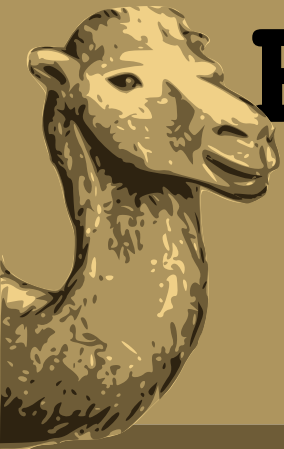


Perl Mongers im Ruhrgebiet



Ruhr . pm

Image::Magick::Thumbnail::PDF

Thumbnail und Docshots von PDF-Dokumenten erstellen

Autor: Veit Wahlich

E-Mail: veit AT ruhr.pm.org

Datum: 8. Oktober 2007

<http://ruhr.pm.org/>

Dieses Dokument wurde veröffentlicht unter der Lizenz

Creative Commons Attribution-Noncommercial-NoDerivs 2.0 Germany

Die Lizenz sowie entsprechende Übersetzungen sind einsehbar unter:
<http://creativecommons.org/licenses/by-nc-nd/2.0/de/>

Zusammenfassend ergeben sich hieraus die folgenden Rechte:



Sie dürfen das Werk vervielfältigen, verbreiten und öffentlich zugänglich machen.

Diese Rechte werden Ihnen unter den folgenden Bedingungen gewährt:



Namensnennung. Sie müssen den Namen des Autors/Rechteinhabers in der von ihm festgelegten Weise nennen (wodurch aber nicht der Eindruck entstehen darf, Sie oder die Nutzung des Werkes durch Sie würden entlohnt).



Keine kommerzielle Nutzung. Dieses Werk darf nicht für kommerzielle Zwecke verwendet werden.



Keine Bearbeitung. Dieses Werk darf nicht bearbeitet oder in anderer Weise verändert werden.

Im Falle einer Verbreitung müssen Sie anderen die Lizenzbedingungen, unter welche dieses Werk fällt, mitteilen.

Jede der vorgenannten Bedingungen kann aufgehoben werden, sofern Sie die Einwilligung des Rechteinhabers dazu erhalten.

Diese Lizenz lässt die Urheberpersönlichkeitsrechte unberührt.



Ruhr.pm

Image::Magick::Thumbnail::PDF

- erzeugt Bitmap-Grafiken aus PDF-Dateien
- verwendet Image::Magick als Backend
- verfügbar im CPAN
- aktive Entwicklung
 - demnächst Unterstützung für animierte GIF-Dateien
 - aktuelles Release 1.11 vom 28.08.2007
- Autor: Leo Charre <leocharre@cpan.org>



Ruhr . pm

Vorteile

- sehr einfach zu verwenden
- wenige Dependencies
- wahlweise Optimierung für Dokumente mit feinen Linien
 - Zeichnungen
 - gescannte Dokumente
- unterstützt viele Grafikformate



Ruhr . pm

Nachteile

- unflexibles API
 - kein objekt-orientiertes Interface
 - kein Zugriff auf das Image::Magick-Objekt
- eingeschränkte Konfigurationsmöglichkeiten
 - keine Konfiguration der Rahmenfarbe
 - eingeschränkte Benennung der Ausgabe-Dateinamen
 - keine Konfiguration des Normalisierungsgrades



Ruhr.pm

Synopse

```
use Image::Magick::Thumbnail::PDF
    qw(create_thumbnail);

$output = create_thumbnail(
    $input, $seite, $output,
    {
        restriction => $maxkantenlaenge,
        normalize   => $bool,
        frame       => $randbreite,
        quality     => $qualitaet
    }
);
```

\$input:

Pflicht, Dateiname des PDF-Dokuments

\$seite:

Option, Nummer der abzubildenden Seite oder 'all_pages' für alle Seiten

\$output:

Option, Dateiname(n) der ausgegebenen Seite(n), bei 'all_pages' ist der Rückgabewert ein Array-Ref



Ruhr.pm

Synopse

```
use Image::Magick::Thumbnail::PDF
    qw(create_thumbnail);

$output = create_thumbnail(
    $input, $seite, $output,
    {
        restriction => $maxkantenlaenge,
        normalize   => $bool,
        frame       => $randbreite,
        quality     => $qualitaet
    }
);
```

restriction => \$maxkantenlaenge
Option, definiert die maximale
Länge der längsten Kante der
erzeugten Pixmap in Pixeln

normalize => \$bool
Option, Linienoptimierung inaktiv
wenn false

frame => \$randbreite
Option, Breite eines Rands in Pixeln

quality => \$qualitaet
Option, Qualität/Kompression von
JPEG/PNG/MIFF wie Image::Magick



Ruhr.pm

Einfache Beispiele

```
use Image::Magick::Thumbnail::PDF
    qw(create_thumbnail);

my($input,$output,$seite)=
    ('beispiel.pdf','beispiel.jpg',3);

create_thumbnail($input, $seite, $output,
    { restriction => 200,
      normalize   => 0,
      frame       => 0,
      quality     => 80 } );

$output=create_thumbnail(
    $input, 'all_pages'
);
```

Erstellt eine rahmenlose JPEG-Grafik mit 80% Qualität der dritten Seite von beispiel.pdf, deren längste Kante 200px misst. Linienoptimierung ist deaktiviert.

