

The contour package

Harald Harders (h.harders@tu-bs.de)
Morten Høgholm (moho01ab@student.cbs.dk)

File version v2.14, date 2004/11/18, printed September 27, 2012

Abstract

This package generates a colored contour around a given text in order to enable printing text over a background without the need of a color box around the text. By default this is done by placing copies of the text around the text itself and can be switched to use real outlines if supported by the backend driver.

Contents

1	Introduction	2
2	User interface	2
3	Examples	3
4	Advanced internals	3
5	Acknowledgements	4
6	The implementation	4
6.1	The package	4
6.2	Driver Files	7
6.2.1	Dvips	7
6.2.2	pdfTeX	8
6.2.3	VT _E X	9
6.2.4	DVIPSONE	10
6.3	Configuration Files	11

Copyright

Copyright 1999–2004 Harald Harders, Morten Høgholm.

This program can be redistributed and/or modified under the terms of the LaTeX Project Public License Distributed from CTAN archives in directory `macros/latex/base/lppl.txt`; either version 1 of the License, or any later version.

1 Introduction

Sometimes it is necessary to print text over a background that is not pure white, e.g., in gnuplot plots. It is not always wished to plot the text in a rectangular box since this can cover important information or look poor. This package provides a solution to cover less space with the text and ensure readable text in the same time.

The used technique is quite simple. By default, in a circle around the original text position the same text is printed evenly distributed 16, 32, or a given number times. The default radius for the circle is 0.03 em. If requested and supported by the used driver (dvips, pdfTEX, VTEX, DVIPSONE) a real outline can be chosen instead of text copies.

2 User interface

To use this package place

```
\usepackage[<options>]{contour}
```

in the preamble of your document.

If loaded without package option or with the option `copies` the contour is printed by placing copies of the text around the original text, as described below. This can be changed by the option `outline`. If specified this option the contour is printed by a real outline of the text instead of copies. This increases speed as well as quality¹ and reduces the file size. But some prerequisites have to be fulfilled:

- The backend driver has to be supported. Currently, dvips, pdfTEX, VTEX, and DVIPSONE are supported. Normally, the file `contour.cfg` tries to find out which driver has to be loaded.² But you can also specify the driver by giving one of the package options `dvips`, `xdvi`, `pdftex`, `vtex`, or `dvipsone`. `xdvi` refers to `dvips` as driver. In `xdvi` and `DVIWindo`, the contours are silently ignored.
- Outline (vector) fonts have to be used. With dvips, this means that Type 1 fonts have to be used. With pdfTEX, Type 1 or TrueType fonts are supported. If you are using METAFONT fonts you have to use copies (package option `copies`).

`\contour` The package provides the commands

```
\contour{<color>}{<text>}
\contour[<number>]{<color>}{<text>}
\contour*{<color>}{<text>}
```

which produce the text *<text>* with a *<color>*ed contour around. The text itself is typeset in the normal active color. The normal `\contour` command produces 16 copies of the text while the `*` variant makes 32 copies. If *<number>* is given, the given number of copies is generated around the text. If *<number>* is `auto` instead of a number, the number of copies is calculated automatically depending on the font size.

`\contourlength` The radius of the circle (= thickness of the contour) can be changed using the

¹Especially in Acrobat Reader, the quality is better.

²DVIPSONE cannot be determined automatically.

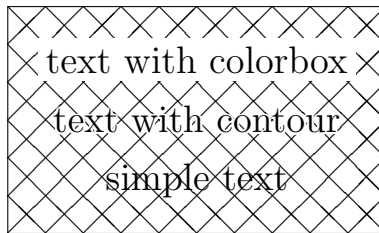


Figure 1: Comparison of the commands `\colorbox` and `\contour`

command `\contourlength{<length>}`, where *<length>* is a length understood by L^AT_EX. The length is not expanded when defining the contour length but when using it. For example when using `0.05em`, `0.05em` of the font used in the command `\contour` are used.

