*А. В. Сорокин*

# О ВОЗМОЖНОМ РАЗВИТИИ МЕТОДА ИЗВЛЕЧЕНИЯ ДАННЫХ ДЛЯ ВОССТАНОВЛЕНИЯ ФАЙЛОВ ФОРМАТА JPEG

## Введение

Метод восстановления файлов из двоичной последовательности данных цифрового носителя или образа носителя путем извлечения или «вырезания» (carving) фрагментов по некоторым признакам подробно описан, например, в книге Брайана Кэрриэ [1]. Данный метод является простым в реализации и позволяет быстро получить удовлетворительный результат в ситуациях, когда файлы на носителе размещены последовательно, без фрагментации. Если же требуется восстановить фрагментированный файл, то ситуация с применимостью данного метода меняется. Так, Г. Сенкевич в своей книге [2] прямо утверждает, что задачи восстановления фрагментированных и нефрагментированных файлов имеют совершенно различные прогнозы успешного решения.

В силу высокой популярности формата графических файлов JPEG и его внутренних особенностей для решения задачи восстановления файлов в сложных случаях представляется оправданной разработка отдельного решения, применимого именно для этого формата. Подтверждением этого может служить ряд работ зарубежных авторов, посвященных изучению конкретных проблемных случаев, например: [3], [4] и др.

## Метод извлечения данных, его недостатки

Метод восстановления файлов, называемый извлечением данных, состоит в поиске пар начальных и конечных сигнатур для известных типов файлов в последовательности данных носителя и представлении последовательности данных, заключенных между этими сигнатурами в качестве восстановленного файла. Иллюстрация работы данного метода приведена на рис. 1.

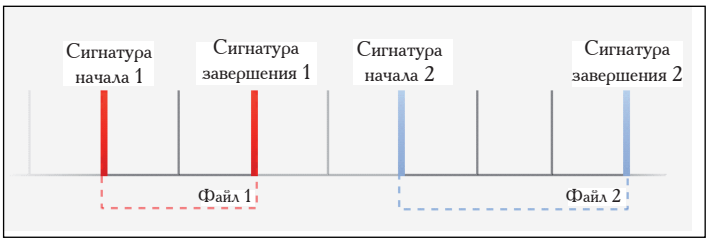


*Рис. 1. Извлечение файлов из последовательности кластеров носителя по сигнатурам начала и завершения*

Из самой сути данного метода нетрудно понять, что успех его работы основывается на том, что восстанавливаемый файл размещен на диске в последовательно адресованные кластеры. Если же данное условие будет нарушаться, добиться корректного результата не удастся. Подробнее трудности применения данного метода и возможные пути его дальнейшей эволюции рассмотрены в работе [5].

## Авторский подход

Для доработки неудовлетворительных результатов работы метода извлечения данных, а также для ускорения работы реализаций алгоритма «умного извлечения» (Smart Carving) — дальнейшего развития метода извлечения данных, предложенного в работах [6] и [7], — был предложен метод определения принадлежности произвольного кластера цифрового носителя данных файлу формата JPEG, использующего кодировочные таблицы Хаффмана, соответствующие описанным в стандарте формата [8]. Данный метод и оценки его ошибок первого и второго рода приведены в работе [9].

**Предварительная классификация кластеров**

Наиболее общим алгоритмом восстановления фрагментированных файлов произвольного формата, является перебор всех возможных последовательностей кластеров носителя до обнаружения последовательности, составляющей искомый файл. При этом предварительный отбор кластеров, относящихся к файлам того же типа, что и искомый, снижает время получения требуемого результата. Для форматов файлов, не использующих каких-либо преобразований файлов (таких как сжатие или шифрование), такой отбор может быть реализован на основе их содержимого, а для форматов, использующих сжатие, в частности формата JPEG, такой отбор оказывается затруднен. В силу того что файл данного формата содержит большинство значений яркости и цвета пикселей в форме разницы с другими значениями, а также переменной длины кодов, указывающих на порядок интерпретации некодированных значений, возможность визуализации отдельного кластера файла формата JPEG представляется сомнительной. В названных выше работах [5] и [6] предлагается алгоритм восстановления, заключающийся в автоматическом поиске точки фрагментации файла, считываемого последовательно от сигнатуры начала. После обнаружения начала фрагмента полным перебором среди всех оставшихся кластеров можно найти продолжение файла и продолжить извлечение файла. При обнаружении сигнатуры окончания файла процедура будет завершена, при обнаружении следующей точки фрагментации будет вновь запущен поиск продолжения восстанавливаемого файла. Для ускорения работы данного алгоритма можно использовать авторский метод для отделения множества кластеров, предположительно относящихся к файлам формата JPEG, с тем чтобы при необходимости поиска продолжения файла данного формата рассматривать их в первую очередь.

