

# Datalogics

## Content Extraction From PDF Files

***Matt Kuznicki  
Chief Technical Officer  
Datalogics***

***PDF Days – December 2014  
PDF Association***



# Outline

Image Files as Document Representations

Advantages of PDF for Content Preservation

Extracting Content: Examples

Common Pitfalls

Summary



# In the Beginning... We Had Pictures

Image file formats have a long history in archiving and document management workflows:

- Images are the native format of scanners and digital cameras
- Image files are easy to produce and consume – little interpretation is needed in most cases
- Images are easy for humans to interpret visually
- Platform and environment compatibility is reasonable



# Drawbacks of Images for Content Storage

Image Formats Have Several Key Drawbacks:

- The collection of dots (pixels) discards information in the original document  
Compression can exacerbate this dramatically
- File size can increase dramatically from the original
- Document content can no longer be edited
- Images capture only what a document *looks like*, not what a document *contains* or *means*
- ***Users – human and otherwise – who cannot see images cannot derive any value from them***



# Images – They're Just Dots

## Drawbacks of Images



# Images – They're Just Dots

## Drawbacks of Images





# PDF – the Portable Document Format

PDF was created by Adobe Systems as a format to store content in an extractable manner, and to store information about that content.

PDF is an open, international standard (ISO 32000) and is freely implementable. Many tools, commercial and open source, are available.

Different flavors of PDF (PDF/A, PDF/UA) represent restrictions and best practices that enable more reliable and accurate content extraction.



# PDF – the Portable Document Format

What can a PDF contain?

- Textual content: words, sentences, numeric values, tables
- Line art: curves, lines, shapes
- Pictures: raster images
- Multimedia and interactivity
- Markup and comments
- Display and navigation instructions: layers, bookmarks
- Attachments and embedded files



# PDF – the Portable Document Format

PDFs can contain not just content, but information about the content within:

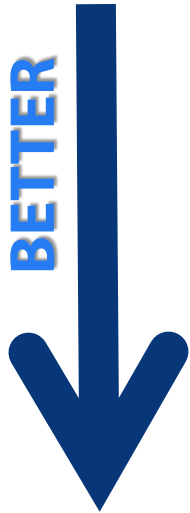
- Semantic meaning and alternate information (tagged PDF)
- Notes and annotations
- Metadata & properties
- Measurements and geospatial information
- Attachments, including source documents

PDFs are able but not guaranteed to store this information



# Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Example: Text Extraction via OCR

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Example: Text Extraction via OCR

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

9

©2014 Datalogics Incorporated

Text can be retrieved from PDFs in a variety of ways:

- OCR process on a printed I rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

<;>2014 Datalogics Incorporated.



# Example: Text Extraction via OCR

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

©2014 Datalogics Incorporated

9

This is a best-case example.  
More complicated pages  
lead to more errors.

Text can be retrieved from PDFs in a variety of ways:

- OCR process on a printed I rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

<;>2014 Datalogics Incorporated.



# Example: Text Extraction via Content Stream

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Example: Text Extraction via Content Stream

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

It is important to use a PDF processor that can read and understand tagging information

### Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:

BETTER

\$OCR \$Basic \$Intelligent \$Processing  
process scraping of  
on a of analysis PDF  
printed / content and re- tagging  
rendered streams assembly information  
rendition of page



# Example: Text Extraction via Content Stream

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

The order of text in the content stream may be quite different from how the human eye interprets the page

It is important to use a PDF processor that can read and understand tagging information

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:

BETTER

\$OCR \$Basic \$Intelligent \$Processing  
process scraping of  
on a of analysis PDF  
printed / content and re- tagging  
rendered streams assembly information  
rendition of page



# Example: Text Extraction, Tagged PDF

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Example: Text Extraction, Tagged PDF

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:

A list of text content extraction methods, in increasing capability:

\$OCR process on a printed / rendered rendition

\$Basic scraping of content streams

\$Intelligent analysis and re-assembly of pagecontent

\$Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Example: Text Extraction, Tagged PDF

## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:



- OCR process on a printed / rendered rendition
- Basic scraping of content streams
- Intelligent analysis and re-assembly of page content
- Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information

Text is Extracted in the Reading Order Specified by the PDF Creator.

