# МАТЕМАТИЧЕСКИЕ МЕТОДЫ СТЕГАНОГРАФИИ

УДК 519.7

## ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ МЕТОДОВ СТЕГОАНАЛИЗА ПРИ ПОМОЩИ ПРЕДВАРИТЕЛЬНОЙ ФИЛЬТРАЦИИ КОНТЕЙНЕРОВ[1]

В. А. Монарёв*, А. И. Пестунов**

*Институт вычислительных технологий СО РАН, г. Новосибирск, Россия
**Новосибирский государственный университет экономики и управления, г. Новосибирск, Россия

Предлагается новый подход к стегоанализу, названный «предварительной фильтрацией», который заключается в том, чтобы перед финальным обнаружением добавить этап отбора «хороших» контейнеров, наличие/отсутствие внедрённой информации в которых может быть определено более достоверно, чем во всём множестве. При этом размер данного подмножества, точнее, его доля по отношению ко всему контрольному множеству может рассматриваться как дополнительная характеристика метода стегоанализа. Предлагаются три конкретных метода для предварительной фильтрации, которые названы «наивный метод», «простая классификация» и «комбинированная классификация». Приводятся результаты экспериментов по предварительной фильтрации изображений из известного множества BOSSbase v1.01.

**Ключевые слова:** *стегоанализ, ошибка обнаружения, адаптивное внедрение, HUGO, ансамблевый классификатор.*

DOI 10.17223/20710410/32/6

## ENHANCING STEGANALYSIS ACCURACY VIA TENTATIVE FILTERING OF STEGO-CONTAINERS

V. A. Monarev*, A. I. Pestunov**

*Institute of Computational Technologies SB RAS, Novosibirsk, Russia
**Novosibirsk State University of Economics and Management, Novosibirsk, Russia

**E-mail:** viktor.monarev@gmail.com, pestunov@gmail.com

We introduce a new approach to steganalysis called “the tentative filtering” and consisting in inserting an additional filtering phase before the final classification in order to select those containers where stego-information can be reliably detected. The size of this “good” subset of containers can be considered as an additional characteristic of the detector. We introduce three methods for implementing the tentative filtering: the naive method, the simple classification, and the combined classification. The experiments demonstrate that it is possible to select about 35 % of BOSSbase v1.01 images, for which HUGO 0.4 bpp is detected with the error less than 0.003, while the error over the whole set is 0.141. It is also demonstrated that it is possible to select

[1]Работа поддержана грантом РФФИ, проект № 14-01-31484 (мол_а).

about 5 % images, for which HUGO 0.1 bpp is detected with the error less than 0.05, while the whole set gives the error 0.37 (which is not quite a reliable detection).

**Keywords:** *steganalysis, detection error, image features, SRM, adaptive steganography, HUGO, ensemble classifier.*

## Введение

Вычисление ошибки обнаружения (detection error) по формуле $P_{\mathrm{E}} = (P_{\mathrm{FA}} + P_{\mathrm{MD}})/2$, где $P_{\mathrm{FA}}$ — вероятность ложного срабатывания (False Alarm), а $P_{\mathrm{MD}}$ — вероятность пропущенного обнаружения (Missed Detection), в последнее время является одним из наиболее распространённых подходов к оценке точности методов стегоанализа [1 – 5].

Проблема заключается в том, что на практике статистическая модель контейнеров редко бывает известной и ошибку обнаружения не удаётся вычислить аналитически. Обычно она вычисляется экспериментально и равняется отношению числа контейнеров, на которых метод стегоанализа отработал правильно, к числу всех контейнеров в контрольном множестве. Для этой цели специалисты по стегоанализу используют известные стандартизованные множества контейнеров, например BOSSbase [6, 7], BOWS2 [8] или NRCS [9]. Тем не менее ошибка обнаружения, вычисленная по различным множествам, может отличаться из-за их специфичных свойств (наличия шумов, степени сжатия и пр.) [10]. Более того, в рамках одного множества контейнеров могут выделяться подмножества, свойства которых различаются, что приводит к различиям в ошибках обнаружения, если вычислять их по этим подмножествам отдельно.

В настоящей работе предлагается новый подход к стегоанализу, который назван «предварительная фильтрация». Идея подхода заключается в том, чтобы перед финальным обнаружением добавить этап отбора тех контейнеров (назовём их «хорошими»), наличие/отсутствие внедрённой информации в которых может быть определено более достоверно, чем во всём множестве. Другими словами, ошибка обнаружения, вычисленная по отобранному подмножеству, будет ниже, чем ошибка, вычисленная по всему контрольному множеству. При этом размер данного подмножества, точнее, его доля по отношению ко всему контрольному множеству может рассматриваться как дополнительная характеристика метода стегоанализа. Аналогичная ситуация имеет место в криптографии при разработке атак в предположении использования слабых ключей, когда размер множества слабых ключей является одним из показателей эффективности атаки [11, 12]. Предварительная фильтрация позволит не только снизить ошибку обнаружения, но и более тонко оценивать точность методов стегоанализа, поскольку появляется возможность выбирать контейнеры, свойства которых подходят для заданного метода стегоанализа.

Предварительную фильтрацию следует рассматривать как общий подход к стегоанализу, поэтому конкретных методов в рамках этого подхода может быть разработано довольно много. В данной работе предлагается три возможных варианта: наивный метод, простая классификация и комбинированная классификация, являющаяся комбинацией первых двух методов. Эксперименты показали, что предварительная фильтрация позволяет выбрать порядка 35 % изображений из известного множества BOSSbase v1.01 [7], для которых метод адаптивной стеганографии HUGO 0,4 битов на пиксель (б/п) обнаруживается с ошибкой менее 0,003, в то время как ошибка, вычисленная по всему множеству, составляет 0,141. Показано также, что из всего множества можно выбрать порядка 5 % изображений, для которых HUGO 0,1 б/п определяется с ошибкой менее чем 0,05, тогда как ошибка по всему множеству составляет 0,37.

