The Impact of Information Fusion in Steganalysis on the Example of Audio Steganalysis

Christian Kraetzer and Jana Dittmann
Otto-von-Guericke University of Magdeburg, PO Box 4120, 39016 Magdeburg, Germany
Contact email: christian.kraetzer@iti.cs.uni-magdeburg.de

ABSTRACT

Information fusion tries to determine the best set of experts in a given problem domain and devise an appropriate function that can optimally combine the decisions of the individual experts. Only few systematic approaches to information fusion exist so far in the signal processing field of steganalysis.
Under the basic assumption that steganalysis can be seen as a statistical pattern recognition process like biometrics, a state of the art five level information fusion model known from biometrics is transferred to steganalysis as well as statistical detectability evaluations for watermarking algorithms and its applicability is evaluated in practical testing.
The primary test goal for these evaluations is to measure the impact of fusion on the classification accuracy. Therefore a match and decision level fusion are performed here for three selected data hiding algorithms (one steganography and two watermarking), two feature extractors and five different classifiers. For the test heterogeneous audio test sets are used for content independent training and testing. The secondary test goal of this work is to consider the impact of the key selection assumption on the accuracy of the classification in steganalysis.
The results show for the test cases an increase of the classification accuracy for two of the three tested algorithms by match level fusions, no gain by decision level fusion and a considerably small impact of the key selection assumption on the statistical detectability.

1. MOTIVATION AND INTRODUCTION

Steganalysis based on statistical models is used to classify digital assets into unmodified objects and objects modified by a data hiding algorithm. Some quite mature approaches especially in the image domain not only show high classification accuracies (>99%) but also allow for message length estimations. Other domains, like the here considered audio steganalysis, have not yet reached the same degree of maturity as their image counterpart.
The approach presented within this document is focusing with information fusion on a technique so far rather uncommon to steganalysis. The goal of using fusion is to improve the quality in steganalysis (measured here in classification accuracy) and thereby improve its value as a detection mechanism for hidden embedding of information into digital objects, especially in a domain like audio where few reliable detection approaches exist so far.
In contrast to previous work on fusion in steganalysis (Kharrazi et al.[10]) we focus on the question: How can the detection performance (measured in detection accuracy) on selected data hiding algorithms be improved by fusion in steganalysis? To address this question we transfer a five level fusion model from the state of the art in biometrics to the domain of audio steganalysis with the goal to increase the detection performance (instead of aiming for a stronger universality of the steganalysis approach like Kharrazi et al.) and show how the overall steganalysis process would benefit from fusion operations on the example of match and decision level fusion. For the practical implementation of the fusion a steganalysis approach which has been successfully employed in audio steganalysis[11] and audio forensics[13] in the past is combined with an approach adapted form image steganalysis. Based on this background it can be assumed that the results derived in practical testing here can also be transferred back into the field of audio forensics and thereby help to establish trust (in terms or authenticity and integrity) in digital objects.
The primary test goal defined for the evaluations performed here is to measure the impact of fusion on the classification accuracy. For the evaluation of this goal a match level and a decision level fusion of the two mentioned steganalysers (AAST (AMSL Audio Steganalysis Toolset) and AudioRS) and five different classifiers is performed for three selected data hiding algorithms under the hypothesis that a complete file is either “marked” or “unmarked” by an information hiding algorithm (binary decision). The secondary test goal is to consider the impact of the key selection assumption on the accuracy of the classification in steganalysis.

Both security mechanisms steganalysis and media forensics are of uttermost importance for other IT disciplines like for example secure data storage or long term archiving where establishing trust in the authenticity and integrity of communication or storage environments, as well the digital objects within these environments, is a necessity for any security concept or business model. Hidden channels within an archiving environment pose not only the imminent threat

of the misuse of such a system for hidden communication but also the potential threat of steganographically inserted malicious code[9] which might later violate e.g. the authenticity or integrity of stored objects.
To achieve the goal of improving the value of steganalysis as a secure and reliable detection mechanism e.g. for secure storage applications this work shows as result of the performed tests for example in match level fusion an increase of the classification accuracy for two of the three tested algorithms. Comparing the performance of the five evaluated classifiers in the match level fusion performed here, then the AdaBoost and linear logistic regression models seem to outperform the SVM, the Bayesian classifier as well as the used decision tree. The results for decision level fusion are not able to show any gain on this fusion level, indicating that a late fusion might not be the optimal choice for the used steganalysis approaches. Also a considerably small impact of the key selection assumption on the statistical detectability of the tested algorithms is shown. Here the tests show only in 16.6% of the non-fusion test cases a significant deviation in the results between the two tested key scenarios, all of them either for the decision tree or the logistic regression model. No such differences are seen in the fusion tests.

The document is structured as follows: Section 2 describes the used information fusion model, which originates in biometrics and is here transferred to steganalysis for the exemplary domain of audio. All five fusion levels and the corresponding signal processing steps are introduced briefly. In section 3 the complete test scenario, including the test goals, test setup and the procedure for the practical evaluations, is described. Here the subsection containing the test setup specifies the choices for: The test sets used, the three data hiding algorithms, feature computation steps and the five exemplarily chosen classifiers (with the corresponding output normalisation/weighting strategies). Section 4 contains the test results from the practical tests and section 5 concludes the document and shows perspectives for future work.

2. PATTERN RECOGNITION, STEGANALYSIS AND FUSION

If using a definition given by Bebis[2], then **pattern recognition** is in general the study of how machines can observe their environment, learn to distinguish patterns of interest from their background signals and make sound and reasonable decisions about categories of the patterns. Therefore the key objectives in pattern recognition are to process the sensed data to eliminate noise, hypothesise the models that describe each class population and, given a sensed pattern, choose the best-fitting model for the assignment to the class associated with the model.
From the various main pattern recognition areas[2] (template matching, statistical pattern recognition, structural pattern recognition, syntactic pattern recognition, artificial neuronal networks, etc) the approach of statistical pattern recognition is considered here for its application in steganalysis. This approach assumes that the patterns to be recognised (here the impact of the data embedding by data hiding algorithms/techniques) are represented in a feature space and tries to build a statistical model for pattern generation in this space. Figure 1 shows the general statistical pattern recognition scheme.

