Increased Robustness in Context Detection and Reasoning Using Uncertainty Measures: Concept and Application

Martin Berchtold and Michael Beigl

Distributed and Ubiquitous Systems, TU Braunschweig
Muehlenpfordtstr. 23, 38106 Braunschweig, Germany

Abstract. This paper reports on a novel recurrent fuzzy classification method for robust detection of context activities in an environment using either single or distributed sensors. It also introduces a classification of system architectures for uncertainty calculation in general. Our proposed novel method utilizes uncertainty measures for improvement of detection, fusion and aggregation of context knowledge. Uncertainty measurement calculations are based on our novel recurrent fuzzy system. We applied the method in a real application to recognize various applause (and non applause) situations, e.g. during a conference. Measurements were taken from mobile phone sensors (microphone, accel. if available) and acceleration sensory attached to a board marker. We show that we are able to improve robustness of detection using our novel recurrent fuzzy classifier in combination with uncertainty measures by ∼30% on average. We also show that the use of multiple phones and distributed recognition in most cases allows to achieve a recognition rate between 90% and 100%.

1 Introduction

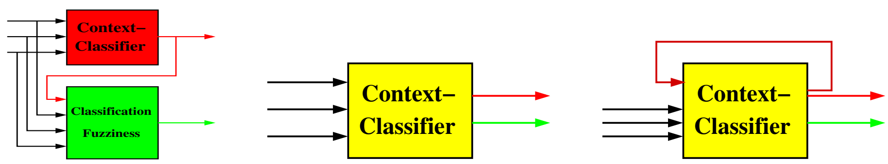
The detection of surrounding situations or contexts has been an interesting area of research for almost a decade. Robust context recognition could have many applications in office or industrial environments. In this paper we focus on a more playful application area, that is nevertheless very challenging: the detection of clapping events. The recognition system we present does not assume any a-priori knowledge regarding the sensors being used or their placement. As sensors we use mobile phones with microphones and optional acceleration sensors. The phones may be carried in a pocket or rest on the table. Our system is able to handle unsteady detection quality, aggregate classifications from different sources and still classifies situations correctly to a high percentage. In context recognition, measures to express the confidence of a detected context can be very helpful to improve the overall robustness. Some authors, e.g. Bucholz et al. [1] refer to this confidence level as "Quality of Context (QoC)". [2] shows the design of quality extensions for context ontologies and how fuzzy set theory can be used for context ontology matching under uncertainty. None of these publications describes a method how such a quality could be derived. We show

M. Tscheligi et al. (Eds.): AmI 2009, LNCS 5859, pp. 256–266, 2009.
© Springer-Verlag Berlin Heidelberg 2009

how systems can be designed that deliver a QoC measure, although we use the term "Uncertainty" instead of "Quality" in reference to the wording in classical AI literature. Support for reasoning about uncertain contexts with probabilistic logic, fuzzy logic and Bayesian networks is described in [3]. How to model uncertainty in context-aware computing is described in [4], but the method for uncertainty measure calculation is not described and there is also no evidence given how uncertainty measures can improve robustness in reasoning. We will present uncertainty measures, their computation and also evaluate their benefits in this paper. In our proposed approach uncertainty measures are derived using a recurrent fuzzy inference system (sec. 3). We also evaluate how uncertainty can be used throughout the further inference processes - e.g. fusion and aggregation of contexts - to increase reliability of classification (sec. 4+5). Furthermore, the paper contains a first system architecture typification for systems and classifiers that are able to produce uncertainty measures (sec. 2).

2 Various Methods of Calculating Uncertainty Measures

There are three general methods of computing an uncertainty value in a context classification system. These three methods correspond to possible system architecture styles or types for uncertainty measurement derivation, as shown in figure 1. Which of these styles are suitable depends on the classification method, but also on the specific setting in an application context. The most general architecture style is Parallel Uncertainty Calculation (fig. 1(a)). In this system architecture style a context classifier works in parallel to an uncertainty detector (here called classification fuzziness). The uncertainty classifier thus behaves like an independent observer that constantly evaluates the output of the context classifier. Such systems are useful if methods for classification and evaluation of the classification differ. [5] shows that this approach is very beneficial for filtering contexts. A more compact classification is the Implicit Uncertainty System Architecture (fig. 1(b)). An examplary implementation of this architecture style are Fuzzy Inference Systems (FIS)[6]. Here, fuzziness from within the mapping FIS can be used to derive the uncertainty level. E.g. in a TSK-FIS the outcome requires interpretation of the mapping outcome using a membership function.