## 1. Предварительные замечания и обозначения

Предлагаемый подход (и методы в его рамках) может применяться к любым контейнерам, но поскольку все эксперименты проводились на изображениях, то во избежание разночтений далее вместо термина «контейнер» везде используется «изображение».

### 1.1. Задача бинарной классификации в стеганографии

Задача бинарной классификации заключается в том, чтобы отнести заданное изображение к одному из двух классов — пустое или заполненное, причём стегоаналитик действует по следующему сценарию [3]:

1) имеет доступ к изображениям, которые обладают статистическими свойствами, схожими с теми, которые используются для внедрения информации;
2) знает алгоритм внедрения и точный размер внедряемого сообщения (обычно он измеряется в битах на пиксель);
3) знает, какой объект он должен исследовать.

В рамках настоящей работы не затрагивается количественный стегоанализ [13, 14], когда стегоаналитик не знает размера внедряемого сообщения.

Современные подходы к решению задачи бинарной классификации состоят из двух основных этапов: выделения признаков (feature extraction) из изображения и непосредственно классификации [3]. При этом предполагается, что у стегоаналитика имеется в распоряжении некоторое число пустых и заполненных изображений, составляющих обучающее множество (следует из первого требования сценария: стегоаналитик может внедрить случайную информацию в пустые изображения). Далее стегоаналитик действует по следующему алгоритму:

1) извлечь признаки из изображений, составляющих обучающее множество;
2) обучить классификатор различать признаки пустых и заполненных изображений;
3) извлечь признаки очередного изображения из контрольного множества и с помощью обученного классификатора отнести его к классу пустых/заполненных.

### 1.2. Ансамблевый классификатор и его элементы

Предлагаемые в работе методы опираются на идею применения ансамблевых классификаторов к задачам стегоанализа [4]. Ансамблевые классификаторы называют «отличной альтернативой методу опорных векторов» из-за их хорошей производительности и конкурентоспособной эффективности [15]. Эти классификаторы, в частности, применялись победителями известного конкурса по стегоанализу BOSS competition [15]. Схема работы ансамблевого классификатора, как она описана в [4], следующая:

1) взять $d$ признаков (таких, как SRM [1], SPAM [10], PSRM [16] и т. д.);
2) получить $L$ случайно выбранных подмножеств из множества всех признаков, каждое из которых состоит из $d_{\text{sub}} < d$ признаков;
3) обучить $L$ элементов ансамблевого классификатора на обучающем множестве различать пустые/заполненные изображения.

Пусть $N_{\text{votes}}(z)$ — число элементов ансамбля, голосующих за принадлежность изображения $z$ классу пустых изображений:

$$N_{\text{votes}}(z) = \sum_{l=1}^{L} B_l(z).$$

Каждый элемент ансамбля работает следующим образом:

$$B_l(z) = \begin{cases} 1, & \text{элемент } l \text{ голосует за то, что } z \text{ — пустое,} \\ 0, & \text{элемент } l \text{ голосует за то, что } z \text{ — заполненное.} \end{cases}$$

Решение о принадлежности очередного тестового изображения к тому или иному классу принимается согласно следующему правилу:

$$\text{Ensemble-Rule}(L, N_{\text{votes}}) = \begin{cases} 1, & \text{если } N_{\text{votes}} > L/2, \\ 0, & \text{если } N_{\text{votes}} < L/2, \\ \text{random}\{0,1\} & \text{иначе.} \end{cases}$$

### 1.3. Формирование обучающего и контрольного множеств

Далее при проведении экспериментов обучающее и контрольное множества формируются на основе одной из наиболее известных баз изображений BOSSbase v1.01 [6, 7], которая часто используется специалистами по стеганографии и стегоанализу в качестве источника изображений. Данная база содержит 10000 чёрно-белых 8-битовых изображений размера 512×512 пикселей.

Обозначим обучающее и контрольное множества через $\mathcal{X}^p$ и $\mathcal{Y}^p$ соответственно, где $p$ указывает на размер внедрения в битах на пиксель, и опишем процесс их формирования:

1) всё множество BOSSbase v1.01 разделено на два подмножества $\mathcal{X}_0$ и $\mathcal{Y}_0$, где $|\mathcal{X}_0| = 8000$ и $|\mathcal{Y}_0| = 2000$;
2) посредством случайного внедрения $p$ б/п во все изображения из $\mathcal{X}_0$ и $\mathcal{Y}_0$ получены множества $\mathcal{X}_1^p$ и $\mathcal{Y}_1^p$ соответственно;
3) обучающее множество формируется как $\mathcal{X}^p = \mathcal{X}_0 \cup \mathcal{X}_1^p$;
4) контрольное множество формируется как $\mathcal{Y}^p = \mathcal{Y}_0 \cup \mathcal{Y}_1^p$.

Таким образом, $|\mathcal{X}^p| = 16000$, $|\mathcal{Y}^p| = 4000$ и в каждом множестве половина изображений пусты, половина заполнены. Далее индекс $p$, обозначающий размер внедрения, будем опускать (это не должно вызвать путаницы у читателя), и множества обозначаются через $\mathcal{X} = \mathcal{X}_0 \cup \mathcal{X}_1$ (обучающее) и $\mathcal{Y} = \mathcal{Y}_0 \cup \mathcal{Y}_1$ (контрольное).