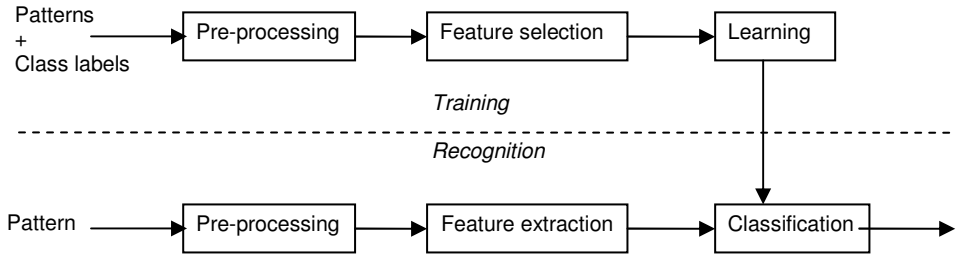

Figure 1: General statistical pattern recognition scheme (based on Bebis[2])

One of the research fields where applied signal processing and statistical pattern recognition are extensively employed is the fields of biometrics[18] and HCI[21]. Having emerged in the 1960s and early 1970s (see e.g. Atal[1] for biometric speaker verification/identification), biometrics achieved until now some maturity from which other (similar) pattern recognition problems like steganalysis can benefit. The idea of a knowledge transfer from biometrics to steganalysis is not a new one. One previous attempt is presented by Kharrazi et al.[10]. In their paper the authors propose to transfer a concept called information fusion from biometrics to image domain steganalysis. **Fusion,** which is a fairly common technique in biometrics, has the goal to determine the best set of experts in a given problem domain and devise an appropriate function that can optimally combine the decisions rendered by the individual experts[18].

From the numerous fusion concepts known in biometrics two different ones shall be briefly considered here. The first one was presented by Sanderson and Paliwal[19] in 2002 and is used in a simplified version in the considerations by Kharrazi et al.[10]. It uses a model which distinguishes into pre-classification (sensor and feature level) and post-classification (measurement, rank and abstract/decision level) information fusion. Where pre-classification fusion refers to combining information prior to the application of any classifier (or matching algorithm), while in post-classification the information is combined after the decisions of the classifiers have been obtained.

In their paper Kharrazi et al. limit themselves to three different operations: First, the transfer of the aforementioned fusion model from biometrics to the image steganalysis domain, second the practical evaluation of the impact of a fusion of three different steganalysers (two universal, one specific) on the classification performance for two image steganography techniques (fusion results presented ranging from worse than the best individual technique to better than all techniques – depending on the tested algorithm), and third, the question whether the fusion of steganalysers might lead to the same classification results as a truly "global" universal steganalyser (trained with a training set containing samples for all available steganographic techniques). In the test results of the third presented evaluation a reduction of the classification result by choosing an universal or fused detector instead a of specific one is seen (results achieved are between 3 and 7% worse), while at the same time it is indicated that the scalability of the steganalysis increases (complexity decreases).

The second fusion approach to be mentioned here is the one used by Ross, Nandakumar, and Jain[18]. In this approach a five level fusion model (sensor, feature, match, rank and decision level fusion) is employed. This latter fusion approach by Ross et al. is chosen for the considerations on information fusion in steganalysis in this work because it has a finer granularity and incorporates (amongst other benefits) a more appropriate model for dynamic classifier selection. It is formalised and visualised by Oermann et al.[16] and enhanced here by adding the corresponding signal processing operations between the fusion levels (see Figure 2).

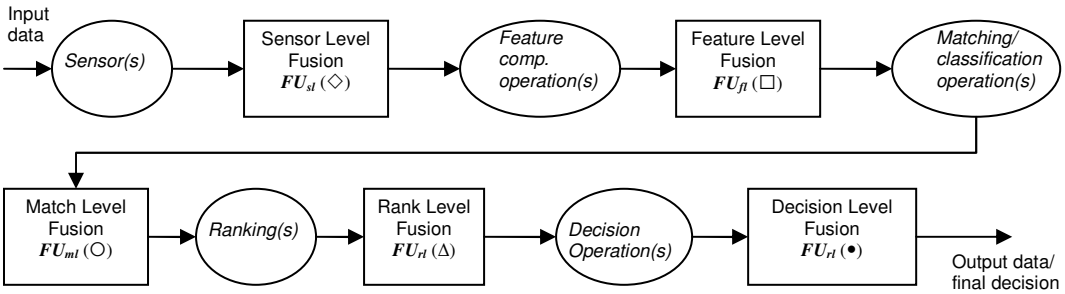

Figure 2: Overview of the five signal processing steps and the five different fusion levels (with their corresponding fusion operators; based on Oermann et al.[16])

The five fusion levels[18] (sensor ($\boldsymbol{FU_{sl}}$), feature ($\boldsymbol{FU_{fl}}$), match ($\boldsymbol{FU_{ml}}$), rank ($\boldsymbol{FU_{rl}}$), and decision level fusion ($\boldsymbol{FU_{dl}}$)) used in this model with their corresponding generic fusion operators (◇,□,○,Δ,•) as well as the required signal processing operations (signal acquisition at sensor level, feature computation, classification/matching, ranking and decision making) can be summarised as follows (note: detailed examples on how fusion on these levels is performed in biometrics are presented by Ross et al.[18]):

- **Sensor level fusion** ($\boldsymbol{FU_{sl}}$): entails the consolidation of evidence presented by multiple sources of raw data before they become subject to feature extraction[18].
- **Feature level fusion** ($\boldsymbol{FU_{fl}}$): involves consolidating the evidence presented by different feature sets of the same source. The following requirements for feature level fusion have to be considered: The features have to be related (i.e. really belong to one source), they must be of the same type (e.g. a variable length and a fixed length feature set should not be joined), and should be considered under the knowledge of the course-of-dimensionality problem (i.e. the number of samples for a training set has to reflect the number of features).
- **Match level fusion** ($\boldsymbol{FU_{ml}}$; also known as score level fusion or classification level fusion): a fusion on matching score level implies a consolidation of matching scores (respectively classification results) gained from separate comparisons/classification of reference data and test data for each source. Because fusion on this

level is the most commonly applied technique it biometrics it incorporates an separate chapter in the work of Ross et al.[18].