(a) Parallel Uncertainty Calculation Architecture

(b) Implicit Uncertainty Calculation Architecture

(c) State Dependent Uncertainty Calculation Architecture

Fig. 1. Different system architecture styles to compute uncertainty of a context classification

The disadvantage of this method is, that only the fuzziness of the mapping model can be detected, with only small variations among different classes. The uncertainty of the system configuration itself will thus not be taken into account when calculating the uncertainty level. The third architecture style is State Depended Uncertainty calculation 1(c). An implementation example of this architecture are recurrent fuzzy classification systems. Here it might be possible to solve problems arising in Implicit Uncertainty Calculation Architectures. Although this paper focuses on the use of uncertainty measures for filtering, we will also show in our system implementation how the measure is computed. Our system implements filter-like behavior and provides a fuzzy uncertainty on which a fusion or aggregation step is significantly improved when using uncertainty measures.

3 Offline Identification and Online System

3.1 Offline Identification Algorithm for RFIS-Classifier

A Recurrent Fuzzy Inference System (RFIS) is used to map sensor data features onto a classifiable linear set. The general idea behind Recurrent Fuzzy Systems (RFS) can be found in [7]. This soft system needs to be identified upon an annotated training feature set via a combination of a clustering algorithm and linear regression. Usually the identification of a Fuzzy Inference System (FIS) needs only one step of clustering, but since we use a recurrent one, each new mapping result leads to a new set of input data, upon which another iteration of clustering needs to be performed. The algorithm for identifying the RFIS consists of the following steps:

1. Data Separation: The training data is separated according to the class the data pairs belong to. Clustering on each subset delivers rules that can be assigned to each class. **2. Subtractive Clustering:** Subtractive clustering [8] per subset identifies the number of rules and the membership functions of each rule's antecedent without having to declare how many clusters there are. **3. Least Squares:** A linear regression identifies the linear functional consequence of the rules. The least squares method minimizes the quadratic error, which is the quadratic distance between the desired output and the actual output of the TSK-FIS classifier for the training data set. Minimizing the quadratic error leads to an overdetermined linear equation to be solved. **4. Recurrent Data Set:** The recurrent TSK-FIS is obtained over a data set that has the output of the previously identified FIS shifted by one, so the first data pair of the training set has a zero in the recurrent dimension. All data pairs for time $t > 1$ have the output of the FIS mapping of $t-1$ in the recurrent dimension. For this data set the steps 1 to 3 are repeated. **5. Stop Criterion:** We could not find a general stop criterion, since two demands need to be met. The resulting classifier needs to have high accuracy and the outcome needs to have an uncertainty level that is of profit for reasoning. Therefore the developer has to decide, according to a separate check data set, what good results for the classifier and its uncertainty levels are. The steps 1 to 4 are repeated and graphically observed until the developer recognizes a good outcome.

The RFIS identified through this algorithm is the key component of the sensor data classifier. This RFIS also provides the desired fuzzy uncertainty described previously.

3.2 Online Recurrent Fuzzy Classifier

The online recurrent fuzzy classifier consists of several steps of processing from a real world value to a tuple of class and fuzzy uncertainty. The first step is the sensory, that converts the real world signal into a digital measurement. Secondly, the desired features are extracted from the measurement. In the third step the Recurrent Fuzzy Inference System (RFIS) maps the features onto a classifiable linear set. The outcome of the mapping at time t gets fed back as part of the input at $t+1$. The linear set gets fuzzily classified according to designated fuzzy numbers in the last step. All steps are diagrammed in figure 2.