### 1.4. Используемые в классификации признаки

Как известно, в методах машинного обучения применяются признаки, извлекаемые из объектов классификации. В настоящей работе в качестве таких признаков берутся SRM-признаки (Spatial Rich Model) [1], позволяющие создавать одни из наиболее эффективных методов стегоанализа. Их более новый вариант, PSRM-признаки (Projection Spatial Rich Model) [16], снижают ошибку обнаружения лишь незначительно, но при этом повышают сложность методов, существенно замедляя работу. Размерность пространства SRM-признаков составляет 34,671. Программа для извлечения этих признаков из изображений взята с сайта [17].

### 1.5. Элементы ансамбля

Существуют разные варианты выбора элементов ансамбля, но в экспериментах мы следуем рекомендациям [4] и используем линейный дискриминант Фишера (Fisher Linear Discriminant) [18] в силу его быстрого обучения и хороших результатов, которые показывают методы стегоанализа на его основе. Применяются два типа элементов ансамбля, которые обозначаются соответственно через

$$B_l,\ l = 1, \ldots, L \quad \text{и} \quad B'_m,\ m = 1, \ldots, M.$$

Таким образом, в первом случае их число равно $L$, а во втором — $M$. Каждому элементу приписывается по 800 случайно выбранных SRM-признаков.

### 1.6. Алгоритм внедрения информации

Эффективность предлагаемого подхода исследуется посредством его применения к обнаружению информации, внедрённой с помощью адаптивного метода Highly Undetectable Steganography (HUGO) [19]. На сегодняшний день этот метод считается наиболее трудно обнаружимым (см., например, результаты [16], где HUGO сравнивается с другими методами адаптивного внедрения, такими, как WOW [20] и UNIWARD [21]). HUGO базируется на ±1-стеганографии (LSB matching), но при его использовании места для внедрения выбираются не случайно, а вероятностно в зависимости от SPAM-признаков [19]. Такая модификация позволяет увеличить размер внедряемого сообщения приблизительно в 7 раз по сравнению с внедрением при помощи ±1-стеганографии с сохранением уровня стойкости (другими словами, при такой же ошибке обнаружения).

### 1.7. Лучшие результаты обнаружения метода HUGO

Для того чтобы оценить эффективность предварительной фильтрации, очевидно, необходимо сравнить ошибку обнаружения, вычисленную по всему множеству, с ошибкой обнаружения, вычисленной по подмножеству, полученному после фильтрации. При этом необходимо иметь реализацию ансамблевого классификатора, чтобы вычислять эти ошибки. Однако поскольку классификатор имеет достаточно много параметров, а в литературе они не всегда приводятся, то мы реализовали ансамблевый классификатор самостоятельно. Для проверки правильности реализации и достоверности вычисленных для него ошибок обнаружения мы сравнили ошибку обнаружения для этой реализации с ошибками обнаружения лучших реализаций [16]. Данные, приведённые в табл. 1, показывают, что значения ошибок обнаружения для нашей реализации согласуются с существующими данными. Далее при сравнении результатов предварительной фильтрации будем ориентироваться на ошибки обнаружения для нашей реализации (0,44, 0,37 и 0,141), поскольку ошибка обнаружения для «хорошего» подмножества вычисляется с помощью неё же.

Таблица 1

**Лучшие результаты обнаружения метода HUGO (ансамблевый классификатор)**

| Размер внедрения, б/п | Ошибка обнаружения, вычисленная по всему контрольному множеству (BOSSbase v1.01) | |
|---|---|---|
| | Результаты из [16] при различных параметрах | Наша реализация, SRM-признаки, $L = 500$ |
| 0,05 | — | 0,44 |
| 0,10 | 0,3564–0,3757 | 0,37 |
| 0,20 | 0,2397–0,2701 | — |
| 0,40 | 0,1172–0,1383 | 0,141 |

Далее $\mathcal{Y}^{\text{good}}$ — подмножество «хороших» изображений, которые отбираются после предварительной фильтрации, а $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ — ошибка обнаружения, вычисленная по данному подмножеству. В экспериментах сравниваются $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ и $P_{\text{E}}(\mathcal{Y})$ со стратегический целью снизить $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ и увеличить $|\mathcal{Y}^{\text{good}}|$ — размер $\mathcal{Y}^{\text{good}}$.

## 2. Описание и экспериментальное обоснование предлагаемого подхода

### 2.1. Базовая идея

Основной идеей, на которую опираются все методы, предлагаемые в работе, является естественное предположение о том, что если для некоторого изображения $z$ значение $N_{\text{votes}}(z)$ близко к 0 или к $L$, то можно быть более уверенным в решении, чем в случае, когда это значение далеко от 0 и от $L$. Другими словами, если элементов ансамбля, проголосовавших за наличие информации, очень мало, то, скорее всего, её там действительно нет, а если таких голосов много, то, скорее всего, она там есть.

Данная идея непосредственно реализована в первом предлагаемом методе, который выбирает «хорошие» изображения, для которых

$$N_{\text{votes}}(z) \leqslant T^{\text{left}} \quad \text{или} \quad N_{\text{votes}}(z) \geqslant T^{\text{right}}$$

для некоторых заданных порогов $T^{\text{left}}$ и $T^{\text{right}}$. Следующий метод, названный «простой классификацией», заключается в том, чтобы обучить некий дополнительный классификатор различать между «хорошими» и «плохими» изображениями и использовать его для выбора «хороших» изображений. Наконец, третий метод является комбинацией первых двух.

