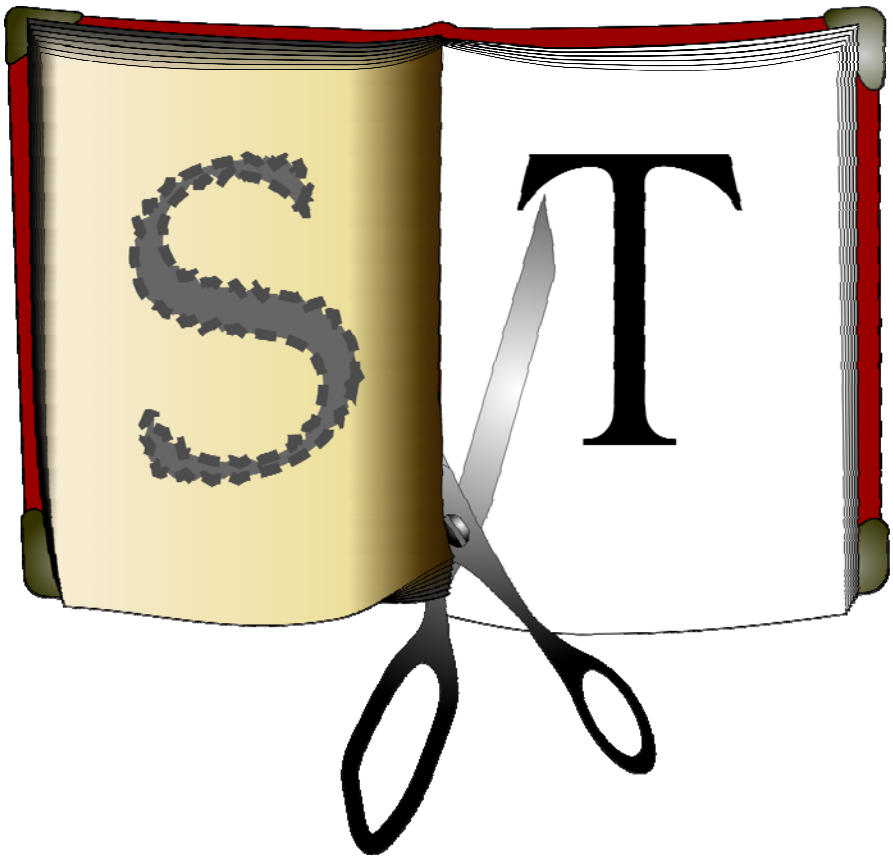
Scan Tailor

Руководство пользователя



2012

1. О программе

Scan Tailor (tailor по-английски - портной) – это интерактивный инструмент для пост-обработки сканированных страниц. Он делает такие операции как разрезание страниц, компенсация наклона, добавление/удаление полей, и другие. Вы даете ему необработанные сканы, а в результате получаете страницы, готовые для печати или сборки в PDF или DjVu файл.

Сканирование, оптическое распознавание символов, а также сборка многостраничных документов не входят в задачи проекта.

Программа разрабатывается как для Windows, так и для GNU/Linux (и других Unix-подобных систем).

1.1 Полезные ссылки

- Программа ScanTailor – загрузка дистрибутива программы.
- Тема на форуме Ru-Board - основной топик обсуждения. Здесь можно скачать и последние рабочие бета-версии программы.
- Тема на форуме Натахаус
- Контактная информация автора wiki-документации.
- Исходная wiki-документация.

2. Быстрое начало

1. Создаем новый проект. Добавляем в проект сканы, при необходимости задав им верное разрешение (DPI).
2. При необходимости, вручную правим ориентацию страниц на стадии Исправление ориентации.
3. Переходим на стадию Разрезка страниц, выбираем преобладающий тип разреза, применяем его ко всем страницам, затем, в случае необходимости, выбираем тип разреза для страниц с другим типом разреза. Запускаем пакетную обработку, по окончании обработки на страницах с ошибочно определенной линией разреза поправляем ее расположение.
4. Переходим на стадию Полезная область и там запускаем пакетную обработку. Ждем, пока пакетная обработка не завершится.
5. Пробегаем глазами по ленте предпросмотра в поисках неверно определенных областей. Кликаем на такие страницы и правим вручную. Если ошибка вызвана неправильной компенсацией наклона на предыдущей стадии, переходим на эту стадию и исправляем проблему там, после чего возвращаемся на стадию "Полезная область" и убеждаемся, что рамка на этот раз определилась правильно. Удобно при этом пользоваться сортировкой по ширине и по высоте полезной области (выпадающий список в нижней части полосы предпросмотра).
6. Переходим на стадию Макет страницы.
7. Для страниц обложки выключаем режим "Выровнять с другими страницами".
8. Выбираем наиболее типичную страницу книги и задаем для нее поля таким образом, чтобы они оказались максимально приближенными к полям бумажной книги. Применяем настройку полей ко всем страницам.
9. Затем обращаем внимание на пунктирные линии (если они есть). Если они достаточно далеко от сплошных, значит есть страницы, для которых суммарный размер полезной области и полей получился большим, чем размер страницы бумажной книги. Для обнаружения таких страниц удобно воспользоваться сортировкой по ширине и высоте. Для правки полезной области придется возвращаться на соответствующую стадию.
10. В режимах сортировки по высоте и ширине пробегаем глазами по верхним страницам на ленте предпросмотра в поисках неполных страниц. Для них обычно требуется изменить параметры центрирования, либо сдвинуть одно из полей (не забыв разорвав связь между парными полями).
11. Задаем для полей страниц обложки нулевые значения.
12. Переходим на этап Стадия "Вывод", по-умолчанию для всех страниц уже выбран режим Черно-белый. Используя Shift и Ctrl выделяем на ленте предпросмотра страницы, в которых есть одновременно и цвет и полутоновые иллюстрации, и применяем к ним режим Смешанный. Затем выделяем страницы обложки и полностраничные иллюстрации, и применяем к ним режим Цветной/Серый. Внимание, не уменьшайте DPI вывода! При выводе на 600 DPI получаются красивые гладкие буквы, при этом при правильном кодировании размер djvu не увеличивается.
13. Запускаем пакетную обработку. По окончании проверяем результат. Если для страниц в смешанном режиме область текста или иллюстраций определена неверно, ее можно исправить в режиме Зоны картинок. Повторно запускаем пакетную обработку (на этот раз она пройдет гораздо быстрее, чем в первый)
14. Обработанные файлы будут записаны в директорию, указанную при создании проекта (по-умолчанию - в подпапку out). Об их кодировании в формате DjVu читаем в статье Создание качественных DjVu методом вклейки иллюстраций

3. Установка и первый запуск

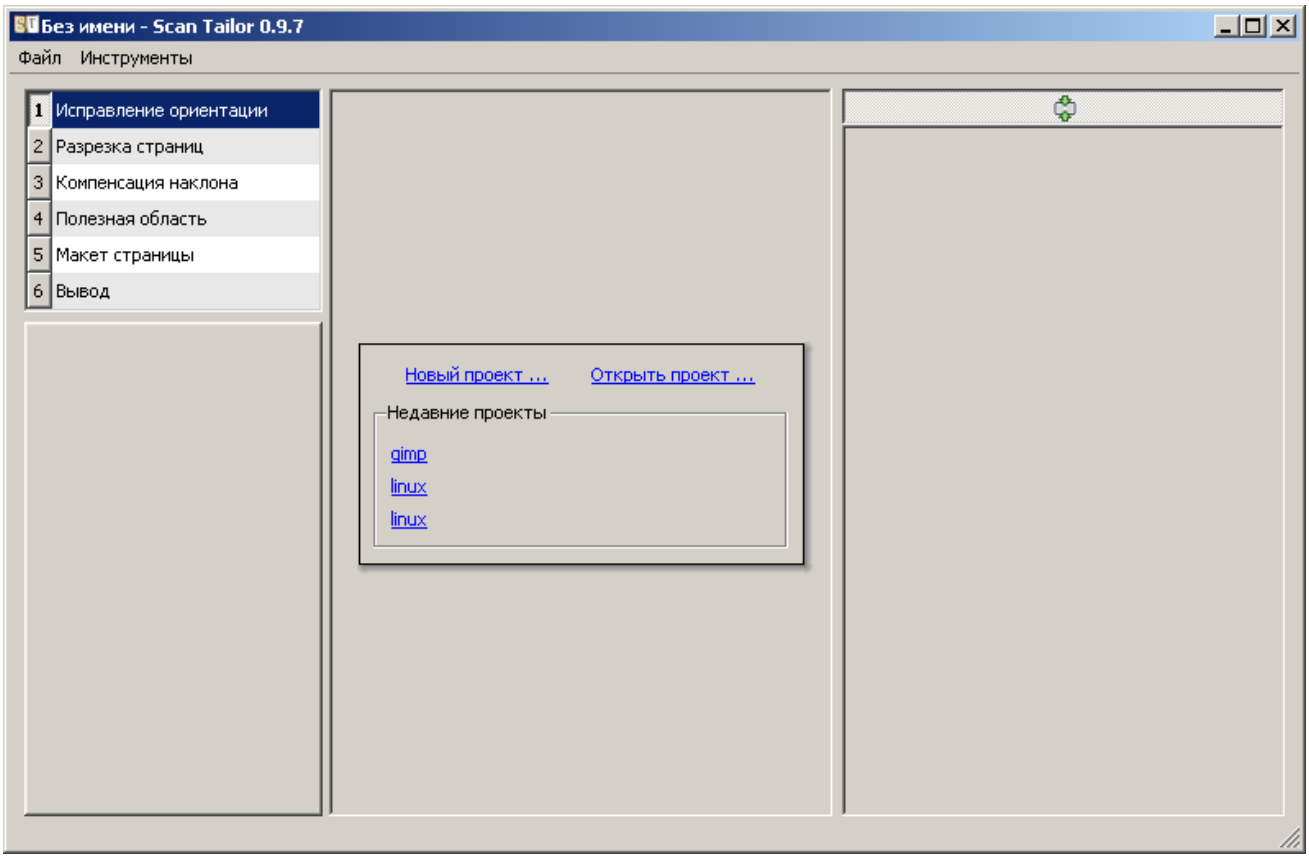
Для Windows
Версии для Windows поставляются в виде инсталлятора. Установите программу и запускайте ее через меню.

Для GNU/Linux
Как минимум следующие дистрибутивы имеют Scan Tailor в оффициальных репозиториях:

- Ubuntu
- Fedora
- OpenSUSE
- AltLinux

Если в вашем дистрибутиве его нет, или вы хотите самую последнюю версию, тогда вам придется собрать его из исходников. После сборки вы сможете запустить Scan Tailor через Alt+F2, введя там комманду "scantailor" (без кавычек).

После запуска, главное окно программы выглядит примерно так:

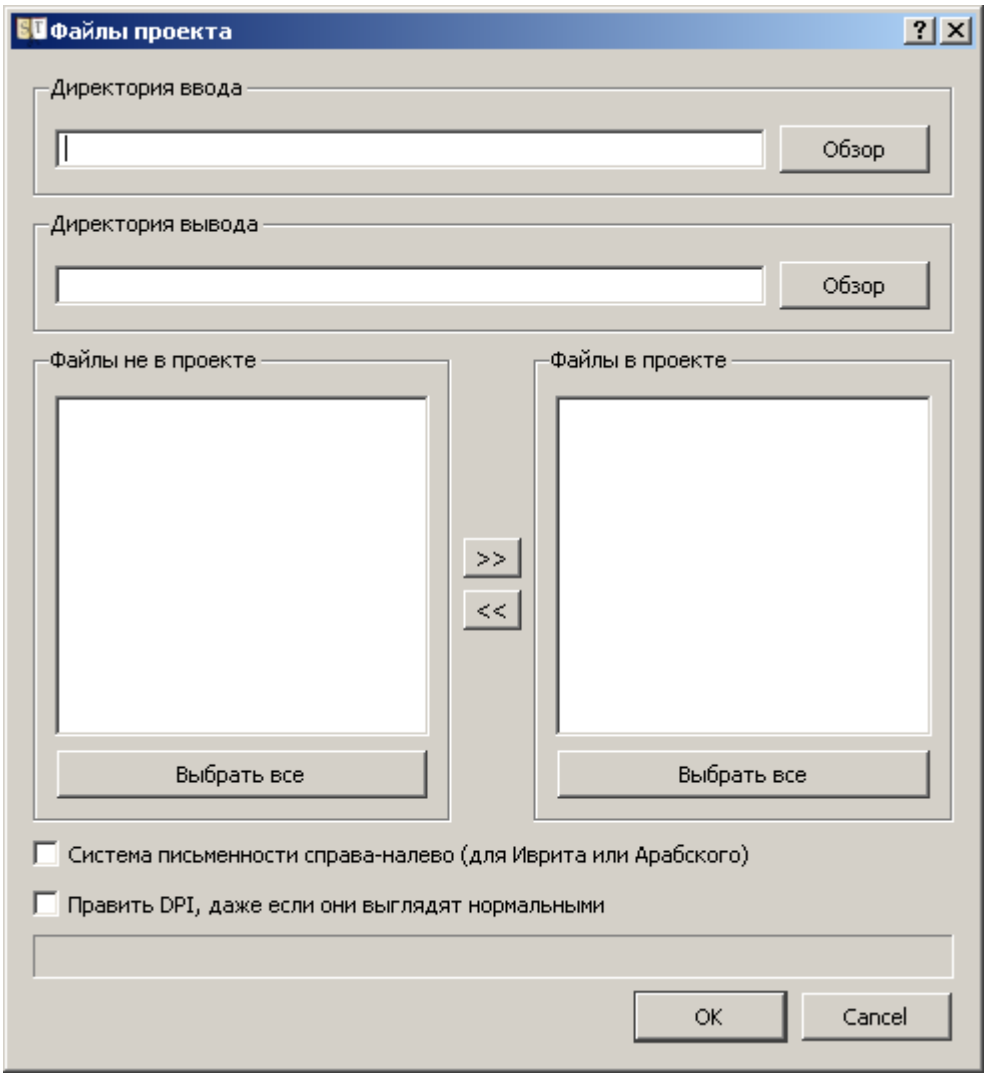


На центральной панели мы видим пункты "Новый проект ...", "Открыть проект ..." а также список недавних проектов, если таковые были.

4. Создание проекта

Итак, давайте создадим наш первый проект. В первую очередь нам понадобятся какие-нибудь исходные материалы - Scan Tailor требует наличие хотя бы одного файла в проекте. Таким исходным материалом может быть сканированная страница книги (можно разворот) или журнала.

Жмем на "Новый проект ..." - открывается диалог "Файлы проекта".



"Директория ввода" – папка, где находятся исходные сканы.

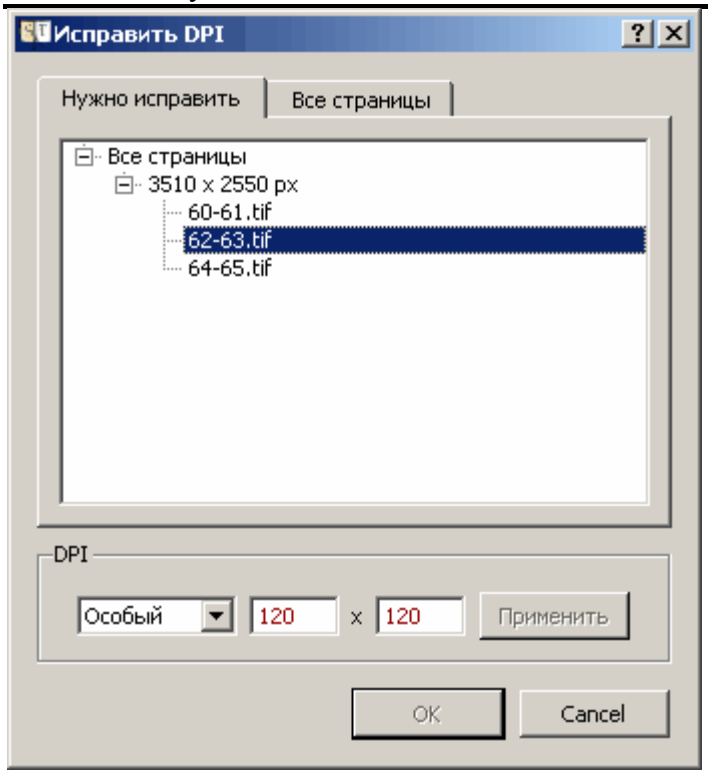
"Директория вывода" – папка, куда будут сохранены обработанные сканы.

"Файлы не в проекте" – файлы, находящихся в папке, указанной в "Директории ввода", но при этом не внесенные в проект.

"Файлы в проекте" – файлы, внесенные в проект, необязательно из директории ввода. В проект можно вносить файлы из разных директорий, для чего надо внести в проект файлы из одной директории ввода, потом сменить ее, внести файлы оттуда, и так далее.

Кнопки ">>" и "<<" между "Файлы не в проекте" и "Файлы в проекте" перемещают файлы из одного списка в другой. Перемещаются не все файлы, а только засвеченные. Можно засветить все файлы кнопкой "Select All", можно засвечивать или снимать засветку индивидуально - через Ctrl+Click, или диапазонами - через Shift+Click.

Внеся файлы в проект (а поддерживаются файлы *.tif, *.tiff, *.png, *.jpg, *.jpeg), жмем OK. Обычно на этом процесс создания проекта заканчивается, но если в проекте попались файлы с неуказанными или явно неправильными DPI, тогда открывается диалог "Исправить DPI".

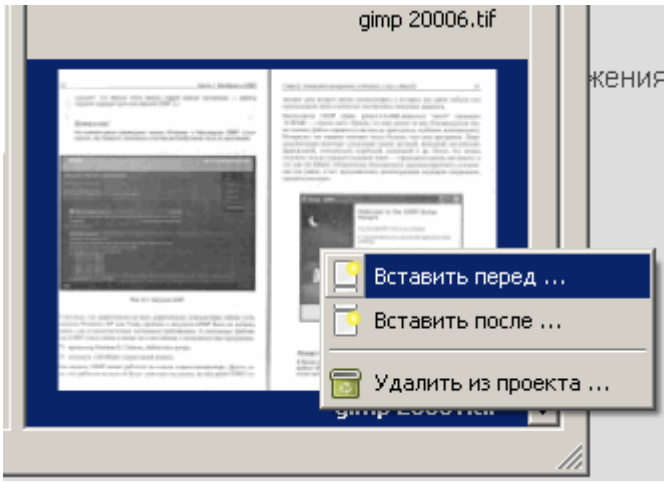


Если вы не до конца понимаете назначение термина "DPI", то прежде чем продолжить, вам следует почитать Ликбез по DPI, в противном случае идем дальше.

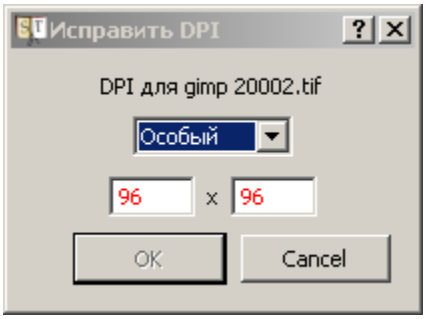
Во вкладке "Нужно исправить" перечислены только те файлы, для которых DPI не заданы или явно неправильны. Во вкладке "Все страницы" (Не путать с одноименным узлом раскрывающегося списка) перечислены все файлы вообще. У них тоже можно менять DPI. Если мы знаем, что все файлы в проекте имеют DPI 300 x 300, тогда можно одним махом задать этот DPI для всех файлов. Для этого переходим к узлу "Все страницы" (не путать со вкладкой), указыаем DPI, жмем применить. Также можно указывать DPI для групп файлов, имеющих одинаковые пиксельные размеры, а также и для отдельных файлов. Файлы, для которых были указаны DPI, пропадают из вкладки "Нужно исправить". Когда эта вкладка совсем опустеет, станет активна кнопка OK. Нажав ее, процесс создания проекта будет завершен.

DPI, которые вы указываете, не пишутся в файлы - они только запоминаются в проектном файле Scan Tailor.