- **Rank level fusion** (FU_{rl}): which is of importance especially for identification problems, has the goal to consolidate the ranked outputs of individual classification systems in order to derive a consensus rank for each identity known.
- **Decision level fusion** (FU_{dl}): if a fusion is applied on decision level then each subsystem draws completely autonomous decisions, which are then combined. The operator ($\bullet$) for this decision combination could be Boolean functions (like AND or OR), (weighted) majority voting, Bayesian decisions, etc.

Instead of performing immediate fusion steps (or earliest possible stage fusions) at the levels identified above, this paper focuses on late fusion operations, e.g. instead of performing a sensor level fusion the signals by two different sensors they are processed in parallel until match or decision level and fused there. Table 1 identifies the fusion operators used in this document. Note: sensor- and rank level fusion are not considered in this work, because for sensor level only the original reference signals are used and rank level fusion is not applicable because the classification problem evaluated in this work is a two-class classification.

Table 1: Fusion levels, processing operations and corresponding fusion operators evaluated in this work

Level used In this work	**Exemplary signal processing operations and their options used**	**Immediate fusion step / fusion operators used**
Feature computation	One or more feature extractors working in different domains; different choices of features (global, segmental and local), normalisation or weighting techniques, etc. **In this paper:** computation of different segmental (in time, frequency and Mel-cepstral domain) features by one feature extractor	Feature set fusion for the *AAST* feature extractor as unweighted concatenation of the three different sub-feature sets computed
Classification/ matching	Usage of different classifiers like support vector machines, neuronal networks, decision trees, etc, while at the same considering pre-processing (e.g. composition of training/testing sets) as well as post-processing (e.g. normalisation/weighting or distance computation in matching) techniques. **In this paper:** context independent[11] training and test set generation strategies for two different feature extractors (with corresponding output normalisation strategies) and five classifiers	For the intra-window feature extractor *AAST* a match level fusion is performed by weighted majority vote; for the *AudioRS* feature extractor no match level fusion is possible due to the fact that it computes global features.
Decision making	Usage of decision operations based on different basic assumptions, like binary or multi-class decisions. **In this paper:** Weighting of the input based on model quality estimations.	The fusion operator used in this work is a (weighted) majority voting in a two-class decision problem

3. THE TEST SCENARIO

Within this section the complete test scenario (consisting of the test goals, the test setup and a description of the test procedure) is described to evaluate the different fusion approaches established in Table 1.

3.1. Test Goals

The primary test goal defined for the evaluations performed here is to measure the impact of fusion on the classification accuracy. For the evaluation of this goal a match level and a decision level fusion of two different steganalysers (one using intra- (segmental) and the other inter-window (global) features) and five different classifiers is performed for the three selected data hiding algorithms. In the fusion tests the best feature extractor & classifier combinations for each algorithm (and key scenario) are determined. The secondary test goal is to consider the impact of the key selection assumption on the accuracy of the classification in steganalysis. To evaluate the impact of key selection two different scenarios are compared. In the first key selection scenario all files in the test sets are marked with one fixed key. In the second scenario each file is marked with one individual key.

3.2. Test Setup

This section describes the practical test setup. This includes the used audio test sets, the three used data hiding algorithms, feature computation steps, the five exemplarily chosen classifiers used here (with the corresponding output normalisation/weighting strategies) and the decision generation process.

3.2.1. Audio Test sets used in training and testing

For the tests performed here the AMSL Audio Test Set (*aats389*) from Kraetzer et al.[14,11], containing 389 PCM coded audio files with an average duration of 28.55s, is split into two completely independent parts. The first part *aats389_Part1*, which contains the larger part of the files (366), is only used for training purposes. The smaller part *aats389_Part2*, which contains one file per genre from *aats389* (except for "silence" genre) is used in testing to establish an a priori classifier confidence or model quality estimation for weighting on the classifier output in the tests. A new audio test set (*testset24*) is generated for the actual tests, containing exactly one file per genre present in *aats389*. This new test set has a duration per file of approximately 30 seconds. Table 2 below summarises the test sets used.

Table 2: Audio test sets used

Name of the test set	Syntactical properties	Number of files	Avg. duration
aats389_Part1	PCM16, 44.1kHz, stereo	366	27.1s
aats389_Part2	PCM16, 44.1kHz, stereo	23	30s
testset24	PCM16, 44.1kHz, stereo	24	30s

3.2.2. Data hiding algorithms used

In this paper a set of three data hiding algorithms (the steganography algorithm AMSL LSB stego, short ***AS1*** and the watermarking algorithms AMSL Spread Spectrum Watermarking ***AW1*** and Wasp ***AW3*** – for algorithm descriptions see Kraetzer et al.[14]) is used for the generation of training and test data. Both classes of data hiding algorithms (steganography and digital watermarking) are present in the tests performed and can be compared in their statistical detectability. Each of the three algorithms is working in a different domain: *AS1* is a time domain LSB algorithm, *AW1* a frequency domain spread spectrum technique and *AW3* a wavelet domain algorithm.

For the embedding two different key selection strategies are compared. The first ("fixed key") uses exactly one predefined key ("UniversityOfMagdeburg") for the generation of the marked versions of the training and test files – i.e. in all files the message is embedded using the same key. The second key selection strategy ("variable key") uses the MD5-hash value of the filename for each file in a test set as the key for embedding – therefore it uses for each file in the test set a unique key.