### 2.2. Метод 1: наивный

Идея, воплощённая в данном методе, заключается в определении двух порогов $T^{\text{left}}$ и $T^{\text{right}}$, таких, что $T^{\text{left}}$ близок к 0, а $T_{\text{right}}$ близок к $L$ (количеству элементов ансамбля), и разделении контрольного множества $\mathcal{Y}$ на «хорошее» и «плохое» подмножества согласно этим порогам следующим образом (алгоритм 1): $\mathcal{Y} = \mathcal{Y}^{\text{good}} \cup \mathcal{Y}^{\text{bad}}$, где

$$\mathcal{Y}^{\text{good}} = \{y \in \mathcal{Y} : N_{\text{votes}}(y) \leqslant T^{\text{left}} \text{ или } N_{\text{votes}}(y) \geqslant T^{\text{right}}\}, \quad \mathcal{Y}^{\text{bad}} = \mathcal{Y} \setminus \mathcal{Y}^{\text{good}}.$$

---

**Алгоритм 1.** Наивный метод фильтрации Naive-Method$(\mathcal{Z}, T^{\text{left}}, T^{\text{right}})$

**Вход:** $\mathcal{Z}$ — множество, из которого выделяются «хорошие» изображения, $T^{\text{left}}$ и $T^{\text{right}}$ — левый и правый пороги соответственно

**Выход:** $\mathcal{Z}^{\text{good}} \subseteq \mathcal{Z}$ — подмножество «хороших» изображений

1: Обучить элементы ансамбля $B_l, \ldots, B_L$ на подмножествах обучающей выборки $\mathcal{X}_0$ и $\mathcal{X}_1$ различать пустые/заполненные изображения

2: Для каждого изображения $z \in \mathcal{Z}$ вычислить число элементов ансамбля, проголосовавших за то, что изображение является пустым: $N_{\text{votes}}(z) = \sum_{l=1}^{L} B_l(z)$

3: Сформировать подмножество

$$\mathcal{Z}^{\text{good}} = \{z \in \mathcal{Z} : N_{\text{votes}}(z) \leqslant T^{\text{left}} \text{ или } N_{\text{votes}}(z) \geqslant T^{\text{right}}\}$$

---

При проведении экспериментов в качестве параметра $\mathcal{Z}$ в алгоритм 1 передавалось контрольное множество $\mathcal{Y}$. Рабочая гипотеза о том, что данная фильтрация позволит выделить подмножество $\mathcal{Y}^{\text{good}}$, такое, что $P_{\text{E}}(\mathcal{Y}^{\text{good}}) < P_{\text{E}}(\mathcal{Y})$, в целом оправдалась, однако по-настоящему впечатляющими результаты оказались только при анализе внедрения HUGO 0,40 б/п. Во-первых, множество «хороших» изображений оказалось достаточно большим (более 32 % от $\mathcal{Y}$). Во-вторых, ошибка обнаружения снизилась значительно (в зависимости от размера внедрения она лежит в промежутке

0,0016–0,0042), что приблизительно в 50 раз меньше, чем $P_{\mathrm{E}}(\mathcal{Y}) = 0{,}141$ (см. табл. 1). Для внедрения HUGO 0,05 и 0,10 б/п ошибка $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ также меньше, чем $P_{\mathrm{E}}(\mathcal{Y})$, но различие не такое существенное и, кроме того, размеры подмножеств $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ крайне малы.

Таблица 2

**Наивный метод ($T^{\mathbf{left}} = 1$, $T^{\mathbf{right}} = L - 1$, % — доля (в %) «хороших» изображений)**

| | HUGO 0,05 б/п | | | HUGO 0,10 б/п | | | HUGO 0,40 б/п | | |
|---|---|---|---|---|---|---|---|---|---|
| $L$ | $\lvert\mathcal{Y}^{\mathrm{good}}\rvert$ | % | $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ | $\lvert\mathcal{Y}^{\mathrm{good}}\rvert$ | % | $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ | $\lvert\mathcal{Y}^{\mathrm{good}}\rvert$ | % | $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ |
| 100 | 50 | 1,25 | 0,260 | 271 | 6,76 | 0,140 | 1651 | 41,28 | 0,0042 |
| 200 | 31 | 0,76 | 0,258 | 176 | 4,40 | 0,125 | 1462 | 36,55 | 0,0041 |
| 300 | 23 | 0,56 | 0,304 | 137 | 3,43 | 0,124 | 1384 | 34,60 | 0,0022 |
| 400 | 16 | 0,40 | 0,313 | 117 | 2,93 | 0,120 | 1323 | 33,08 | 0,0023 |
| 500 | 14 | 0,35 | 0,286 | 106 | 2,65 | 0,123 | 1285 | 32,13 | 0,0016 |

### 2.3. Метод 2: простая классификация

С расчётом увеличить размер подмножества $\mathcal{Y}^{\mathrm{good}}$, особенно для внедрения HUGO 0,05 и 0,10 б/п, мы решили обучить дополнительный ансамблевый классификатор, который мог бы различать «хорошие» и «плохие» изображения (алгоритм 2). Идея метода заключается в том, чтобы сначала разделить обучающее множество на «хорошее» и «плохое» подмножества с помощью наивного метода (алгоритм 1), а затем обучить дополнительный классификатор на этом разбиении.

**Алгоритм 2.** Метод простой классификации Simple-Classification($\mathcal{Z}, T^{\mathrm{left}}, T^{\mathrm{right}}$)

**Вход:** $\mathcal{Z}$ — множество, из которого выделяются «хорошие» изображения, $T^{\mathrm{left}}$ и $T^{\mathrm{right}}$ — левый и правый пороги соответственно

**Выход:** $\mathcal{Z}^{\mathrm{good}} \subseteq \mathcal{Z}$ — подмножество «хороших» изображений

1: Получить подмножество «хороших» изображений из обучающего множества с помощью наивного метода $\mathcal{Z}^{\mathrm{good}} :=$ Naive-Method($\mathcal{Z}, T^{\mathrm{left}}, T^{\mathrm{right}}$)
2: Получить подмножество «плохих» изображений $\mathcal{Z}^{\mathrm{bad}} := \mathcal{Z} \setminus \mathcal{Z}^{\mathrm{good}}$
3: Обучить элементы дополнительного ансамбля $B'_1, \ldots, B'_M$ различать «хорошие»/«плохие» изображения на разбиении $\mathcal{Z}^{\mathrm{good}}$ (класс 0) и $\mathcal{Z}^{\mathrm{bad}}$ (класс 1)
4: Получить подмножество $\mathcal{Z}^{\mathrm{good}} = \{z \in \mathcal{Z} : \text{Ensemble-Rule}(M, N'_{\mathrm{votes}}(z)) = 0\}$