`\contournumber` By default, `\contour` generates 16 copies of the text. You can change this by using `\contournumber`, e.g.,

```
\contournumber{27}
```

If you specify `auto` instead of a number, `\contour` will use the automatically calculated number of copies. You may also use the package option `auto` to reach this behaviour:

```
\usepackage[auto]{contour}
```

If using outlines the number of copies is ignored.

3 Examples

The command

```
\colorbox{black}{This text is not visible}
\contour{white}{but this is.}
```

produces:

but this is.

Another example is shown in Figure 1.

In Figure 2, you can see how the output depends on the number of copies.

4 Advanced internals

Normally, it is not planned to switch between outline and copy mode. If you really have to do it you can use the boolean `\ifcon@outline`. `\con@outlinetrue` switches on outlines, while replacing `true` by `false` switches them off. Don't forget to surround it by `\makeatletter` and `\makeatother` to allow `@` in command names.

See the difference between copy and outline mode:

Copies Outline

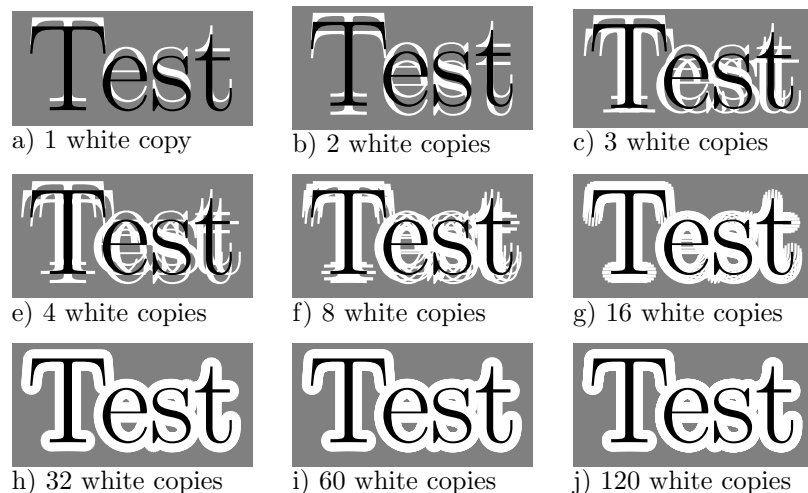


Figure 2: Contours with different numbers of copies with a large distance between text and copies

5 Acknowledgements

Thanks to Richard Pfeiffer who had the idea for this package and wrote some code that did it.

Thanks to Dietrich Grau who asked me for support of DVIPSONE and who did the testing of all my code ideas for this driver since I do not have Y&Y \TeX .

6 The implementation

Heading of the package:

```

1 <package>\NeedsTeXFormat{LaTeX2e}
2 <dvips>\ProvidesFile{dvips.cnt}
3 <pdfTeX>\ProvidesFile{pdfTeX.cnt}
4 <vTeX>\ProvidesFile{vTeX.cnt}
5 <dvipsone>\ProvidesFile{dvipsone.cnt}
6 <cfgfile>\ProvidesFile{contour.cfg}
7 <package>\ProvidesPackage{contour}
8 <package | drv | cfgfile> [2004/11/18 v2.14 Print contoured text (HH, MH)]
9 <*package>

```

6.1 The package

This package requires the color package from the graphics bundle:

```
10 \RequirePackage{color}
```

This package requires the trig package:

```
11 \RequirePackage{trig}
```

A command that makes `\color` inactive prevent the background text color from being changed:

```
12 \def\con@coloroff{%
```

```

13 \def\@undeclaredcolor[##1]##2{%
14 \def\@declaredcolor##1{%
15 }

```

Set text to a specified relative position without using space:

```

16 \newcommand*\con@put[3]{%
17 \rlap{\hskip#1\raisebox{#2}[0pt]{#3}}%
18 }

```

\contourlength Define the commands for changing the base length:

```

19 \newcommand*\contourlength[1]{\def\con@base@length{#1}}
20 \contourlength{0.03em}

```

\contournumber Define the commands for changing the number of text copies:

```

21 \newcommand*\contournumber[1]{\def\con@default@copies{#1}}
22 \contournumber{16}