All files used in the evaluations are marked by the data hiding algorithms with 100% capacity (message to be embedded Goethes' "Faust"). No evaluations on test sets with reduced message lengths or an estimation of message lengths, such as presented e.g. by Fridrich et al.[6], is considered here - these are topics for further work.

Table 3: Algorithm parameterisations used

Alg.	Embedding domain	capacity	message	fixed key	variable key
AS1	Time	100%	Goethes' "Faust"	UniversityOfMagdeburg	*md5sum*(filename)
AW1	Frequency				
AW3	Wavelet				

3.2.3. Feature extractors used

Two different feature extractors are considered here. The first is the AMSL Audio Steganalysis Toolset[12] (***AAST***) in its current version 1.04 (build 20071005), computing 7 intra-window features in time domain ($sf_{variance}$, $sf_{covariance}$, $sf_{entropy}$, $sf_{lsbratio}$, $sf_{lsbfliprate}$, sf_{mean}, sf_{median}), 56 intra-window features in Mel-cepstral domain (28 Mel-cepstral-domain coefficients (MFCCs) and filtered Mel-cepstral-domain coefficients (FMFCCs)) and 35 intra-window features in frequency domain (11 Formants and a 24 feature Bark scale histogram). Due to previous results[12] on feature selection on the three data hiding algorithms the complete features from all three domains are fused into a 98 dimensional intra-window feature vector for the evaluated files. In the tests performed here for each file 200 consecutive and non-overlapping windows á 1024 samples per window are processed by *AAST*.

The second feature extractor used here is an audio adaptation of the RS-Analysis (Regular/Singular analysis or dual statistics) approach of Fridrich et al.[7] called for the rest of this work ***AudioRS***. The implementation used for the tests is adapted by the authors from the ImageRS incorporated by Kathryn Hempstalk into the open source project Digital Invisible Ink Toolkit[8]. In contradiction to *AAST*, which is an intra-window feature extractor, *AudioRS* is an inter-window (global) feature extractor returning one 19 dimensional feature vector per file instead of one per window.

3.2.4. Pre-processing for classification

For feature extractor *AAST*, which computes features in three different domains for one window of audio material, the resulting features are fused at $\boldsymbol{FU_{fl}}$ using as fusion operator (□) the concatenation of the three resulting feature vectors into one feature vector (*fv*) per window. For the second feature extractor *AudioRS* no fusion on this level is required since it computes only one *fv* per file. Another pre-processing operation to be performed is the normalisation required by the SVM classifier used. For this SVM-normalisation the appropriate tool of the libsvm[4] package is applied. The other four classifiers (see section 3.2.5) do not require any pre-processing operations.

3.2.5. The five exemplarily selected classifiers

In the tests performed five different classifiers (prototype based and information theoretic classification approaches) are compared in their performance on a classical two-class classification problem proposed by applied steganalysis on the three different information hiding algorithms described in section 3.2.2.
The support vector machine technique, which is a common choice in steganalysis, is a two class classification approach based on Vapnik's[20] statistical learning theory. The two-class SVM based classifier libsvm[4] is used with its default parameters for kernel (RBF) and kernel parameters γ and *c*. This classifier and its parameterisation are in the following identified as ***SVM***.
The following four classifiers are taken from the WEKA toolset[22] and are used with their standard parameters: The used Bayesian classifier (a Naïve Bayes implementation) computes classifications using a probabilistic approach, i.e. it tries to compute conditional class probabilities and then predict the most probable class. For a detailed description of the classification process of Naïve Bayes classifiers see Borgelt et al.[3]. It is identified in the following as ***Bayes***. The classifier ***SimpleLogistics*** is used for building linear logistic regression models. A class for performing additive logistic regression with simple regression functions as base learners is used for fitting the logistic models. The optimal number of iterations to perform is cross-validated, which leads to automatic attribute selection. For more information see Landwehr et al.[15]. The classifier ***ADABoost*** is a class for boosting a nominal class classifier using the AdaBoost M1 method[5]. ***J48*** is a class for generating a C4.5 decision tree[17].

3.2.6. Post-processing

For the intra-windows features, which are also evaluated window by window, either a match level or a decision level fusion has to be performed. In the first case each window of each *AAST* & classifier combination is considered individually in the matching of the file to the classes "marked" and "unmarked". In the second case for each *AAST* & classifier combination a decision has to be made for each file and these decision results are then used as input to the fusion. Here a majority vote is used as mechanism to derive this decision. This means that if for at least *n/2+1* windows of a file the classifier assumes the file to be modified by an data hiding algorithm, the complete file is classified as "marked/modified". This principle is visualised in Figure 5 below showing in detail the decision level fusion stage of the used test setup.
Generally, the classification of the files in *aats389_Part2* against the model generated on *aats389_Part1* is used for model quality estimation, i.e. the accuracy achieved in this classification is used to derive the weights for the corresponding feature extractor and classifier combination in the fusion.

3.3. Test Procedure

Based on Figure 1 and the descriptions on the test setup given in section 3.2, Figure 3 describes the signal processing operations prior to the match and decision level fusions. The tests performed in this work contain at pre-processing stage in Figure 1 no signal processing operations since the un-modified versions of the marked (fixed or variable key, fixed message at 100% algorithm capacity) and original files are directly used as input for the feature extraction for both classifiers.