Как показали эксперименты (табл. 3), простая классификация действительно позволила значительно увеличить размер подмножества $\mathcal{Y}^{\mathrm{good}}$ по сравнению с наивным методом (табл. 2), хотя ошибка обнаружения $P_{\mathrm{E}}(\mathcal{Y}^{\mathrm{good}})$ уменьшилась незначительно по сравнению с результатами, приведёнными в табл. 1. Более того, для внедрения HUGO 0,40 б/п она даже возросла.

### 2.4. Метод 3: комбинированная классификация

Для получения более эффективного метода предварительной фильтрации, который позволил бы снизить ошибку обнаружения и увеличить размер множества «хороших» изображений (как минимум для HUGO 0,05 б/п и HUGO 0,10 б/п, поскольку для HUGO 0,40 б/п наивный метод уже показал впечатляющие результаты), создана комбинация двух предложенных выше методов. Идея комбинированного метода заключается в том, чтобы выделить «хорошие» подмножества наивным методом и методом

Таблица 3

**Простая классификация (% — доля «хороших» изображений)**

| $T^{\text{left}}$ | $T^{\text{right}}$ | HUGO 0,05 б/п | | | HUGO 0,10 б/п | | | HUGO 0,40 б/п | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ |
| 1 | 499 | 292 | 7 | 0,353 | 1198 | 30 | 0,244 | 1903 | 48 | 0,0189 |
| 2 | 499 | 280 | 7 | 0,346 | 1139 | 28 | 0,227 | 2022 | 51 | 0,0218 |
| 3 | 499 | 455 | 11 | 0,365 | 1158 | 29 | 0,225 | 2061 | 52 | 0,0213 |
| 1 | 498 | 232 | 6 | 0,332 | 1510 | 38 | 0,273 | 1911 | 48 | 0,0167 |
| 2 | 498 | 337 | 8 | 0,359 | 1189 | 30 | 0,230 | 2132 | 53 | 0,0225 |
| 3 | 498 | 528 | 13 | 0,371 | 1302 | 33 | 0,247 | 2009 | 50 | 0,0184 |
| 1 | 497 | 284 | 7 | 0,357 | 1171 | 29 | 0,243 | 2045 | 51 | 0,0220 |
| 2 | 497 | 340 | 9 | 0,362 | 1204 | 30 | 0,236 | 2091 | 52 | 0,0210 |
| 3 | 497 | 347 | 9 | 0,378 | 1182 | 30 | 0,228 | 2048 | 51 | 0,0215 |

простой классификации, а затем взять их пересечение. Согласно рабочей гипотезе, при удачном подборе порогов $T_1^{\text{left}}$ и $T_1^{\text{right}}$ для наивного метода и $T_2^{\text{left}}$ и $T_2^{\text{right}}$ для метода простой классификации окажется возможным выбрать достаточно большое «хорошее» подмножество $\mathcal{Y}^{\text{good}}$, для которого ошибка обнаружения $P_{\text{E}}(\mathcal{Y}^{\text{good}})$ будет заметно меньше, чем по всему контрольному множеству. Описание метода приведено в алгоритме 3.

**Алгоритм 3.** Метод комбинированной классификации Combined-Classification$(\mathcal{Z}, T_1^{\text{left}}, T_1^{\text{right}}, T_2^{\text{left}}, T_2^{\text{right}})$

**Вход:** $\mathcal{Z}$ — множество, из которого выделяются «хорошие» изображения, $T_1^{\text{left}}$ и $T_1^{\text{right}}$ — пороги для наивного метода, $T_2^{\text{left}}$ и $T_2^{\text{right}}$ — пороги для простой классификации

**Выход:** $\mathcal{Z}^{\text{good}} \subseteq \mathcal{Z}$ — подмножество «хороших» изображений

1: $\mathcal{Z}_1^{\text{good}} = $ Naive-Method$(\mathcal{Z}, T_1^{\text{left}}, T_1^{\text{right}})$
2: $\mathcal{Z}_2^{\text{good}} = $ Simple-Classification$(\mathcal{Z}, T_2^{\text{left}}, T_2^{\text{right}})$
3: $\mathcal{Z}^{\text{good}} = \mathcal{Z}_1^{\text{good}} \cap \mathcal{Z}_2^{\text{good}}$

Схема экспериментов следующая: фиксировалась максимально допустимая ошибка $P_{\text{E}}^*$ и подбирались пороги, обеспечивающие максимальный размер подмножества «хороших» изображений $\mathcal{Y}^{\text{good}}(T_1^{\text{left}}, T_1^{\text{right}}, T_2^{\text{left}}, T_2^{\text{right}})$ при условии, что $P_{\text{E}}(\mathcal{Y}^{\text{good}}) \leqslant P_{\text{E}}^*$.

Более формально, $(T_2^{\text{left}}(P_{\text{E}}^*), T_2^{\text{right}}(P_{\text{E}}^*)) = \arg\max_{t^{\text{left}}, t^{\text{right}}} |\mathcal{Y}^{\text{good}}(T_l^{\text{left}}, T_l^{\text{right}}, t^{\text{left}}, t^{\text{right}})|$ при условии, что $P_{\text{E}}(\mathcal{Y}^{\text{good}}(T_l^{\text{left}}, T_l^{\text{right}}, t^{\text{left}}, t^{\text{right}})) \leqslant P_{\text{E}}^*$.