```

Define new offset length:

```

23 \newlength{\con@length}

```

Define the angles for using ‘<#1>’ copies unless they’ve previously been defined.
We use ‘trig’ for this.

```

24 \newcommand*\con@define@copyangles[1]{%
25 \edef\con@tempa{auto}%
26 \edef\con@tempb{#1}%

```

Set the number of copies automatically.

```

27 \ifx\con@tempa\con@tempb
28 \expandafter\ifdim\f@size pt<15pt\relax
29 \edef\con@tempb{16}%
30 \else
31 \expandafter\ifdim\f@size pt<25pt\relax
32 \edef\con@tempb{32}%
33 \else
34 \expandafter\ifdim\f@size pt<50pt\relax
35 \edef\con@tempb{60}%
36 \else
37 \edef\con@tempb{120}%
38 \fi
39 \fi
40 \fi
41 \fi
42 \@tempcnta=\con@tempb\relax
43 \ifundefined{con@\number\@tempcnta @copies}{%
44 \global\@namedef{con@\number\@tempcnta @copies}{}%
45 \@tempdima=360\p@
46 \divide\@tempdima by \@tempcnta
47 \@tempdimb=\z@
48 \@tempcntb=\z@
49 \loop
50 \ifnum\@tempcntb<\@tempcnta
51 \edef\con@temp@fdim{\strip@pt\@tempdimb}%
52 \CalculateSin{\con@temp@fdim}%
53 \CalculateCos{\con@temp@fdim}%
54 \advance\@tempcntb \@ne

```

```

55         \advance\@tempdimb \@tempdima\relax
56     \repeat
57 }{}%
58 \let\con@tempa\@undefined
59 \let\con@tempb\@undefined
60 }

\contour The starred version of \contour prints 32 copies.
61 \newcommand*\contour{\@ifstar{\@contour[32]}{\@contour}}
62 \newcommand*\@contour[3][\con@default@copies]{%
63     \ifvmode
64         \leavevmode
65     \fi
66     \setlength\con@length{\con@base@length}%

Print outline or copies?
67     \ifcon@outline
68         \typeout{contour: Using real outline for '#3'\on@line.}%
69         \@contour@outline{#2}{#3}%
70     \else
71         \begingroup

Calculate the copy angles.
72         \con@define@copyangles{#1}%
73         \typeout{contour: Using \the\@tempcnta\space copies for '#3'\on@line.}%
74         \@tempdima=360\p@
75         \divide\@tempdima by \@tempcnta\relax
76         \@tempdimb=\z@
77         \@tempcntb=\z@

Set color and switch off color command inside argument.
78         \color{#2}%
79         \con@coloroff

Print the copies.
80     \loop
81     \ifnum\@tempcntb<\@tempcnta\relax
82         \edef\con@temp@fdim{\strip@pt\@tempdimb}%
83         \con@put
84             {\UseSin{\con@temp@fdim}\con@length}%
85             {\UseCos{\con@temp@fdim}\con@length}%
86             {#3}%
87         \advance\@tempcntb \@ne\relax
88         \advance\@tempdimb\@tempdima\relax
89     \repeat
90 \endgroup

Print the main text.
91     \mbox{#3}%
92 \fi
93 }

By default, no driver is active.
94 \providecommand*\con@driver{\@empty}

Boolean for using outline or copies.
95 \newif\ifcon@outline

```

Options:

```

96 \DeclareOption{auto}{\contournumber{auto}}
97 \DeclareOption{dvips}{\def\con@driver{dvips.cnt}}
98 \DeclareOption{xdvi}{\ExecuteOptions{dvips}}
99 \DeclareOption{pdftex}{\def\con@driver{pdftex.cnt}}
100 \DeclareOption{vtex}{\def\con@driver{vtex.cnt}}
101 \DeclareOption{dvipsone}{\def\con@driver{dvipsone.cnt}}
102 \DeclareOption{outline}{\con@outline>true}
103 \DeclareOption{copies}{\con@outline>false}