Extracted Alternate Text from the Arrow Graphic



## Extraction of Textual Content

Text can be retrieved from PDFs in a variety of ways:

A list of text content extraction methods, in increasing capability:

- \$OCR process on a printed / rendered rendition
- \$Basic scraping of content streams
- \$Intelligent analysis and re-assembly of pagecontent
- \$Processing of PDF tagging information

It is important to use a PDF processor that can read and understand tagging information



# Extraction of Images

Images can exist in a variety of data formats in PDFs:

- Some images may be directly exported
- Different images may be in different color spaces, even on the same page
- Most images in a PDF require some transformation in order to be repurposed
- Some images in a PDF may need color conversion, resampling or other changes for downstream processes
- Images can contain metadata and alternate text, but this is not required

It's imperative to use a PDF processor that can understand and properly process all types of images in a PDF file

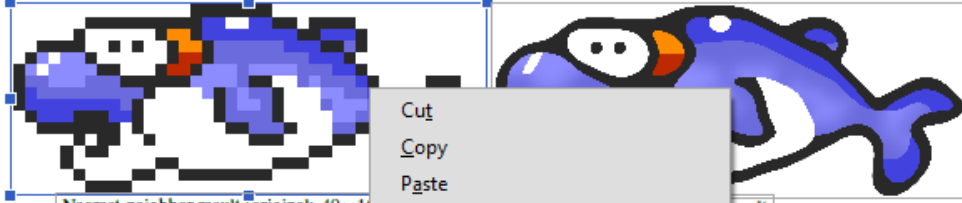


# Extraction of Images – Problem Example

## Depixelizing Pixel Art

Johannes Kopf  
Microsoft Research

Dani Lischinski  
The Hebrew University



Nearest-neighbor result (original: 40x10)

result

Figure 1: Naive upsampling of pixel art images leads to a smooth, resolution-independent vector representation from the image, which is suitable for modern computer graphics. The result is a smooth, resolution-independent vector representation of the original image (see Figure 1).

### Abstract

We describe a novel algorithm for extracting independent vector representation from pixel art images. The algorithm magnifies the results by an arbitrary amount and converts them into regions with smoothly varying colors. Our algorithm resolves pixel-scale input and converts them into regions with smoothly varying colors that are crisply separated by piecewise-smooth contours. In the original image, pixels are represented on a square grid, where diagonal neighbors are only connected through the point. This causes thin features to become visually blurry under magnification by conventional means, and creates ambiguities in the connectedness and separation of diagonal features. The key to our algorithm is in resolving these ambiguities by reshaping the pixel cells so that neighboring pixels to the same feature are connected through edges, thus resolving the feature connectivity under magnification. We also resolve aliasing artifacts and improve smoothness by fitting to contours in the image and optimizing their control points.

**Keywords:** pixel art, upscaling, vectorization

**Links:** [DL](#) [PDF](#) [WEB](#)

### 1 Introduction

Pixel art is a form of digital art where the details in the image are



# Extraction of Images – Problem Example

## Depixelizing Pixel Art

Johannes Kopf  
Microsoft Research

Dani Lischinski  
The Hebrew University

Nearest-neighbor result (original: 40x10)

Figure 1: Naive upsampling of pixel art images leads to a blurry result. Our algorithm extracts an independent vector representation from the image, which is suitable for magnifying the results by an arbitrary amount.

### Abstract

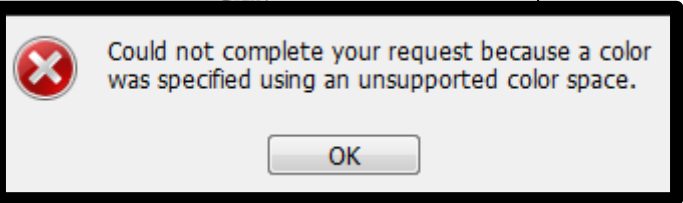
We describe a novel algorithm for extracting independent vector representation from pixel art images. Our algorithm resolves pixel-scale input and converts them into regions with smoothly varying colors. This causes thin features to become visually indistinct under magnification by conventional means, and creates artifacts in the connectedness and separation of diagonal lines. The key to our algorithm is in resolving these ambiguities by reshaping the pixel cells so that neighboring pixels to the same feature are connected through edges, thus preserving the feature connectivity under magnification. We also address aliasing artifacts and improve smoothness by fitting to contours in the image and optimizing their control points.