В табл. 4–7 представлены результаты экспериментов по комбинированной классификации. Параметры: табл. 4 — HUGO 0,1 б/п, $L = 500$, $M = 1$; табл. 5 — HUGO 0,05 б/п, $L = 500$, $M = 11$, $T_1^{\text{left}} = 10$, $T_1^{\text{right}} = 490$; табл. 6 — HUGO 0,4 %, $L = 500$, $M = 11$, $T_1^{\text{left}} = 20$, $T_1^{\text{right}} = 480$; табл. 7 — HUGO 0,1 б/п, $L = 500$, $M = 11$.

В табл. 7 показано, что из всего множества BOSSbase можно выбрать 5 % изображений, для которых ошибка обнаружения стеганографии HUGO 0,1 б/п не превосходит 0,05, в то время как ошибка по всему множеству составляет 0,37 (см. табл. 1), что нельзя считать достоверным обнаружением, поскольку значение близко к 0,5 (случайному угадыванию).

Таблица 4

| | $T_1^{\text{left}}=1,\ T_1^{\text{right}}=499$ | | | | $T_1^{\text{left}}=2,\ T_1^{\text{right}}=498$ | | | |
|---|---|---|---|---|---|---|---|---|
| $P_{\text{E}}^*$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ |
| 0,04 | 187 | 5 | 1 | 489 | 202 | 12 | 2 | 490 |
| 0,05 | 230 | 6 | 2 | 485 | 251 | 12 | 2 | 481 |
| 0,06 | 252 | 6 | 4 | 485 | 303 | 12 | 17 | 490 |
| 0,07 | 346 | 8 | 19 | 483 | 391 | 12 | 27 | 481 |
| 0,08 | 401 | 10 | 33 | 489 | 463 | 12 | 30 | 464 |

Таблица 5

| $P_{\text{E}}^*$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ |
|---|---|---|---|---|
| 0,15 | 21 | 0,5 | 0 | 487 |
| 0,18 | 28 | 0,7 | 3 | 487 |
| 0,21 | 58 | 1,5 | 20 | 486 |
| 0,24 | 92 | 2,3 | 42 | 470 |

Таблица 6

| $P_{\text{E}}^*$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ |
|---|---|---|---|---|
| 0,00000 | 1481 | 37 | 1 | 492 |
| 0,00125 | 1655 | 41 | 9 | 492 |
| 0,00225 | 1680 | 42 | 9 | 490 |
| 0,00325 | 1853 | 46 | 99 | 492 |

Таблица 7

| | $T_1^{\text{left}}=1,\ T_1^{\text{right}}=499$ | | | | $T_1^{\text{left}}=2,\ T_1^{\text{right}}=498$ | | | | $T_1^{\text{left}}=10,\ T_1^{\text{right}}=490$ | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $P_{\text{E}}^*$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ | $\lvert\mathcal{Y}^{\text{good}}\rvert$ | % | $T_2^{\text{left}}$ | $T_2^{\text{right}}$ |
| 0,01 | 0 | 0 | — | — | 0 | 0 | — | — | 0 | 0 | — | — |
| 0,02 | 157 | 3,9 | 1 | 471 | 0 | 0 | — | — | 0 | 0 | — | — |
| 0,03 | 181 | 4,5 | 2 | 464 | 175 | 4,4 | 1 | 485 | 0 | 0 | — | — |
| 0,04 | 191 | 4,8 | 2 | 455 | 227 | 5,7 | 1 | 464 | 203 | 5,1 | 2 | 490 |
| 0,05 | 209 | 5,2 | 3 | 438 | 255 | 6,4 | 2 | 455 | 284 | 7,1 | 2 | 470 |
| 0,06 | 267 | 6,7 | 27 | 455 | 336 | 8,4 | 28 | 464 | 334 | 8,4 | 27 | 490 |
| 0,07 | 293 | 7,3 | 45 | 464 | 388 | 9,7 | 45 | 455 | 460 | 11,5 | 27 | 455 |
| 0,08 | 354 | 8,9 | 90 | 438 | 456 | 11,4 | 93 | 442 | 518 | 13,0 | 33 | 434 |
| 0,09 | 378 | 9,5 | 90 | 403 | 482 | 12,1 | 96 | 418 | 567 | 14,2 | 33 | 403 |
| 0,10 | 386 | 9,7 | 98 | 403 | 499 | 12,5 | 98 | 401 | 626 | 15,7 | 93 | 453 |
| 0,11 | 388 | 9,7 | 98 | 401 | 499 | 12,5 | 98 | 401 | 710 | 17,6 | 95 | 403 |

### 2.5. Предварительная фильтрация «на лету»

Описания предлагаемых методов предварительной фильтрации, приведённые выше, выполнены в терминах множеств и подмножеств, однако эти методы очевидным образом могут быть применены и к отдельным изображениям. Для этого вместо формирования подмножества $\mathcal{Z}^{\text{good}}$ можно тестировать каждое очередное изображение, что даст возможность проводить предварительную фильтрацию «на лету».

## 3. Возможные применения предварительной фильтрации и пути дальнейших исследований

Предварительная фильтрация может быть использована для повышения практической значимости слабых методов стегоанализа, которые дают ошибку обнаружения, близкую к 0,5 (вероятности случайного угадывания). Предварительная фильтрация может позволить отобрать некоторое подмножество «хороших» изображений, на котором ошибка обнаружения будет меньше. Например, подобные результаты получены в экспериментах с HUGO 0,05 б/п, когда ошибка по всему контрольному множеству равна 0,37 (см. табл. 1), что вряд ли может считаться достоверным обнаружением, а предварительная фильтрация позволила отобрать подмножество (хотя и не слишком большое) со значительно меньшей ошибкой (см. табл. 5).

