ML for Security. Лабораторная работа 2. Методы стегоанализа

Суть задачи лабораторной работы

Провести стегоанализ некоторой стеганографической системы. Для этого реализовать саму стегосистему, метод стегоанализа и провести исследование качества работы метода при различной заполненности контейнера. Следует подчеркнуть, что «метод стегоанализа» в базовых вариантах определяет общий принцип, но не конкретный вид вектора признаков и конкретную модель классификатора. И то и другое нужно выбрать самостоятельно, пояснив собственное решение при сдаче работы.

Варианты задания

Вариант определяется парой *{стеганографическая система; метод стегоанализа}*. В каждой категории предлагаются базовые (более простые варианты) и усложнённые (исследовательские; некоторые из них – с непредсказуемым результатом). Усложнённые выделены курсивом. Преподаватель не принуждает выбирать усложнённый вариант.

Лабораторную работу можно выполнять по одному или вдвоём.

На потоке не должно быть одинаковых сочетаний *{стеганографическая система; метод стегоанализа}*, реализуемых более чем двумя коалициями студентов. Для выполнения этого требования при выборе варианта нужно вписать его в общую таблицу.

Варианты атакуемых стеганографических систем (8 базовых и 3 усложнённых):

- НЗБ-встраивание: {последовательная запись; запись по псевдослучайным координатам}, {1-я битовая плоскость, 2-я битовая плоскость}
- ±1-встраивание: {последовательная запись; запись по псевдослучайным координатам}, {1-я битовая плоскость, 2-я битовая плоскость}
- *JSteg*
- *F5*
- *HUGO*

Варианты используемых методов стегоанализа (3 базовых и 4 усложнённых):

- Методы частоты переходов
- Метод длин серий
- Метод пар значений
- *Метод Sample Pairs*
- *Метод SPAM*
- *Метод AUMP*
- *Метод SRNet*
- *Собственный набор признаков*

Параметры, в зависимости от которых нужно исследовать качество стегоанализа:

- Заполненность контейнера q – число бит, делённое на число пикселей изображения. Минимальный набор вариантов – от 20% до 100% с шагом 20%. Можно взять шаг 10%.
- Конкретная версия вектора признаков – определяется студентом самостоятельно на основе выбранного варианта. Приветствуется сравнительное исследование более одной версии.

Неизменяемые параметры методов стеганографии и стегоанализа:

- Развёртка двумерной области в одномерный вектор: по умолчанию – серпантинная. Кому сложно, можно ограничиться построчной.
- $QF = 50$ для методов на основе JPEG.

Порядок выполнения лабораторной работы

Входными данными, необходимыми для выполнения лабораторной работы, являются K полутоновых изображений одного размера.

1. Реализовать процедуру расчёта векторов признаков, используемых для стегоанализа.
2. Выполнить имитацию работы стегосистемы для первых $K/2$ изображений: встроить в каждое изображение в качестве стеганографической информации отдельную реализацию равномерного белого шума (число бит определяется текущим значением q). Вторую половину изображений не менять.
3. Произвести обучение классификатора (модель классификатора выбрать самостоятельно) по выборке, содержащей первые 70 % изображений каждого из двух типов (со встраиванием и без. То есть общий объём обучающей выборки составляет $K \cdot 0{,}7$.
4. Применить обученный классификатор на оставшихся 30 % изображений и оценить качество классификации по мере Accuracy. Вывести результат в виде графиков зависимости Accuracy от q.
5. Повторить пп. 2-4 для других значений q и (при желании) для других векторов признаков.

В работе можно использовать любой из перечисленных наборов изображений:

- BOSS (10000 изображений, формат PGM): http://agents.fel.cvut.cz/stegodata/BossBase-1.01-cover.tar.bz2
- BOWS-2 (10000 изображений, формат PGM): http://bows2.ec-lille.fr/BOWS2OrigEp3.tgz
- BOWS-2 (выборка из 1000 изображений, формат TIFF): https://mega.nz/file/q0NBiTJD#81n-vQ6nBeDTUNWc7S_MhATunDZwYPyzLxFcdIr7z5M

Теоретические основы лабораторной работы

НЗБ-встраивание

Встраивание информации в наименее значимые биты контейнера (или сокращённо НЗБ-встраивание) – исторически один из первых и, пожалуй, наиболее известный широкой публике подход, который может применяться как для стеганографии, так и для защиты сигналов цифровыми водяными знаками. Он очень прост и позволяет встроить достаточно большое количество информации без сколько-нибудь заметных искажений контейнера, однако методы, использующие данный подход, как правило, обладают низкой стойкостью к искажениям носителя информации и относительно легко могут быть подвергнуты стегоанализу, поэтому имеют весьма ограниченную применимость. Тем не менее, НЗБ-встраивание вполне подходит для задач, в которых отсутствуют жёсткие требования по стойкости к отдельным видам атак.