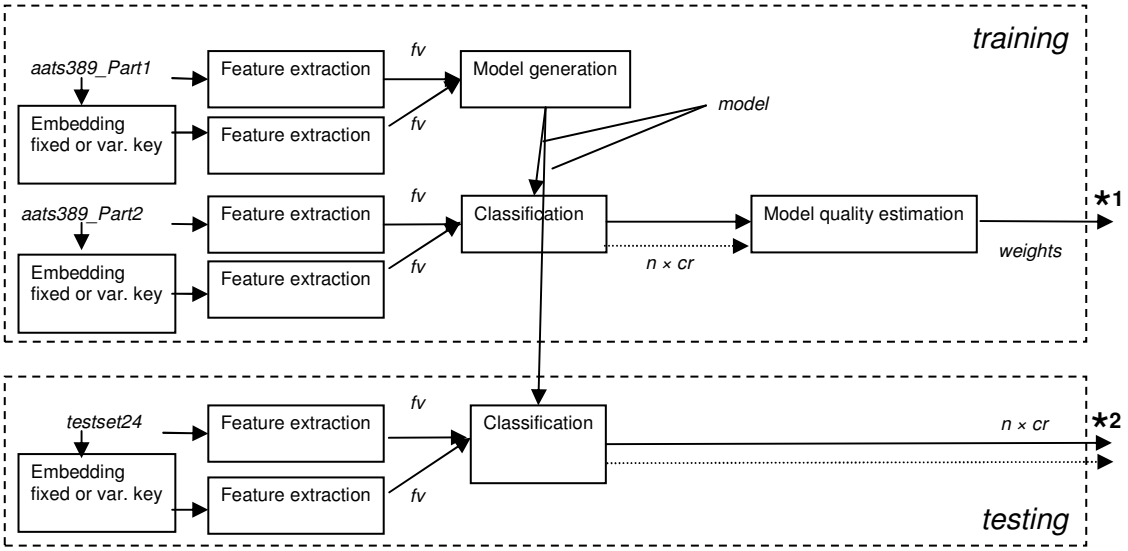

Figure 3: Signal processing operations performed by each feature extractor and classifier combination in training and testing (fixed payload, fixed and variable key)

Figure 3 shows the signal processing operations performed for each combination of feature extractor and classifier tested here. For the intra-window feature extractor *AAST* one feature vector *fv* per window is processed, leading in the training to *n* (for the tests performed here *n*=200 is chosen) classification results *cr* (symbolised in the figure by the dotted arrow) for model quality estimation and in the testing to *n* classification results *cr* to be post-processed on match or decision level into one decision per file. For the global feature extractor *AudioRS* only one *fv* is returned per file, leading to exactly one *cr* each for model quality estimation and classification at decision level (symbolised in the figure by the solid arrow).

The two different kinds of input to the match level fusion are marked in Figure 3 with ✶1 and ✶2. The components ✶1 are the weights from the model quality estimation and ✶2 the classification results from testing (*n* in case of *AAST* and exactly one in case of *AudioRS*).
Figure 4 shows the match level fusion (only possible for the intra-window feature extractor) performed for the output of *AAST*. The inputs for this fusion operation are the five outputs of the *AAST* & classifier combinations. The fusion operator ○ is a weighted sum, where the binary weights for this fusion are set to "1" when an *AAST* & classifier combination is considered significant (in the model quality estimations the ratio between "marked" and "unmarked" decisions is close to equal – remark: both are equally represented in the testing sets) and to "0" otherwise (the classifier tends too much into classifying everything as one of the two classes). The output of the match level fusion (✶3) is a decision on the class of each file which can be used as an additional input to the next fusion stage (together with a confidence measure, which can act as weight in a following decision level fusion; ✶4).

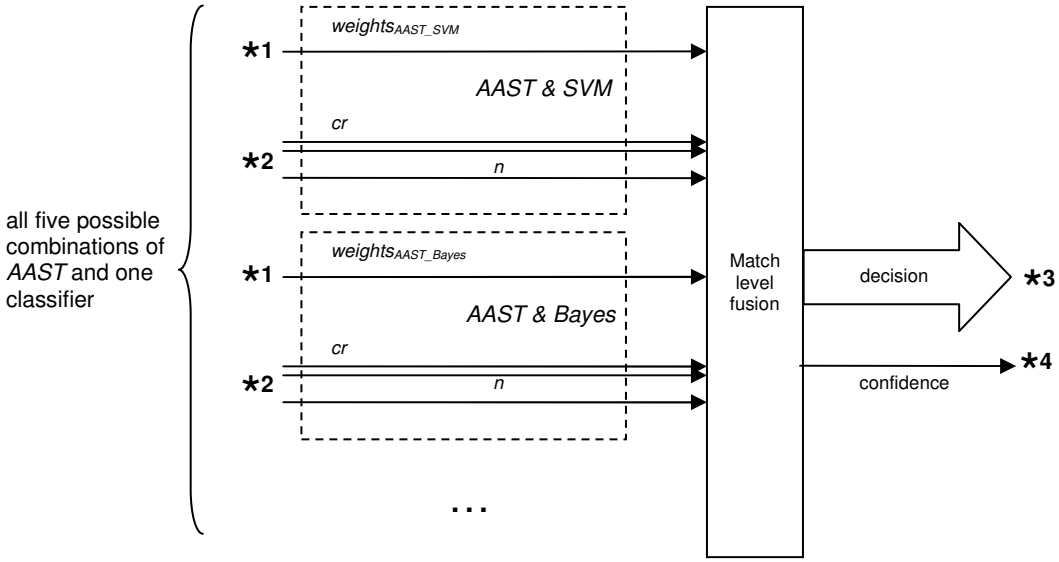

Figure 4: Match level fusion based on the previous signal processing operations

Figure 5 shows the decision level fusion in the tests performed here based on the input generated by the previous signal processing operations. The ten possible combinations of feature extractor and classifier (⋆1) with the corresponding weights (⋆2) are used in parallel to generate the input for the fusion. An additional input is generated by the match level fusion (⋆3 and ⋆4). The decision level fusion operator (•) used in the tests presented here is basically a asymmetric weighted majority vote (asymmetric since in each decision the positive and negative cases can be weighted individually) using the model quality estimations as weights.