Ещё одним потенциальным применением предварительной фильтрации может стать выбор наиболее достоверного метода обнаружения для заданного изображения или множества изображений. Например, если в распоряжении стегоаналитика имеется несколько методов обнаружения, то он может разбить контрольное множество на несколько подмножеств, каждое из которых обеспечивает низкую ошибку для определённого метода, и проводить стегоанализ этих подмножеств соответствующим методом. Итоговая ошибка обнаружения у такой схемы может быть ниже, чем у различных методов в отдельности.

На эффективность обнаружения информации влияют как минимум два фактора: метод обнаружения и специфические свойства изображения, поэтому, если один метод работает лучше на одном множестве изображений, то это не гарантирует того, что он будет работать лучше и на другом множестве. Таким образом, сравнивая несколько различных методов обнаружения скрытой информации, разумно проверять их на нескольких множествах с различными свойствами. Предварительная фильтрация может стать подходом, который позволит выделять такие множества, причём делить можно не только на «хорошие» и «плохие», но и более тонко, скажем, на «очень хорошие», «хорошие», «плохие» и «очень плохие». Как показано в экспериментах, с помощью выбора параметров можно задать требуемый размер этих множеств.

Если после предварительной фильтрации доля «хороших» изображений достаточно велика, то можно составить схему, которая будет иметь более высокую производительность, чем некий высокоточный, но медленный метод. Например, если один метод обнаружения работает медленно (как метод опорных векторов), но является эффективным, а другой метод работает быстрее (как ансамблевый классификатор), то с помощью быстрого метода можно осуществить предварительную фильтрацию, а затем медленный метод будет обрабатывать только «хорошие» изображения.

Вычисление ошибки обнаружения по подмножеству аналогично разработке атак на криптосистемы в предположении использования слабых ключей [11, 12]. При этом размер множества слабых ключей является дополнительной характеристикой атаки: чем больше множество, тем эффективнее атака. Таким же образом размер подмножества «хороших» изображений может считаться дополнительной характеристикой метода обнаружения.

## Заключение

В работе предложен новый подход к стегоанализу, подразумевающий предварительную фильтрацию изображений, подлежащих проверке на наличие в них скрытой информации. Данный подход заключается в том, что перед финальным этапом стегоанализа производится отбор изображений с целью выделить подмножество тех из них, которые обеспечат как можно меньшую ошибку обнаружения (во всяком случае, меньше ошибки, вычисленной по всему контрольному множеству). Предварительная фильтрация может быть реализована различными способами; здесь предложены три возможности: наивный метод, метод простой классификации и метод комбинированной классификации (комбинация первых двух).

Согласно проведённым экспериментам, предварительная фильтрация довольно чувствительна к выбору параметров (порогов), поэтому результаты представлены в виде таблиц, отражающих зависимости ошибки обнаружения от этих параметров. По нашему мнению, наиболее интересными результатами являются следующие. Эксперименты показали, что предварительная фильтрация позволяет выбрать порядка 35 % изображений из BOSSbase v1.01, для которых метод адаптивной стеганографии HUGO

0,4 б/п обнаруживается с ошибкой менее 0,003, в то время как ошибка, вычисленная по всему множеству, составляет 0,141. Показано также, что из всего множества можно выбрать порядка 5 % изображений, для которых HUGO 0,1 б/п определяется с ошибкой менее чем 0,05, тогда как ошибка по всему множеству составляет 0,37.

В работе описаны эти и другие потенциальные применения предварительной фильтрации, такие, как расширенное определение точности методов обнаружения скрытой информации, возможность выбирать размер «хорошего» подмножества за счёт настройки параметров, возможность деления множества изображений на подмножества с разными свойствами.

## ЛИТЕРАТУРА


1. *Fridrich J.* Rich models for steganalysis of digital images // IEEE Trans. Information Forensics and Security. 2012. V. 7. No. 3. P. 868–882.
2. *Fridrich J., Kodovsky J., Holub V., and Goljan M.* Steganalysis of content-adaptive steganography in spatial domain // Proc. 13th Information Hiding Workshop. LNCS. 2011. V. 6958. P. 102–117.
3. *Ker A., Bas P., Bohme R., et al.* Moving steganography and steganalysis from the laboratory into the real world // Proc. 1st ACM Workshop on Information Hiding and Multimedia Security. N. Y., USA, 2013. P. 45–58.
4. *Kodovsky J., Fridrich J., and Holub V.* Ensemble classifiers for steganalysis of digital media // IEEE Trans. Information Forensics and Security. 2011. V. 7. No. 2. P. 434–444.
5. *Monarev V. and Pestunov A.* A known-key scenario for steganalysis and a highly accurate detector within it // Proc. 10th IEEE Intern. Conf. Intelligent Information Hiding and Multimedia Signal Processing. Kitakyushu, 2014. P. 175–178.
6. `www.agents.cz/boss` — Break our steganographic system. 2015.
7. *Bas P., Filler T., and Pevny T.* Break our steganographic system — the ins and outs of organizing BOSS // Proc. 13th Information Hiding Workshop. LNCS. 2011. V. 6958. P. 59–70.
8. `bows2.ec-lille.fr` — Break our watermarking system, second edition. 2015.
9. `photogallery.nrcs.usda.gov/res/sites/photogallery` — NRCS photo gallery. 2015.
10. *Pevny T., Bas P., and Fridrich J.* Steganalysis by subtractive pixel adjacency matrix // IEEE Trans. Information Forensics and Security. 2010. V. 5. No. 2. P. 215–224.
11. *Biryukov A., Nakahara J., Preneel B., and Vanderwalle J.* New weak-key classes of IDEA // LNCS. 2002. V. 2513. P. 315–326.
12. *Kara O. and Manap C.* A new class of weak keys for Blowfish // FSE'2007. LNCS. 2007. V. 4593. P. 167–180.
13. *Pevny T.* Detecting messages of unknown length // Proc. 8th Intern. Conf. Media Watermarking, Security and Forensics. 2011. P. 1–12.
14. *Monarev V. and Pestunov A.* A new compression-based method for estimating LSB replacement rate in color and grayscale images // Proc. 7th IEEE Intern. Conf. Intelligent Information Hiding and Multimedia Signal Processing. Dalian, 2011. P. 57–60.
15. *Fridrich J., Kodovsky J., Holub V., and Goljan M.* Breaking HUGO — the process discovery // Proc. 13th Information Hiding Workshop. LNCS. 2011. V. 6958. P. 102–117.
16. *Holub V. and Fridrich J.* Random projections of residuals for digital image steganalysis // IEEE Trans. Information Forensics and Security. 2013. V. 8. No. 12. P. 1996–2006.
17. `dde.binghamton.edu/download/feature_extractors` — Feature extractors for steganalysis. 2015.
18. *Duda R., Hart P., and Stork D.* Pattern Classification. 2nd ed. N. Y.: John Wiley & Sons Inc., 2001.