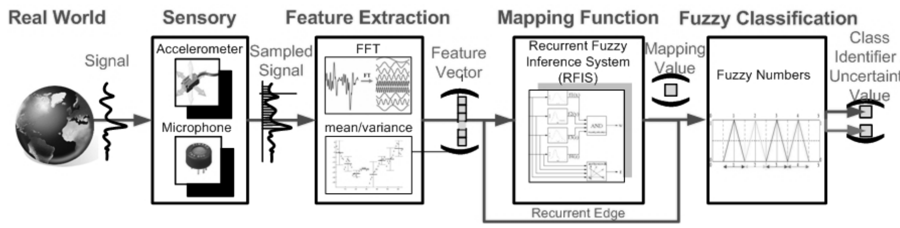


Fig. 2. Online system architecture for classification and fuzzy uncertainty

1. Feature Extraction: The features used for activity recognition with acceleration measurements are mostly variance and mean values, since they are easily calculated and give good classification results. These features were used to preprocess the accelerometer data from the "OpenMoko" phones and the "Freescale ZSTAR". For audio data the standard extraction method is a "Fast Fourier Transformation (FFT)", which is also used to extract the frequency features for the audio classification. The audio sources are the microphones of the "OpenMoko". **2. Recurrent FIS Mapping:** Takagi, Sugeno and Kang [9] (TSK) fuzzy inference systems are fuzzy rule-based structures, which are especially suited for automated construction. Within a TSK-FIS, the consequence of the implication is not a functional membership to a fuzzy set, but a constant or linear function. A TSK-FIS is used to map the extracted features onto a linear set, whose values can be assigned to a class identifier in a separate classification process. The outcome of the mapping at time t is fed back as additional input dimension for the TSK-FIS mapping at $t + 1$. The recurrency not only delivers the desired uncertainty level, but also stabilizes and improves the mapping accuracy. **3. Fuzzy Classification:** The outcome of the TSK-FIS mapping needs to be assigned to one of the classes the projection should result in. This assignment is done fuzzy, so the result is not only a class identifier, but also a membership identifying the fuzzy uncertainty of the classification process. Each class identifier is interpreted as a triangular shaped fuzzy number. The mean of the fuzzy number is the identifier itself, with the highest membership of one. The crisp decision, which identifier is the mapping outcome, is carried out based on the highest degree of membership to one of the class identifiers. The overall output of the RFIS mapping and the classification is a tuple $(C_{\mathcal{A}}, \mu_{\mathcal{A}})$ of a class identifier and the membership to it.

4 Reasoning with Uncertainty

The reasoning about uncertain contexts is residing on the second level of information processing, where context information has been already inferred from sensor data. On this level logics or ontologies are usually used to infer new contextual knowledge. After our classification process we end up with tuples of class identifiers and fuzzy uncertainty measures, e.g. $(C_{\mathcal{A}}, \mu_{\mathcal{A}})$. The usual fuzzy modus ponens used to derive new knowledge has various definitions throughout the literature, e.g. [10]. These inference methods are complex and need a brief introduction, if they are used. Since we focus on fuzzy uncertainty and how it can improve accuracy, the consequences information content and the further inference, we used a different, simpler method to prove our point. The inference of the contexts is done crisp with simple propositional logic and the derivation of the uncertainty is done accordingly through a fuzzy t-norm/t-conorm.

4.1 Fusing Equal Contexts with Uncertainty

The idea behind the fusion of contexts is to use equal contexts from different sources in order to achieve mutual confirmation. In the crisp case contexts get fused based solely on their occurrence in the same time period. Although the overall probability is improved with each mutual confirmative crisp context included in the fusion, the reliability of each fusion member and outcome can vary strongly. This variation in reliability of each context is lost in the merging of crisp contexts and all fusions with the same number of members have the same probability of correctness. Fusion based on a fuzzy uncertainty level has a lot more to offer. If the fusion is done fuzzy according to the uncertainty level of each context, the confidence is not lost in the fusion process. Even if many mutual confirmative contexts each have a low confidence level, the fused context gains reliability. For the fusion of uncertainty we used the fuzzy equivalence to a crisp disjunction, the t-conorm. Many different t-conorms appear throughout the literature, we decided to use the probabilistic sum $(S_P(x, y) = x + y - x \cdot y)$. The result of the probabilistic sum is higher than each input of the t-conorm, which suits our understanding of the fusion process. An example for the fusion of context $C_{\mathcal{C}}$ out of the two contexts $C_{\mathcal{A}}$ and $C_{\mathcal{B}}$ and the fusion of the fuzzy uncertainty accordingly, is the following:

$$\left.\begin{array}{rcl} C_{\mathcal{A}} \vee C_{\mathcal{B}} & \rightarrow & C_{\mathcal{C}} \\ S_P(\mu_{\mathcal{A}}, \mu_{\mathcal{B}}) & = & \mu_{\mathcal{A}} + \mu_{\mathcal{B}} - \mu_{\mathcal{A}} \cdot \mu_{\mathcal{B}} \\ & = & \mu_{\mathcal{C}} \end{array}\right\} (C_{\mathcal{A}}, \mu_{\mathcal{A}}) \vee (C_{\mathcal{B}}, \mu_{\mathcal{B}}) \rightarrow (C_{\mathcal{C}}, \mu_{\mathcal{C}}) \tag{1}$$

4.2 Aggregate New Contextual Knowledge with Uncertainty

Through aggregation contextual knowledge of many sources is combined to new contextual facts. The crisp decision is a typical application for propositional and predicate logic. The antecedent part in combination with the rule determines the conclusion, which is the typical modus ponens inference. In this kind of inference the reliability of each input source is not taken into account. Also the general uncertainty of each context classification is not considered. These uncertainties have a huge impact on the outcome of the inference process. We will show how a fuzzy aggregation can improve the outcome in the evaluation section. For the aggregation of new contexts we used simple propositional logic and for the inference of the uncertainty we used a t-norm. Since the result of the fuzzy inference needs to be less reliable than any of the inputs, the best t-norm is in our understanding the product norm ($T_P(x, y) = x \cdot y$). An example for the aggregation of context $C_\mathcal{C}$ out of the contexts $C_\mathcal{A}$ and $C_\mathcal{B}$ and inference of the fuzzy uncertainty according to the context aggregation, is the following:

$$\left.\begin{array}{rcl} C_\mathcal{A} \wedge C_\mathcal{B} & \rightarrow & C_\mathcal{C} \\ T_P(\mu_\mathcal{A}, \mu_\mathcal{B}) & = & \mu_\mathcal{A} \cdot \mu_\mathcal{B} \\ & = & \mu_\mathcal{C} \end{array}\right\} (C_\mathcal{A}, \mu_\mathcal{A}) \wedge (C_\mathcal{B}, \mu_\mathcal{B}) \rightarrow (C_\mathcal{C}, \mu_\mathcal{C}) \tag{2}$$

5 Evaluation - “Detecting Acclamation”

To evaluate the classification, fusion and aggregation we used parts of the office scenario. The aim was to show that, compared to the simple fusion of context classes, the reliability for fusion of identical context classes from different sources improves, if a uncertainty value is used as weight. The second argument for using an uncertainty value in the inference process is the aggregation of new contextual knowledge. The weighted aggregation should show improvement in reliability towards crisp inference. We used two ”OpenMoko Freerunner” devices and one ”Freescale ZSTAR” demo as sensor data sources. For the ”Freerunner's” two recurrent FIS (RFIS) classifiers were used, with the following classes each:

1. 10-point FFT, 1000-sample window, audio at 4kHz ⇒ 10-dim. input vector

”silence”	(class no. 1)	⇒	no audio except noise
”talking to audience”	(class no. 2)	⇒	speech data
”knocking appreciation”	(class no. 3)	⇒	knocking on table
”clapping applause”	(class no. 4)	⇒	clapping hands

2. variance and mean, 8-sample window, two 3-axis accel. ⇒ 12-dim. input vector

”lying still”	(class no. 1)	⇒	no movement of device
”knocking appreciation”	(class no. 2)	⇒	knocking on table with device next to it
”sitting”	(class no. 3)	⇒	device in users pocket whilst sitting
”standing”	(class no. 4)	⇒	device in users pocket whilst standing
”walking”	(class no. 5)	⇒	device in users pocket whilst walking

The ”ZSTAR” was attached to a board marker and was running also a RFIS classifier, classifing on the following classes:

3. variance and mean, 8-sample window, two 3-axis accel. ⇒ 6-dim. input vector

"lying still"	(class no. 1) ⇒	no movement of device
"knocking appreciation"	(class no. 2) ⇒	knocking on table with marker next to it
"sitting"	(class no. 3) ⇒	marker in users pocket whilst sitting
"standing"	(class no. 4) ⇒	marker in users pocket whilst standing
"writing"	(class no. 5) ⇒	writing on whiteboard

Data was recorded on several controlled test runs with five subjects. A sequence of the classes was simulated to reflect a conference event.