Рассмотрим оценку такой модификации алгоритма. Пусть носитель имеет N кластеров, а сам файл состоит из M кластеров, причем он разбит на s фрагментов, состоящих из $m_i$ кластеров каждый. Тогда $\sum_{i=1}^{s} m_i = M$. Пусть также на носителе имеется еще некоторое количество файлов формата JPEG, занимающих L кластеров. При использовании немодифицированного алгоритма для восстановления файла потребуется произвести

$$\sum_{i=1}^{s-1}(N-\sum_{j=1}^{i} m_i) \qquad (1)$$

опробований отдельных кластеров и проверок того, обнаруживается ли фрагментация. При каждом запуске поиска продолжения файла опробованию подлежат все кластеры, кроме тех, для которых уже установлено, что они входят в искомый файл. Если предположить, что M << N, то сумму (1) можно заменить произведением

$$(s-1)N. \qquad (2)$$

Для модифицированного алгоритма число опробований будет зависеть от результатов предварительной классификации кластеров носителя на предположительно принадлежащие файлам формата JPEG и предположительно таким файлам не принадлежащие. Согласно приведенным в работе [8] результатам, для гипотез $H_0$ = {кластер принадлежит файлам формата JPEG} и $H_1$ = {кластер не принадлежит файлам формата JPEG} вероятности ошибок при этом будут составлять $\alpha = 0{,}035$ для ошибки первого рода и $\beta = 0{,}29$ для ошибки второго рода. Тогда в первый класс попадут $(M + L)(1 - \alpha) + (N - M - L) * \beta$ кластеров. Остальные кластеры попадут во второй класс. Общее же число опробований при этом составит, с учетом предположения M<<N,

$$((M+L)(1-\alpha)+(N-M-L)*\beta)*(s-1)(1-\alpha)+N*(s-1)*\alpha. \qquad (3)$$

Выделив в выражении (3) множитель (s — 1), получим:

$$(s-1)((M+L)(1-\alpha)+\beta(N-M-L))(1-\alpha)+\alpha N). \qquad (4)$$

Сравнивая выражения (2) и (4), можно установить, что при указанных значениях α и β модифицированный алгоритм потребует меньшего числа опробований, которое тем меньше, чем меньше значение отношения $\frac{M+L}{N}$.

### Доработка результатов восстановления фрагментированных файлов формата JPEG

Результатом работы алгоритма извлечения данных с фрагментированным файлом JPEG может быть последовательность данных, содержащая все кластеры искомого файла, а также кластеры, ему не принадлежащие. При этом особенности формата допускают ситуации, когда наличие постороннего фрагмента не вызывает ошибку при преобразовании последовательности данных файла в графическую информацию, представляемую пользователю. Видимое же пользователем изображение не соответствует изображению в искомом файле, пример чего приведен на рис. 2.

*Рис. 2. Результат извлечения фрагментированного файла формата JPEG*

Для исключения из таких результатов применения метода извлечения данных к фрагментированным файлам формата JPEG может быть применен алгоритм определения принадлежности конкретных кластеров данному формату. Такая возможность может быть реализована в программном средстве, позволяющем пользователю производить указанную доработку.

Рабочее окно прототипа такого программного средства показано на рис. 3.

