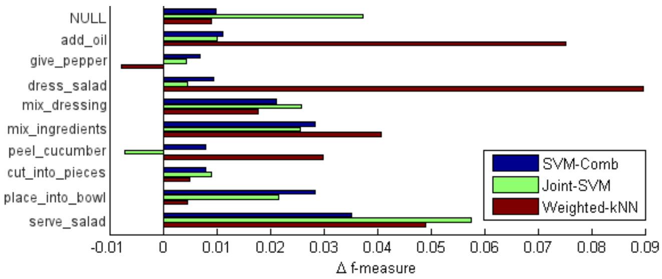


Figure 6: Performance gains for each activity.

6. DISCUSSION & FUTURE WORK

In this paper we presented three methods for user-adaptive activity recognition: combining classifiers that were trained separately on generic and user-specific data, jointly training a single SVM from generic and user-specific data, and a weighted K-nearest-neighbor formulation with different probability mass assigned to generic and user-specific samples. These methods were evaluated on a multi-modal activity recognition dataset of food preparation activities.

The experiments confirmed that adapting an activity recognition system to a target user can considerably increase recognition accuracy. Via randomized control trials we have further shown that this performance increase is indeed attributable to user idiosyncrasies as opposed to being a mere consequence of additional data available for training. Variation in performance increase after user adaptation across individuals indicates that users with execution style that deviates from the norm benefit most from adaptation. Similarly, user-adaptation is particularly advantageous for tasks that may be accomplished with a wide range of execution strategies, such as many tasks involved in food preparation.

Which of the presented methods for user-adaptation to apply in a particular scenario depends on the specific constraints on storage space and computation time. If there are no constraints on storage space and time for classifier retraining, jointly training a single SVM from generic and user-specific data is expected to achieve higher recognition performance than the other presented methods. The weighted K-nearest-neighbor approach would be recommended in the extreme case where only minimal computation time for initial classifier training and user-adaptation is available. If both storage space and retraining time are limited but not

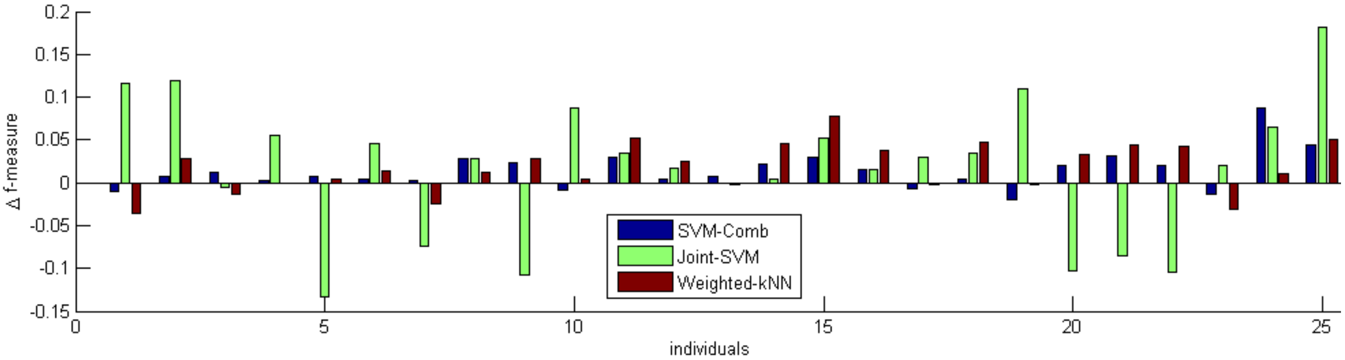


Figure 5: Variation in performance gained through user-adaptation across individuals.

negligible, combining classification results from separately trained SVMs should be chosen out of the three methods discussed in this paper. While it showed slightly lower performance gain from user-adaptation than joint SVM training and weighted K-nearest-neighbor classification, absolute recognition performance was comparable to the best observed performance. This model provides a reasonable compromise that may be particularly useful in practical applications.

Future work includes evaluation of the presented models with more than a single training sequence of user-specific data, and in the context of recognizing activities involved in multiple recipes. Further directions for improved recognition of food preparation tasks include temporal models that effectively exploit the recipe structure, and adding visual appearance features for differentiating between activities with respect to the ingredients.

7. ACKNOWLEDGEMENTS

This research is funded by RCUK Digital Economy Research Hub EP/G066019/1 SIDE: Social Inclusion through the Digital Economy, from December 2010 until May 2014. We would like to thank Kyle Montague for useful suggestions.

8. REFERENCES

[1] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys*, 43(3):16:1 – 16:43, 2011.

[2] L. Bao and S. S. Intille. Activity recognition from user-annotated acceleration data. In *Proceedings of the 2nd International Pervasive Computing Conference, Linz/ Vienna, Austria*, pages 1–17, 2004.

[3] A. Behera, D. C. Hogg, and A. G. Cohn. Egocentric activity monitoring and recovery. In *The 11th Asian Conference on Computer Vision (ACCV'11), Daejeon, Korea*, 2012.

[4] D. Figo, P. C. Diniz, and D. R. Ferreira. Preprocessing techniques for context recognition from accelerometer data. *Personal and Ubiquitous Computing*, 14(7):645 – 662, 2010.

[5] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2/3):107–123, 2005.

[6] A. Niculescu-Mizil and R. Caruana. Obtaining calibrated probabilities from boosting. In *Conference on Uncertainty in Artificial Intelligence (UAI'05), Edinburgh, UK*, 2005.

[7] A. Nosary, L. Heutte, and T. Paquet. Unsupervised writer adaptation applied to handwritten text recognition. *Pattern Recognition*, 37:385–388, 2004.

[8] C. Pham and P. Oliver. Slice&Dice: recognizing food preparation activities using embedded accelerometers. *Ambient Intelligence, LNCS*, 5859:34–43, 2009.

[9] T. Plötz, N. Y. Hammerla, and P. Olivier. Feature learning for activity recognition in ubiquitous computing. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, pages 1729 – 1734, 2012.

[10] S. Stein and S. J. McKenna. Accelerometer localization in the view of a stationary camera. In *Proceedings of the 9th Conference on Computer and Robot Vision (CRV'12), Toronto, Ontario, Canada*, pages 109 – 116, 2012.

[11] S. Stein and S. J. McKenna. Combining embedded accelerometers with computer vision for recognizing food preparation activities. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), Zürich, Switzerland*, 2013.

[12] Y. Tang and R. Rose. Rapid speaker adaptation using clustered maximum-likelihood linear basis with sparse training data. *IEEE Transactions on Audio, Speech, and Language Processing*, 16:607–616, 2008.

[13] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'11), Colorado Springs, Colorado, USA*, 2011.

[14] T.-F. Wu, C.-J. Lin, and R. C. Weng. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, 5:975–1005, 2004.

[15] J. Zhang, M. Marszalek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. *International Journal of Computer Vision*, 73(2):213–238, 2007.