5.1 Fuzzy Classifiers vs. Recurrent Fuzzy Classifiers

One feature of the recurrent fuzzy classifier is the desired classifications fuzzy uncertainty, the other one is the improvement of the classification process towards normal non-recurrent classifiers. To show the improvements in accuracy, we compared a normal FIS based classifier with our RFIS classification process. FIS uses the same algorithm as RFIS, except there is only one iteration of clustering and linear regression. The feedback of the RFIS stabilizes the classification process significantly. The most incorrect classifications are made when there is a change from one class to another one. To evaluate this disadvantage we used a check data set that reflects this insufficiency. The check data set consists of subsets (30 data pairs each) of class specific patterns (many subsets per class), which where randomly ordered: **1. "OpenMoko" audio** - 1530 training data pairs (TDP) (382,5 sec) → ∼51 successive class changes (SCC); 1500 check data pairs (CDP) (375 sec) → ∼50 SCC. **2. "OpenMoko" acc**. - 1410 TDP (352,5 sec) → ∼47 SCC; 1770 CDP (442,5 sec) → ∼59 SCC. **3. "ZStar" acc.** - 660 TDP (165 sec) → ∼22 SCC; 450 CDP (112,5 sec) → ∼15 SCC.

The feedback before the first classification is always 0, which means not identifying any class. Despite these challenges RFIS performed significantly better for all three classifiers than FIS. The confusion matrices for both phone classifiers, FIS and RFIS classifiers are shown in table 1 and 2. Table 1 shows the results of the accelerometer data classifier, where the overall correct classifications of FIS are ∼62% and for RFIS ∼94%. This shows an improvement of about 32%. The results of the RFIS audio classifier show even more improvement. As displayed in table 2, the RFIS classifier shows an enhancement from ∼24% to ∼92%. The classification accuracy of the FIS classifier indicates that the patterns are not separable with this method. The improvement for the "ZSTAR" attached to a

Table 1. Conf. mat. of FIS ∼62% (left) and RFIS ∼94% (right) accel. class. phone

designat. classes	classes classified onto 1	2	3	4	5
1	0	100.00	0	0	0
2	0.74	99.26	0	0	0
3	0	8.33	90.00	1.67	0
4	0	0	0.3344	99.67	0
5	2.33	0.33	0.33	10.33	86.67

designat. classes	classes classified onto 1	2	3	4	5
1	90.50	3.00	2.33	4.17	0
2	0.74	96.67	2.5926	0	0
3	0	0.33	94.67	3.67	1.33
4	0	0	0	99.67	0.33
5	0	0.33	0.67	8.00	91.00

Table 2. Confusion mat. of FIS ∼24% (left) and RFIS ∼92% (right) for audio class

		classes class. onto			
		1	**2**	**3**	**4**
desig. classes	**1**	0.0	98.50	1.50	0.0
	2	0.0	35.74	63.65	0.60
	3	0.0	22.71	65.22	12.07
	4	0.0	14.16	63.01	22.83

		classes classified onto			
		1	**2**	**3**	**4**
desig. classes	**1**	88.83	5.50	3.17	2.50
	2	0.20	99.80	0.0	0.0
	3	0.0	7.25	82.61	10.14
	4	0.0	0.0	7.31	92.69

board marker is not as significant as with the phone classifiers, but still amounts about 2% (from ∼88% to ∼90%). This results shows the advantage of recurrent classifiers in the field of sensor data processing.