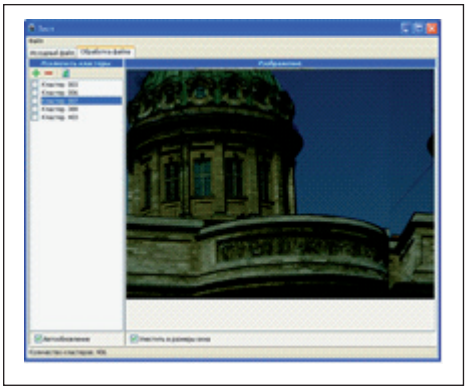

*Рис. 3. Фрагментированный файл в процессе исключения посторонних кластеров*

Результат работы алгоритма извлечения данных, показанный на рис. 2, загружается в программу, после чего запускается процесс поиска сигнатур в каждом кластере загруженного файла. Для каждого из них принимается решение об отнесении к классу вероятно принадлежащих или вероятно не принадлежащих данному файлу, после чего список кластеров, отнесенных ко второму классу, предлагается пользователю. Пользователь затем может самостоятельно устанавливать список кластеров, которые требуется исключить из загруженного файла, при необходимости дополняя данный список дополнительными кластерами — вводя их номера вручную. Например, обнаруживая в списке некоторую последовательность кластеров за исключением одного кластера, можно предположить, что вся последовательность является посторонним фрагментом, а не вошедший в нее кластер был ошибочно отнесен к первому классу. В этом случае пользователь может исключить всю последовательность, постепенно добиваясь корректного отображения содержимого графического файла. Результат такой обработки файла показан на рис. 4.

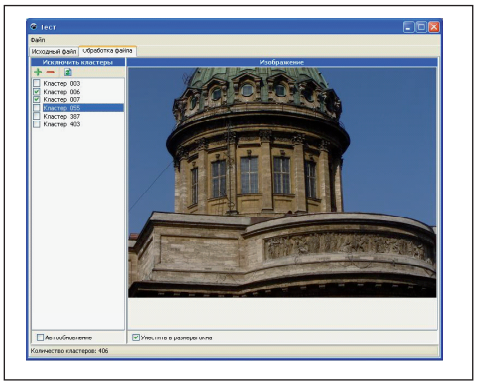

*Рис. 4. Результат исключения посторонних кластеров*

Таким образом, авторский метод определения принадлежности произвольного кластера цифрового носителя файлам формата JPEG с кодовыми таблицами, соответствующими стандарту формата [7], может быть применен двумя способами. Во-первых, для ускорения работы алгоритма, позволяющего собирать файлы данного формата, состоящие из нескольких фрагментов, и, во-вторых, для исключения посторонних кластеров из последовательностей данных носителя, заключенных между сигнатурами начала и окончания файла JPEG, обнаруженных в результате извлечения данных.

## Заключение

Восстановление графических файлов формата JPEG в случаях значительной фрагментации требует специальных методов преодоления возникающих сложностей с обнаружением и размещением в правильном порядке отдельных фрагментов или кластеров. Предложенный автором способ классификации кластеров на основе вероятной принадлежности файлам данного формата позволяет значительно ускорить или дополнить существующие методики их восстановления. При этом наилучший результат достигается в том случае, когда доля файлов формата JPEG среди всего объема носителя невелика. Применив его вместе с алгоритмом «умного извлечения» данных, можно достичь снижения числа операций, требуемого для получения корректного результата, а вместе со стандартным алгоритмом извлечения данных — позволить пользователю парировать его недостатки.

## СПИСОК ЛИТЕРАТУРЫ:

1. *Кэрриэ Б.* Криминалистический анализ файловых систем. СПб.: Питер, 2007.
2. *Сенкевич Г. Е.* Искусство восстановления данных. СПб.: БХВ-Петербург, 2011.
3. *Pal A., Shanmugasundaram K., Memon N.* Automated Reassembly of Fragmented Images // Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003.
4. *Sencar H. T., Memon N.* Identification and Recovery of JPEG Files with Missing Fragments // Digital Investigation. 2009. № 6. P. 88–98.
5. *Сорокин А. В.* Пределы применимости современных программных средств восстановления данных // Промышленные АСУ и контроллеры. 2013. № 1. С. 54–57.
6. *Pal A., Memon N.* Automated Reassembly of File Fragmented Images using Greedy Algorithms // IEEE Transactions on Image processing. February 2006. P. 385–393.
7. *Pal A., Sencar T., Memon N.* Detecting File Fragmentation Point Using Sequential Hypothesis Testing // Digital Investigations. 2008. № 5. P. S2–S13.
8. CCITT T.81 (09 /92) Digital Compression and Coding of Continuous-tone Still Images. The International Telegraph and Telephone Consultative Committee, ITU. 1993.
9. *Сорокин А. В.* Об определении принадлежности кластеров диска файлам формата JPEG // Проблемы информационной безопасности. Компьютерные системы. 2012. № 4. С. 61–67.