PDF Enhancements Tools

Radim Hatlapatka

Masaryk University, Faculty of Informatics
Brno, Czech Republic
<208155@mail.muni.cz>

October 6th, 2010

EuDML
The EUROPEAN DIGITAL MATHEMATICS LIBRARY

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

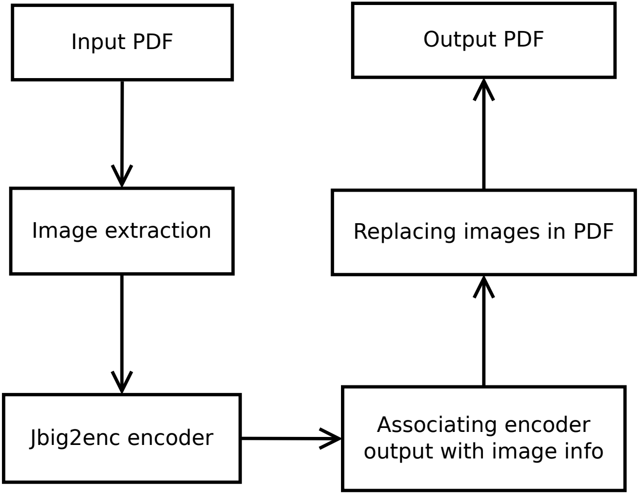
Motivation

- Metadata are not enough
- Fulltexts are needed as well
- Fulltexts must be stored ⇒ PDFs are necessary
- PDFs must be stored and transfered to the user
- Cost for storing and transfering is suitable to reduce
- Lots of PDF documents in EuDML contain images with scanned text
- Suitable tools `pdfJbIm` and `pdfsizeopt` for PDF optimization in EuDML

PdfJbIm

- Open-source tool written in Java for re-compression of bitonal images in PDF
- Uses benefits of standard JBIG2 which is supported in PDF since version `1.4` (Acrobat 5)
- Uses improved jbig2enc with symbol coding used for text area

PdfJbIm workflow



Input PDF

Image extraction

Jbig2enc encoder

Associating encoder output with image info

Replacing images in PDF

Output PDF

Jbig2enc

- Open-source encoder written in C/C++
- Uses open-source library Leptonica for manipulation with images
- Output in format suitable for PDF

JBIG2 and jbig2enc basic principle

- Page segmented to several regions based on type of data (text, image, generic)
- For each region is used specific coding
- Text area segmented to connected components (symbols)
- For each new symbol is created a representant and instances of this symbol are just pointers to the representant

Improvement of jbig2enc – motivation

- Number of symbols recognized for a page is several times greater than of born digital documents
- Our improvement reduces size of output image in average for further 10 percent without visible loss

Improvement of jbig2enc

- Comparing representative symbols
  - Two symbols are considered equivalent if there is not found a big enough difference to form a line or a point
- Unification of two equivalent symbols to one

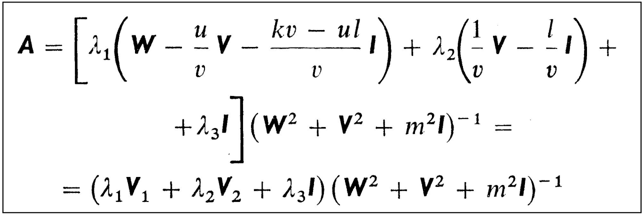
Improvement of jbig2enc

- Comparing representative symbols
  - Two symbols are considered equivalent if there is not found a big enough difference to form a line or a point
- Unification of two equivalent symbols to one

Improvement of jbig2enc

- Comparing representative symbols
  - Two symbols are considered equivalent if there is not found a big enough difference to form a line or a point
- Unification of two equivalent symbols to one

