Comparison of Real-Time Classification Systems for Arrhythmia Detection on Android-based Mobile Devices

Heike Leutheuser[1], *Student Member, IEEE, EMBS*, Stefan Gradl[1], Patrick Kugler[1], *Student Member, IEEE, EMBS*, Lars Anneken[2], Martin Arnold[2], Stephan Achenbach[2], and Bjoern M. Eskofier[1], *Member, IEEE, EMBS*

***Abstract*— The electrocardiogram (ECG) is a key diagnostic tool in heart disease and may serve to detect ischemia, arrhythmias, and other conditions. Automatic, low cost monitoring of the ECG signal could be used to provide instantaneous analysis in case of symptoms and may trigger the presentation to the emergency department. Currently, since mobile devices (smartphones, tablets) are an integral part of daily life, they could form an ideal basis for automatic and low cost monitoring solution of the ECG signal. In this work, we aim for a real-time classification system for arrhythmia detection that is able to run on Android-based mobile devices. Our analysis is based on 70% of the MIT-BIH Arrhythmia and on 70% of the MIT-BIH Supraventricular Arrhythmia databases. The remaining 30% are reserved for the final evaluation. We detected the R-peaks with a QRS detection algorithm and based on the detected R-peaks, we calculated 16 features (statistical, heartbeat, and template-based). With these features and four different feature subsets we trained 8 classifiers using the Embedded Classification Software Toolbox (ECST) and compared the computational costs for each classification decision and the memory demand for each classifier. We conclude that the C4.5 classifier is best for our two-class classification problem (distinction of normal and abnormal heartbeats) with an accuracy of 91.6%. This classifier still needs a detailed feature selection evaluation. Our next steps are implementing the C4.5 classifier for Android-based mobile devices and evaluating the final system using the remaining 30% of the two used databases.**

I. INTRODUCTION

The electrocardiogram (ECG) is a key diagnostic tool in heart disease and may serve to detect ischemia, arrhythmias, and other conditions. Since mobile devices such as smartphones or tablets are an integral part of daily life [1], [2], they could form an ideal basis for a mobile ECG monitoring and arrhythmia classification application. The PhysioNet/Computing in Cardiology Challenge 2011 [3] had the target of developing an efficient mobile algorithm for displaying a diagnostically useful 12-lead ECG recording.

Even with the deployment of applications that enabled the possibility of displaying the 12-lead ECG recording, the electrode application in case of 12-lead ECG recording or the interpretation beyond quality assessment is limited to specifically trained medical personnel. An automatic, low-cost monitoring solution of the ECG signal, even in the home environment, would be a major advantage. It could be used to provide instantaneous analysis in case of symptoms and may trigger—or prevent—presentation to the emergency department or an outpatient unit. Such a solution should operate continuously over an extended period of time (e.g. 12 hours) and should be fully portable and as unobtrusive as possible as not to interfere with daily life activities.

In the following, the work of two research groups [4], [5] performing an embedded arrhythmia classification on mobile devices like smartphones are summarized. Yen et al. [4] used a wavelet decomposition to build a noise-tolerant algorithm and calculated higher-order statistics features. For the classification of seven different ECG beat types, a back propagation neural network was used. They evaluated their algorithm using 15 selected records from the MIT-BIH Arrhythmia database [6] and achieved an average recognition rate of 98.34%.

Oresko et al. [5] presented two smartphone-based algorithms for classification of cardiovascular diseases. For the classification of five different ECG beat types, a multilayer perceptron was used. They evaluated their algorithm using 5421 heart beats from the MIT-BIH Arrhythmia database [6] using three-fold cross-validation with the five classes uniformly distributed in the three folds. Recognition rates between 81% and 99% were achieved. These two classification systems achieved high recognition rates, but their results were based on few heartbeats/records.

In a preliminary version of this work [7] we showed a real-time ECG monitoring and arrhythmia detection application using Android-based mobile devices. In this previous work, the MIT-BIH Arrhythmia [6] and the MIT-BIH Supraventricular Arrhythmia [8] databases were used. The QRS detection algorithm identified over 99.5% of the heartbeats in real-time. The arrhythmia classification was based on a decision tree classifier with a sensitivity of 89.5% and a specificity of 80.6%. Some datasets had to be excluded as they were for example too noise prone.