19. *Pevny T., Filler T., and Bas P.* Using high-dimensional image models to perform highly undetectable steganography // Proc. 12th Information Hiding Workshop. LNCS. 2010. V. 6387. P. 161–177.
20. *Holub V. and Fridrich J.* Designing steganographic distortion using directional filters // Proc. 4th IEEE Intern. Workshop on Information Forensics and Security. Tenerife, 2012. P. 234–239.
21. *Holub V. and Fridrich J.* Digital image steganography using universal distortion // Proc. 1th ACM Workshop on Information Hiding and Multimedia Security. N. Y., USA, 2013. P. 59–68.

## REFERENCES

1. *Fridrich J.* Rich models for steganalysis of digital images. IEEE Trans. Information Forensics and Security, 2012, vol. 7, no. 3, pp. 868–882.
2. *Fridrich J., Kodovsky J., Holub V., and Goljan M.* Steganalysis of content-adaptive steganography in spatial domain. Proc. 13th Information Hiding Workshop, LNCS, 2011, vol. 6958, pp. 102–117.
3. *Ker A., Bas P., Bohme R., et al.* Moving steganography and steganalysis from the laboratory into the real world. Proc. 1st ACM Workshop on Information Hiding and Multimedia Security, N. Y., USA, 2013, pp. 45–58.
4. *Kodovsky J., Fridrich J., and Holub V.* Ensemble classifiers for steganalysis of digital media. IEEE Trans. Information Forensics and Security, 2011, vol. 7, no. 2, pp. 434–444.
5. *Monarev V. and Pestunov A.* A known-key scenario for steganalysis and a highly accurate detector within it. Proc. 10th IEEE Intern. Conf. Intelligent Information Hiding and Multimedia Signal Processing, Kitakyushu, 2014, pp. 175–178.
6. `www.agents.cz/boss` — Break our steganographic system. 2015.
7. *Bas P., Filler T., and Pevny T.* Break our steganographic system — the ins and outs of organizing BOSS. Proc. 13th Information Hiding Workshop, LNCS, 2011, vol. 6958, pp. 59–70.
8. `bows2.ec-lille.fr` — Break our watermarking system, second edition. 2015.
9. `photogallery.nrcs.usda.gov/res/sites/photogallery` — NRCS photo gallery. 2015.
10. *Pevny T., Bas P., and Fridrich J.* Steganalysis by subtractive pixel adjacency matrix. IEEE Trans. Information Forensics and Security, 2010, vol. 5, no. 2, pp. 215–224.
11. *Biryukov A., Nakahara J., Preneel B., and Vanderwalle J.* New weak-key classes of IDEA. LNCS, 2002, vol. 2513, pp. 315–326.
12. *Kara O. and Manap C.* A new class of weak keys for Blowfish FSE'2007. LNCS. 2007, vol. 4593, pp. 167–180.
13. *Pevny T.* Detecting messages of unknown length. Proc. 8th Intern. Conf. Media Watermarking, Security and Forensics, 2011, pp. 1–12.
14. *Monarev V. and Pestunov A.* A new compression-based method for estimating LSB replacement rate in color and grayscale images. Proc. 7th IEEE Intern. Conf. Intelligent Information Hiding and Multimedia Signal Processing, Dalian, 2011, pp. 57–60.
15. *Fridrich J., Kodovsky J., Holub V., and Goljan M.* Breaking HUGO — the process discovery. Proc. 13th Information Hiding Workshop, LNCS, 2011, vol. 6958, pp. 102–117.
16. *Holub V. and Fridrich J.* Random projections of residuals for digital image steganalysis. IEEE Trans. Information Forensics and Security, 2013, vol. 8, no. 12, pp. 1996–2006.
17. `dde.binghamton.edu/download/feature_extractors` — Feature extractors for steganalysis. 2015.
18. *Duda R., Hart P., and Stork D.* Pattern Classification. 2nd ed. N. Y., John Wiley & Sons Inc., 2001.

19. *Pevny T., Filler T., and Bas P.* Using high-dimensional image models to perform highly undetectable steganography. Proc. 12th Information Hiding Workshop, LNCS, 2010, vol. 6387, pp. 161–177.
20. *Holub V. and Fridrich J.* Designing steganographic distortion using directional filters. Proc. 4th IEEE Intern. Workshop on Information Forensics and Security, Tenerife, 2012, pp. 234–239.
21. *Holub V. and Fridrich J.* Digital image steganography using universal distortion. Proc. 1th ACM Workshop on Information Hiding and Multimedia Security, N. Y., USA, 2013, pp. 59–68.