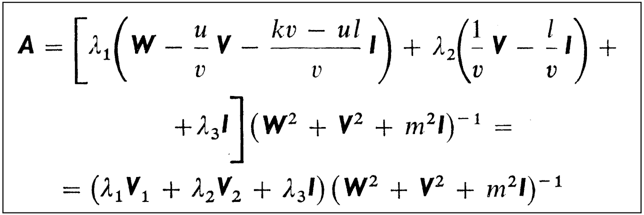
Image Before and After Compression



$$\boldsymbol{A} = \left[\lambda_1\left(\boldsymbol{W} - \frac{u}{v}\boldsymbol{V} - \frac{kv - ul}{v}\boldsymbol{I}\right) + \lambda_2\left(\frac{1}{v}\boldsymbol{V} - \frac{l}{v}\boldsymbol{I}\right) + \right.$$

$$\left. + \lambda_3\boldsymbol{I}\right]\left(\boldsymbol{W}^2 + \boldsymbol{V}^2 + m^2\boldsymbol{I}\right)^{-1} =$$

$$= \left(\lambda_1\boldsymbol{V}_1 + \lambda_2\boldsymbol{V}_2 + \lambda_3\boldsymbol{I}\right)\left(\boldsymbol{W}^2 + \boldsymbol{V}^2 + m^2\boldsymbol{I}\right)^{-1}$$

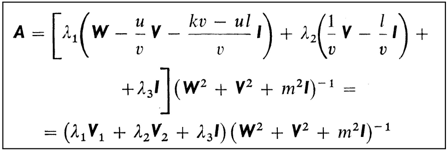


$$\boldsymbol{A} = \left[\lambda_1\left(\boldsymbol{W} - \frac{u}{v}\boldsymbol{V} - \frac{kv - ul}{v}\boldsymbol{I}\right) + \lambda_2\left(\frac{1}{v}\boldsymbol{V} - \frac{l}{v}\boldsymbol{I}\right) + \right.$$

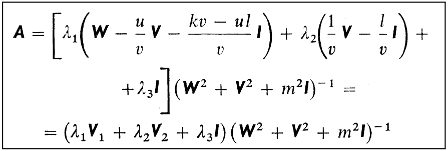
$$\left. + \lambda_3\boldsymbol{I}\right]\left(\boldsymbol{W}^2 + \boldsymbol{V}^2 + m^2\boldsymbol{I}\right)^{-1} =$$

$$= \left(\lambda_1\boldsymbol{V}_1 + \lambda_2\boldsymbol{V}_2 + \lambda_3\boldsymbol{I}\right)\left(\boldsymbol{W}^2 + \boldsymbol{V}^2 + m^2\boldsymbol{I}\right)^{-1}$$

Image Before and After Compression (cont.)



$$\mathbf{A} = \left[\lambda_1\left(\mathbf{W} - \frac{u}{v}\mathbf{V} - \frac{kv - ul}{v}\mathbf{I}\right) + \lambda_2\left(\frac{1}{v}\mathbf{V} - \frac{l}{v}\mathbf{I}\right) + + \lambda_3\mathbf{I}\right](\mathbf{W}^2 + \mathbf{V}^2 + m^2\mathbf{I})^{-1} = = (\lambda_1\mathbf{V}_1 + \lambda_2\mathbf{V}_2 + \lambda_3\mathbf{I})(\mathbf{W}^2 + \mathbf{V}^2 + m^2\mathbf{I})^{-1}$$



$$\mathbf{A} = \left[\lambda_1\left(\mathbf{W} - \frac{u}{v}\mathbf{V} - \frac{kv - ul}{v}\mathbf{I}\right) + \lambda_2\left(\frac{1}{v}\mathbf{V} - \frac{l}{v}\mathbf{I}\right) + + \lambda_3\mathbf{I}\right](\mathbf{W}^2 + \mathbf{V}^2 + m^2\mathbf{I})^{-1} = = (\lambda_1\mathbf{V}_1 + \lambda_2\mathbf{V}_2 + \lambda_3\mathbf{I})(\mathbf{W}^2 + \mathbf{V}^2 + m^2\mathbf{I})^{-1}$$