In this work, we exploit potential arrhythmia classification algorithms for use on Android devices using the complete MIT-BIH Arrhythmia [6] and MIT-BIH Supraventricular Arrhythmia [8] databases. The challenge of limited hardware resources in smartphones is addressed using the Embedded Classification Software Toolbox (ECST) [9]. We aim for a classification system with high classification accuracy and simultaneously with low computational complexity.

[1]Digital Sports Group, Pattern Recognition Lab, Department of Computer Science, Friedrich-Alexander University Erlangen-Nürnberg (FAU), Erlangen, Germany.

[2]Department of Cardiology, University Hospital Erlangen, Erlangen, Germany.

Corresponding author: H. Leutheuser

`Heike.Leutheuser@cs.fau.de`

II. METHODS

A. Data

PhysioNet [10] provides free access to a large connection of recorded physiological signals. From this website, we used the MIT-BIH Arrhythmia [6] and the MIT-BIH Supraventricular Arrhythmia [8] databases. The recordings were digitized with a sample rate of 360 Hz. The MIT-BIH Arrhythmia [6] database consists of 48, and the MIT-BIH Supraventricular Arrhythmia [8] database of 78 half-hour ECG recordings. We used the available beat annotations provided by PhysioNet [10]. In this work, only the raw ECG signals of lead II (adapted from Einthoven) were used.

B. Beat Classification

Our aim was to give indications for present abnormal heart beats. Hence, we fused all annotations provided for the two databases into the classes normal and abnormal beat. The abnormal beat class consisted of: (left/right) bundle branch block beats; atrial/aberrated atrial/nodal (junctional)/supraventricular (atrial or nodal) premature beats; ectopic (atrial or nodal) beats; (R-on-T) premature ventricular contractions; atrial/nodal (junctional)/supraventricular (atrial or nodal)/ventricular escape beats; paced beats; fusion of paced/ventricular and normal beats; unclassifiable/unclassified beats.

C. QRS Detection

This QRS detection equalled the QRS detection algorithm illustrated in the preliminary paper as with this algorithm 99.59% heartbeats of the MIT-BIH Arrhythmia [6] and 99.58% heartbeats of the MIT-BIH Supraventricular [8] database were detected [7]. The raw ECG signal was preprocessed with digital filters according to Pan & Tompkins [11] consisting of a Bandpass filter, five-point differentiation, squaring operation, and moving window integration. The single QRS complexes were then extracted using a threshold-based method. The threshold was calculated in applying a moving average filter of size 150 ms to the output of the Pan-Tompkins algorithm.

D. Feature Extraction

We calculated 16 features for each heartbeat (Table I). These features were divided into the three groups statistical features, heartbeat features, and template-based features. The statistical features and the template-based features [12] were calculated using 400 ms windows around the R-peak (150 ms before and 250 ms after the R-peak).

We generated two templates (used for the template-based features) from the first six detected R-peaks in a fully autonomous approach. First, the individual waveform areas of each 400 ms heartbeat were calculated. Second, the heartbeats were ordered according to the smallest difference to the average waveform area of the six heartbeats (starting with heartbeats smaller than the average value in case of equality in value). Third, the Pearson correlation between two consecutive heartbeats (of this ranked order) was calculated and the first two heartbeats with a Pearson correlation higher than 0.95 were chosen as templates. If none of the above heartbeats fulfilled these conditions, the first two heartbeats in the ranked order were chosen. Both templates were adapted over time to refrain from expert-supervised selection of normal beats [7].