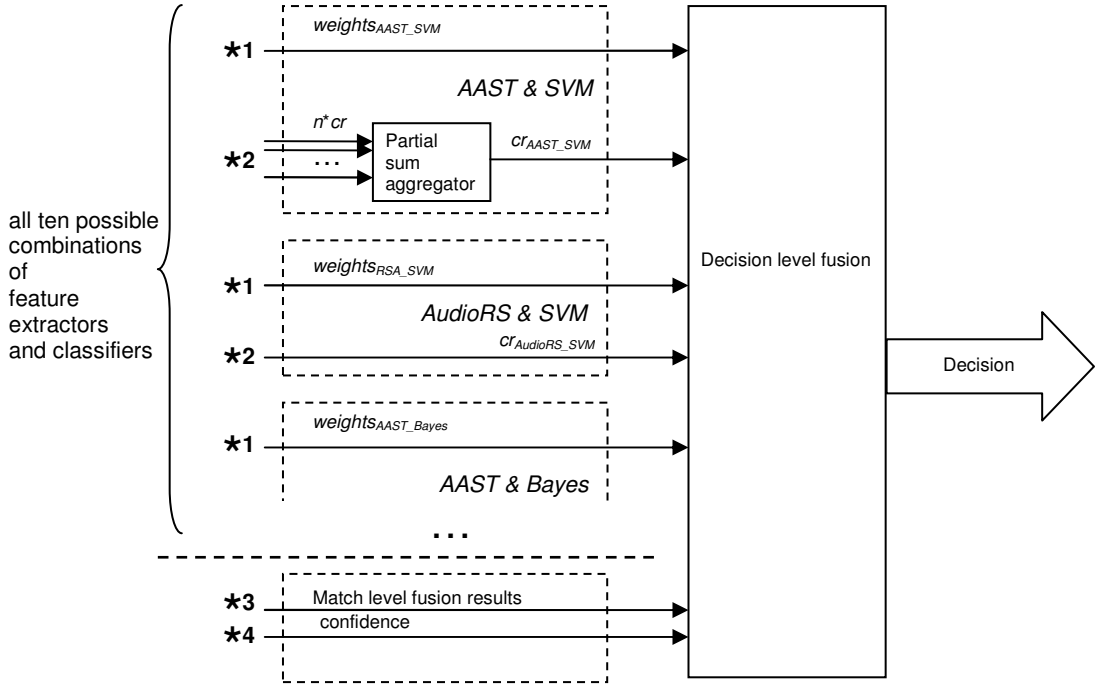

Figure 5: Decision level fusion based on the previous signal processing operations and the output of the match level fusion

4. TEST RESULTS

This section introduces the test results achieved based on the test scenario described in section 3. First, as a base for comparisons and to establish the required model quality estimations, the results of the analyses without decision level fusion are presented. Second, the main research goal of this paper – to show the impact of fusion in steganalysis – is addressed. The results for the secondary test goal – the evaluation of the impact of the key scenario – are compared for each test.

4.1. Steganalysis results without fusion

The complete results for the steganalysis for all three considered algorithms are, due to space constraints, presented in a separate file (APPENDIX_A.pdf; available at: http://omen.cs.uni-magdeburg.de/itiamsl/cms/front_content.php?idart=238). In this section the results are briefly summarised to act as point of reference for the remaining tests. If comparing in the results the performance of the feature extractors the results show a better performance for *AAST* in nearly all cases. If the performance of the classifiers is evaluated, then the *ADABoost* and *SimpleLogistics* outperform the other algorithms in most cases while the Bayesian classifier achieves very low results in nearly all tested cases. When evaluating the results algorithm by algorithm it can be stated that for *AS1* no significant classification results could be achieved for this algorithm. It shows the highest result for the model quality estimation with 52.2% (fixed key) and for the testing with 54.2% (for the variable key scenario). The evaluations for *AW1* and *AW3* show more promising results. Table 4 and Table 5 summarise from the complete results the best results achieved (in terms of classification accuracy) in the steganalysis without fusion on the algorithms and identifies the feature extractor and classifier combination which achieve these results.

Table 4: Best results achieved in model quality estimation (train. set: *aats389_Part1*, test set: *aats389_Part2*)

Alg.	fixed key		variable key	
	Accuracy (per file)	**Feature extractor & classifier**	**Accuracy (per file)**	**Feature extractor & classifier**
AS1	52.2%	*AudioRS & SimpleLog.*	50.0%	*AudioRS & SimpleLog.*
AW1	89.1%	*AAST & ADABoost*	93.5%	*AAST & J48*
AW3	60.9%	*AAST & SVM*	60.9%	*AAST & SVM*

Considering the results from testing presented in Table 5 it can be seen that the best accuracies achieved are for *AW1* approximately the same and for *AW3* higher (by 15%-17%) than the ones found in model quality estimation (Table 4). This result for *AW3* seems to be an outlier; in general (average over all ten feature extractor & classifier combinations) no significant better performance on the test set *testset24* can be noticed.

Table 5: Best results achieved in testing (train. set: *aats389_Part1*, test set: *testset24*)

Alg.	fixed key		variable key	
	Accuracy (per file)	**Feature extractor & classifier**	**Accuracy (per file)**	**Feature extractor & classifier**
AS1	52.1%	*AudioRS & SimpleLog.*	54.2%	*AudioRS & Bayes*
AW1	89.6%	*AAST & ADABoost*	89.6%	*AAST & ADABoost*
AW3	75.0%	*AAST & ADABoost*	77.1%	*AAST & ADABoost*

From the test results achieved, the choice of the key scenario seems to have little influence on the performance of feature extractor and classifier. Only in 5 out of the 30 direct comparisons (*AW1*: *AAST & SimpleLogistics, AAST & J48, AudioRS & J48*; *AW3*: *AudioRS & SimpleLogistics, AudioRS & J48*) between fixed and variable key the difference can be considered significant (>2%) and should be subjected to further research.

4.2. Steganalysis results with fusion

In this section the fusion steganalysis results are presented based on the test procedure introduced in section 3.3: first, for the match level fusion results computed by the five classifiers on the *AAST* output, second for a decision level fusion on *AAST*, *AudioRS* and the match level fusion output, based on the model quality estimations introduced in 4.1.

4.2.1. Match level fusion

The results for the match level fusion for the intra-window feature extractor *AAST* are presented in Table 6. The weights for and classifier output in this fusion are set to "1" when an *AAST* & classifier combination is considered significant (in the model quality estimations the ratio between "marked" and "unmarked" decisions is close to equal) and to "0" else (the classifier is classifying nearly everything as one of the two classes) – see Table 6 columns titled "weights".