5.2 How Fusion with Uncertainty Improves the Accuracy

The aim in this evaluation is to indicate the improvement of fuzzy context fusion towards normal crisp fusion in overall accuracy. To show this, the fusion of context classes which vary in classification correctness and according to that in accuracy needs to be made. The differentiation between the classes "lying still" and "knocking appreciation" of the marker classifier provides the desired uncertainty and shaky classification. The fusion with a more precise classification of the context "knocking appreciation" should improve the overall classification. Improvement is achieved through filtering upon the fused uncertainty level. The aim is to sort out the false classifications according to a lower uncertainty level. How a threshold for filtering can be found was shown in [5] and is generally known as "receiver operator characteristics". Another classification qualifies for fusion, the classification of the audio data on "knocking appreciation" when the phone is carried in the pocket. This classification should also improve if being fused with the same classification of a phone lying freely on a table. The following combinations of contexts, devices and device states are fused:

1. Phone A is lying on the table and phone B is in users pocket. Both should recognize context class "knocking appreciation" through the audio classifier.

2. Phone A is lying on the table recognizing "knocking appreciation" through audio and the board marker is also lying on the table recognizing the same class through the accelerometer classifier.

In the following plots the different fuzzy uncertainty levels of the fused contexts are plotted along with the samples from the test data set. The bounded areas which are signed out with "correct classified" show time periods the contexts actually happened. Mean values of the fuzzy uncertainty for correct and incorrect classifications are plotted in the figures as dashed lines. The greater the distance between these dashed mean lines is, the better correct classifications can be separated from incorrect ones. The results of fusion (1) can be seen in figure 3(a). The filtering on the uncertainty level at threshold $\tau = 0.9$ improves the accuracy by about 6% from ∼90% to ∼96%. In this example samples are not as clearly separable as in the following one, but still an improvement can be achieved. The

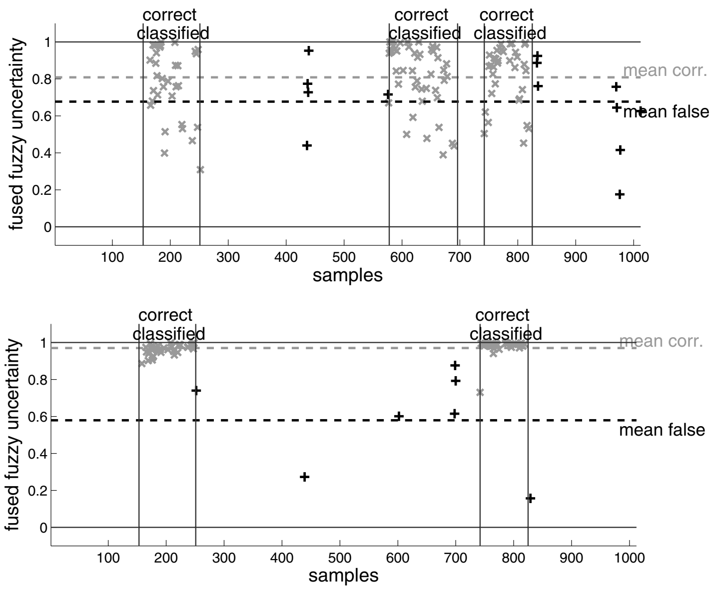


Fig. 3. Fusion (1-top) of fuzzy uncertainty for phone A and phone B classifying on "knocking app.", with correct class. marked 'x' (gray) and incorrect '+' (black). Fusion (2-bottom) of uncertainty for phone B and marker class. on "knocking app.".

results of fusion (2) are shown in figure 3(b), where the filtering on threshold $\tau = 0.9$ improves the accuracy by about 3% from ∼97% to 100%. The problem with filtering is, that along with incorrect classifications also some correct ones are filtered out. Also the amount of classifications is reduced. The trade-off can be influenced through the developer via the threshold level. In our experience it is better to exclude some correct classifications from the following reasoning process or the application using the contexts, than have incorrect classifications result in faulty system states. The reduction of samples is of less significance, since much more samples are processed than needed in most applications.

5.3 Aggregated Contextual Knowledge Improved with Uncertainty

In the last section we have shown that filtering upon the fuzzy uncertainty after a fusion improves accuracy. An aggregation of new context classes is improved through the filtering on the uncertainty level. Since aggregation combines different contextual knowledge to new information, the reliability depends on every part of the input. The following combination of contexts, devices and device states are aggregated to new contexts:

1. Phone A is lying on table recognizing "clapping applause" with audio classifier and the board marker is in a users pocket classifying on "standing" which is resulting in the implication "standing ovations".