На стадиях обработки "Исправление ориентации" и "Разрезка страниц" предусмотрена возможность добавления/удаления файлов из проекта с помощью всплывающего меню, вызываемого щелчком ПК мыши на нужной странице ленты предпросмотра:



При попытке добавления файла с явно некорретным DPI программа попросит его исправить:



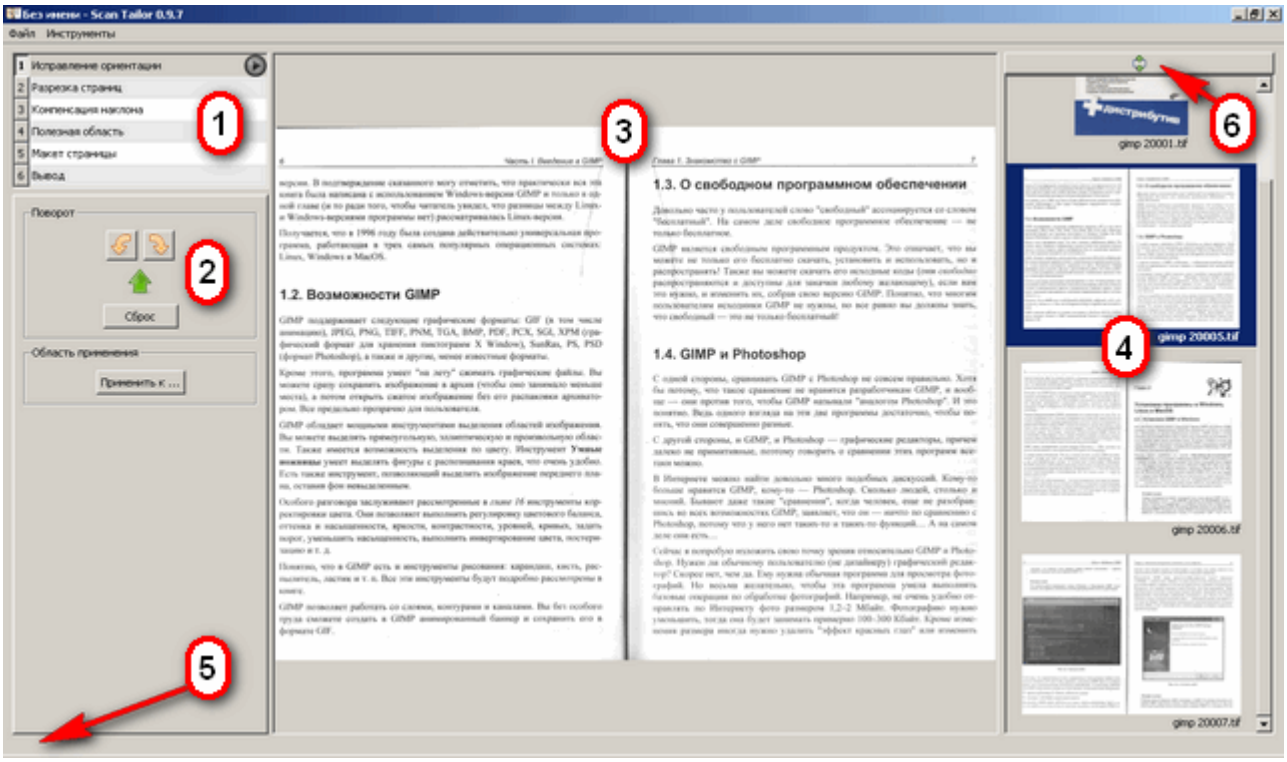
Имейте в виду, что на данный момент нет способа поменять DPI для файла в уже созданном проекте, как и способа удалить или добавить одновременно группу файлов в существующий проект. Появление этих возможностей ожидается в будущих версиях.

5. Концепция обработки сканов в Scan Tailor

Прежде всего необходимо понять общие концепции работы с программой.

Обработка сканированных страниц в **Scan Tailor** напоминает конвейер на заводе. И там и там есть стадии, на каждой из которых производится какая-то определенная манипуляция с изделием (сканом).

Посмотрим на главное окно программы:



Слева сверху мы имеем список стадий **(1)**, а ленту предпросмотра **(4)** как раз можно считать тем самым конвейером. Важно понять, что как и на обычном конвейере, если изделие добралось до определенной стадии, значит все предыдущие стадии уже были пройдены, причем в строго определенной последовательности. Если вы скажем сразу встали на стадию "Полезная область", то страницы, с которыми вы работаете на этой стадии все равно проходят все предыдущие стадии, просто вы этого не видите. В любой момент можно вернуться на любую из предыдущих стадий чтобы посмотреть, и при необходимости поправить то, что со сканом было сделано на этой стадии.

Круглые кнопки "Play" напротив каждой стадии как раз запускает пакетную обработку, т.е. вышеописанный конвейер. Страницы, одна за другой, проходят все стадии обработки, включая текущую стадию. "Проходят все стадии обработки" - не обязательно означает "обрабатываются". Например если скан уже был разрезан, то когда он вновь попадет на стадию разрезки - заново его резать не будут, если конечно не было изменено что-то важное, например его ориентация. Остановить пакетную обработку можно в любой момент большой круглой кнопкой "Stop", появляющейся во время обработки на главной рабочей области **(3)**.

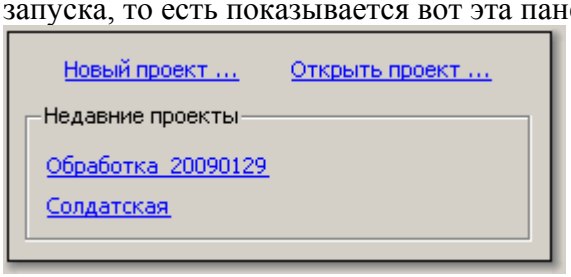
Еще одна маленькая деталь - никакие из стадий, кроме последней стадии "Вывод" не записывают на диск получившихся изображений. Дело в том, что все стадии кроме последней - чисто аналитические. Их результат - не новое изображение, а новая информация об исходном изображении.

Остальные области главного окна:
(2) Параметры обработки для текущей стадии.
(3) Главная рабочая область, в которой отображается само изображение, а также инструменты для манипуляции с ним, зависящие от текущей стадии.
(5) Строка подсказки - здесь можно увидеть интерактивную подсказку по управлению программой.
(6) Кнопка "Держать активную страницу в поле зрения". В нажатом состоянии переключает программу в режим отслеживания текущей обрабатываемой страницы на ленте предпросмотра.

6. Меню программы

Меню "Файл"
Новый проект ...
Открыть проект ...
Сохранить проект
Сохранить проект как ...
Закрыть проект
Выход

Все это не требует объяснений, за исключением пункта "Закрыть проект". Эта комманда закрывает проект, но не всю программу. Главное окно приводится к виду, который оно имеет сразу после запуска, то есть показывается вот эта панель:



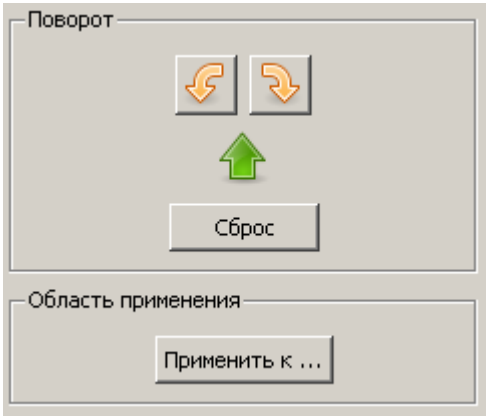
Меню "Инструменты":
Режим отладки – предназначен только для разработчиков. После обработки отдельной страницы, в центральной области появляются вкладки с промежуточными результатами обработки.
Настройки - предназначен для дополнительной настройки программы. В настоящий момент настройки позволяют лишь попробовать включить опцию "Использовать 3D ускорение для интерфейса пользователя". Данная опция некорректно работает на некоторых моделях видеокарт, и потому по-умолчанию отключена.

7. Стадии обработки

7.1 Стадия "Исправление ориентации"

На данной стадии можно повернуть скан на угол, кратный 90 градусов. То есть положить на бок или перевернуть вверх ногами. Стадия **Исправление ориентации** - ручная, то есть программа не умеет сама определять правильную ориентацию сканов - это должен сделать пользователь. Это также означает, что запускать пакетную обработку на данной стадии бесполезно - это будет холостой прогон.

Панель параметров для данной стадии выглядит так:

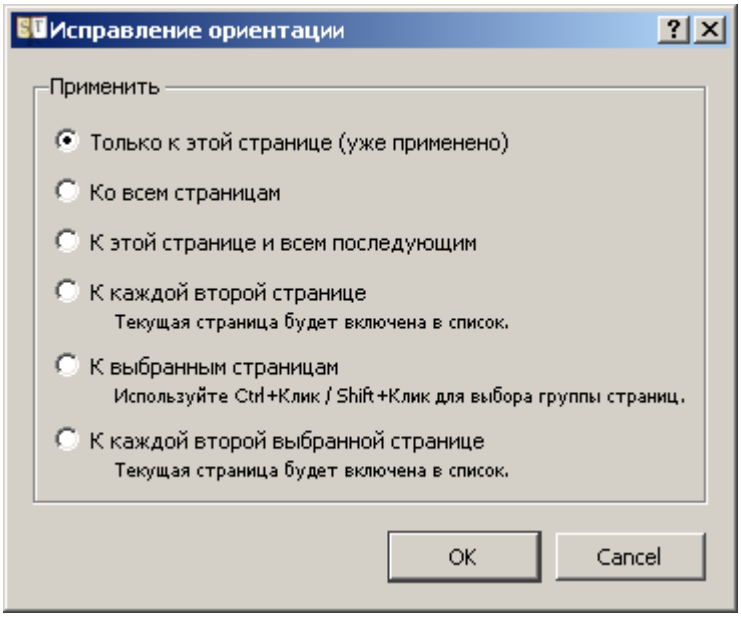


7.1.1 Поворот

Кнопки с желтыми стрелками поворачивают скан на 90 градусов в ту или иную сторону. Зелёная стрелка показывает, куда в данный момент повернут скан. Кнопка *Сброс* возвращает скан в исходное положение - зеленая стрелка будет указывать вверх.

7.1.2 Область применения

Применить к ... – позволяет применить поворот не только к данной странице (что делается автоматически), но и к другим. При нажатии открывается такой диалог:



Первые три пункта в списке не требуют пояснений. Вариант "Только к этой странице" выбирается по-умолчанию.

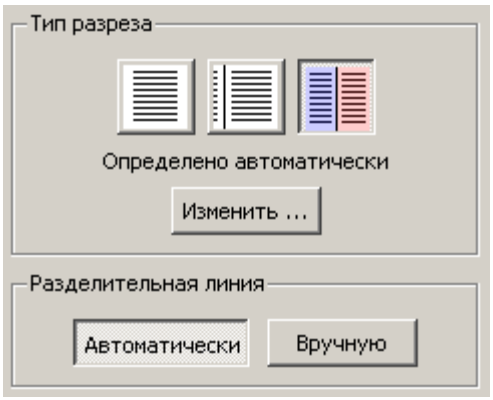
Пункт "К каждой второй странице" применит поворот либо к каждой четной, либо к каждой нечетной странице в соответствии с тем, является ли четной или нечетной текущая страница.

Последние два пункта активизируются только при наличии двух и более выбранных страниц на ленте предпросмотра, причем для активизации пункта "К каждой второй выбранной странице" выбранный интервал должен быть непрерывным (поэтому выбирать интервал для данного случая удобнее с помощью Shift + Клик).

7.2 Стадия "Разрезка страниц"

На этой стадии определяется, нужно ли, и если нужно то как, разрезать скан.

Панель параметров на этой стадии выглядит так:



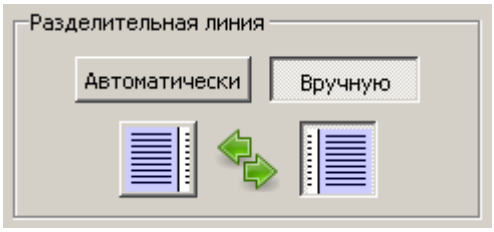
Тип разреза слева направо:

1. Одностраничный скан без каких-либо частей соседней страницы. Такие сканы обычно получаются на специализированных сканерах для книг.
2. Одностраничный скан, в который попала часть соседней страницы.
3. Двухстраничный скан.

Тип разреза определяется автоматически, хотя может быть задан и вручную. С помощью кнопки *"Изменить ..."*, вручную заданный тип разреза можно применить ко всем страницам сразу. Этой же кнопкой можно вернуть автоматический выбор типа разреза.

Разделительная линия также может определяться автоматически или задаваться вручную, но ее нельзя применить к другим страницам.

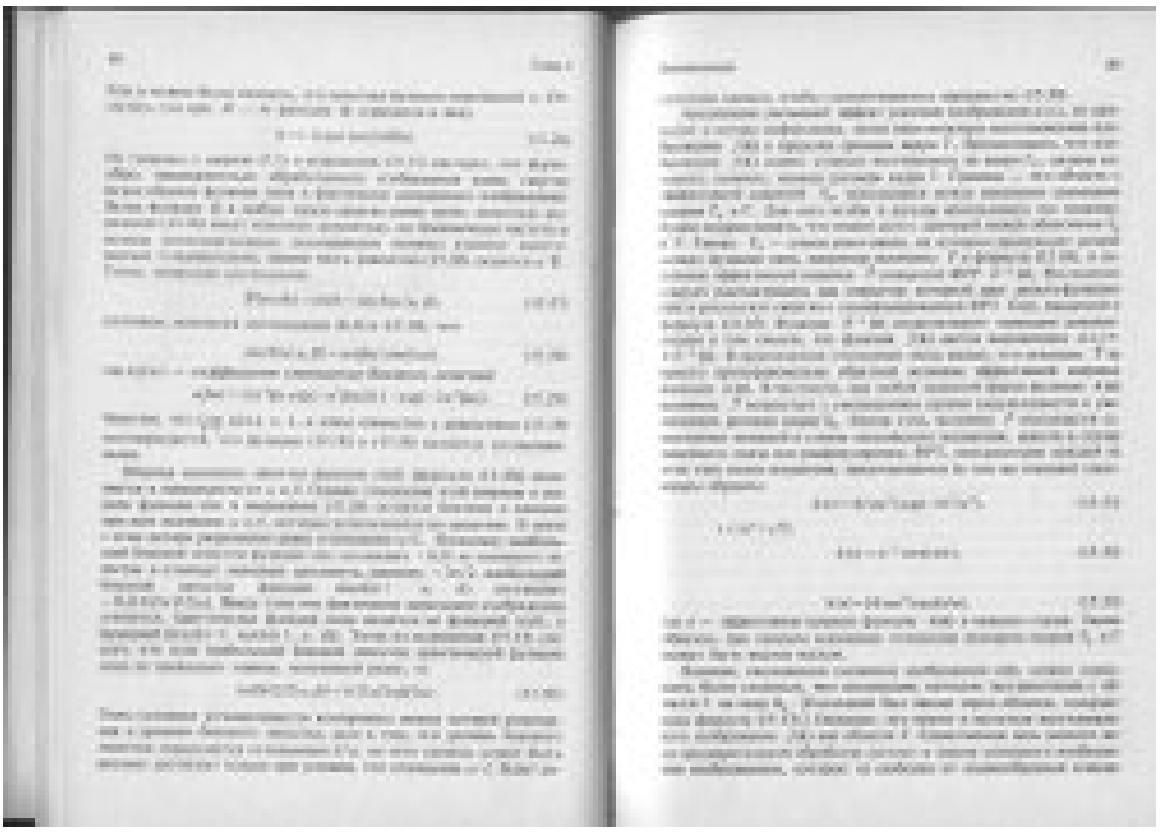
В случае, если выбран 2-й тип разреза, появится возможность вручную указать полузную область: слева или справа от линии разреза. Зеленая стрелка - это на самом деле кнопка (что становится очевидным при наведении на нее мышкой), которая как раз и переключает полезную область.



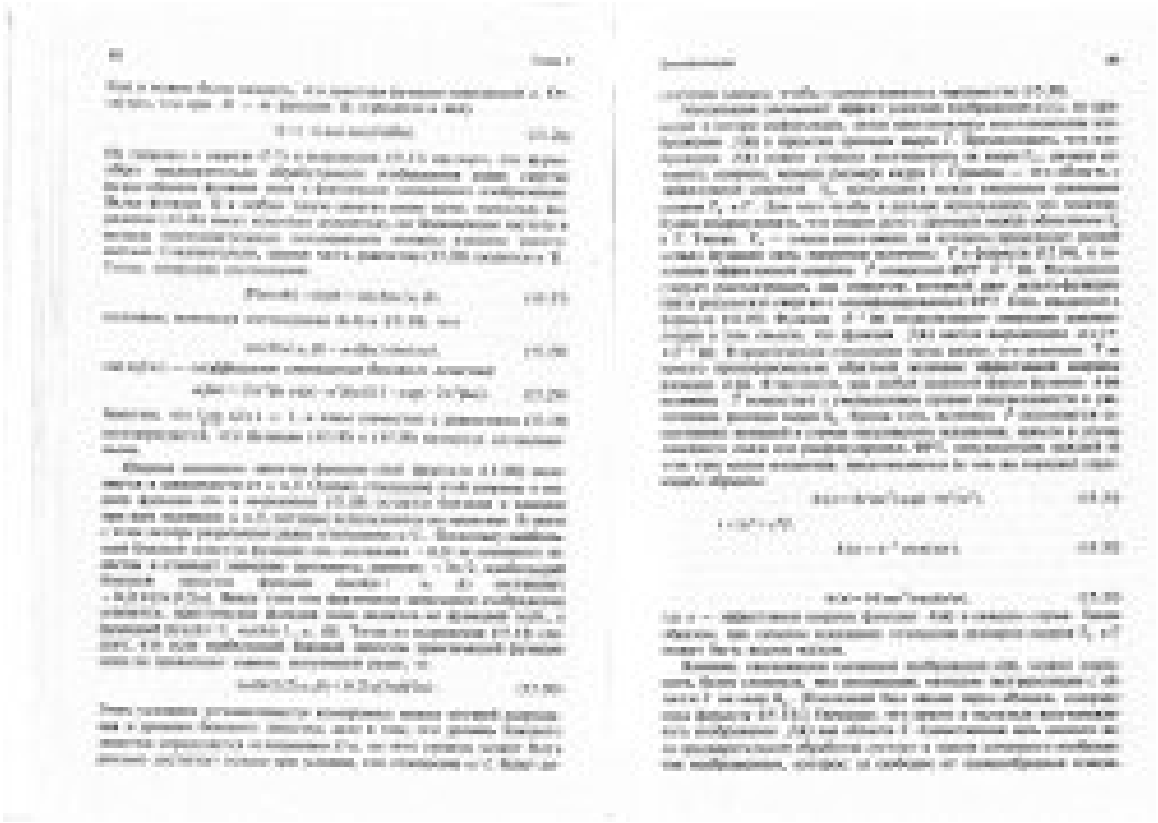
Чтобы увеличить шансы на правильное автоматическое определение типа разреза и разделительной линии, старайтесь следовать этим правилам при сканировании:

- Если делаете одностраничный скан, делайте его в портретной, а не в ландшафтной ориентации.
- При сканировании выбирайте наиболее сырой режим, то есть такой, при котором программа сканирования не будет пытаться ничего улучшать.