```

Load configuration file if existing.

```

104 \InputIfFileExists{contour.cfg}{%
105   \typeout{Loading configuration file 'contour.cfg'.}%
106 }{%
107   \typeout{No configuration file 'contour.cfg' found.}%
108 }

```

Process the options.

```

109 \ProcessOptions\relax

```

Load the driver file.

```

110 \expandafter\ifx\con@driver\@empty
111   \ifcon@outline
112     \PackageError{contour}{Chosen package option 'outline' but no
113       driver defined}{Leave out the 'outline' option or define a driver}%
114   \else
115     \PackageWarning{contour}{No driver defined (which does not matter
116       when using copies)}%
117   \fi
118 \else
119   \InputIfFileExists{\con@driver}{%
120     \typeout{contour: Using driver file '\con@driver'.}%
121   }{%
122     \PackageError{contour}{Driver file '\con@driver' does not exist}{}%
123   }%
124 \fi
125 \end{package}

```

6.2 Driver Files

6.2.1 Dvips

`\@contour@outline` Prints the text and contour using real outlines. `\@contour@outline{<color>}{<text>}`

```

126 <*dvips>
127 \newcommand*\@contour@outline[2]{%
128   \begin{group}

```

Double the width of the contour since the inner half is overprinted by the normal text; convert pt to bp.

```

129   \setlength\con@length{2\con@length}%
130   \setlength\con@length{0.99626400996\con@length}%

```

Set the contour color and disable color command.

```

131   \color{#1}%
132   \con@coloroff

```

PostScript preamble to print an outline for the text.

```

133   \special{ps:
First, save all graphics settings to avoid side effects.
134   gsave
Start a new path and choose a round pen.
135       newpath
136       1 setlinejoin
137       1 setlinecap
Set the line width and scale it according to the PostScript scale.
138       Resolution 72 div DVImag mul
139       \strip@pt\con@length\space mul setlinewidth
Save show to be able to restore it later.
140       /cntorigshow /show load def
Redefine the show command that prints a text to do the outline instead of the
text.
141       /show { false charpath } def
142   }%
Typeset the outline text.
143   \rlap{#2}%
PostScript postamble.
144   \special{ps:
Finally, do the outline.
145       stroke
Restore the original settings.
146       /show /cntorigshow load def
147       grestore
148   }%
149   \endgroup
Print the main text.
150   \mbox{#2}%
151 }
152 </dvips>

```

6.2.2 pdfTeX

\@contour@outline Prints the text and contour using real outlines. `\@contour@outline{<color>}{<text>}`

```

153 <*pdfTeX>
154 \newcommand*\@contour@outline[2]{%
155   \begingroup
Double the width of the contour since the inner half is overprinted by the normal
text; convert pt to bp.
156   \setlength\con@length{2\con@length}%
157   \setlength\con@length{0.99626400996\con@length}%
Set the contour color and disable color command.
158   \color{#1}%
159   \con@coloroff

```


PDF preamble.

```

160    \pdfliteral{%
Save the graphics settings.
161    q
Choose a round pen.
162    1 j
163    1 J
Switch text to print an outline instead of fill.
164    1 Tr
Set the line width.
165    \strip@pt\con@length\space w
166    }%
Typeset the outline text.
167    \rlap{#2}%
PDF postamble.
168    \pdfliteral{%
Restore original settings.
169    Q
170    }%
171    \endgroup
Print the main text.
172    \mbox{#2}%
173 }
174 </pdfTeX>

```

6.2.3 VTeX

`\@contour@outline` Prints the text and contour using real outlines. `\@contour@outline{<color>}{<text>}`

```

175 <*vTeX>
176 \newcommand*\@contour@outline[2]{%
177   \begingroup
Double the width of the contour since the inner half is overprinted by the normal
text; convert pt to bp.
178   \setlength\con@length{2\con@length}%
179   \setlength\con@length{0.99626400996\con@length}%
Set the contour color and disable color command.
180   \color{#1}%
181   \con@coloroff
PostScript preamble to print an outline for the text.
At this point, VTeX does an ugly job since it interprets this code immediately
and thus puts the path itself into the output instead of just let the printer do the
work. I do not know how to change this.
182   \special{pS:
First, save all graphics settings to avoid side effects.
183     save