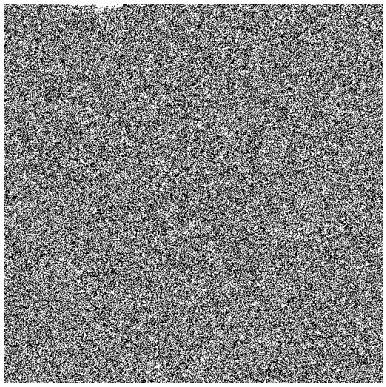
Основная идея метода заключается в том, что любое полутоновое изображение может быть представлено в виде совокупности битовых плоскостей. Так, контейнер $C(n_1, n_2)$ *будет иметь вид*:

$$C(n_1, n_2) = C_1(n_1, n_2) + C_2(n_1, n_2) \cdot 2 + \ldots + C_K(n_1, n_2) \cdot 2^{K-1}, \qquad (1)$$

где $C_k(n_1, n_2) \in [0,1]$ – битовые плоскости, k – номер битовой плоскости, $K = 8$ – их количество.

Наименее и наиболее значащими битовыми плоскостями являются соответственно C_1 и C_8: если изменить значение бита $C_1(n_1, n_2)$, то яркость изменится на единицу; если же изменить значение бита $C_8(n_1, n_2)$, то яркость изменится на 128. Различие между младшими и старшими битовыми плоскостями хорошо заметно на рис. 1. Младшие битовые плоскости выглядят как слабокоррелированный шум. Осмысленные детали, как правило, начинают проступать лишь с

четвёртой битовой плоскости. Это означает, что наименее значимые битовые плоскости можно модифицировать с целью встраивания скрытого сообщения или цифрового водяного знака.

а)   б)

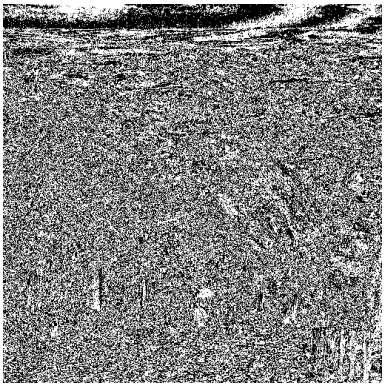
в)   г)

Рис. 1 – Битовые плоскости полутонового изображения: а) исходное изображение; б) 1-я битовая плоскость; в) 4-я битовая плоскость; г) 8-я битовая плоскость

Далее будем рассматривать лишь случай встраивания информации в одну p-ю битовую плоскость. Тогда носитель информации будет иметь вид:

$$C^W(n_1, n_2) = C_1^W(n_1, n_2) + \ldots + C_K^W(n_1, n_2) \cdot 2^{K-1}, \qquad (2)$$

где $C_k^W(n_1, n_2) = C_k(n_1, n_2)$ для всех $k \neq p$.

Существует достаточно большое количество систем НЗБ-встраивания, которые отличаются способом формирования битовой плоскости $C_p^W(n_1, n_2)$. Ниже мы рассмотрим две таких системы: стеганографическое НЗБ-встраивание (LSB replacement) и ±1-встраивание в полутоновые изображения (LSB Matching).

Стеганографическое НЗБ-встраивание

При стеганографическом встраивании внутри контейнера $\mathrm{C}(\mathrm{n}_1, \mathrm{n}_2)$ передаётся бинарный вектор $\mathbf{b}$ длины $\mathrm{N_b} \leq \mathrm{N_1 N_2}$. Как правило, встраивание происходит путём замены бит. В простейшем случае информация заносится в НЗБ последовательно:

$$C_p^W(n_1, n_2) = b_{n_1 \cdot N_2 + n_2}, \qquad (3)$$

где $\mathrm{b_i}$ – i-й элемент вектора $\mathbf{b}$. Однако такое встраивание легко поддаётся стегоанализу, то есть легко обнаруживается на основе анализа статистических характеристик наименее значимой битовой плоскости (НЗБП), использованной для встраивания информации.