Table 6: Match level fusion results (*AAST*; training set: *aats389_Part1*, test set: *testset24*) and error rates (TP=true positive, TN=true negative, FP=false positive, FN=false negative)

Alg.	Key scenario	*weights*					Accuracy (per file)	Error rates			
		SVM	*Bayes*	*Simp.Log.*	*ADABoost*	*J48*		TP	TN	FP	FN
AS1	fixed	1	1	0	0	0	50.0%	50.0%	0%	0%	50.0%
	variable	1	1	0	1	0	50.0%	50.0%	0%	0%	50.0%
AW1	fixed	1	0	1	1	1	93.8%	50.0%	43.8%	0.0%	6.3%
	variable	1	0	1	1	0	91.7%	50.0%	41.7%	0.0%	8.3%
AW3	fixed	1	0	1	1	0	75.0%	50.0%	25.0%	0.0%	25.0%
	variable	0	1	1	1	0	79.2%	45.8%	33.3%	4.2%	16.7%

As can be seen in the comparison between Table 6 and the results for steganalysis without fusion presented in section 4.1 the results are improved for *AW1* and *AW3* in three out of four cases. when combining the output of the *SVM*, *SimpleLogistics*, *ADABoost* and *J48* classifiers the result for *AW1* with fixed key rises from 89.6% (*AAST & ADABoost*, see Table 5) to 93.8%. When combining *SVM*, *SimpleLogistics* and *ADABoost* the result for *AW1* and variable keys rises from 89.6% to 91.7%. For *AW3* and fixed keys the result after match level fusion is with 75% the same as without fusion, while for variable keys the result is improved from 77.1% to 79.2% by combining *Bayes*, *SimpleLogistics* and *ADABoost.* The results for *AS1* resulted in all cases in an accuracy of 50% which equals the probability for guessing in this two-class classification problem.

4.2.2. Decision level fusion

Table 7 summarises briefly the best results achieved in decision level fusion when fusing for the three algorithms the output of the 10 feature extractor & classifier combinations and the output of the match level fusion for the *AAST* feature extractor. The weights for the fusion are set to "1" (the input is used) when a feature extractor & classifier combination is considered significant (≥52%) in the model quality estimations introduced in section 4.1, otherwise the weight is set to "0" (the input is not used).

Table 7: Summary of the best classification results achieved in decision level fusion (including the output of the match level fusion as one source for input; training set: *aats389_Part1*, test set: *testset24*) and error rates (TP=true positive, TN=true negative, FP=false positive, FN=false negative)

Alg.	Key scenario	*weights*											Accuracy (per file)	Error rates			
		AAST					*AudioRS*										
		SVM	*Bayes*	*Simp.Log.*	*ADABoost*	*J48*	*SVM*	*Bayes*	*Simp.Log.*	*ADABoost*	*J48*	*Match lev. Fus.*		TP	TN	FP	FN
AS1	fixed	0	1	1	1	1	1	1	0	1	1	0	52.1%	50.0%	2.1%	47.9%	0.0%
	variable	1	1	1	1	1	0	1	0	1	1	0	54.2%	37.5%	16.7%	33.3%	12.5%
AW1	fixed	1	1	1	1	1	0	1	1	0	0	1	93.8%	50.0%	43.8%	6.3%	0.0%
	variable	0	1	1	1	1	0	1	0	1	0	1	91.7%	50.0%	41.7%	8.3%	0.0%
AW3	fixed	0	1	1	0	0	0	0	0	0	0	1	75.0%	50.0%	25.0%	25.0%	0.0%
	variable	1	1	1	1	1	0	0	0	0	0	1	79.2%	45.8%	33.3%	16.7%	4.2%

As can be seen in the comparison between these decision level fusion results and the results from non-fusion steganalysis and match level fusion, the results presented in Table 7 match for each algorithm exactly the highest result achieved before (the same fact was established in test without using the output of the match level fusion as additional input). No increase in the classification accuracies could be achieved in the tests performed here. The reason for this is assumed to be the low reliability of the individual experts to be fused here. A different outcome would be expected if the individual feature extractor & classifier combinations had an accuracy >90%.

5. SUMMARY

For the primary test goal of showing the impact of fusion it can be said, that for the match level fusion results, which were obtained for the intra-window feature extractor *AAST* and using five classifiers, an increase of the classification accuracy is achieved for two of the three algorithms evaluated here. In the best test case it rose from 89.6% (*AAST & ADABoost*, see Table 5) to 93.8%.
When summarising the results for the decision level fusion the overall classification accuracy was not improved the conducted decision level fusion tests compared to non-fused or match level fusion results. This is consistent with the practical results on decision level fusion obtained by Kharazzi et al.[10] where the fusion accuracies are also only in few test cases better than the non-fused results.

As shown by the results on match level, fusion has to be considered a promising method to improve the detection accuracy in steganalysis and closely related sciences. In general we consider fusion as a useful method for improving the value of steganalysis as a reliable security mechanism for secure storage applications, e.g. in long term archiving environments. When comparing the results for the steganography algorithm and the two watermarking algorithms in the test set, the (expected) result was that the watermarking algorithms show a far higher detectability than the steganography algorithm. If this is expanded into the field of media forensics, were the *AAST* with the *SVM* classifier was in the past successfully employed for microphone classification[11], the high classification accuracies achieved there for the microphones support the assumption that this application might even more benefit from the introduced fusion approach than the steganography and watermark detection performed here.

When summarising the results for the secondary test goal (the impact of key selection) the results show a rather small impact of the key selection strategies tested on the classification results. Especially the results for the non-fusion analyses (see APPENDIX_A.pdf; available at:
http://omen.cs.uni-magdeburg.de/itiamsl/cms/front_content.php?idart=238) support the assumption that the choice of the key has in average for the three tested algorithms only limited impact on the classification accuracy. Only in 5 out of the 30 direct comparisons between fixed and variable key the difference can be considered significant and should be subjected to further research.