Примеры хороших сканов:



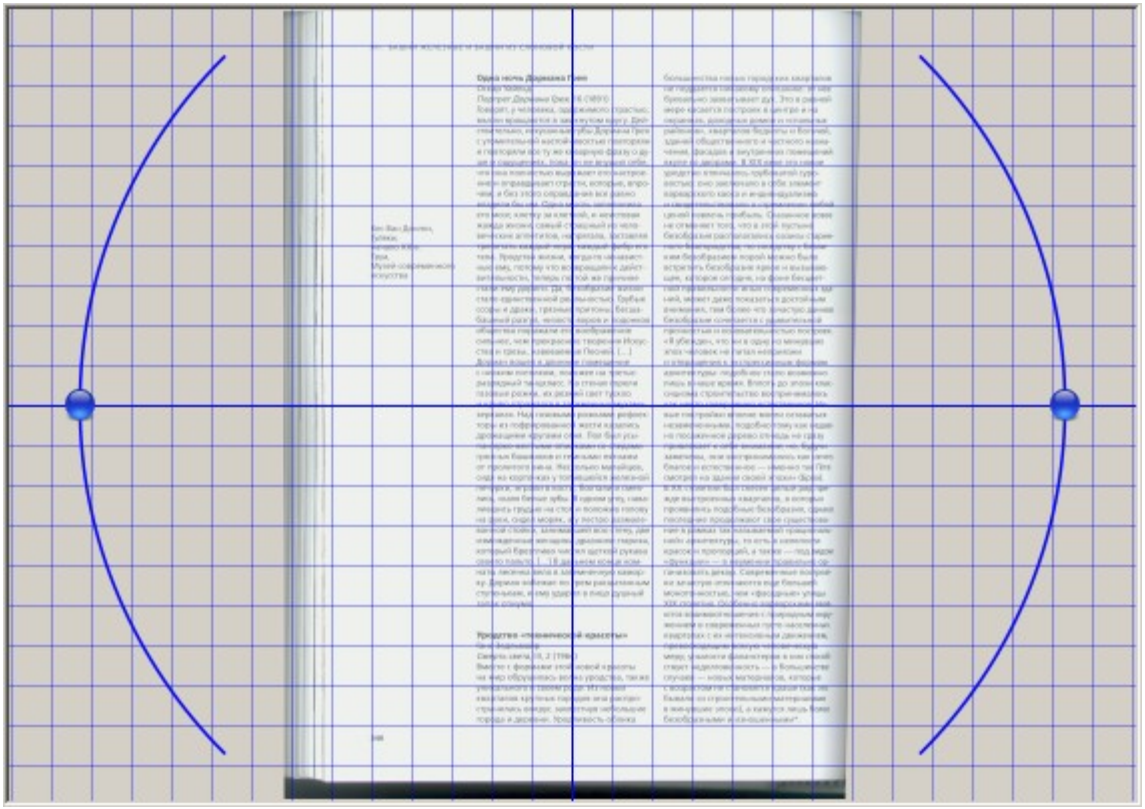
Пример плохого скана (предварительно обработанного программным обеспечением сканера):



7.3 Стадия "Компенсация наклона"

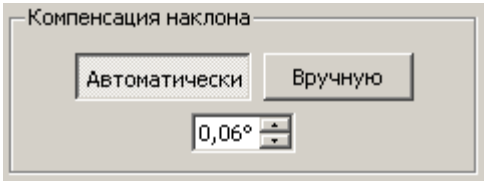
На этой стадии определяется угол, на который надо повернуть страницу, чтобы строки стали строго горизонтальными. Поскольку компенсация делается простым вращением, такие искажения как "кривые хвосты" на этой стадии исправить нельзя. Угол наклона определяется автоматически, но имеется возможность задать его и вручную.

Вот как выглядит рабочая зона в этом режиме:



Изображение можно вращать, перетаскивая мышкой рукоятки, которые по краям.

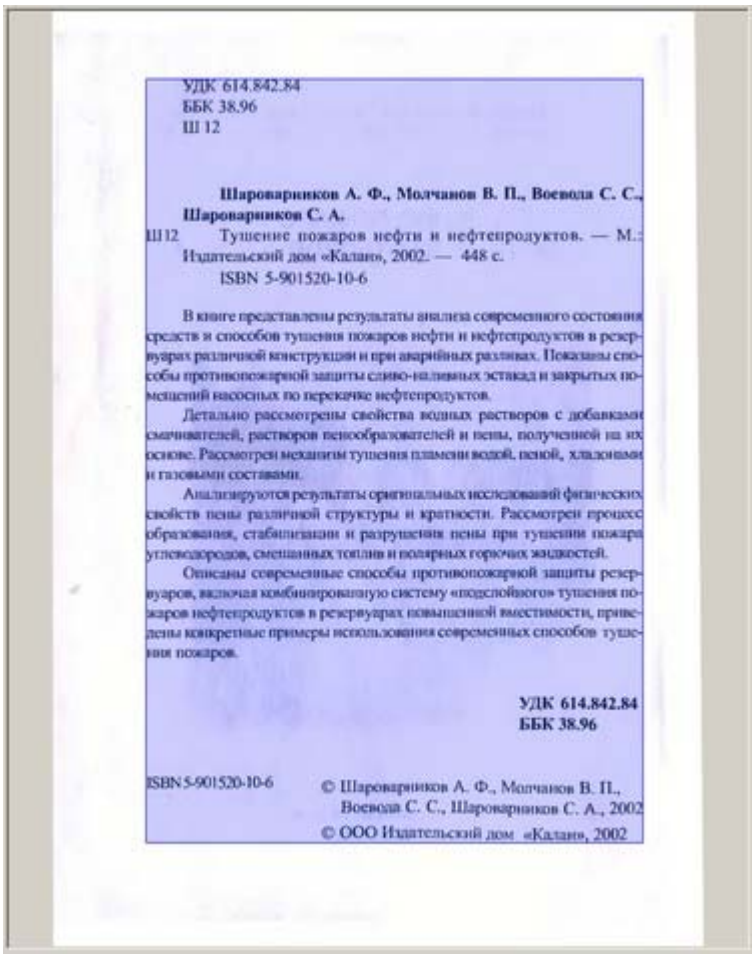
Вот так выглядит панель опций:



Здесь можно явно задать угол поворота в градусах. Положительные углы будут вращать изображение против часовой стрелки, отрицательные - по часовой стрелке.

Для тонкой подгонки угла, удобно кликнуть мышкой по текстовой части поля ввода угла, после чего использвать колесо мыши для его подгонки. При этом нажатие клавиши Ctrl увеличивает шаг вращения.

7.4 Стадия "Полезная область"



УДК 614.842.84
ББК 38.96
Ш 12

Шароварников А. Ф., Молчанов В. П., Воевода С. С., Шароварников С. А.
Ш12 Тушение пожаров нефти и нефтепродуктов. — М.: Издательский дом «Калан», 2002. — 448 с.
ISBN 5-901520-10-6

В книге представлены результаты анализа современного состояния средств и способов тушения пожаров нефти и нефтепродуктов в резервуарах различной конструкции и при аварийных разливах. Показаны способы противопожарной защиты сливо-наливных эстакад и закрытых помещений насосных по перекачке нефтепродуктов.

Детально рассмотрены свойства водных растворов с добавками смачивателей, растворов пенообразователей и пены, полученной на их основе. Рассмотрен механизм тушения пламени водой, пеной, хладонами и газовыми составами.

Анализируются результаты оригинальных исследований физических свойств пены различной структуры и кратности. Рассмотрен процесс образования, стабилизации и разрушения пены при тушении пожаров углеводородов, смешанных топлив и полярных горючих жидкостей.

Описаны современные способы противопожарной защиты резервуаров, включая комбинированную систему «послойного» тушения пожаров нефтепродуктов в резервуарах повышенной вместимости, приведены конкретные примеры использования современных способов тушения пожаров.

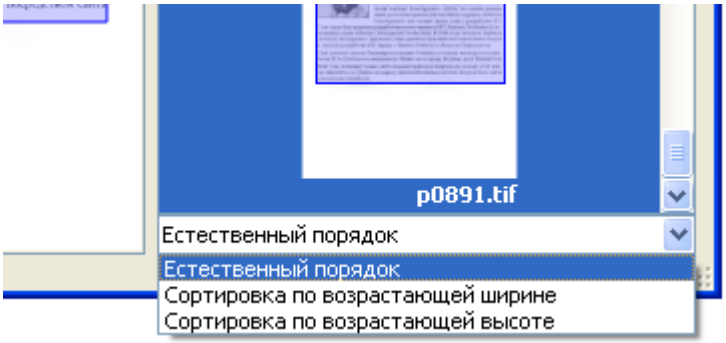
УДК 614.842.84
ББК 38.96

ISBN 5-901520-10-6

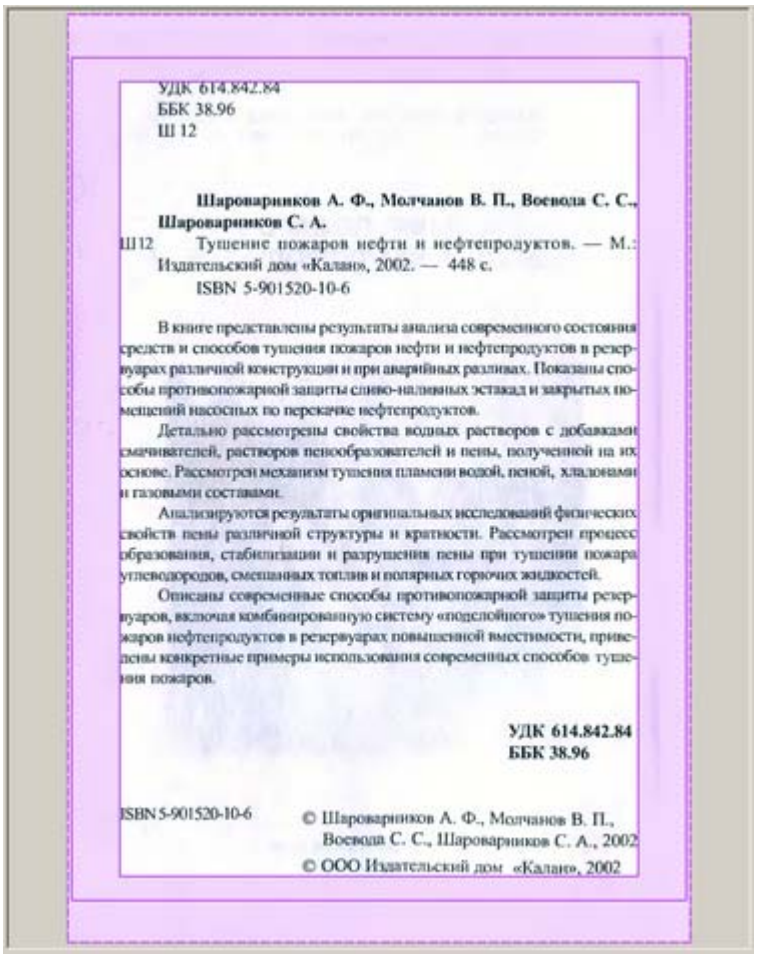
На этой стадии определяется область с "полезным" содержанием (залито цветом). Зачем вообще нужно определять эту область? Во первых для того, чтобы определить размеры страницы на выходе. К полезной области будут добавлены поля, и внешняя граница этих полей как раз и задаст размеры выходного файла. Во вторых, чтобы в вывод не попала линия сгиба или другой мусор с краев. Строго говоря, попадет мусор на полях в вывод или нет, зависит от режима вывода. В большинстве режимов поля заливаются белым.

Если область определилась неверно, можно поправить ее вручную, потянув мышкой за ее край. Бывает также, что на странице где совсем нет полезного содержимого, Scan Tailor все равно находит полезную область, или наоборот - не находит там, где она есть. В таком случае можно вручную создать или удалить область, кликнув правой кнопкой мыши по изображению, и выбрав нужный пункт меню.

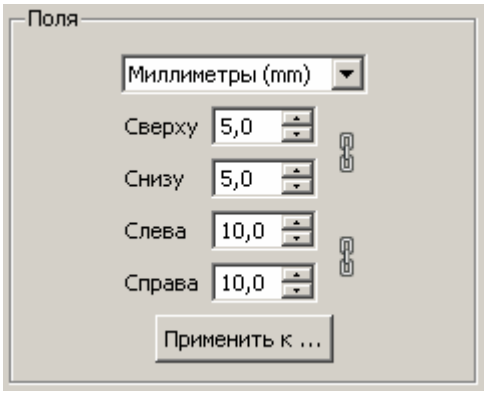
Для поиска неверно определенных областей удобно использовать сортировку по ширине/высоте полезной области, включить которую можно с помощью раскрывающегося списка под панелью предпросмотра:



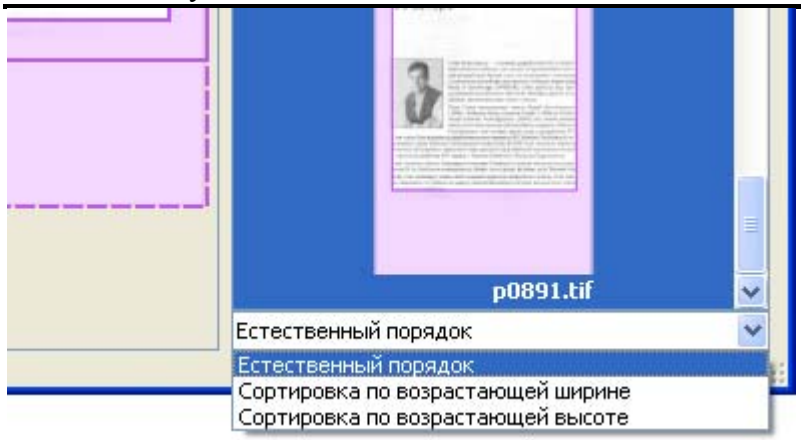
7.5 Стадия "Макет страницы"



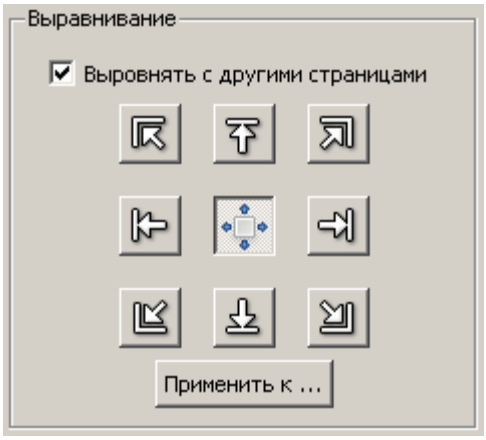
На этой стадии к полезной области добавляются поля. Есть два типа полей - жесткие и мягкие. **Жесткие поля** - это то, что между сплошными линиями. Они задаются пользователем. Можно либо потянуть за любую сплошную линию - хоть внешнюю, хоть внутреннюю, либо задать для полей числовые значения.



Мягкие поля - это то, что между сплошной и пунктирной линией. Эти поля добавляются автоматически, чтобы довести размер страницы до размера других страниц. Если вы видите пунктирную линию - это значит, что где-то в проекте есть страница с такой шириной (полезная область + жесткие поля) и (возможно другая) с такой высотой. Найти такие страницы можно, отсортировав страницы с помощью раскрывающегося списка:



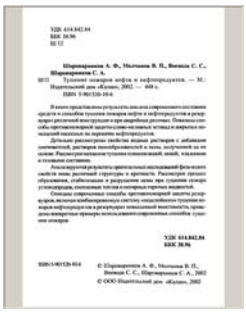
То есть одна большая страница вызывает появление мягких полей у всех остальных страниц, если только для них не отключено выравнивание.



Параметры выравнивания как раз и определяют, добавлять ли мягкие поля, и если добавлять, то с каких сторон.

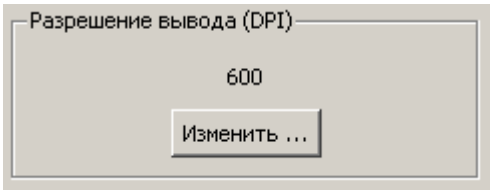
7.6 Стадия "Вывод"

На этой стадии создается и записывается на диск результирующее изображение страницы. Результат также отображается в центральной области программы.

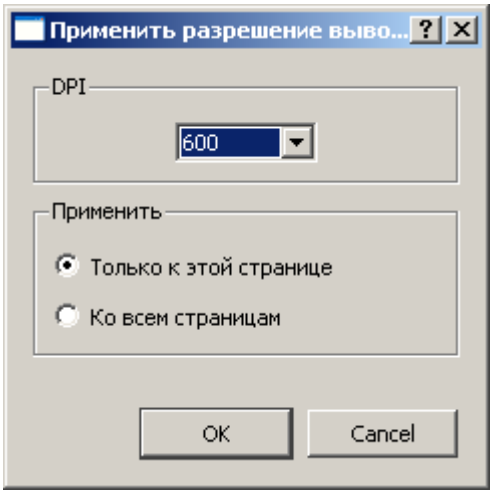


В отличии от других стадий, стадия "Вывод" становится доступной только после прохождения всеми страницами стадий "Полезная область" или "Макет страницы". Так происходит потому, что размеры страниц на выводе зависят друг от друга. Скажем если попалась крупная страница, то у всех остальных наращиваются поля (подробнее это описано в документации к стадии Макет страницы). Поэтому важно знать конечные размеры страниц, а узнать это можно только обработав их всех на стадиях "Полезная область" или "Макет страницы". Почему достаточно стадии Полезная область? Потому что все параметры на стадии "Макет страницы" задаются вручную, либо берутся значения по умолчанию. Таким образом в любой момент времени известны все параметры для всех страниц.

Настраиваются следующие параметры вывода:



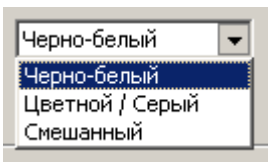
Разрешение вывода (DPI) – кнопокой "Изменить" можно вручную указать разрешение для выходных файлов:



Обратите внимание, что хотя для ввода поддерживаются несимметричные DPI (горизонтальное DPI не равно вертикальному), для вывода такой поддержки нет.

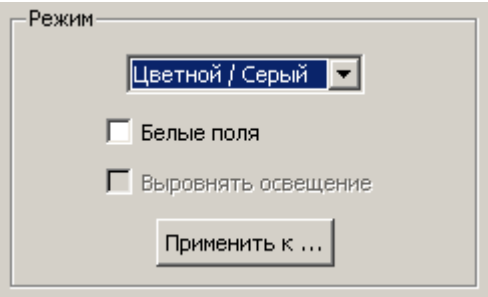
По умолчанию выставлено 600 DPI. В некоторых случаях бывает достаточно 300.

Режим – выбирается режим вывода готовых страниц:



Черно-белый режим объяснений не требует.

Для режима **Цветной / Серый** доступны дополнительные настройки:

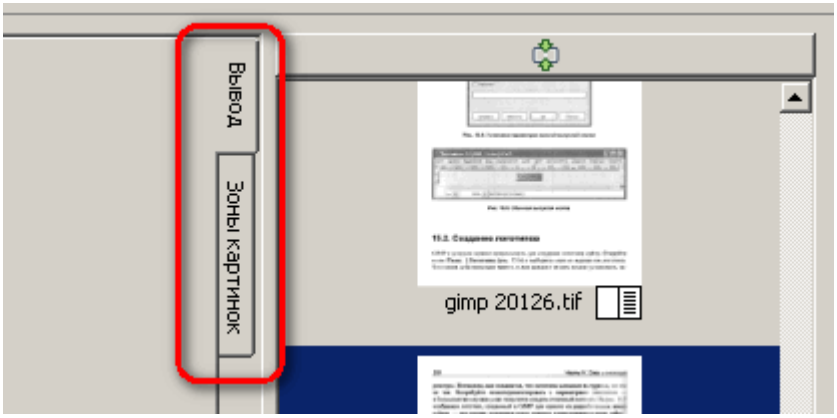


Поля можно залить белым или оставить как есть. Если поля заливаются белым, то становится доступной опция *Выровнять освещение*. Эта опция нормализует цвет фона, доводя его до белого, а также нормализует контраст, увеличивая его в затененных областях.

Смешанный режим применяется для проектов, в которых присутствуют сканы с полутоновыми картинками (в градациях серого или цветные). Картинки будут автоматически обнаружены и выведены как есть, точнее как в режиме "Цветной / Серый" с включенным выравниванием освещения. Остальная часть страницы выводится в черно-белом виде.
Автоматическое определение картинок работает достаточно хорошо, но если картинка где-либо плавно переходит в фон - результат может быть неудовлетворительным. В этом случае необходимо создать и настроить Зоны картинок. Обратите внимание - создание зон картинок возможно только в смешанном режиме.

7.6.1 Зоны картинок

Данный режим отображения возможен только в смешанном режиме вывода. Переключение в режим "зоны картинок" осуществляется с помощью закладок в правом верхнем углу рабочей области окна программы:



Если автоматика ошиблась, то нужно либо добавить серую область, либо исключить часть автоматически распознанной серой области. Для этого необходимо создать зону - замкнутый многоугольник произвольной формы, в свойствах которого можно указать, какую именно функцию - добавление или исключение серой области - он выполняет.

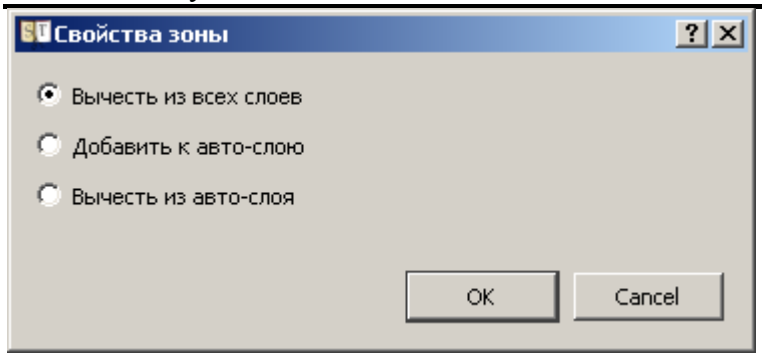
Для создания такого многоугольника достаточно последовательно щелкнуть ЛК мыши в предполагаемых вершинах многоугольника зоны, последний щелчек нужно сделать по первой вершине - это замкнет зону. Для удаления недорисованной зоны достаточно нажать клавишу Del.