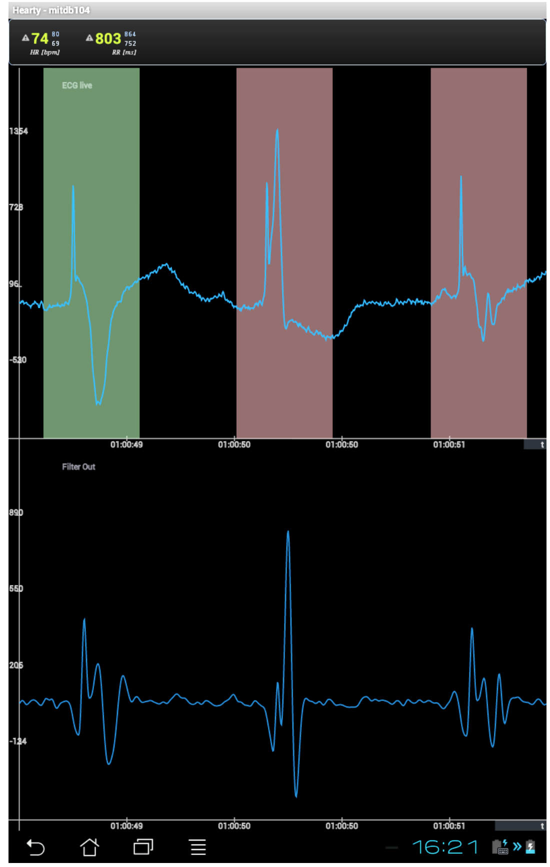

Fig. 1. Screenshot of the current QRS detection and feature extraction application implemented in Java. The green numbers display the current heart rate (left) and the current RR-interval (right). The upper and lower right numbers correspond to the maximum heart rate/RR-interval and minimum heart rate/RR-interval, respectively, that appeared in the signal so far. The upper signal is the current ECG signal with normal heartbeats highlighted in green, abnormal heartbeats in red. The lower signal is the output of the Pan-Tompkins algorithm.

E. Implementation

The previously described algorithms for QRS detection and feature extraction were implemented in Java using the Android SDK 2.3.3 (Google Inc., Mountain View, CA, USA). Fig. 1 shows the screenshot of this Android application. Android is the most widely used smartphone operating system with a market share of 79.3% [13] and Android is an open source operating system.

F. Embedded Software Classification Toolbox

For the design of the classification system, we used the Embedded Classification Software Toolbox (ECST) [9]. In this toolbox, different classification systems can be trained and compared according to their classification rate in each classification decision, and the complexity of the trained

TABLE I

SET OF 16 FEATURES FOR EACH HEARTBEAT WITH THE ASSIGNMENT TO THE FIVE DIFFERENT FEATURE SETS (FS).

Statistical features (FS 2)	Heartbeat features (FS 3)	Template-based features (FS 4)
1. Mean	7. QRS-width*	13. Maximal cross-correlation coefficient to template 1*
2. Minimum value	8. RR-interval*	14. Maximal cross-correlation coefficient to template 2*
3. Maximum value	9. Previous RR-interval	15. Area difference to template 1*
4. Standard deviation	10. QR-amplitude	16. Area difference to template 2*
5. Kurtosis	11. RS-amplitude	
6. Skewness	12. QRST-area	

$\underbrace{\hspace{10cm}}_{\text{FS 1}}$

Features assigned to FS 5 are marked with *.

TABLE II

COMPUTATIONAL COST, ACCURACY (ACC), SENSITIVITY (SENS), AND SPECIFICITY (SPEC) ANALYSIS USING THE ECST [9] FOR THE FEATURE SET 1 WITH ALL FEATURES. THE SHOWN COMPUTATIONAL COSTS ARE NECESSARY FOR ONE CLASSIFICATION DECISION.

Classifier	ACC [%]	SENS [%]	SPEC [%]	Computational Cost $+,-$	$\times$	$\div$	$\sqrt{x}$	e^x	$\leq$	Memory Demand Floats	Integers
AdaBoost M1	74.1	73.7	74.4	10	0	0	0	0	11	20	30
C4.5	91.6	90.9	92.3	0	0	0	0	0	31	1819	10917
Linear Regression	74.6	69.7	82.3	33	32	2	0	0	5	34	0
Multilayer Perceptron	88.1	86.7	89.7	173	162	11	0	11	1	173	0
Naïve Bayes	64.9	60.6	74.9	32	192	64	32	32	1	66	0
Nearest Neighbor	92.8	92.3	93.3	1263781	1348032	0	84252	0	84252	1348032	84253
PART	90.5	89.3	91.7	0	0	0	0	0	1901	1901	4091
SVM	77.0	72.0	84.9	65	32	16	0	0	2	51	0

TABLE III

ACCURACY (ACC), SENSITIVITY (SENS), AND SPECIFICITY (SPEC) ANALYSIS USING THE ECST [9] FOR THE FEATURE SETS (FS) FS 2, FS 3, FS 4, AND FS 5. THE SHOWN COMPUTATIONAL COSTS ARE NECESSARY FOR ONE CLASSIFICATION DECISION.