Based on the results of this paper, further research should be invested into more feature extractors for audio steganalysis, hopefully generating feature extractor & classifier combinations which show a higher relevance (higher classification accuracy) and therefore would allow for additional evaluations, e.g. message length estimations. Besides this most urgent need for further research additional directions would be evaluations on the complexity/scalability of the introduced approach, or an extension from a two-class problem (“marked” or “unmarked”) into a multi-class problem of identification of the used data hiding algorithm – which would also allow for the incorporation of rank level fusion operations. Additionally the fusion approach considered her should be transferred also into media forensics (e.g. for microphone identification).

ACKNOWLEDGEMENTS

The work in this paper has been supported in part by the European Commission through the FP7 ICT Programme under Contract FP7-ICT-216736 SHAMAN. The information in this document is provided as is, and no guarantee or warranty is given or implied that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. Thanks are expressed to Tobias Scheidat from AMSL for many fruitful discussions for applied classification in biometrics. Additional thanks go to Andrey Makrushin (OvGU/FIN/AMSL), Rene Schult (OvGU/FIN/KMD) and Georg Ruß (OvGU/FIN/CI) for the discussions on classification techniques and their help in choosing suitable classifiers for the classification problem at hand.

REFERENCES

[1] B. S. Atal: *Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification.* J. Acoust. Soc. Amer., vol. 55, no. 6, pp. 1304-1312, 1974.

[2] G. Bebis: Lecture CS479/679 *Pattern Recognition* (Spring'06), Introduction to Pattern Recognition. Department of Computer Science and Engineering, University of Nevada, Reno, NV 89557, 2006.

[3] C. Borgelt, H. Timm, and R. Kruse: *Probabilistic networks and fuzzy clustering as generalizations of naïve bayes classifiers*. In Computational Intelligence in Theory and Practice (Advances in Soft Computing), B. Reusch and K.-H. Temme, eds., pp. 121-138, Physica-Verlag, Heidelberg, Germany, Heidelberg, Germany, 2001.

[4] C.-C. Chang and C.-J. Lin: *LIBSVM: a Library for Support Vector Machines*. 2001.

[5] Y. Freund and R. E. Schapire: Experiments with a new boosting algorithm. Proc International Conference on Machine Learning, pp. 148-156, Morgan Kaufmann, San Francisco, 1996.

[6] J. Fridrich, M. Goljan, D. Hogea, and D. Soukal: *Quantitative Steganalysis of Digital Images: Estimating the Secret Message Length*. ACM Multimedia Systems Journal, Special issue on Multimedia Security, Vol. 9(3), pp. 288-302), 2003.

[7] J. Fridrich, M. Goljan, and R. Du: *Detecting LSB steganography in color and gray-scale images*. Mag. IEEE Multimedia (Special Issue on Security), pp. 22-28, Oct.-Dec., 2001.

[8] K. Hempstalk: *Digital Invisible Ink Toolkit*. http://diit.sourceforge.net/, 2005.

[9] T. Hoppe, A. Lang, J. Dittmann: *Evaluierung der Bedrohung durch fortschrittliche Angriffstechniken von Programmen mit Schadensfunktion*. Proc. 10. Deutscher IT-Sicherheitskongress des BSI; SecuMedia Verlag Ingelheim, pp. 31-49, ISBN 978-3-922746-98-0, 2007.

[10] M. Kharrazi, H. T. Sencar, and N. Memon: *Improving steganalysis by fusion techniques: A case study with image steganography*. In Transactions on Data Hiding and Multimedia Security *I*, Y. Q. Shi, ed., Springer LNCS 4300, 2006.

[11] C. Kraetzer and J. Dittmann: *Cover Signal Specific Steganalysis: the Impact of Training on the Example of two Selected Audio Steganalysis Approaches*. Proc. of Security, Forensics, Steganography, and Watermarking of Multimedia Contents X. Electronic Imaging Conference 6819, IS&T/SPIE, 2008.

[12] C. Kraetzer and J. Dittmann: *Impact of Feature Selection in Classification for Hidden Channel Detection on the Example of Audio Data Hiding*. Proc. ACM Multimedia and Security Workshop 2008.

[13] C. Kraetzer, A. Oermann, J. Dittmann and A. Lang: *Digital Audio Forensics: A First Practical Evaluation on Microphone and Environment Classification*. Proc. ACM Multimedia and Security Workshop 2007.

[14] C. Kraetzer and J. Dittmann: *Pros and Cons of Mel-cepstrum based Audio Steganalysis using SVM Classification*. Proceedings of Information Hiding 2007.

[15] N. Landwehr, M. Hall, and E. Frank: *Logistic Model Trees*, Proc.ECML'03, 2003

[16] A. Oermann, T. Scheidat, C. Vielhauer, and J. Dittmann: *Semantic Fusion for Biometric User Authentication as Multimodal Signal Processing*. Lecture Notes in Computer Science (4105), ISSN: 03029743, Springer Verlag, 2006.

[17] R. Quinlan: *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers, San Mateo, CA, 1993.

[18] A.A. Ross, K. Nandakumar, and A.K. Jain: *Handbook of Multibiometrics*. International Series on Biometrics. Springer Verlag, 2006.

[19] C. Sanderson and K.K. Paliwal: *Information Fusion and Person Verification Using Speech and Face Information*, Technical ReportIDIAP 02-33, Martigny, Switzerland, 2002.

[20] V. Vapnik: *The nature of statistical learning theory*, Springer Verlag, New York, 1995.

[21] Vielhauer, C., Schimke, S., Thanasis, V., Stylianou, Y., Fusion Strategies for Speech and Handwriting Modalities in HCI. Proceedings of SPIE Electronic Imaging - Security, Steganography and Watermarking of Multimedia Contents VI, 2005.

[22] I. H. Witten and E. Frank: *Data Mining: Practical machine learning tools and techniques.* 2nd Edition, Morgan Kaufmann, San Francisco, 2005.