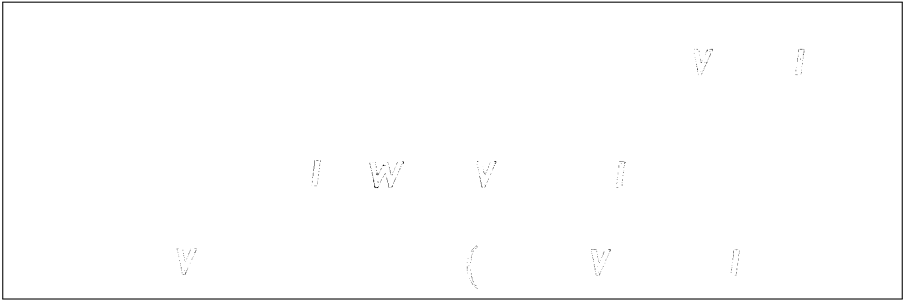
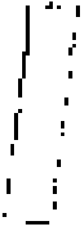
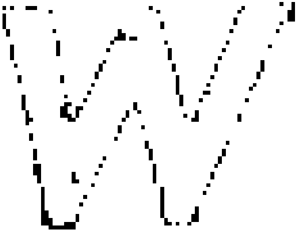


Image Before and After Compression (cont.)





pdfsizeopt.py

- Script written in python by Péter Szabó (Google) [4]
- Uses best practices and Unix tools to optimize size of PDF document
- Optimizes for example fonts, images, removes duplicate and unused objects.

pdfsizeopt.py

- Script written in python by Péter Szabó (Google) [4]
- Uses best practices and Unix tools to optimize size of PDF document
- Optimizes for example fonts, images, removes duplicate and unused objects.

pdfsizeopt.py

- Script written in python by Péter Szabó (Google) [4]
- Uses best practices and Unix tools to optimize size of PDF document
- Optimizes for example fonts, images, removes duplicate and unused objects.

Results

Journal or collection name	Size of original PDFs	After running pdfJbIm	After running both
Equadiff	279.5	194.3 (69.5%)	126.3 (45.1%)
NAFSA	79.5	59.2 (74.4%)	34.4 (42.1%)
Toposym	281.2	178.7 (63.5%)	144.8 (51.4%)
WSAA	469.6	300.2 (63.9%)	210.9 (44.9%)
WSGP	431.9	277.3 (64.1%)	183.1 (42.3%)
Časopis pro Pěst. Mat.	2,906.0	2,172.2 (74.7%)	1,296.1 (44.6%)
Časopis pro Pěst. Mat. Fys.	4,091.6	3,340.5 (81.6%)	1,700.1 (41.5%)
Czech Mathematical Journal	3,369.7	2,127.1 (63.1%)	1,874 (55.6%)

Results (cont.)

Journal or collection name	Size of original PDFs	After running pdfJbIm	After running both
Kybernetika	2,297.9	1,646 (71.6%)	906 (39.4%)
Mathematica Bohemica	472.9	326.7 (69.0%)	234.2 (49.5%)
Mathematica Slovaca	2,725.7	1,895.1 (69.5%)	1,051.4 (38.5%)
Pokroky MFA	2,312.3	1,554.4 (67.2%)	858.4 (37.1%)
Bolzano Collection	534.1	348.5 (65.2%)	280.2 (52.4%)
Dějiny Mat.	170.5	115.7 (67.8%)	75.5 (44.2%)
Single books	170.6	117.1 (68.6%)	72.3 (42.3%)
Totals	**20,592.84**	**14,652.77 (71.1%)**	**9,047.77 (43.9%)**

Summary

- Already functional version
- By combining `PdfJbIm` and pdfsizeopt.py we achieve size reduction of PDF files to less than 44%
- Tools suitable for use in EuDML either as part of EuDML core or as an independent applications
- Still lots of work in image preprocessing and improving perceptually lossless compression (visually lossless)

Further developement

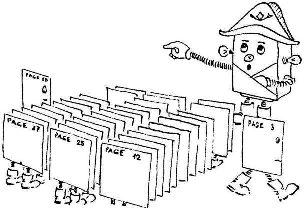
Image preprocessing

- Noise filtering
- Quality image detection

Integration with OCR

- Improve compression ratio by decreasing number of representantive symbols to a number as close as possible to a born digital documents
- Improve quality of output image by choosing the best representant

Questions?



References

Dan Bloomberg:
Leptonica.
<http://www.leptonica.com/>.

R. Hatlapatka:
Websites of the PDF re-compression project.
<http://nlp.fi.muni.cz/projekty/eudml/pdfRecompression/>.

Adam Langley:
Jbig2enc.
<http://github.com/agl/jbig2enc/>.

Péter Szábo:
Optimizing PDF output size of TeX documents.
<http://code.google.com/p/pdfsizeopt/>.