Classifier	FS 2 ACC [%]	SENS [%]	SPEC [%]	FS 3 ACC [%]	SENS [%]	SPEC [%]	FS 4 ACC [%]	SENS [%]	SPEC [%]	FS 5 ACC [%]	SENS [%]	SPEC [%]
AdaBoost M1	60.0	58.0	63.5	74.3	74.1	74.5	70.6	68.0	74.1	74.2	72.2	76.7
C4.5	86.6	85.6	87.8	89.6	88.6	90.8	75.1	73.0	77.7	86.2	85.9	86.5
Linear Regression	57.7	56.0	61.1	68.4	64.4	75.5	67.1	62.0	80.1	67.7	63.2	76.9
Multilayer Perceptron	70.4	65.5	80.2	80.4	76.8	85.2	68.9	66.8	71.5	72.0	70.9	73.3
Naïve Bayes	56.1	54.3	61.4	64.3	51.5	71.6	65.9	60.0	83.6	66.6	60.6	81.8
Nearest Neighbor	87.6	86.9	88.3	90.8	90.5	91.8	71.3	71.5	71.2	85.0	84.7	85.2
PART	81.4	80.1	84.1	87.4	85.5	89.5	72.8	72.3	75.6	84.1	84.5	76.5
SVM	59.5	56.5	68.1	71.2	68.2	75.2	66.8	61.3	82.0	67.0	61.5	82.6

systems are analyzed. This toolbox integrates the WEKA library [14].

We used 70% of each database for the training phase. The remaining 30% were assigned for testing, however, we did not consider the testing dataset in this work. We determined the above mentioned 16 features (Table I). This resulted in 89 datasets with the total number of 196487 heart beats and corresponding feature vectors. Due to the uneven distribution of heart beats in the two classes and the importance of high accuracy for the detection of abnormal beats, we used all feature vectors of the abnormal class (resulted in 42126 feature vectors) and randomly picked the same number of feature vectors of the normal class. Thus, the different classification systems were trained using 84252 feature vectors.

No single classifier is suitable for all classification tasks (No Free Lunch Theorem) [15]. Therefore, the following classification systems were compared [15], [16]: AdaBoost M1, C4.5, Linear Regression, Multilayer Perceptron, Naïve Bayes, Nearest Neighbor, PART, and Support Vector Machine (SVM).

We did not perform a preprocessing step except for the SVM. For the SVM classifier, we applied a normalization step to the features. Further, we did not perform a feature selection step with the features in the ECST. Instead, we compared five different feature sets (FS) (Table I): FS 1 consisted of all 16 features, FS 2 consisted of all statistical features, FS 3 consisted of all heartbeat features, FS 4 consisted of all template-based features, and FS 5 consisted of the features used by Gradl et al. [7] (QRS-width, RR-interval, maximal cross-correlation coefficient to template 1, maximal cross-correlation coefficient to template 2, area difference to template 1, area difference to template 2).

We chose the adjustable parameters as default parameters according to the WEKA data mining software [14]. For evaluation, we applied ten-fold cross-validation.

III. RESULTS

Table II shows the computational cost, accuracy, sensitivity, and specificity analysis using the 8 different classifiers evaluated with the ECST for FS 1. Table III shows the accuracy, sensitivity, and specificity analysis using the 8 different classifiers evaluated with the ECST for FS 2, FS 3, FS 4, and FS 5. The best accuracy of 92.8% was obtained with the kNN classifier using FS 1, followed by the C4.5 classifier (91.6%) using FS 1. The worst accuracy of 56.1% was obtained with the Naïve Bayes classifier using FS 2.

The computational costs per classification step are only shown for the 8 classifiers using FS 1. The computational costs were comparable for each classifier. The fewest computational costs per classification step were used by the AdaBoost M1 (21 operations total) classifier. The smallest memory demand was used by the Linear Regression classifier. The highest computational costs per classification decision and the highest memory demand were needed by the Nearest Neighbor (2,780,317 operations total).

IV. DISCUSSION

In this study, we exploited potential arrhythmia classification algorithms for use on Android devices using the complete MIT-BIH Arrhythmia [6] and the complete MIT-BIH Supraventricular Arrhythmia [8] databases with the ECST [9].

The accuracies for the different classification systems ranged from 56.1% (Naïve Bayes with FS 2) to 92.8% (Nearest Neighbor with FS 1). The sensitivities ranged from 51.5% (Naïve Bayes with FS 3) to 90.9% (C4.5 with FS 1). The specificities ranged from 61.1% (Linear Regression with FS 2) to 93.3% (Nearest Neighbor with FS 1).