Для противодействия простейшим методам стегоанализа прибегают к следующим мерам:

1) заполняют по возможности небольшую часть НЗБ контейнера, т.е. добиваются того, чтобы величина

$$q = \frac{N_b}{N_1 N_2}, \qquad (4)$$

называемая заполненностью контейнера, была существенно меньше 1;

2) заполнение контейнера производят в псевдослучайном порядке, который полностью определяется ключом встраивания $\mathbf{k}$.

Ко второму пункту следует добавить, что ключ сам по себе не содержит последовательности координат пикселей, но однозначно определяет её. Например, ключ может представлять собой начальное значение генератора случайных чисел.

Процедура извлечения информации очевидна и представляет собой чтение битов из заданных ключом координат.

При встраивании информации в *p*-ю битовую плоскость яркость отдельно взятого пикселя либо не меняется, либо меняется ровно на p, причём известно, в какую сторону. Пусть для определённости $p = 2$ и стоит задача встроить в пиксель с яркостью 21 значение 1. Число 21 в двоичной записи имеет вид 10101, то есть во второй битовой плоскости стоит 0. Таким образом, встраивая туда 1, мы прибавляем к текущему значению p и в итоге получаем 23. Однако что произойдёт, если мы не прибавим p, а вычтем? Очевидно, что в этом случае разница между искомым и полученным значением составит $2p$, то есть изменения произойдут в более старших разрядах двоичной записи, в то время как *p*-й бит не претерпит изменений. Действительно, в нашем примере получится число 19, то есть 10011 в двоичной записи. Таким образом, мы имеем два способа изменения значения яркости пикселя, приводящих к идентичным изменениям в нужной битовой плоскости и сопровождаемых равными по абсолютной величине искажениями. Это свойство позволяет несколько модифицировать процедуру стеганографического НЗБ-встраивания путём внесения дополнительной неопределённости, способствующей защите от атак, направленных на обнаружение канала скрытой передачи информации.

±1-встраивание

Рассуждения выше приводились для общего случая *p*-й битовой плоскости. Однако на практике чаще всего ограничиваются рассмотрением случая $p = 1$. Более того, само название данной модификации НЗБ-метода, укоренившееся в научной литературе – ±1-встраивание – уже косвенно говорит о номере битовой плоскости (в общем случае следовало бы говорить о $\pm$-встраивании). Мы приведём формулу встраивания для этого частного случая, однако её обобщение не составит труда.

Итак, пусть $\mathrm{b_i}$ – i-й элемент вектора $\mathbf{b}$,

$$(n_1, n_2) = \big(n_1(\boldsymbol{k}, i), n_2(\boldsymbol{k}, i)\big)$$

– координаты i-го пикселя, в который необходимо встроить бит $\mathrm{b_i}$, а ξ_i – псевдослучайное число, с равной вероятностью принимающее положительные и отрицательные значения (генерация последовательности $\{\xi_\mathrm{i}\}$ также происходит на основе ключа). Тогда встраивание информации будет осуществляться по формуле

$$C^W(n_1, n_2) = \begin{cases} C(n_1, n_2), & C_1(n_1, n_2) = b_i, \\ C(n_1, n_2) + sign(\xi_i), & C_1(n_1, n_2) \neq b_i, \end{cases} \qquad (5)$$

где

$$sign(\xi_i) = \begin{cases} 1, & \xi_i \geq 0, \\ -1, & \xi_i < 0, \end{cases}$$

то есть случайным образом прибавляется или вычитается единица в том случае, если значение бита не совпадает с требуемым. С учётом того, что значения $C^W(n_1, n_2)$ должны принадлежать отрезку от 0 до 255, формула (5) примет вид:

$$C^W(n_1, n_2) = \begin{cases} C(n_1, n_2), & C_1(n_1, n_2) = b_i, \\ C(n_1, n_2) + 1, & C_1(n_1, n_2) \neq b_i \wedge C(n_1, n_2) = 0, \\ C(n_1, n_2) - 1, & C_1(n_1, n_2) \neq b_i \wedge C(n_1, n_2) = 255, \\ C(n_1, n_2) + sign(\xi_i), & \text{иначе.} \end{cases} \qquad (6)$$

Извлечение информации происходит так же, как и в случае стеганографического НЗБ-встраивания.