Когда зона окончательно создана, ее вершины можно перемещать простым перетаскиванием. Для создания новой вершины достаточно кликнуть по грани зоны. Для удаления вершины нужно взять удаляемую вершину, и переместить ее поверх одной из вершин-соседок удаляемой.

При щелчке ПК мышки по зоне появляется всплывающее меню с пунктами Свойства и Удалить.

Выбор пункта Удалить приводит к удалению всей зоны.

Выбор пункта Свойства приводит к появлению окна "Свойства зоны":



При выборе пункта "Вычесть из всех слоев" содержимое данной зоны на выводе станет черно-белым, независимо от того какие еще зоны (ручные или автоматические) пересекаются данной зоной.

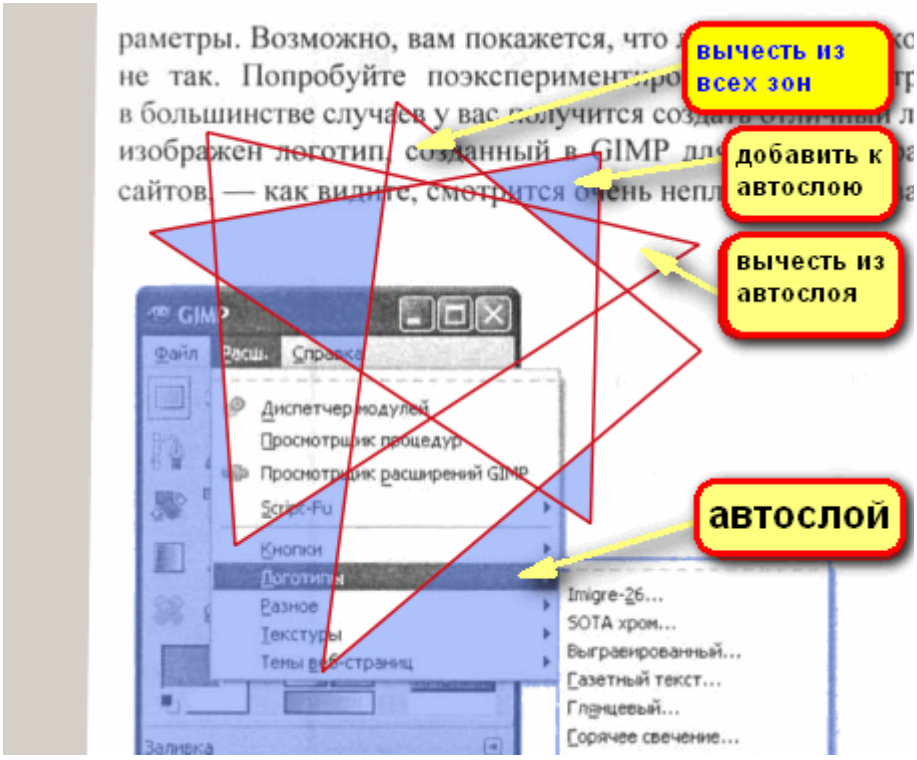
При выборе пункта "Добавить к автослою" содержимое данной зоны на выводе будет в оттенках серого.

При выборе пункта "Вычесть из автослоя" содержимое данной зоны на выводе будет черно-белым там, где оно не пересекается зонами с выбранным пунктом "Добавить к автослою".

Накладывая зоны и комбинируя их свойства можно точно настроить расположение серых и черно-белых участков на выводе. При этом удобно учитывать следующую систему приоритетов зон:

1. Самый высокий - "Вычесть из всех слоев"
2. "Добавить к автослою"
3. "Вычесть из автослоя" - не действует на зону с приоритетом 2, но действует на зону с приоритетом 4.
4. Самый низкий - Автослой.

Ниже на рисунке показана иллюстрация вклада каждого типа зоны в результирующее изображение:

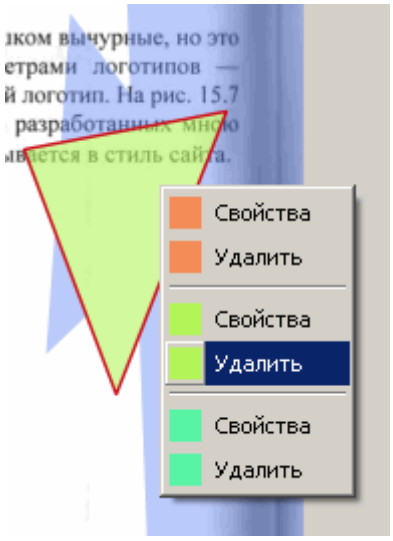


При этом удобным подспорьем в работе с зонами является то, что области, которые после выполнения операции "Вывод" останутся в оттенках серого выделяются мерцающей заливкой. Это сделано для того, чтобы зоны хорошо различались на изображении любого цвета.

Переключение типа зоны делает его типом по умолчанию для вновь создаваемых зон.

В случае наложения зон предусмотрен следующий механизм определения зоны, для которой будет применен пункт выпадающего меню: во всплывающем меню показываются через разделитель

пары пунктов "Свойства" и "Удалить" для каждой из наложенных на данную точку изображения зон. Данные пары для каждой зоны выделены своим цветом, и при наведении на них мышкой соответствующая зона начинает этим цветом подсвечиваться:



Чтобы посмотреть на окончательный результат работы созданных вами зон, щелкните на закладку "Вывод" в правом верхнем углу рабочей области.

7.6.2 Вывод всех файлов сразу

Для этого запустите пакетную обработку через меню. Файлы будут записаны в директорию, которую вы выбрали при создании проекта. К сожалению, в отличии от других стадий, уже выведенные страницы будут выведены повторно, даже если вы ничего в них не меняли. Это особенность будет устранена в будущих версиях. Пока же, при необходимости подправить несколько страниц после пакетного вывода, лучше не запускать пакетную обработку, а встать на каждую из таких страниц вручную. Процесс вывода текущей страницы начинается сразу при переходе на стадию "Вывод" или при переходе на другую страницу на этой стадии.

7.6.3 Формат выходных файлов

Вывод осуществляется в формат TIFF со сжатием LZW без потерь качества.

8. Часто задаваемые вопросы

8.1 Автоматический режим Scan Tailor'а часто ошибается в таких-то ситуациях ...

Можно сказать что все такие ситуации автору известны. В некоторых случаях есть какая-то надежда на улучшение алгоритмов автоматической обработки, в других - нет. Это как с прогнозом погоды - неправильные прогнозы время от времени - вполне нормальное и естественное явление. Заметьте - неправильный прогноз это (обычно) не ошибка метеорологов - это результат ограниченного объема и точности данных, а также сложности самого процесса, которой требуется моделировать. В общем не стоит лишний раз упомянать о проблемах автоматической обработки на форумах или писать баг репорты.

Если ошибки происходят очень часто, это говорит о том, что ваш исходный материал нарушает те или иные предположения, которые делает Scan Tailor. Вот некоторые из них:

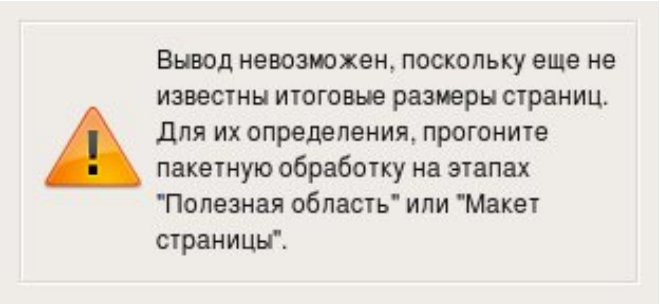
- Scan Tailor предполагает, что вокруг контента есть поля, и чем они больше, тем лучше.
- Scan Tailor предполагает, что на странице есть хотя-бы пара строк текста.
- Scan Tailor предполагает, что DPI прописанный в файлах или указанный вами вручную, соответствует реальному.
- Scan Tailor не любит слишком мелкого и / или расплывчатого текста. Это характерно для снимков камерой.

Довольно часто проблемы возникают на уже чем-то обработанных сканах. Такой исходный материал - неистощаемый источник проблем. Решил одну - осталось бесконечность минус один. Из-за этого решение таких проблем имеет низкий приоритет. В общем хорошую работу на таких сканах следует воспринимать как чудо, а плохую - как данность.

8.2 Что означает вопросительный знак на ленте предпросмотра?

Это означает, что данная страница еще не прошла текущую стадию, и соотвественно мы не можем отобразить ее так, как это полагается на данной стадии. Например если текущая стадия - Компенсация наклона, то страницы, еще не прошедшие эту стадию, будут отображаться на ленте предпросмотра без компенсации наклона и с вопросительным знаком. Также возможна ситуация, что страница проходила текущую стадию, но после этого были сделаны изменения, которые требуют повторного прохода этой стадии. Например страница прошла стадию Разрезка страниц, но после этого вы вернулись на Исправление ориентации и изменили эту самую ориентацию.

8.3 Меня не пускают на стадию Вывод, говоря что надо пройти предыдущие стадии - но я их прошел!



Значит пройдя предыдущие стадии, вы вернулись назад и что-то там изменили. В таком случае придется прогнать пакетную обработку на стадии Полезная область или Макет страницы еще раз. Не волнуйтесь, повторный прогон будет гораздо быстрее первого, потому как изменено было скорее всего немного страниц.

8.4 Ликбез по DPI

DPI (dots per inch - количество пикселей на линейный дюйм) - это связанный с растровым изображением страницы коэффициент, используемый для определения физического размера страницы. Основное применение - вывод страницы на принтер в оригинальном размере. Также применяется программами сканообработки (в том числе ScanTailor) в некоторых алгоритмах. В общем случае DPI изображения задается двумя числами - DPI по горизонтали и DPI по вертикали. Однако в подавляющем числе случаев эти два числа совпадают, поэтому в дальнейшем будем оперировать значением DPI, как единственным числом, определяющим разрешение и по вертикали, и по горизонтали.

DPI вычисляется по простой формуле

DPI = Размер_в_пикселях / Физический_размер_в_дюймах

Таким образом зная DPI отсканированной страницы (например 300dpi) и ее размер в пикселях (например, высота = 2700pix) мы можем узнать, что реальная высота отсканированной страницы составляла 2700 / 300 = 9 дюймов, или 9 * 2.54 = 22.86 см.

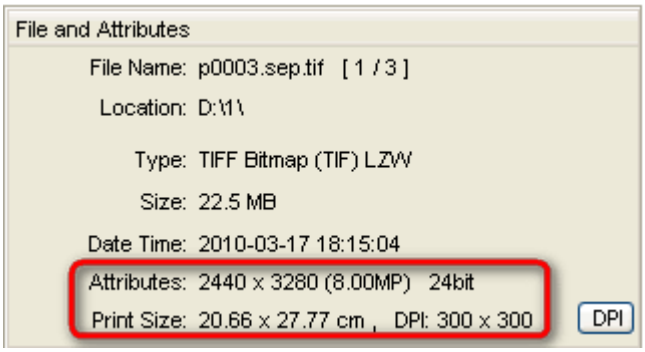
К сожалению, по ряду причин DPI не всегда верен. DPI не прописывают (или произвольно изменяют) некоторые программы. Кроме того, для "сканов", полученных фотографированием источника или через скриншоты страниц книг, предоставляемых для постраничного просмотра некоторыми сетевыми библиотеками определить точное DPI практически не представляется возможным. Типичное прописанное разрешение таких "сканов" равно 72 DPI.

Точно определить DPI таких первоисточников можно только зная физический размер страницы, который в свою очередь определяется по полиграфическим данным о формате книги, обычно указанных на одной из первых страниц. Например, запись вида Формат 60x90/16 означает, что печать производилась на листе размером 60см x 90см с последующей разрезкой на 16 страниц. Таким образом приходим к размеру одной страницы 15 см x 22.5 см с точностью до сантиметра в меньшую сторону.

Если полиграфические данные найти не удалось и книга напечатана стандартным шрифтом со стандартным межстрочным интервалом, то можно воспользоваться "правилом 6-7 строк". Данное правило гласит, что высота 6-7 строк текста примерно равна одному дюйму, соответственно высота 6-7 строк в пикселях равна DPI. То есть, если высота 6-7 строк текста примерно равна 300 пикселям, то разрешение скана можно оценить числом 300 DPI.

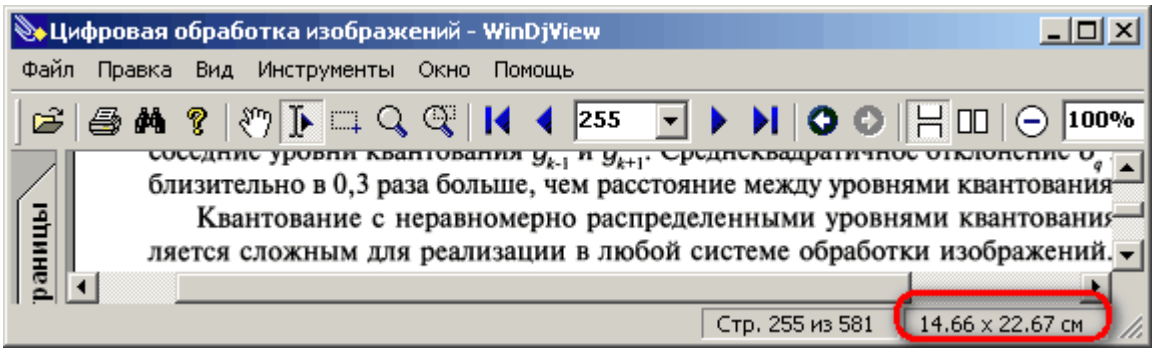
В связи с этим рекомендуем при подготовке сканов обращать внимание на вычисляемый по dpi физический размер страниц книги. При уменьшенном dpi он будет увеличенным и наоборот.

В FastStone Image Viewer физический размер и DPI скана можно посмотреть в полноэкранном режиме на всплывающей справа панели *File and Attributes*.

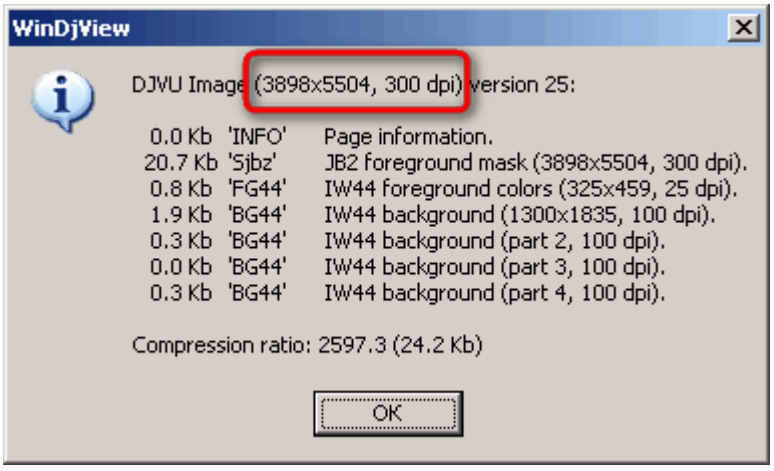


В этом же окне можно быстро задать просматриваемой странице требуемый DPI.

В WinDjView - вычисленный физический размер отображается в правой части строки состояния (возможно, придется предварительно ее включить через пункт меню *Вид -> Строка состояния*).



Пиксельные размеры и DPI DjVu страницы можно посмотреть в окне *Информация о странице*, запускаемом через контекстное меню страницы.



Есть возможность групповым образом задать требуемое значение DPI группе сканов с помощью инструмента *Пакетное преобразование* FastStone Image Viewer.

Одинаковое DPI всем страницам DjVu-книги можно задать программой djvu_toy, однако это не всегда возможно, т.к. полноцветные иллюстрации и обложки после подготовительной обработки обычно имеют меньший dpi (как правило, это разрешение сканирования), чем страницы с текстом, которые в качественно сделанных книгах как правило кодируются в разрешении 600dpi.

8.5 Советы по сканированию

Чтобы получить хороший результат обработки в Scan Tailor и минимизировать ошибки в автоматическом режиме, следуйте следующим правилам при сканировании:

- Не сканируйте в черно-белом режиме - сканируйте в оттенках серого, или, если это необходимо - в цвете.

- Не сканируйте в разрешении ниже 300 DPI. Больше - лучше (особенно для мелкого текста и для текста с тонкими перемычками), но замедляется и сканирование, и обработка. Обычно сканируют в 300 или в 600 DPI.
- Избегайте сканирования в JPEG - это формат с потерями, и дальнейшее пересохранение JPEG в другие форматы уже не вернет качества. Особенно искажаются цвета на границе между цветами.
- При сканировании обычно используют формат TIFF, но будьте внимательны - TIFF может использовать для сжатия jpeg-алгоритмы. Как правило, есть возможность настроить параметры сохранения в формат TIFF, в этом случае следует выбрать способ сжатия LZW - это формат сжатия без потерь. Если такой возможности нет, сканируйте в формате PNG - в нем гарантированно используются алгоритмы сжатия без потерь. В крайнем случае, сканируйте в BMP - файлы получаются несжатые и потому огромные, кроме того они требуют предварительной конвертации в TIFF или PNG, т.к. Scan Tailor не поддерживает формат BMP.
- Если имеет место эффект просвечивания обратной стороны страницы, то для его устранения следует подложить под сканируемую страницу лист черной бумаги. Особенно важно сделать это для страниц с иллюстрациями.
- Избегайте режима сканирования "Документ", и вообще старайтесь отключать все опции по улучшению сканов. Однократное улучшение (ST) всегда лучше двукратного (софт для сканирования, потом ST – см. пример.

9. Сборка из исходников под Linux

9.1 Подготовка

Для начала нам понадобятся исходники Scan Tailor'a. Их можно скачать с официального сайта или взять из Git. В Git лежат самые последние изменения, и он предназначен прежде всего для разработчиков. Если вы решили взять исходники из Git, предполагается, что вы представляете себе, что это такое. Если так, то вам достаточно знать Git URL чтобы вы смогли взять исходники оттуда. Итак, Git URL у нас такой:

```
git://scantailor.git.sourceforge.net/gitroot/scantailor/scantailor
```

Все остальное для сборки должно быть в вашем дистрибутиве, хотя вряд-ли оно все будет установлено по умолчанию. Итак, нам понадобятся:

- Базовый инструментарий для сборки. В Ubuntu есть мета-пакет *build-essential*, который тянет за собой все основные коомпоненты, необходимые для сборки программ из исходников.
- Система сборки CMake. Наверняка есть в репозитории вашего дистрибутиве под именем cmake.
- Qt версии по крайней мере 4.5.0. Она есть в достаточно свежих дистрибутивах. Нужны не только сама библиотека Qt, но и заголовочные файлы, которые как правило находятся в пакетах с именами, заканчивающимеся на -dev или -devel. Например в Ubuntu, нужный нам пакет называется libqt4-dev. Он потянет за собой и саму библиотеку Qt, если она еще не установлена.
- Заголовочные файлы от еще нескольких пакетов, а именно libpng (libpng12-dev), libjpeg (libjpeg62-dev), libtiff (libtiff4-dev), libxrender (libxrender-dev). Последний может также называться libXrender-dev, а раньше он был частью libx11-dev. Все эти пакеты определенно есть в репозитории вашего дистрибутива.