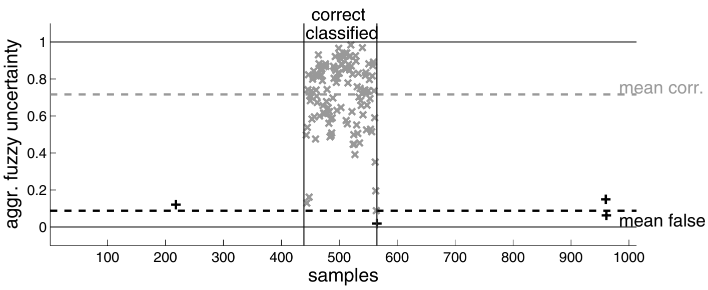


Fig. 4. Uncertainty for aggregation of "standing ovations", correct 'x' & incorrect '+'

The results of the aggregation can be seen in figure 4. For filtering, a threshold $\tau = 0.2$ was chosen, since the fuzzy uncertainty for the test set is spread over the whole interval $[0, 1]$. This circumstance is the result of the product t-norm which was chosen for the aggregation. The accuracy after filtering improves by about 2% from ∼98% up to 100%. The improvement up to 100%, as in the last two examples, is rather unusual. But the examples show that the presented approach is in principle and in practice capable to squeeze out the last 4% (in average) of detection accuraccy to reach absolute correct classification.

6 Conclusion and Future Work

This paper shows how uncertainty measures are created and used in context reasoning applications. Our contribution to the computation of uncertainty measures was a recurrent fuzzy classifier (RFIS) system. The evaluation shows that even in application settings with deliberately unfriendly conditions - especially fast changing contexts - more than 92% recognition rate can be reached. We also infer uncertainty measures and used them for filtering outliers after data fusion and aggregation. This approach boosts our classification result about 4% to almost 100% recognition rate. The shown application (acclamation detection) requires only a one-step fusion process. It is to be expected that the effect of using uncertainty measurement in applications with complex fusion and aggregation processes will be even more prominent. Future work will also research the inclusion of probabilities combined with the investigated fuzziness in the inference process and the utilization of recurrence in classification.

Acknowledgements. This work is partially funded by the State of Niedersachsen in the "IT-Ecosystems" project and the European Commission founded project "Cooperative Hybrid Objects Sensor Networks (CHOSeN)". We also wish to thank Matthias Budde for his corrections on grammar and general understanding of this paper and Henning Guenther for his technical support.

References

1. Buchholz, T., Kuepper, A., Schiffers, M.: Quality of context: What it is and why we need it. In: Workshop of the HP OpenView University Association (2003)
2. Preuveneers, D., Berbers, Y.: Quality extensions and uncertainty handling for context ontologies. In: W. on Context and Ont. Theory, Practice and Appl. (2006)
3. Ranganathan, A., Al-Muhtadi, J., Campbell, R.H.: Reasoning about uncertain contexts in pervasive computing environments. IEEE Pervasive Computing (2004)
4. Truong, B.A., Lee, Y.K., Lee, S.Y.: Modeling uncertainty in context-aware computing. In: Computer and Information Science, ICIS (2005)
5. Berchtold, M., Decker, C., Riedel, T., Zimmer, T., Beigl, M.: Using a context quality measure for improving smart appliances. In: IWSAWC (2007)
6. Berchtold, M., Riedel, T., Beigl, M., Decker, C.: Awarepen - classfication probability and fuzziness in a context aware application. Ubiq. Intell. and Comp. (2008)
7. Gomni, V., Bersini, H.: Recurrent fuzzy systems. IEEE Fuzzy Systems (1994)
8. Chiu, S.: Method and software for extracting fuzzy classification rules by subtractive clustering. IEEE Control Systems Magazine, 461–465 (1996)
9. Tagaki, T., Sugeno, M.: Fuzzy identification of systems and its application to modelling and control. Syst., Man and Cybernetics (1985)
10. Weisbrod, J.: Unscharfes schliessen. Diss. zur Kuenstlichen Intelligenz (1996)