Методы стегоанализа

Задача и направления стегоанализа

Под *стегоанализом* обычно понимается атака на стеганографические системы, целью которой является обнаружение канала скрытой передачи информации. Также обычно выделяют *целенаправленный стегоанализ* (target steganalysis), при проведении которого считаются известными используемые стеганографические методы и протоколы, и *слепой стегоанализ*, не ориентированный на какие-либо методы.

Поскольку результатом проведения стегоанализа для какого-либо цифрового носителя информации является бинарный ответ: есть встраивание или нет, – то в сущности задача стегоанализа может быть сведена к задаче классификации объекта на два соответствующих класса. В этом случае её решение будет включать два этапа:

1) выбор информативных признаков;
2) классификация векторов признаков с обучением.

На втором этапе может использоваться любой известный классификатор. Выбор конкретного решения может зависеть от характера векторов признаков, их длины, разделимости, количества имеющихся для обучения данных и прочих факторов.

Рассмотрим некоторые популярные методы выбора признаков для решения задачи стегоанализа методов стеганографического встраивания информации в наименее значимые биты полутоновых изображений, а именно собственно НЗБ-встраивания и ±1-встраивания. Все они используют следующее предположение: стеганографическое встраивание разрушает корреляционные связи между соседними отсчётами цифрового сигнала – контейнера. Таким образом, эти признаки должны отражать коррелированность сигнала в пространстве сокрытия.

Метод пар значений

Одним из самых простых и наиболее известных методов целенаправленного стегоанализа НЗБ-встраивания является метод гистограмм пар значений.

Данный метод стегоанализа использует расчёт статистики хи-квадрат для проверки гипотезы о виде распределения яркости контейнера. Пусть для простоты проверяется наличие встраивания информации в первую битовую плоскость. Тогда теоретически значения яркости, отличающиеся только младшим битом (0 и 1, 2 и 3, 4 и 5…), должны быть равновероятны. Таким образом, метод заключается в расчёте эмпирической гистограммы анализируемого изображения $h_i^e, i = 0..255$, а также соответствующей ей теоретической:

$$h_i^t = \frac{h_{2\cdot\lfloor i/2 \rfloor}^e + h_{2\cdot\lfloor i/2 \rfloor+1}^e}{2}.$$

На рис. 2 показан пример эмпирической и теоретической гистограмм. Соответствие эмпирической гистограммы теоретической в классическом методе проверяется посредством расчёта статистики хи-квадрат для чётных отсчётов гистограммы. Однако в настоящей лабораторной работе вместо использования данного метода в формате проверки статистической гипотезы предлагается его трансформировать в формат обучения с учителем. Для этого необходимо сформировать вектор признаков, учитывающий отмеченные на рис. 2 особенности, и обучить классификатор на примерах данных.

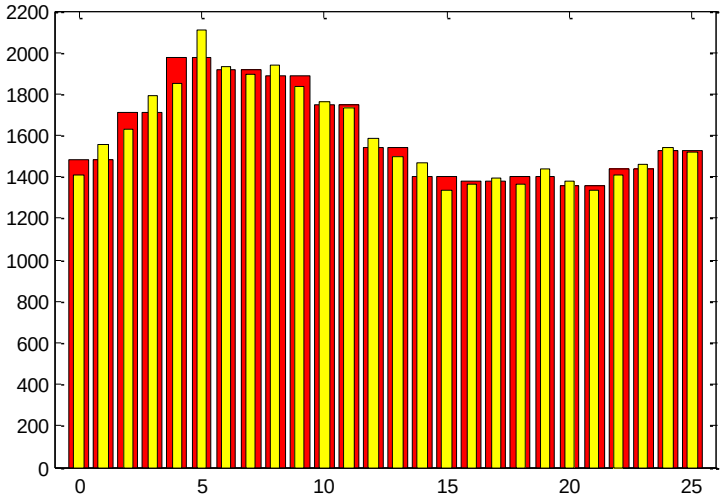


Рис. 2 – Пример эмпирической гистограммы изображения (жёлтый цвет) и соответствующей ей теоретической гистограммы (красный цвет)

Признаки на основе развёртки двумерной области

Существует группа методов, использующих развёртку двумерной битовой плоскости в одномерную последовательность нулей и единиц с последующим расчётом некоторых её статистических характеристик. При одномерной развёртке близкие на плоскости пиксели должны располагаться как можно ближе в результирующей последовательности. Наиболее часто используются две развёртки: построчная и серпантинная. Они проиллюстрированы на рис. 3. Очевидно, что серпантинная развёртка имеет преимущество по сравнению с построчной, поскольку не содержит разрывов.