9.2 Сборка

Открываем консоль (терминал), идем в директорию, куда распаковали исходники Scan Tailor'a, и даем команду "cmake ." (там точка в конце):

```
cd "/home/username/путь/к/ST"
cmake .
```

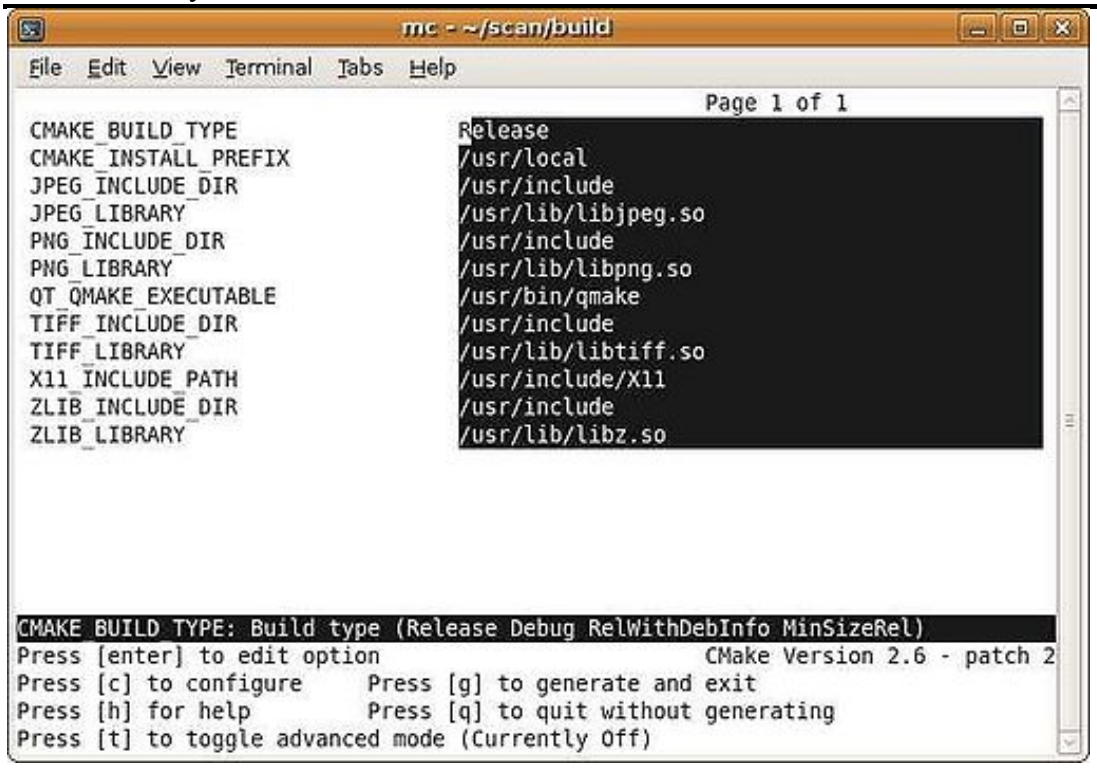
Если все прошло успешно, то ближе к концу вы увидите строки:

```
-- Configuring done
-- Generating done
```

В противном случае какие-то необходимые компоненты не были найдены. Что именно не было найдено, должно быть понятно из сообщений об ошибках. В некоторых случаях компонент имеется, но система сборки все равно не может его найти. В этом случае надо запустить CMake в интерактивном режиме:

```
ccmake .
```

И потом вручную указать пути к тем или иным не найденным файлом. ccmake - консольное интерактивное приложение, так что не помешают некоторые инструкции.



Предположим, что путь к библиотеке LibTIFF не был найден. В таком случае последовательность действий будет такая:

1. Клавишами со стрелками наводим курсор на строку TIFF_LIBRARY.
2. Enter.
3. Вводим правильный путь.
4. Enter.
5. [Правим таким же образом остальные ненайденные пути].
6. Жмем c
7. Жмем g

Когда CMake наконец отработал успешно, можно приступать к самой сборке:

```
make
```

Если сборка завершается с ошибками (не доходит до 100%), задайте вопрос на форуме.

Теперь остается сделать:

```
sudo make install
```

На этом этапе вас спросят ваш пароль, так как инсталляция требует прав рута.

На этом сборка и инсталляция закончены, и Scan Tailor готов к запуску. К сожалению, данный процесс инсталляции не предусматривает создания пункта меню, так что вам придется либо создать его самостоятельно, либо запускать его через Alt+F2 набрая там комманду "scantailor" (без кавычек).

10. Формат DjVu

Для эффективного кодирования DjVu необходимо хотя бы в общих чертах понимать его структуру.

В отличие от большинства других форматов упаковки изображений, формат DjVu использует одновременно несколько алгоритмов сжатия, наиболее оптимизированных каждый для своей задачи, размещая результаты сжатия в отдельных слоях изображения. В терминах DjVu такие слои называют чанками (chunks).

Слой **mask** используется для хранения информации о резко очерченных элементах изображения. Кодируется в высоком (300-600dpi) разрешении с помощью алгоритма JB2.

Слой **foreground** используется для хранения информации о цвете элементов, выделенных с помощью слоя mask. Кодируется в минимальным разрешении (порядка 25dpi) wavelet-алгоритмом IW44.

Слой **background** используется для хранения не резко очерченной части изображения (не вошедшей в слой mask). Кодируется в среднем разрешении (100-300dpi) wavelet-алгоритмом IW44.

Для объяснения структуры DjVu иногда пользуются следующей "некомпьютерной" аналогией технологии формирования изображения в формате DjVu, как технология рисования плаката художником-агитатором:

слой "background" - фон плаката, рисуется кистями среднего размера (например, небо и радуга)

слой "mask" - аккуратный трафарет для основного лозунга (или логотипа)

слой "foreground" - основной лозунг, накатывается широким валиком через трафарет - особой точности тут не нужно, через трафарет сложно промахнуться

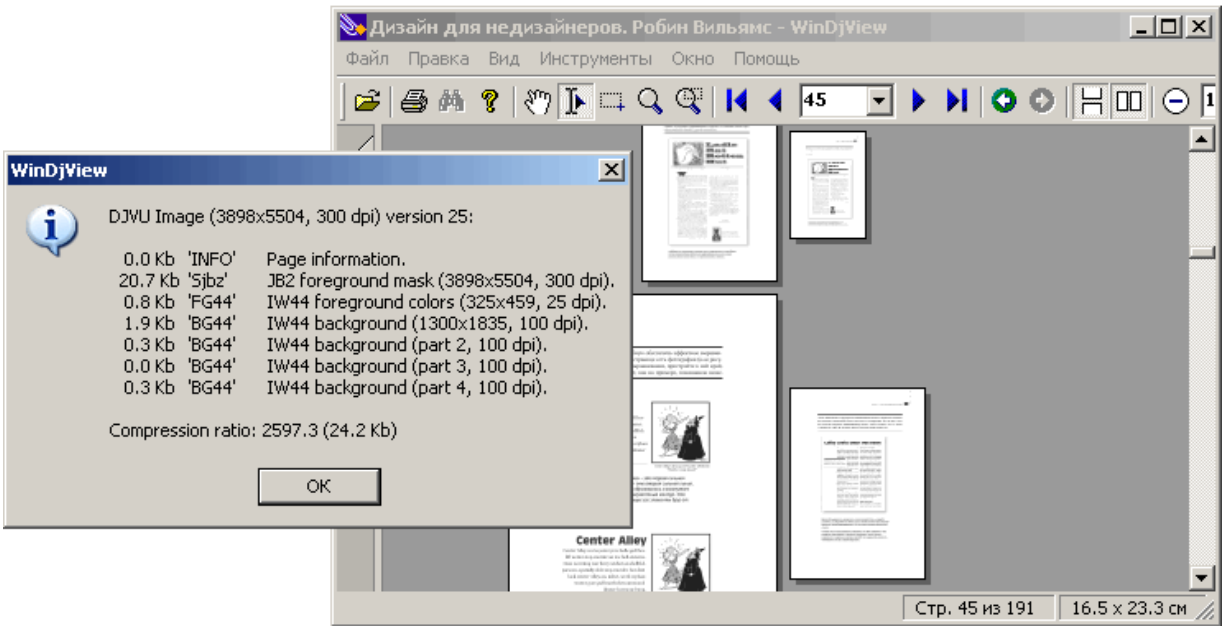
10.1 Словари djbz

Эти словари используются алгоритмом JB2 для замены похожих друг на друга элементов изображения (например, букв) одним элементом из словаря. Можно создавать один такой словарь для нескольких страниц (в предельном случае - один на всю книгу). Суммарный размер JB2-слоев книги при кодировании в режиме "один словарь на страницу" и в режиме "один словарь на книгу" для источника с хорошо проведенной бинаризацией (увеличение масштаба до 600dpi, сглаживание, буквы не касаются друг друга) может отличаться в несколько раз. Однако во втором случае затруднен просмотр книги через интернет (приходится дожидаться загрузки всей книги), кроме того значительно замедляется просмотр книг на слабых машинах. Обычно придерживаются золотой середины - один словарь на 20-50 страниц.

10.2 Просмотр информации о структуре djvu-файла с помощью WinDjView

С помощью пункта меню Вид -> Режим можно выбрать отображаемый слой djvu-документа. Цветной - все слои, Черно-белый - это mask, Передний план - mask, раскрашенный цветами foreground, Задний план - background. При экспорте страниц (соответствующей командой выпадающего меню) также сохраняется выбранный режимом слой.

Также много полезного можно узнать, выбрав пункт "Информация о странице" в контекстном меню страницы. Информация о странице, закодированной методом вклейки иллюстраций, выглядит следующим образом.



Обратите внимание, как "пляшут" размеры страниц книги на приведенном скриншоте. Это - результат неверного задания DPI.

Посмотреть размер используемых в djvu-файле словарей можно, например, с помощью пункта меню Файл -> Информация о документе.. в WinDjView. Если в появившемся окне установить галочку "Все файлы", то появившиеся в списке страниц файлы djbz и есть файлы словаря.

Кстати, с помощью данного окна можно легко определить самые проблемные в плане размера страницы вашего djvu-файла путем сортировки по столбцу "Размер".

10.3 Программы, используемые при DjVu-кодировании

10.3.1 Некоммерческие программы

10.3.1.1 Djvu Imager

Open source программа, позволяющая не только кодировать сканы в фото-качестве, но и вставлять полученные djvu-страницы как IW44-слой в djvu-файлы, оставляя JB2-слой нетронутым.

10.3.1.2 Djvu Libre

Open source комплект консольных утилит, часто используемых сторонними разработчиками для создания своих djvu-приложений. Имеются утилиты для кодирования в профилях photo и bitonal. Однако размер получающихся bitonal djvu завышен, т.к. словарь примитивов создается отдельно для каждой страницы.

10.3.1.3 Djvu Solo 3.1

Довольно древняя (2000 года выпуска), но зато бесплатная для некоммерческого использования (речь идет о noncom-версии) программа. Имеет встроенный автосегментор (на профилях scanned и clean), а также профили кодирования bitonal (один словарь примитивов на 10 страниц) и photo. Минимум настроек при кодировании (однако некоторые настройки можно изменять вручную путем правки .conf-файлов в папке profiles программы). Из недостатков - при пакетном добавлении нескольких страниц требуется ручная коррекция их последовательности (либо выделение добавляемых страниц начиная с последней). Не поддерживаются djvu-файлы последних поколений (например, с многоцветным слоем foreground).

10.3.1.4 Minidjvu

Open source программа, позволяет создавать bitonal djvu из монохромных файлов. Позволяет настраивать агрессивность алгоритма и размер словаря примитивов. Основной недостаток - низкая скорость работы.

10.3.2 Коммерческие пакеты программ и утилиты

В основном представлены продуктами компании LizardTech и ее преемника - компании Caminova. Эти продукты имеют максимально широкие возможности настроек кодирования, а также как правило более удобный пользовательский интерфейс. Существуют также утилиты для упрощенного кодирования, базирующиеся на входящих в коммерческие пакеты утилитах кодирования, а также djvu-принтеры.

Все эти продукты и большинство утилит предельно подробно описаны в статье Семейство программ Document Express от компании LizardTech для работы с файлами в формате DjVu. Общее описание, настройки, советы.

11. Классическая методика создания DjVu - кодирование всей книги в одном профиле

11.1 Профиль bitonal

используется при кодировании книг, не содержащих иллюстраций, или содержащих одноцветные без заливки, так называемые line-art, иллюстрации. Вся информация кодируется алгоритмом JB2. Слой background не создается. Результат кодирования корректно обработанных в ST (с повышением dpi и сглаживанием контура букв) практически идеален и не требует улучшения.

Иногда рекомендуют кодировать в данном профиле книги с растровыми иллюстрациями, предварительно обработанными диффузионным одноцветным алгоритмом. Однако, т.к. в результате такой обработки получают среднее качество иллюстраций, муар при просмотре и увеличенный размер djvu-файла, то к подобным рекомендациям следует относиться очень осторожно.

Кодирование в профиле bitonal книг с необработанными полутоновыми иллюстрациями является грубейшей ошибкой новичков. Иллюстрации в этом случае превращаются в черно-белые пятна, практически не содержащие информации.

11.2 Профиль photo

используется при кодировании источников, практически не содержащих текстовой части, например фотоальбомов или каталога репродукций. Вся информация кодируется алгоритмом IW44 и размещается в слое background.

Довольно часто данный профиль используется для кодирования обложек книг. Обычно при этом остальные страницы книг кодируется в профиле bitonal и потом объединяются с djvu-файлами обложки в Djvu-редакторах, либо утилитой djvm пакета Djvu Libre.

Основной недостаток кодирования в профиле photo - размер djvu (около 200-300 Кб на страницу), из-за которого результирующий размер книги при кодировании необработанных сканов может достигать сотен мегабайт. Однако для кодирования малостраничных журналов в djvu такой профиль вполне применим, хотя и не дает особых преимуществ в сравнении с кодированием в pdf.

Кодирование необработанной книги книги в профиле photo - неплохой выбор для новичка, опасающегося испортить книгу неграмотной обработкой. В случае необходимости такую книгу можно будет перекодировать оптимальным образом лишь немного потеряв в качестве.

11.3 Профили с алгоритмом автосегментации

например, **scanned** и **clean** в Djvu Solo, используются при кодировании журналов, комиксов и т.п., т.е источников, в который текстовая часть (или контуры рисунков) неотделимо связана с иллюстративной (фотографиями, рисунками). Алгоритм автосегментации сам определяет, какую часть изображения в какой слой поместить. Дает минимальный размер результирующего djvu, но результат сильно зависит от интеллектуальности автосегментора и предварительной обработки сканов.

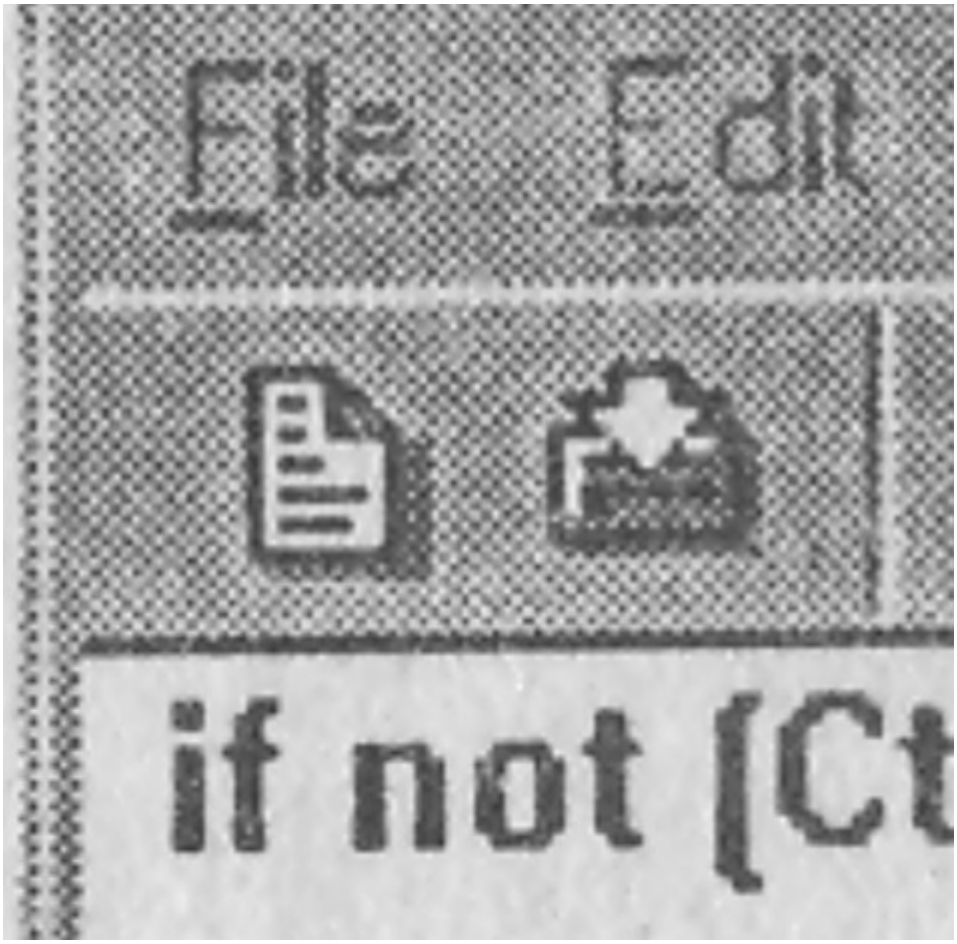
Кодирование сканов с растровыми иллюстрациями в профиле, использующем автосегментацию приводит к получению djvu приемлемого, благодаря предобработке в CT, но не идеального качества. Проблемы обработки растровых иллюстраций подробно рассмотрены ниже.

11.4 Проблемы классической методики

11.4.1 Проблемы растра на изображении

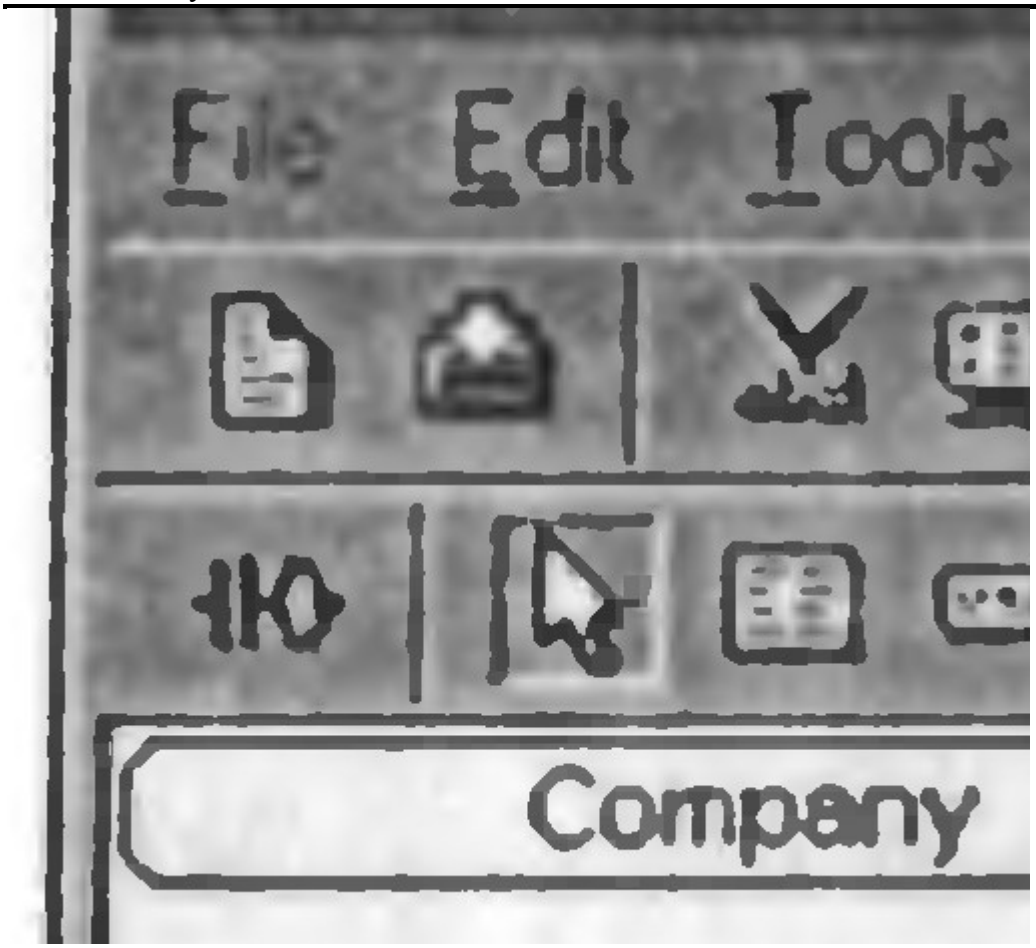
Как правило, технология создания книг такова, что изображения печатаются с разрешением, значительно меньшим, чем разрешение текста. В результате на скане буквы текста выглядят как

равномерно залитые области, а участки с изображениями представляют собой матрицу точек с резко отличающейся яркостью - растр. Это порождает несколько проблем для djvu-кодера. Т.к. точки достаточно контрастны, то при кодировании JB2-алгоритмом они определяются как элементы текста, увеличивая размер словаря примитивов. Для вейвлетного алгоритма IW44 такие точки также представляют проблему. В конечном счете все это приводит к дополнительному увеличению размера итогового файла и ухудшению его качества.