Erstellt eine PNG-Datei von jeder Seite mit Standardeinstellungen. Die Dateinamen finden sich in dem anonymen Array \$output.



Ruhr.pm

Linienoptimierung/Normalisierung

```
use Image::Magick::Thumbnail::PDF
    qw(create_thumbnail);

create_thumbnail(
    'topologie.pdf',
    'topologie.png',
    {
        restriction => 200,
        frame       => 0,
        normalize   => 0
    }
);

create_thumbnail(
    'topologie.pdf',
    'topologie_normalized.png',
    {
        restriction => 200,
        frame       => 0,
        normalize   => 1
    }
);
```

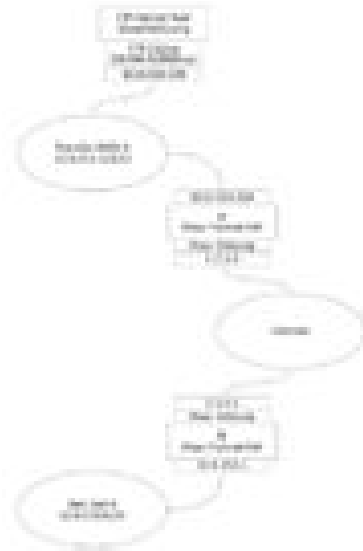
Normalisierung dient zur besseren Darstellung von feinen Linien wie z.B. in Zeichnungen oder gescannten Dokumenten. Bei anderen Dokumenten, z.B. mit eingebetteten Fotos, kann es sich eher negativ auswirken.

Hier wird das Dokument einmal ohne (topologie.png) und einmal mit Normalisierung (topologie_normalized.png) gespeichert.

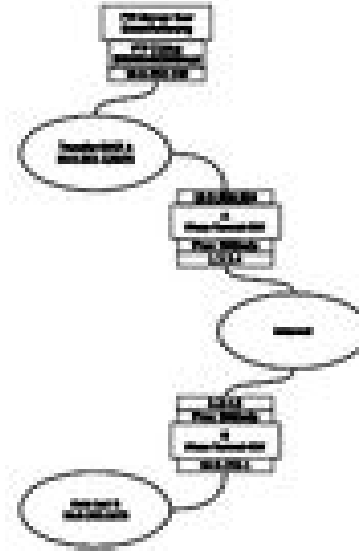


Ruhr . pm

Linienoptimierung/Normalisierung



topologie.png
vergrößert



topologie_normalized.png
vergrößert



Ruhr.pm

Einen Dokument-Index erstellen

```
print(<<__EOF);
<html>
  <body>
    <h1>Index der PDF-Dateien</h1>
    <p>
__EOF

foreach(glob('*.*pdf')){
  create_thumbnail($_, $_.'.png', 1,
    { restriction => 150,
      frame       => 1,
      normalize   => 0   }
  );
  print(<<__EOF);
    <a href="$_"></a>
__EOF
}

print(<<__EOF);
  </p>
</body>
</html>
__EOF
```

Erzeugt für jede Datei mit der Endung .pdf im aktuellen Verzeichnis eine PNG-Grafik.

Dabei wird auf stdout ein Index der erzeugten Dateien in HTML ausgegeben.

Eine ausführlichere Version dieses Listings hängt dem Dokument an.



Ruhr . pm

Einen Dokument-Index erstellen





Ruhr.pm

Einen Docshot erstellen

```
my $file='regexp.pdf';
my $shots=create_thumbnail(
    $file, 'all_pages',
    { restriction => 800,
      frame => 1,
      normalize => 0    });

print(<<__EOF);
<html>
  <body>
    <h1>$file</h1>
    <p>
__EOF

print(<<__EOF) foreach(@{$shots});
  
__EOF

print(<<__EOF);
  </p>
</body>
</html>
__EOF
```

Erzeugt für jede Seite der Datei regexp.pdf eine PNG-Grafik mit 800px maximaler Seitenlänge und 1px Rand.

Auf stdout wird anschliessend ein HTML-Dokument ausgegeben, das alle Grafiken darstellt.

Eine ausführlichere Version dieses Listings hängt dem Dokument an.



**Vielen Dank
für Eure Aufmerksamkeit**



Ruhr . pm

Links

- Image::Magick::Thumbnail::PDF im CPAN
 - <http://search.cpan.org/~leocharre/Image-Magick-Thumbnail-PDF-1.11/>
- Image::Magick/PerlMagick
 - <http://www.imagemagick.org/script/perl-magick.php>
- Präsentation, Beispiele und RPMs
 - <http://ruhr.pm.org/treffen/artikel.psp?id=45>