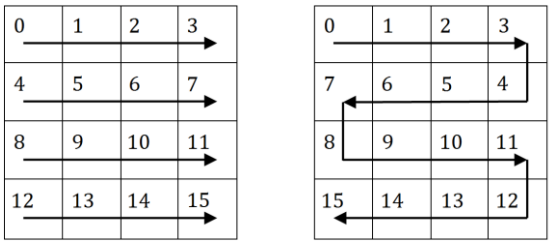


Рис. 3 – Развёртки двумерной области 4×4: построчная (слева), серпантинная (справа)

Обозначим полученную последовательность $\{\beta_k\}_{k=0..N-1}$, где $N = N_1 N_2$.

Первый вариант её дальнейшего использования заключается в расчёте относительной частоты переходов между соседними отсчётами последовательности:

$$\pi_{00} = \frac{1}{N-1}\sum_{k=0}^{N-2}\gamma_k^{00}, \tag{7}$$

где

$$\gamma_k^{00} = \begin{cases} 1, & (\beta_k = 0) \wedge (\beta_{k+1} = 0), \\ 0, & \text{иначе.} \end{cases} \tag{8}$$

По аналогичным формулам находятся также π_{01}, π_{10}, π_{11} , в совокупности образуя вектор из четырёх признаков:

$$(\pi_{00}, \pi_{01}, \pi_{10}, \pi_{11}). \tag{9}$$

В случае заполненного контейнера частоты переходов должны быть достаточно близкими, в то время как в пустом контейнере частоты переходов из 0 в 0 и из 1 в 1 значительно превышают частоты переходов двух других видов. На рис. 4 показан пример диаграммы частот переходов для разных битовых плоскостей пустого контейнера в сравнении с заполненной битовой плоскостью, рассчитанных по последовательности, полученной при помощи построчной развёртки. Статистика пустого контейнера считалась по классическому изображению “Lenna”.

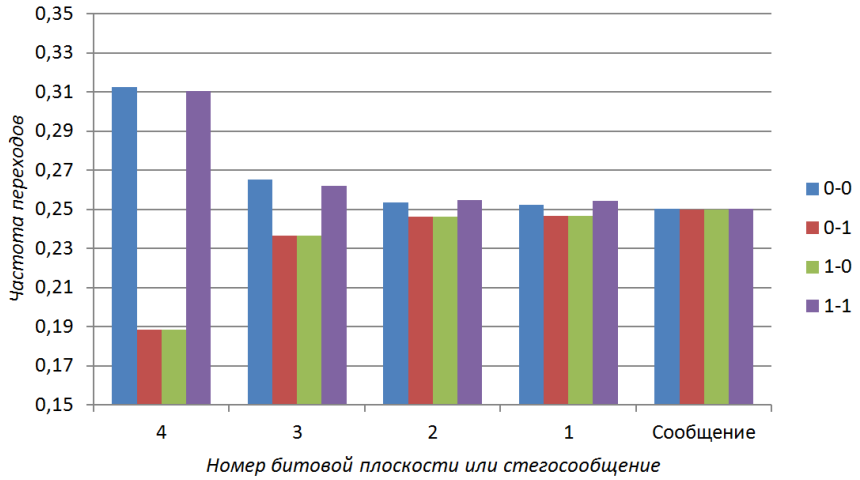


Рис. 4 – Диаграмма частоты переходов для пустого и заполненного контейнера

Другой способ формирования признаков по двоичной последовательности – расчёт числа серий разной длины. Серией является фрагмент последовательности, состоящий из одинаковых значений (неважно, единиц или нулей) и ограниченный другими значениями или границей последовательности. Достаточно информативной характеристикой последовательности является статистика, отражающая число серий различной длины. Будем обозначать её $\{s_i\}_i$, где $i > 0$ – длина серии. Иногда эту статистику нормируют на длину последовательности N, чтобы получить значения, не зависящие от объёма контейнера:

$$\{v_i\}_i = \left\{\frac{s_i}{N}\right\}_i. \qquad (10)$$

В табл. 1 приведён пример статистики числа серий последовательностей, полученных при помощи построчной развертки первой битовой плоскости пустого и заполненного контейнера. Статистика пустого контейнера считалась по изображению “Lenna”.