11.4.2 Дефекты при автосегментации djvu-кодировщика

Известно, что djvu-кодировщик использует различные подходы к обработке текстовых зон и зон с изображениями. В первом случае он использует алгоритм сжатия чёрно-белых изображений JB2, во втором - вейвлетный алгоритм сжатия фона IW44. К сожалению, информация о зонах, сформированная вами в процессе подготовки сканов в CT пока что никак не распознается djvu-принтерах, поэтому они пытается сами распознать зоны с изображениями, используя те или иные алгоритмы автосегментации. В случае, если на входе такого алгоритма текст или line-art рисунок, можно, выбрав соответствующий профиль кодирования вообще исключить создание фона. Однако, если на входе - растровый рисунок, то точки растра затрудняют работу алгоритма, в результате чего часть рисунка, содержащая текст или полезные детали, могут переместиться в фон и "раствориться" там. Кроме того многие точки растра переместятся на передний план, создав характерный шум.



11.4.3 Выводы

Таким образом, для создания качественных djvu-файлов из сканов с растровыми иллюстрациями необходимо решить основную проблему - избавиться от растра. Дальнейшая обработка зависит от содержимого исходных сканов и использованных методов борьбы с растром.

Если использовались неадаптивные алгоритмы, например фильтр Гаусса, обычно используемый при обработке фотоиллюстраций, то с позиций сохранения качества как правило лучше вставить иллюстрацию целиком в фон используя метод вклейки иллюстраций.

Если использовались адаптивные (практически не затрагивающие резкие участки изображения) алгоритмы, например интеллектуальное размытие, обычно используемое при обработке частично-растровых иллюстраций, например, чертежей, диаграмм или скриншотов, то можно попробовать положиться на автоматическую сегментацию. Если результат кодирования после автоматической сегментации окажется неудовлетворительным, всегда можно вклеить иллюстрации по нижеприведенной методике.

В случае, когда отделить иллюстрации от текста не представляется возможным, например, при обработке богато иллюстрированных журналов, использование кодирования в профиле с автоматической сегментацией - это практически единственный способ получить более-менее качественные djvu приемлемого размера.

При использовании метода вклейки иллюстраций необходимо учитывать, что для богато иллюстрированных книг он дает увеличенный объем djvu-файла в сравнении с djvu-файлом, закодированном с использованием автоматической сегментации, практически нивелируя основное преимущество формата djvu перед pdf.

12. Кодирование фотоиллюстраций

12.1 Методы борьбы с растром

Простейший способ борьбы с растром - уменьшить разрешение картинки до такой степени, чтобы оно сравнялось с разрешением растра. На уровне кодирования это делается через варьирование степени сжатия алгоритма IW44 (параметр Качество задн. фона в Djvu Imager).

Несколько лучший результат дает фильтр Гаусса. Применяемый радиус размытия зависит от соотношения между разрешением сканирования и разрешением растра. Если они равны, то достаточно будет радиуса в 1-2 пикселя. Если разрешение сканирования в 2 раза больше, придется использовать радиус от 3 до 5 пикселей. Обязательно необходимо визуально контроллировать поиск оптимального радиуса размытия. Лишнего размытия по понятным причинами следует избегать.

Если иллюстрации частично-растровые, т.е. на растровых располагается, например, текст или линии, не имеющие растровой структуры (такое часто можно видеть, например, на скриншотах), то предложенные выше способы могут оказаться непригодными, т.к. заметно снижают четкость таких нерастровых элементов.

Более корректный способ обработки частично-растровых иллюстраций - использование адаптивных алгоритмов - т.е. алгоритмов, сглаживающих только участки с близкими значениями яркости.

Интересные результаты дает *Выборочное гауссово размывание* в GIMP. Если требуется еще более качественный результат, то стоит обратить внимание на алгоритм *Интеллектуальная размытость* комерческой программы Corel PHOTO-PAINT.

Максимального же качества при удалении растра можно достичь с помощью платного плагина *Sattva Descreen* для платного же Photoshop'a, только нужно помнить, что обработка с его помощью будет более эффективной, если производить ее до обработки в ST.

Если применение предложенных выше методов привело к ухудшению резкости элементов изображения, можно после размытия воспользоваться эффектами, увеличивающими резкость картинки.

Как правило, обработка фильтрами уменьшает контрастность и без того не очень контрастной иллюстрации. Для компенсации этого эффекта следует увеличить контрастность результата обработки либо вручную (при этом очень удобно задавать границы яркости с помощью черной и белой пипетки), либо автоматическими средствами, например командой *Цвет -> Авто -> Увеличить контраст* в GIMP или *Настройка -> Автонастройка* в Corel PHOTO-PAINT.

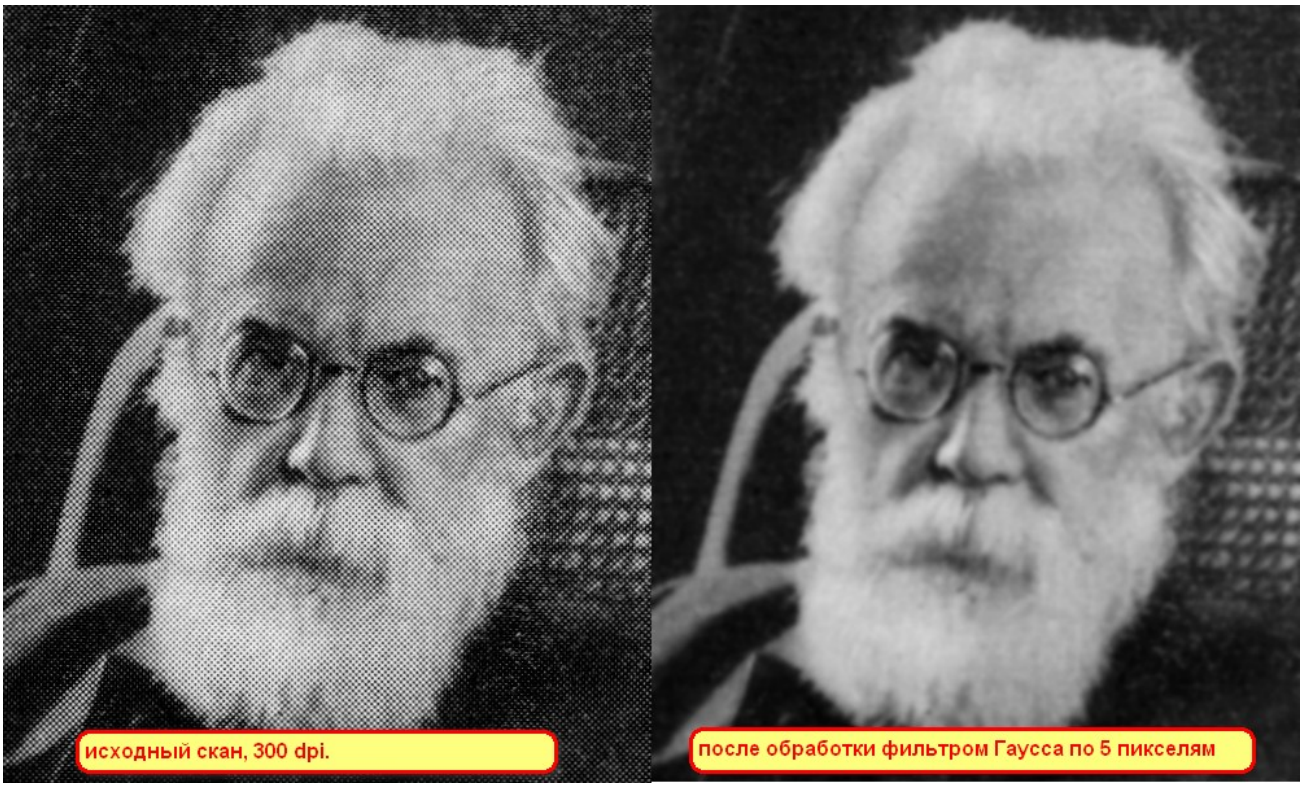
Любой вид обработки в современных редакторах можно автоматизировать:

Групповая обработка в Gimp делается с помощью скриптов на языке Script-Fu.

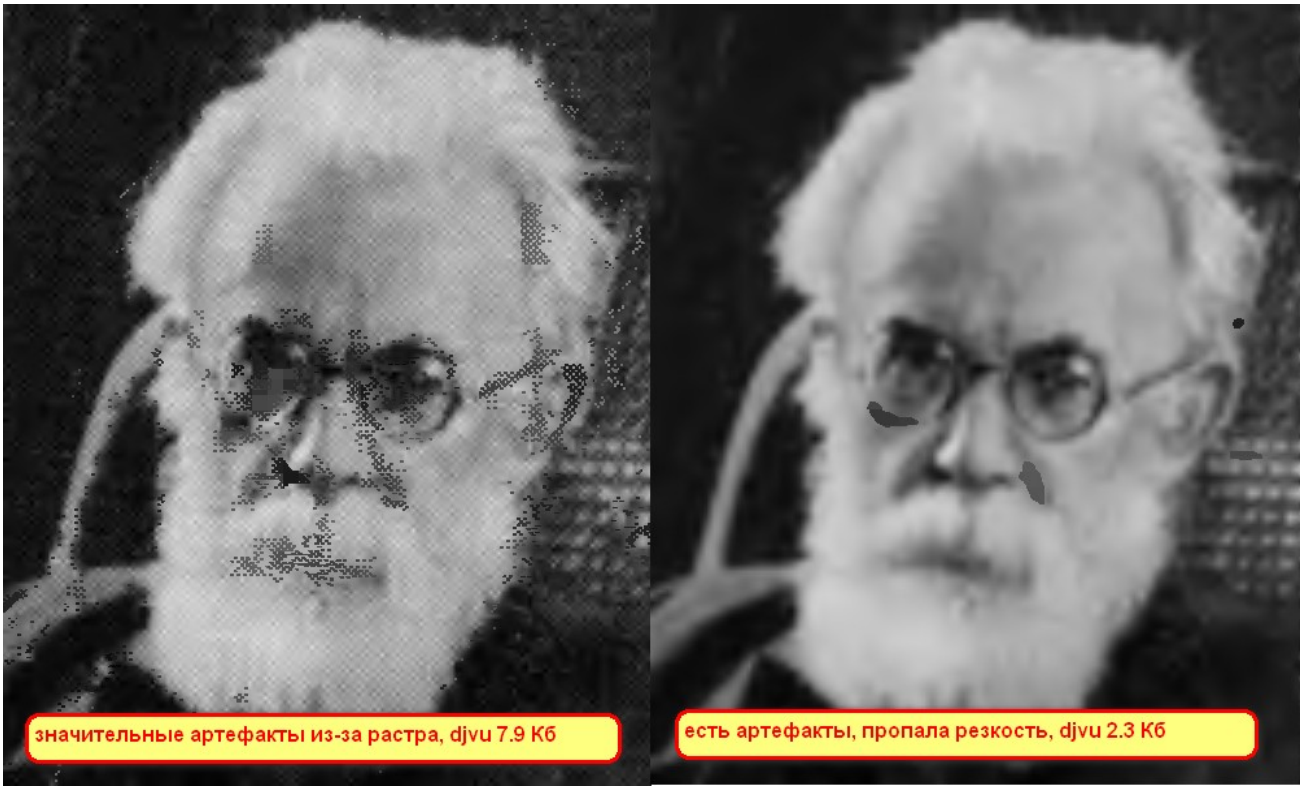
В Corel PHOTO-PAINT аналогичного результата можно добиться, записав нужную последовательность обработки команд в скрипт (Окно -> Окна настройки -> Запись) и воспроизведя ее для требуемого набора файлов (Файл -> Пакетная обработка), существует также VBA-макрос, немного упрощающий процедуру пакетной обработки в Corel для наиболее типичных действий - фильтр гаусса, интеллектуальное размытие, автонастройка.

В Adobe Photoshop используют Actions.

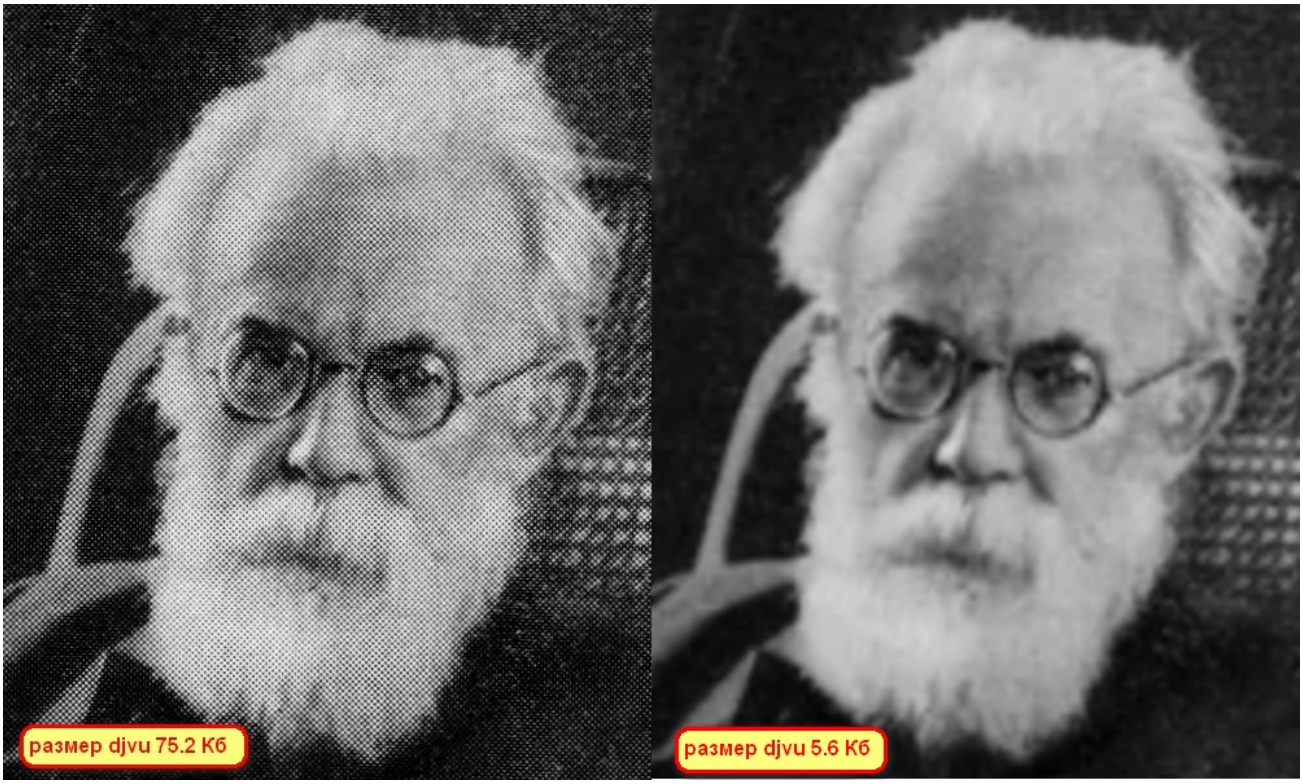
12.2 Пример фотоиллюстрации до и после удаления растра



Кодирование с профилем scanned исходного скана и скана с удаленным растром



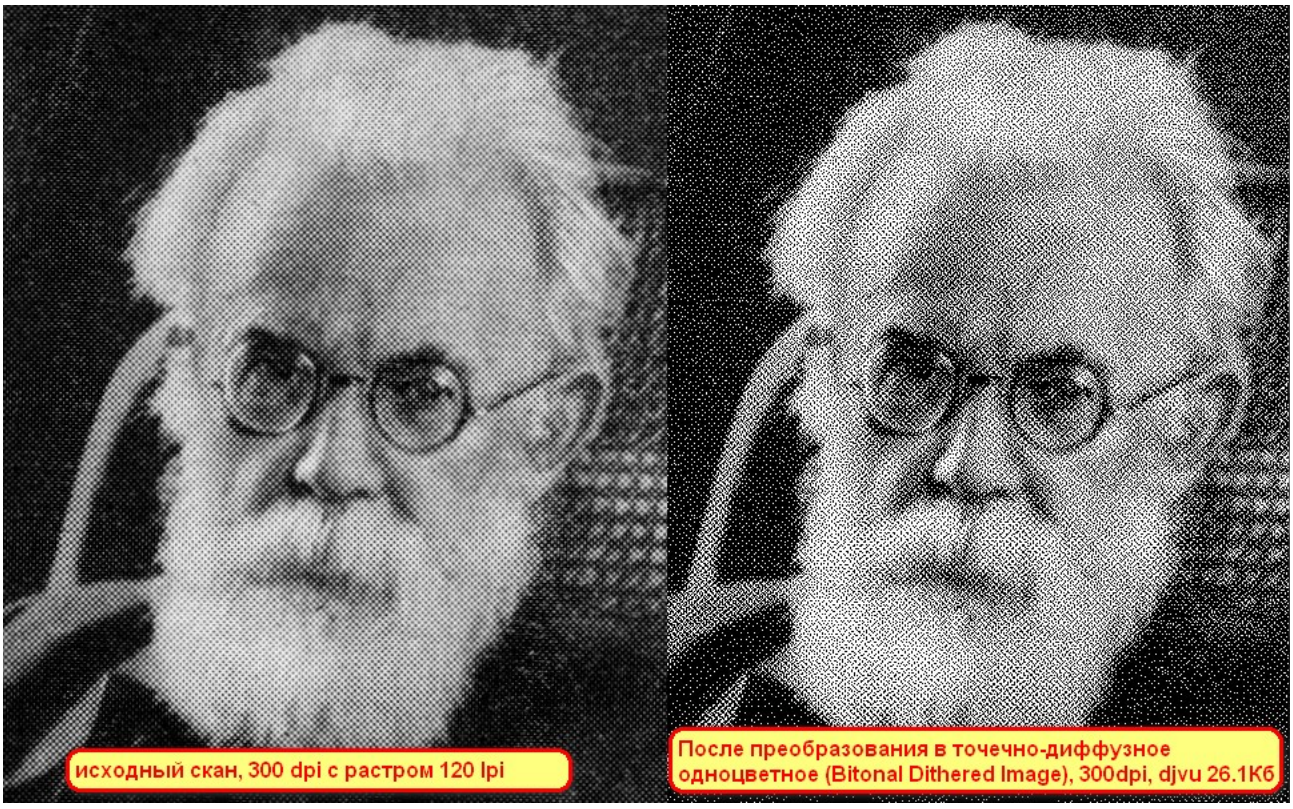
Кодирование с профилем photo исходного скана и скана с удаленным растром



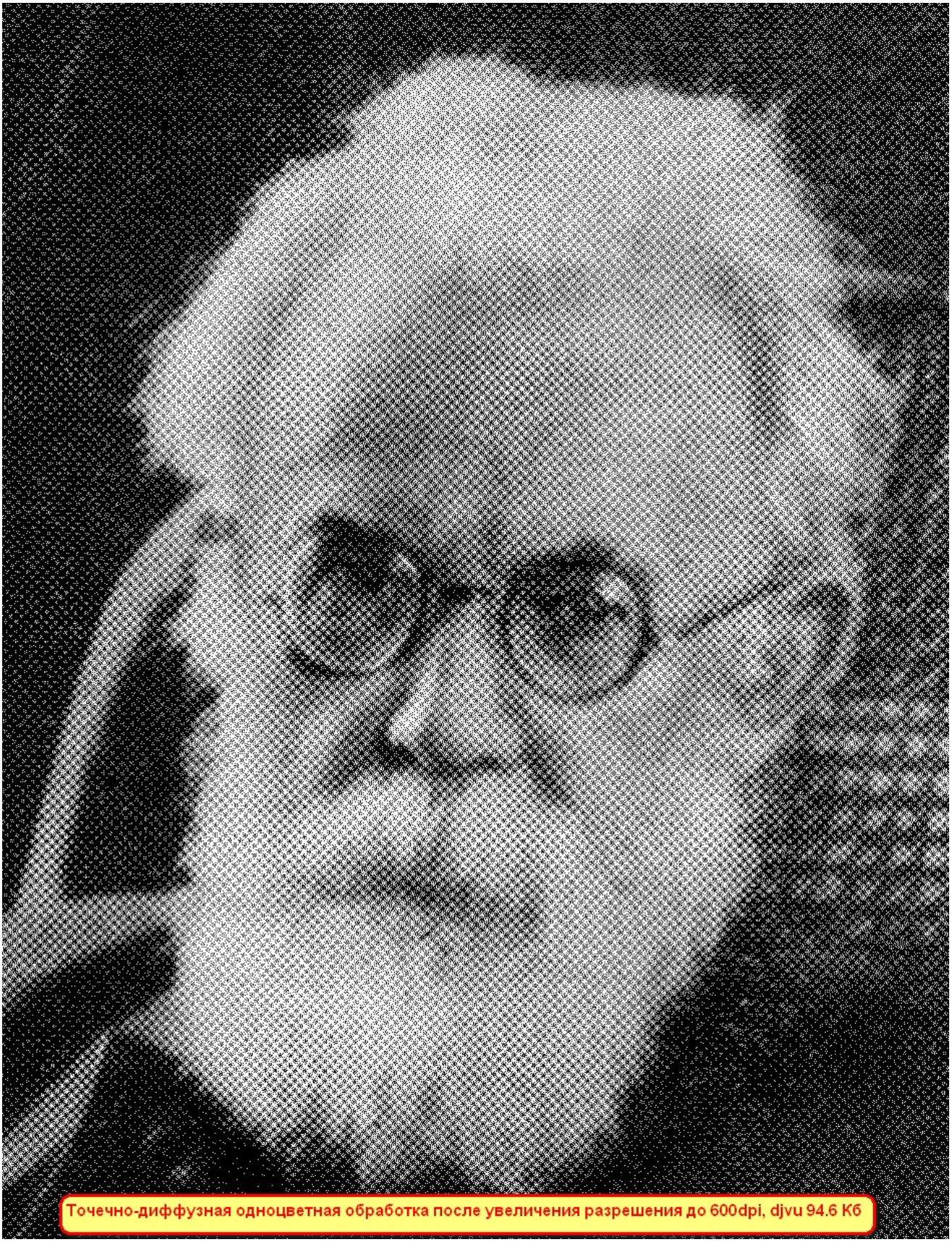
Вывод Все достаточно очевидно - избавляемся от растра и кодируем в профиле photo.