**Keywords:** pixel art, upscaling, vectorization

**Links:** [DL](#) [PDF](#) [WEB](#)

### 1 Introduction

Pixel art is a form of digital art where the details in the image are





# Extraction of Metadata and Properties

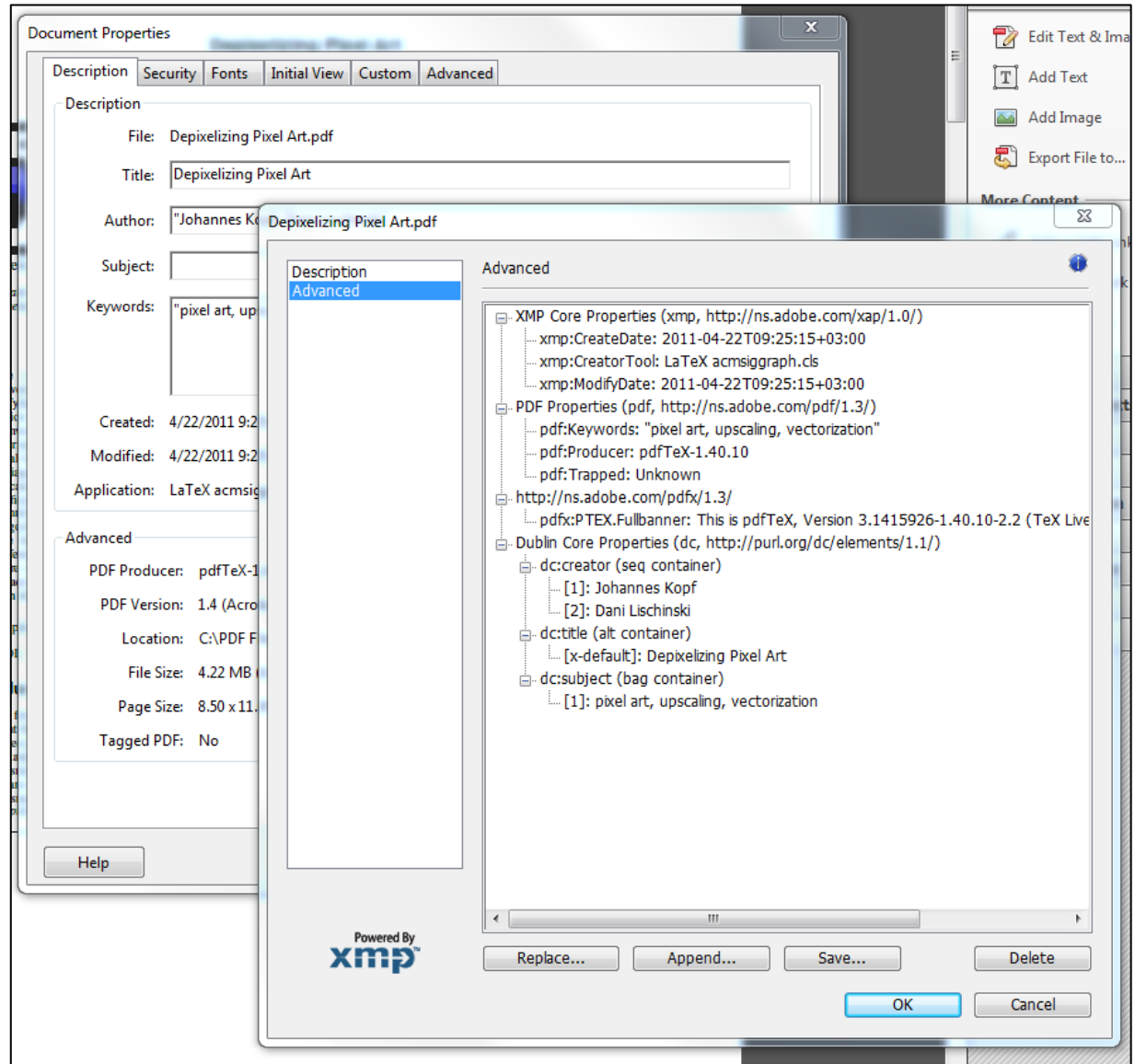
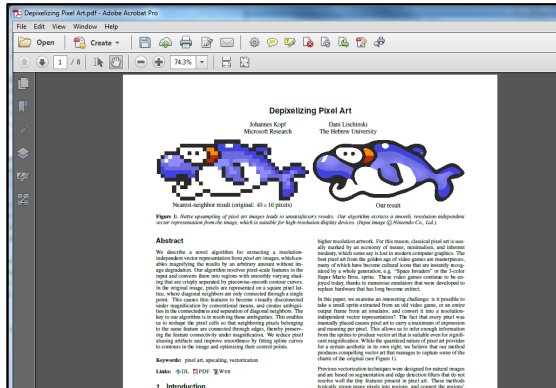
Metadata and properties can exist for a PDF and for the elements stored within:

- Document level information: author, title, ISBN, etc.
- Metadata about images: copyright information, camera information, etc.
- Alternate text descriptions for images and figures
- Creator or workflow specific metadata for any piece of extractable content

Information about how content is structured is contained within Tagged PDF



# Example: Document Metadata





# Common Pitfalls

PDF files are able to contain information that can reliably be extracted – but they do not have to be “nice”:

- Text may be represented in a semantically different way in the PDF from its visual appearance  
In extreme cases, text may accidentally or intentionally be represented in a way that cannot be reliably extracted
- What looks to be one image may be several (or thousands!) placed close together
- Invisible content may be contained within a file

PDF files that are created well are much easier for processors to use for content extraction.



# Summary

PDF is a format for documents and information – not just visual images

- PDF files can be edited and used in workflows, they are not just for end-stage archiving or storage
- PDF files contain reusable, extractable information – they are not just pictures
- PDF files are not guaranteed to be well-behaved or perfectly constructed for content extraction
- PDF standards like PDF/A and PDF/UA help make content extraction more predictable and reliable
- Tools that understand PDF tagging and structure make for better results in content extraction





Thank you for your time!