Табл. 1 – Пример статистики числа серий в пустом и заполненном контейнере

i	Число серий s_i		i	Число серий s_i	
	Пустой контейнер	Заполненный контейнер		Пустой контейнер	Заполненный контейнер
1	64097	65363	12	48	37
2	32462	32741	13	14	14
3	16143	16596	14	7	11
4	8124	8131	15	7	1
5	4155	4093	16	6	3
6	2075	2054	17	3	1
7	1076	964	18	1	0
8	584	539	19	1	0
9	320	245	20	1	0
10	169	138	21	0	0
11	94	63	22	0	0

Как видно из таблицы, число серий малой длины в заполненном контейнере превышает число серий в пустом контейнере, но начиная с некоторого значения i статистика по пустому контейнеру становится выше. Если рассматривать очень большие серии – длиной в несколько десятков отсчётов, то в заполненном контейнере таковые почти всегда отсутствуют, в то время как в пустом время от времени могут появляться.

Наиболее простой способ формирования вектора признаков по статистике числа серий – выбор в качестве признаков некоторого количества

$$\{s_i\}_{i \in I}.$$

При этом множество I также может формироваться различными способами. Некоторым недостатком такого подхода является существенное различие в абсолютных значениях s_i для разных длин i. Эта проблема может решаться нормировки векторов признаков.

Рассмотренные признаки просты для изучения, но не слишком хороши для используемых на практике стеганографических методов. Даже простые модификации системы НЗБ-встраивания позволяют обеспечить стойкость некоторым из рассмотренных признаков. Поэтому в научных работах, вышедших позднее 2005 г., был предложен ряд более эффективных признаков.

Особенности реализации усложнённых вариантов

Методы стеганографического встраивания для JPEG-изображений

При реализации данной группы методов необходимо осуществить не полное JPEG-сжатие, а его имитацию: применить блочное ДКП, поделить полученные значения на элементы матрицы, определяющей шаги квантования для ДКП-коэффициентов, округлить полученные значения. После встраивания ЦВЗ нужно осуществить деквантование, расчёт обратного ДКП и сохранить полученное изображение в файл на диск в 8-битном формате.

При выборе стеганографической системы для JPEG-коэффициентов признаки, используемые для стегоанализа тоже должны рассчитываться для квантованных ДКП-коэффициентов.

JSteg

JSteg представляет собой НЗБ-встраивание для квантованных ДКП-коэффициентов, сохраняемых при JPEG-сжатии. Особенность метода - пара коэффициентов {0, 1} не используется для встраивания информации во избежание существенных искажений исходного контейнера.

F5

В рамках лабораторной работы предлагается реализовать упрощённую версию метода F5.

При встраивании информации пропускаем все ДКП-коэффициенты, имеющие значение 0, а также исключаем из рассмотрения DC-отсчёты каждого блока (коэффициент с координатами (0,0)). Встраивание обсуществляется по сути методом НЗБ, однако в данной системе используется альтернативный способ расчёта наименее значимых бит:

$$LSB_{F5}(x) = \begin{cases} 1 - x \ (mod\ 2), & x < 0, \\ x \ (mod\ 2), & x \geq 0. \end{cases}$$

Метод HUGO

Данный метод не нужно реализовывать. Вместо этого нужно скачать уже готовые результаты встраивания:

- Контейнеры: http://agents.fel.cvut.cz/stegodata/BossBase-1.01-cover.tar.bz2
- Изображения со стегосообщениями: http://agents.fel.cvut.cz/stegodata/BossBase-1.01-hugo-alpha=0.4.tar.bz2

Изображения в приведённых архивах представлены в формате PGM.

Методы стегоанализа

Метод SPAM: http://dde.binghamton.edu/download/spam/ (C++)

Методы Sample Pairs, AUMP: http://dde.binghamton.edu/download/structural_lsb_detectors/ (MATLAB)

Метод SRNet: http://dde.binghamton.edu/download/feature_extractors/download/SRNet.zip (Python)