12.3 Образец иллюстрации, обработанной диффузионным одноцветным алгоритмом

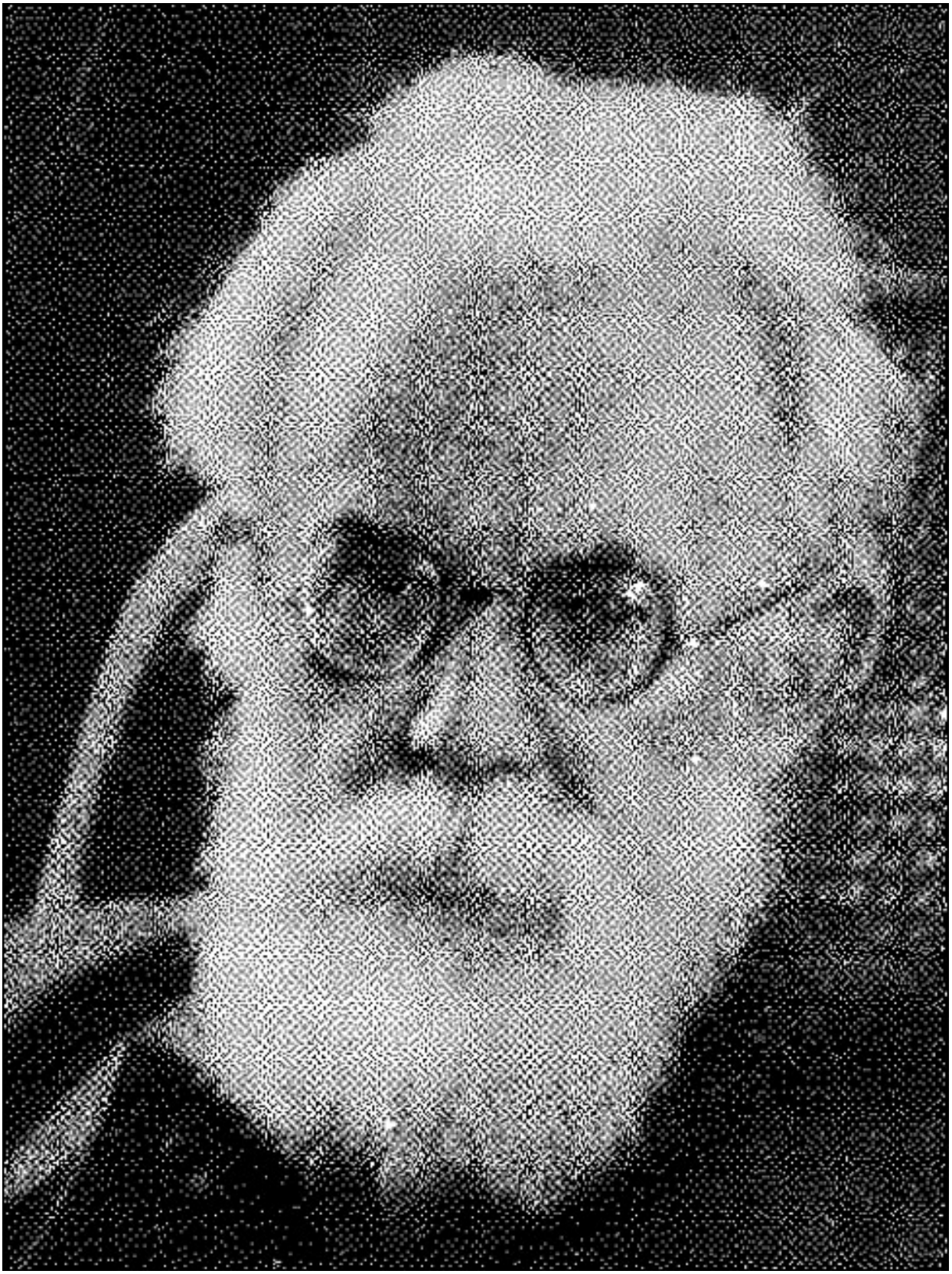
Обработка без увеличения разрешения



Обработка с увеличением разрешения до 600dpi



Образец муара при просмотре в WinDjView

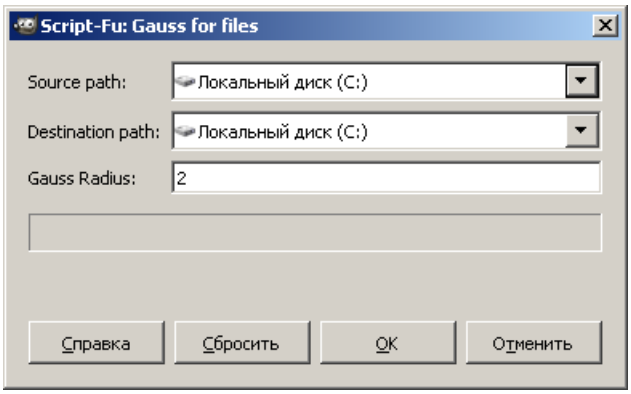


Выводы:

С учетом того, что кодирование в профиле photo той же иллюстрации, но обработанной фильтром Гаусса даст результирующий размер в 5 раз меньший при лучшем качестве и принципиальном отсутствии муара, то вывод однозначен - обработку иллюстраций диффузионным одноцветным алгоритмом применять не следует.

12.4 Фильтр Гаусса

Ниже предложен вариант групповой фильтрации всех tif-файлов в папке методом гаусса с помощью скрипта на языке script-fu. Текст скрипта нужно вставить во вновь созданный текстовый файл с именем "gauss-for-files.scm", который необходимо положить в папку "C:\Program Files\GIMP-2.0\share\gimp\2.0\scripts". После перезапуска GIMP появится пункт меню "Xtns/Utils/Gauss for files..." (Фильтры/Utils/Gauss for files...), при выборе которого появится следующее окно



Итак, ниже сам скрипт:

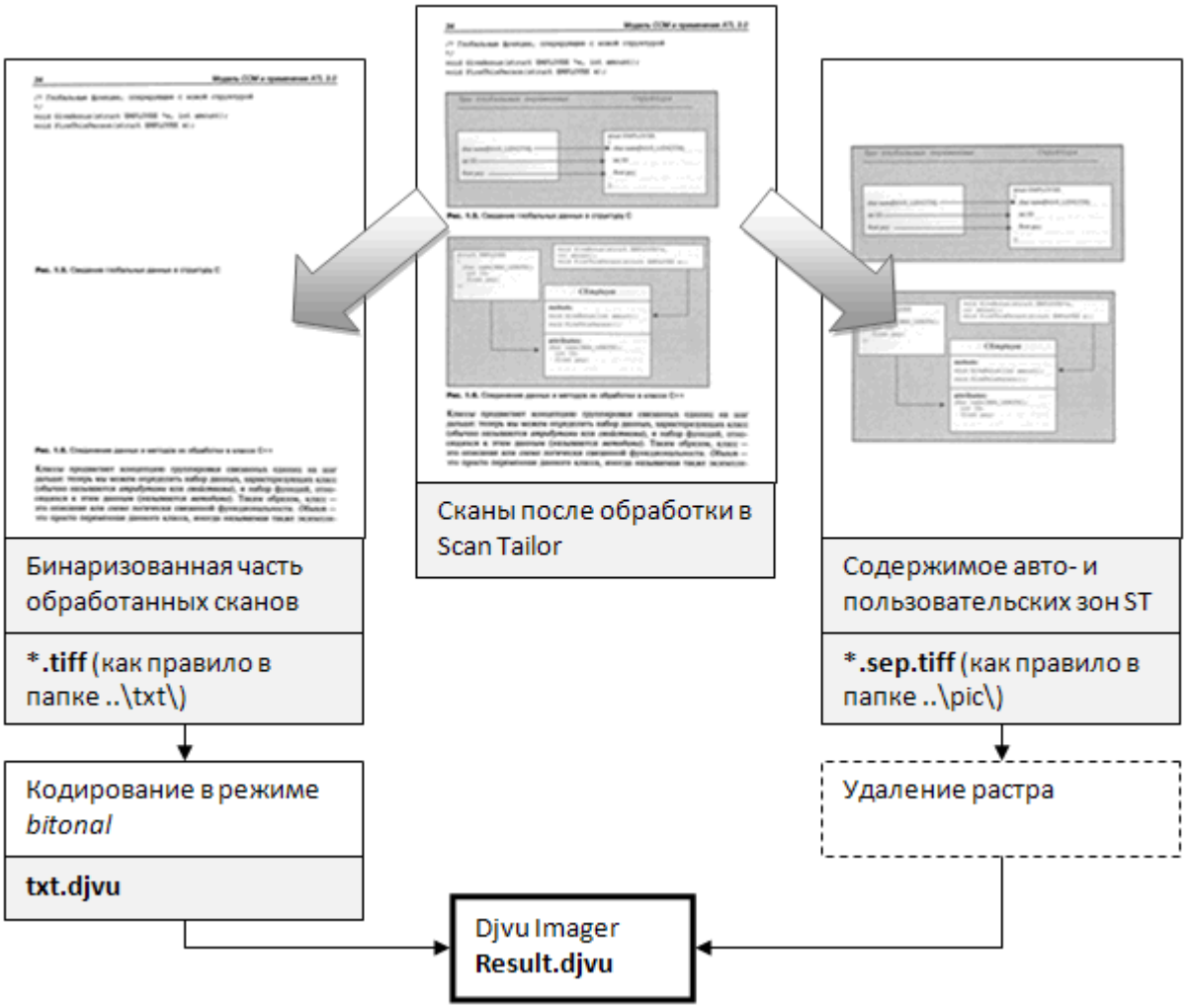
```
(define (script-fu-gauss-for-files source dest radius)
(let* ((filelist (cadr (file-glob (string-append source "\\*.tiff") 1))))
(while (not (null? filelist))
(let*
(
(filename (car filelist))
(image (car (gimp-file-load RUN-NONINTERACTIVE filename filename)))
(drawable (car (gimp-image-get-active-layer image)))
(cname "")
)
(gimp-layer-flatten drawable)
(plug-in-gauss RUN-NONINTERACTIVE image drawable radius radius 0)
(set! cname (string-append dest (substring filename (string-length source) (string-length filename))))
(file-tiff-save RUN-NONINTERACTIVE image drawable cname cname 1)
(gimp-image-delete image)
)
(set! filelist (cdr filelist))
)
)
)

(script-fu-register "script-fu-gauss-for-files"
"<Toolbox>/Xtns/Utils/Gauss for files..."
"Gauss filter"
"Manuele J Sarfatti"
"2009 by Andrey S Stolyarov; GNU GPL"
"November 9, 2009"
""
SF-DIRNAME "Source path" "C:\\"
SF-DIRNAME "Destination path" "C:\\"
SF-VALUE "Gauss Radius" "2"
)
```

13. Создание качественных DjVu методом вклейки иллюстраций

13.1 Общее описание метода

Вместо автоматической сегментации предлагается использовать ручное разделение слоя текста и иллюстраций с последующей вставкой иллюстраций в слой background. Текстовая часть и line-art иллюстрации располагаются в слоях mask и foreground.

Алгоритм такого разбиения следующий (предполагается, что сканы уже обработаны и разбиты на составляющие - текстовая составляющая располагается в папке \txt\, составляющая с иллюстрациями - в папке \pic\):



1. Формируем txt.djvu любым djvu-кодером, используя профиль кодирования bitonal, либо программой minidjvu, указав кодеру в качестве входной папки папку \txt\
2. Избавляемся от растра и при необходимости повышаем контрастность изображений из папки \pic\
3. Переименовываем все файлы, расположенные в \pic\ следующим образом *.tif -> *.sep.tif (при разделении ST Separator'ом этот шаг не нужен).
4. С помощью программы DjVu Imager кодируем в профиле photo содержимое папки \pic\ и вставляем результат в черно-белый txt.djvu, получаем Result.djvu

13.2 Особенности метода

Для иллюстраций при разделении в большинстве случаев следует задать разрешение, равное и меньшее, чем разрешение сканирования (как правило, 300 dpi). Такая возможность осуществляется штатными средствами, заложенными как в CTA, так и в ST Separator. Разрешение текстовых субсканов при этом остается равным 600 dpi.

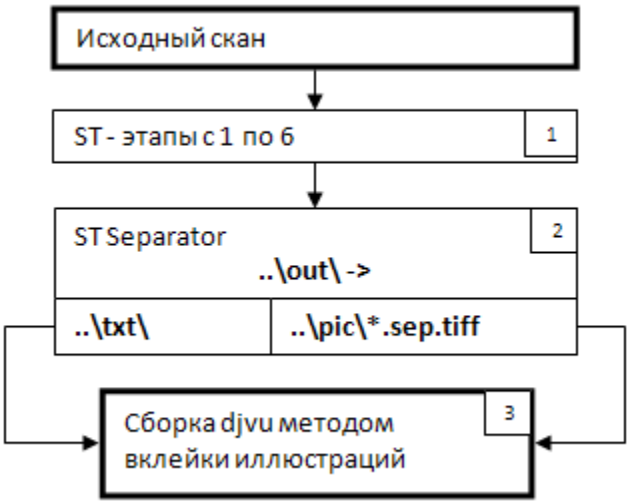
Приведенные далее особенности касаются только случая разделения с помощью STA. ST Separator автоматически их учитывает.

1. Пустые (белые) страницы из папки \pic\ можно (и нужно) удалять, т.к. информации они не несут, а 1-2 КБ в слое background создают.
2. При кодировании составляющих с различным разрешением необходимо следить за кратностью пиксельных размеров соответствующих файлов в папках txt и pic. В случае несоблюдения условия кратности лишние пиксели необходимо обрезать (например, с помощью настройки вкладки "Обрезать" при пакетном преобразовании в FastStone Image Viewer).
3. Для группового переименования файлов удобно использовать соответствующие инструменты Free Commander (для запуска инструмента выделите в Commander'e несколько файлов и нажмите Ctrl+M). Также инструменты группового переименования встроены в популярные просмотрщики графических файлов - IrfanView и FastStone Image Viewer.

13.3 Реализация метода

Используется предложенное U235 и введенное во всех режимах вывода CT начиная с версии 0.9.8.1 ограничение яркости иллюстраций диапазоном [1..254] (это меньше, чем на 1% - т.е. разница между иллюстрацией до и после такого ограничения практически неразличима). Таким образом, пиксели с нулевой и стопроцентной яркостью теперь однозначно относятся к бинаризованной части скана.

Вкратце алгоритм таков:



1. Проходим все этапы ScanTailor, обрабатывая страницы с текстом и иллюстрациями в смешанном режиме.
2. Разделяем каждый файл из выходной папки ScanTailor на два файла - с текстовой и графической информации утилитой ST Separator. На выходе получаем две заполненные файлами папки - \txt\ с текстовой составляющей, и \pic\ - с графической.
3. Собираем final.djvu методом вклейки иллюстраций

Достоинства

1. Метод применим ко всем версиям ScanTailor, начиная с 0.9.8.1
2. Пользователям, использующим стандартный метод работы в ST нет необходимости менять привычный подход к работе в программе
3. Вывод в ST происходит в один проход, большинство ошибок автоопределения зон видны сразу же и исправляются в процессе обработки в ScanTailor

Недостатки

1. Необходимость дополнительной обработки изображений внешней утилитой
2. Увеличенные требования к объему свободного места на жестком диске. Для уменьшения влияния данного недостатка предназначена заложенная в ST Separator возможность выводить иллюстрации в уменьшенном разрешении

Примечания

Утилиту ST Separator (следует учесть, что она требуют установленного .NET 2.0) можно скачать здесь

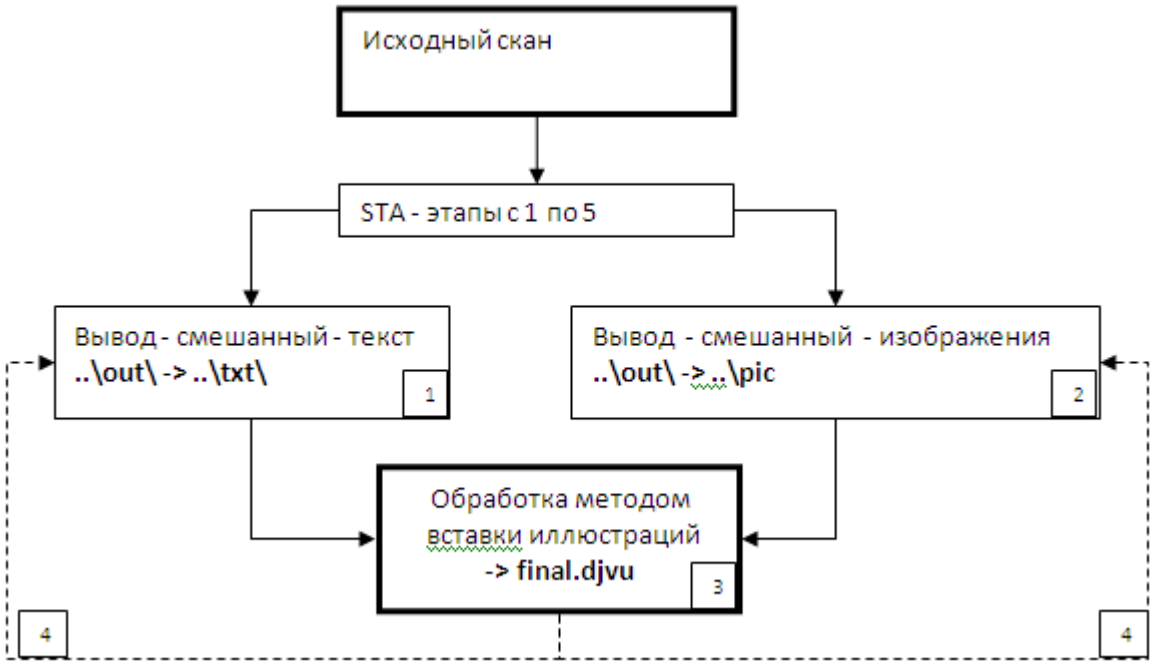
Аналог ST Separator с базовым функционалом, зато не требующий .NET 2.0 - ST Split.

На базе Scan Tailor версии 0.9.7.1 существует патч от anagnost96 (STA), позволяющий создавать субсканы текста и иллюстраций средствами Scan Tailor. Реализация метода вклейки иллюстраций с использованием STA

13.4 Реализация метода вклейки иллюстраций с использованием STA

Используется Scan Tailor версии 0.9.7.1, пропатченный патчем от anagnost96 (далее - STA), самая свежая ссылка на который находится в шапке топика обсуждения Scan Tailor на руборде. Прямая ссылка на патч для версии 0.9.7.1.