The Nearest Neighbor classifier had by far the highest computational costs and the highest memory demands and is hence not suitable for implementation on Android-based mobile devices. The computational costs of the Nearest Neighbor classifier are dependent on the used number of feature vectors. In general, it is always good to have a lot of data for the training of classification systems. Thus, reducing the number of feature vectors in the training dataset should not be favored.

The second best classifier was C4.5. This classifier had low computational costs but the second highest memory demand. The memory demand of this classifier depends on the number of used features. Our suggestion is to use this classifier, as a high accuracy, sensitivity, and specificity were obtained. For decreasing the memory demand, we suggest to apply a smaller number of features.

The best feature subset results was achieved with the heartbeat features (FS 3). We suppose that applying a feature selection procedure could further enhance the classification accuracy, sensitivity, and specificity of the C4.5 classifier and decrease the memory demands.

In the future, we are going to implement the C4.5 classifier for Android-based mobile devices. In this work, we used 70% of the two databases for the training of the different classification systems. Hence, the evaluation of the C4.5 classifier will be based on the remaining 30% of the two databases.

Additionally, we are acquiring long-term ECG recordings in patients suffering from heart conditions. We are further planning to evaluate the QRS detection algorithm and the implemented classification system on these acquired data.

ACKNOWLEDGMENT

This work was funded by the Bavarian Ministry for Economic Affairs, Infrastructure, Transport, and Technology and the European Fund for Regional Development.

REFERENCES

[1] International Telecommunication Union, *Measuring the Information Society*. ITU, 2013.

[2] IDC, "Press release: Smartphones expected to grow 32.7% in 2013 fueled by declining prices and strong emerging market demand," Accessed on February 28th 2014, http://www.idc.com/getdoc.jsp?containerId=prUS24143513.

[3] I. Silva, G. B. Moody, and L. Celi, "Improving the quality of ecgs collected using mobile phones: The physionet/computing in cardiology challenge 2011," *In Proc: Computing in Cardiology*, pp. 273–276, 2011.

[4] T.-H. Yen, C.-Y. Chang, and S.-N. Yu, "A portable real-time ECG recognition system based on smartphone," *In Proc. 35th Annual International Conference of the IEEE EMBC*, pp. 7262–7265, 2013.

[5] J. J. Oresko, Z. Jin, J. Cheng, S. Huang, Y. Sun, H. Duschl, and A. C. Cheng, "A wearable smartphone-based platform for real-time cardiovascular disease detection via electrocardiogram processing," *IEEE Trans Inf Technol Biomed.*, vol. 14(3), pp. 734–740, 2010.

[6] G. Moddy and R. G. Mark, "The impact of the MIT-BIH arrhythmia database," *IEEE Eng Med Biol.*, vol. 20(3), pp. 45–50, 2001.

[7] S. Gradl, P. Kugler, C. Lohmüller, and B. Eskofier, "Real-time ECG monitoring and arrhythmia detection using android-based mobile devices," *In Proc: 34th Annual International Conference of the IEEE EMBC*, pp. 2452–2455, 2012.

[8] S. D. Greenwald, "Improved detection and classification of arrhythmias in noise-corrupted electrocardiograms using contextual information," Ph.D. dissertation, Harvard-MIT Division of Health Sciences and Technology, 1990.

[9] M. Ring, U. Jensen, P. Kugler, and B. M. Eskofier, "Software-based performance and complexity analysis for the design of embedded classification systems," *In Proc: 21st International Conference on Pattern Recognition*, pp. 2266–2269, 2012.

[10] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101(23), pp. e215–e220, 2000 (June 13).

[11] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans Biomed Eng.*, vol. BME-32, no. 3, pp. 230–236, 1985.

[12] V. Krasteva and I. Jekova, "Qrs template matching for recognition of ventricular ectopic beats," *Ann Biomed Eng.*, vol. 35(12), pp. 2065–2076, 2007.

[13] IDC, "Press release: Apple cedes market share in smartphone operating system market as android surges and windows phone gains," Accessed on March 2nd 2014, https://www.idc.com/getdoc.jsp?containerId=prUS24257413.

[14] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA data mining software: An update," *SIGKDD Explor Newsl.*, vol. 11(1), pp. 10–18, 2009.

[15] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2000.

[16] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*. Academic Press, 2006.