Вкратце алгоритм таков:



1. Прогоняем исходные сканы в STA, вывод делаем в 600dpi, в смешанном режиме, в подрежиме только текст, копируем результат работы из папки \out\ в папку \txt\
2. Прогоняем вывод STA в родном разрешении скана (обычно 300 dpi), в смешанном режиме, в подрежиме только изображения, копируем результат работы из папки \out\ в папку \pic\
3. Собираем final.djvu методом вклейки иллюстраций
4. Анализируем полученный djvu, если обнаружены, например, ошибки выделения зон, то исправляем их и проходим этап 2 и последующие заново (это займет меньше времени, так как можно обрабатывать только исправленные файлы).

Достоинства

1. Не нужны дополнительные внешние утилиты, кроме пропатченного Scan Tailor
2. Возможность задания различных значений dpi для каждого из режимов вывода
3. Минимальные требования к объему свободного пространства на жестком диске

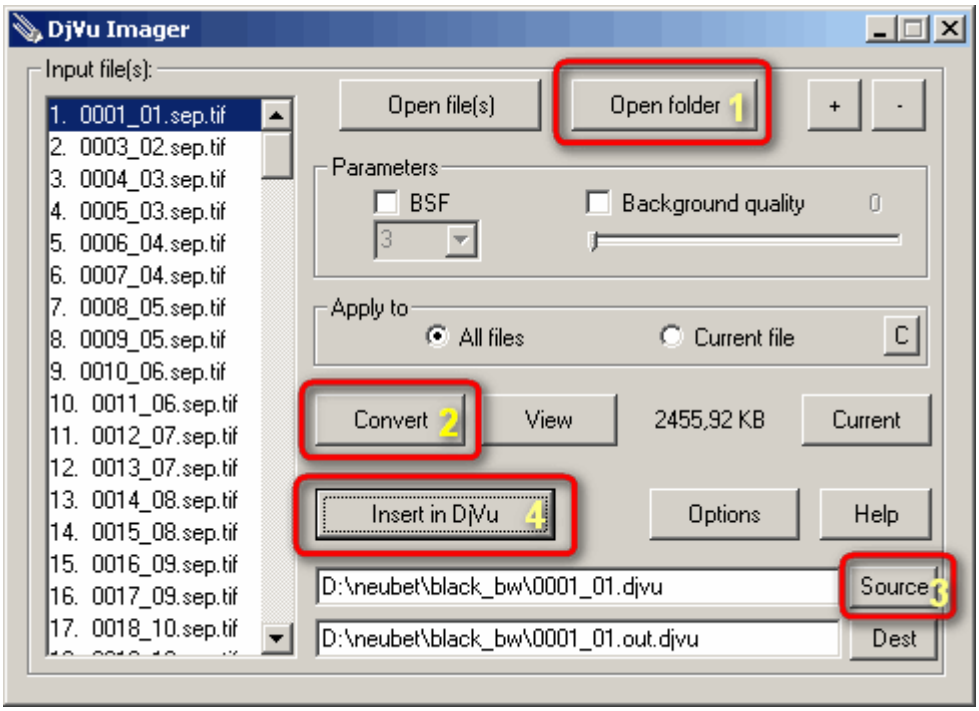
Недостатки

1. Многопроходность - метод изначально ориентирован на проход в два этапа, а в случае ошибок выделения зон количество этапов возрастает
2. Недостаточная наглядность - из-за того, что изображения и текст собираются вместе только в final.djvu становится затруднительной проверка работы механизма автоматического определения зон, что чревато дополнительными проходами. Для борьбы с данным недостатком можно использовать предварительный вывод с низким разрешением (например, 150dpi) в подрежиме текст и изображения.

3. Необходимость в дополнительной работе по перемещению файлов из папки out в дополнительные папки
4. Необходимость переделывать патч в случае изменения кода стадии Вывода, что может ограничить (и ограничивает) применение патча к новым версиям ScanTailor

13.5 Вклейка иллюстраций с помощью DjVu Imager

Вклейка иллюстраций с помощью Djvu Imager осуществляется в 4 шага:



1. Подключаем папку \pic\.
2. Формируем для каждого файла данной папки временные djvu-файлы для вклейки в черно-белый djvu (по файлу на страницу).
3. Указываем местоположение файла bw.djvu.
4. Вклеиваем djvu с иллюстрациями поверх соответствующих черно-белых страниц bw.djvu.

Важно:

1. Имена файлов в \pic\должны соответствовать именам файлов, из которых собирался bw.djvu, **плюс добавлен суффикс "sep"**. Т.е. имя файла 1.djvu должно превратиться в 1.sep.djvu.
2. Размер изображений, из которых собирался bw.djvu должен быть **точно кратен размеру изображений** в \pic\. При выводе под разными разрешениями в Scan Tailor данное условие иногда не выполняется. В случае необходимости можно, например, воспользоваться инструментом "Обрезать" в "Расширенных настройках" инструмента "Пакетное преобразование" программы FastStone Image Viewer.
3. Djvu Imager по умолчанию прописывает полностраничным иллюстрациям DPI=100. Поэтому необходимо перед кодированием установить в настройках Djvu Imager значение DPI, соответсвующее реальному DPI кодируемых субсканов.

14. Дальнейшая обработка готового djvu-файла

14.1 Добавление OCR-слоя

Для добавления OCR-слоя и максимальной автоматизации последующих этапов обработки текст книги необходимо распознать. Из бесплатных программ распознаванием занимается CuneiForm, вставить распознанный данной программой текст в djvu-файл можно утилитой CuneiDjvu.

Коммерческие программы djvu-кодирования, например Document Express Editor, также предлагает при кодировании распознать и добавить в djvu OCR-слой, однако качество результата сравнимо, если не хуже, чем достижимое с помощью CuneiForm.

Наилучшие результаты распознавания дает платная программа FineReader, и именно под него "заточена" программа DjvuOCR, с помощью которой обычно вставляют OCR-слой. К сожалению, под Windows связка FineReader + DjvuOCR - это пока что единственный доступный способ создать качественный OCR-слой.

Также можно отметить, что во многих случаях не обязательно распознавать всю информацию на страницах - достаточно распознать только текстовую составляющую. Это, кроме уменьшения времени распознавания, значительно уменьшит требования к свободному месту на жестком диске. Главное - чтобы оставалось точное соответствие между последовательностью страниц, подаваемых на вход системы распознавания и последовательностью страниц в книге. Для уже готовой книги экспорт только текстовой части страниц можно сделать, например, с помощью WinDjView, перейдя перед экспортом страниц в режим отображения "Черно-белый" (Вид -> Режим -> Черно-белый).

14.2 Распознавание текста с помощью FineReader

14.2.1 Замечания по версиям FineReader

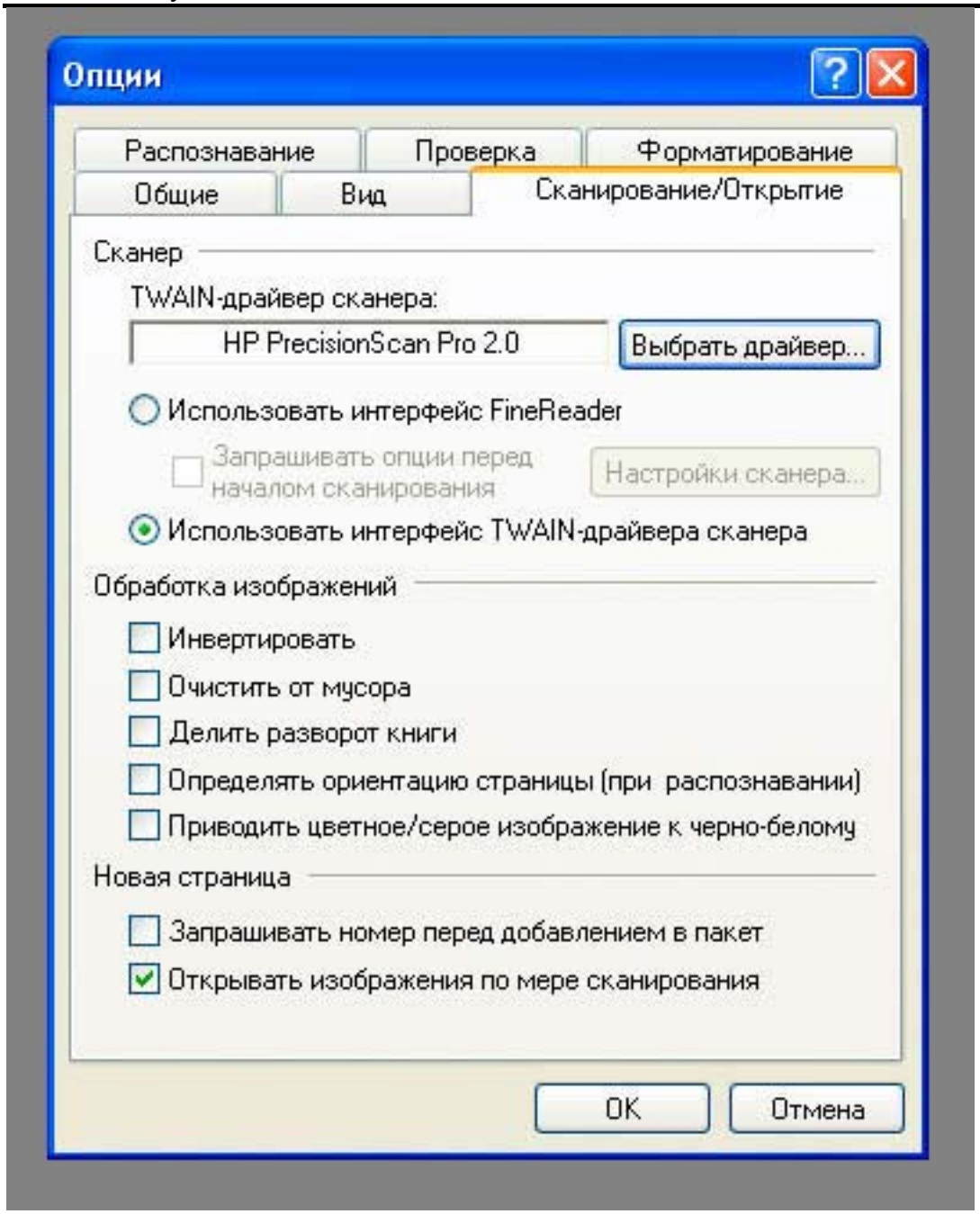
Максимально корректная работа DjvuOCR происходит для версий по восьмую включительно.

FR9 научился использовать многоядерные процессоры, однако увеличение скорости распознавания омрачается проблемой перестановки страниц. Подробнее об этой проблеме можно прочитать в ветке DjvuOCR руборда. Впрочем, нумерация легко исправляется простейшей программой, ссылку на которую можно найти там же.

FR10 с программой DjvuOCR несовместим.

14.2.2 Рекомендации от twdragon

Первое, что нужно сделать - зайти в диалог опций пакета, и сбросить там все флажки на вкладке Сканирование/Открытие в группе Обработка изображений.

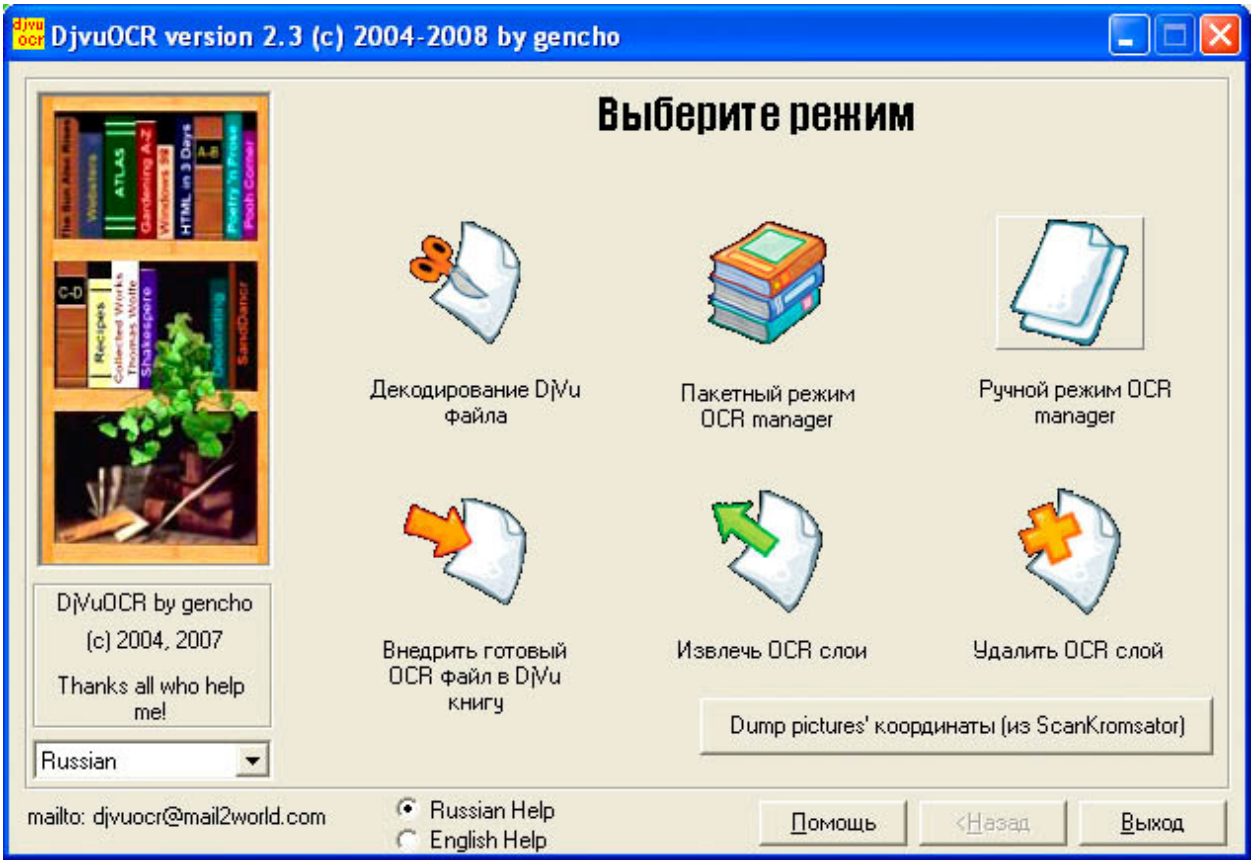


После этого нужно переместить куда-нибудь в известное место сам пакет, чтобы потом легко найти его. Я предпочитаю сохранять в папку, куда выводил изображения страниц ScanKromsator. Когда страницы открыты, можно сразу запускать распознавание.

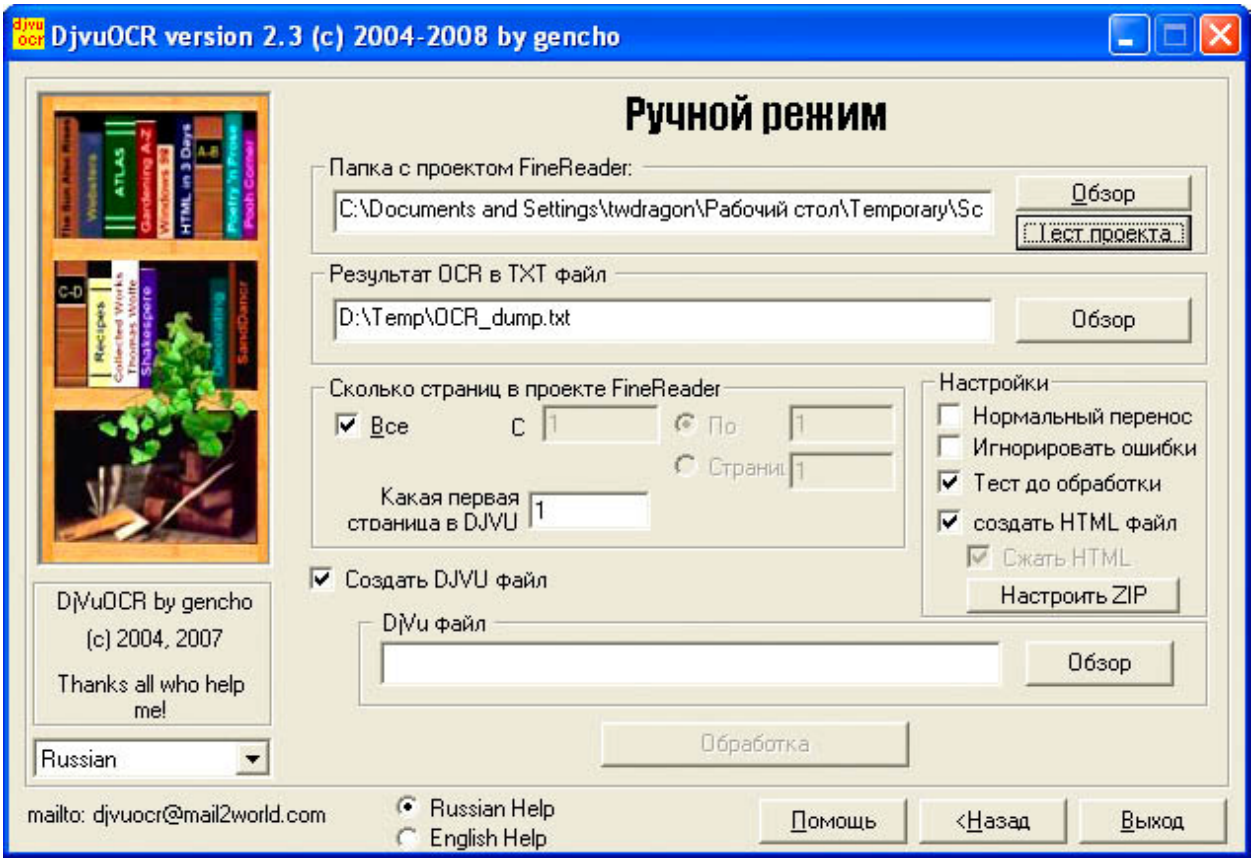
Первичная вычитка в FineReader сводится к легкой коррекции самых заметных ошибок. Главное правило при работе - если вы собираетесь сохранять файл в DjVu, ни в коем случае не удаляйте знаки переноса строки и концевые дефисы абзацев! Тогда внедрить текстовый слой в DjVu-файл можно будет легко и быстро, и не возникнет проблем при модификации готовой книги.

14.3 Добавление OCR-слоя с помощью программы DjvuOCR

Основное окно DjvuOCR выглядит так:



Выбираем режим "Ручной OCR manager"



Здесь нужно указать адрес папки пакета FineReader с распознанной книгой, номер первой страницы пакета в файле DjVu, а также имя самого файла DjVu. Флажок "Создать" не должен

пугать - на самом деле, в существующий файл DjVu просто будет записан невидимый слой с текстами и координатами строк. Когда все параметры заданы, запускаем обработку. Через относительно небольшое время получаем djvu-файл с внедренным в него текстовым слоем.

14.4 Добавление гиперссылок в оглавлении и предметном указателе

Достаточно автоматизированно можно добавлять гиперссылки с помощью программы Djvu Hyperlinks Editor. Однако развитие программы дошло до версии 0.8 и, по-видимому, остановилось. Кроме того, перед внедрением в djvu OCR-слой на страницах, подвергаемых обработке Djvu Hyperlink Edtitor должен быть тщательно вычитан на отсутствие ошибок, особенно это касается ссылок на номера страниц в оглавлении и предметном указателе.

Тем не менее - вот инструкция по работе с DjvuOCR и DjvuHyperlinks Editor, расположенная на сайте infanata.org.

14.5 Вставка оглавления

В плагине LizardTech/Caminova, , на боковой панели WinDjView и DjVuLibre DjView можно отображать встроенное оглавление документа.

Если вас не смущает необходимость работы в системе Windows с установленным .NET Framework 2.0, то отдельное (так называемое bookmark-) оглавление удобнее всего делать с помощью программы Djvu Bookmarker, скачать последнюю версию которой можно здесь, а почитать документацию - здесь.

Также для любых действий со внедренным в djvu содержимым служит утилита djvused.exe, однако она оперирует восьмеричным представлением UTF8, поэтому напрямую работать с таким содержимым без специализированного ПО не представляется возможным.

Кроме того, можно создать html-оглавление в специальном формате и вставить его в djvu-книгу утилитой Djvu Bookmark Tool 2.0.