```

Start a new path and choose a round pen.

```

184      newpath
185      1 setlinejoin
186      1 setlinecap

```

Set the line width.

```

187      \strip@pt\con@length\space setlinewidth

```

Redefine the command that prints a text to do the outline instead of the text.

```

188      /show { false charpath } def
189  }%

```

Typeset the outline text.

```

190  \rlap{#2}%

```

PostScript postamble.

```

191  \special{pS:

```

Finally, do the outline.

```

192      stroke

```

Restore the original settings.

```

193      restore
194  }%
195  \endgroup

```

Print the main text.

```

196  \mbox{#2}%
197 }
198 </vtex>

```

6.2.4 DVIPSONE

`\@contour@outline` Prints the text and contour using real outlines. `\@contour@outline{<color>}{<text>}`

```

199 <*dvipsone>
200 \newcommand*\@contour@outline[2]{%

```

First, print the text. This is a hack and avoids problems when `\contour` is used in some contexts, e.g., at the begin of the text.

```

201  \rlap{#2}%
202  \begingroup

```

Double the width of the contour since the inner half is overprinted by the normal text; convert pt to bp.

```

203  \setlength\con@length{2\con@length}%
204  \setlength\con@length{0.99626400996\con@length}%

```

Set the contour color and disable color command.

```

205  \color{#1}%
206  \con@coloroff

```

PostScript preamble to print an outline for the text.

```

207  \special{ps:

```

First, save all graphics settings to avoid side effects.

```

208      gsave

```

Save the current position to be used for the new path.

```

209      currentpoint

```

Start a new path and go back to the old position.

```

210      newpath
211      moveto

```

Choose a round pen.

```

212      1 setlinejoin
213      1 setlinecap

```

Set the line width and scale it according to the PostScript scale. The factor 65693.4 is chosen by trial and error and may be inexact.

```

214      65693.4 \strip@pt\con@length\space mul setlinewidth

```

Save `show` to be able to restore it later.

```

215      /cntorigshow /show load def

```

Redefine the `show` command that prints a text to do the outline instead of the text.

```

216      /show { false charpath } def
217  }%

```

Typeset the outline text.

```

218      \rlap{#2}%

```

PostScript postamble.

```

219      \special{ps:

```

Finally, do the outline and save the position.

```

220      currentpoint
221      stroke

```

Restore the original settings and position.

```

222      /show /cntorigshow load def
223      grestore
224      moveto
225  }%
226 \endgroup

```

Print the main text.

```

227      \mbox{#2}%
228 }
229 </dvipsone>

```

6.3 Configuration Files

This configuration file is just a copy of a part of `graphics.cfg` from T_EXLive.

```

230 <*cfgfile>
231 % Select an appropriate default driver
232 \begingroup
233   \chardef\x=0 %
234   % check pdfTeX
235   \@ifundefined{pdfoutput}{-}{%
236     \ifcase\pdfoutput
237     \else
238       \chardef\x=1 %
239     \fi
240   }%
241   % check VTeX

```

```

242 \ifundefined{OpMode}{\{%
243   \chardef\x=2 %
244 }%
245 \expandafter\endgroup
246 \ifcase\x
247   % default case
248   \ExecuteOptions{dvips}%
249 \or
250   % pdfTeX is running in pdf mode
251   \ExecuteOptions{pdftex}%
252 \else
253   % VTeX is running
254   \ExecuteOptions{vtex}%
255 \fi
256 \</cfgfile>

```

Change History

1.04		for V _T E _X	1
	General: Avoide usage of	2.12	
	\textversion etc.	1	General: Full outline support for
2.00			V _T E _X
	General: Allow arbitrary numbers	2.13	
	of text copies	1	\@contour@outline: More robust
2.10			mechanism to change Postscript
	General: Bugfix: enable switch-		internals for dvips
	ing colors inside \contour argu-	2.14	
	ment	4	General: Add support for DVIP-
	Print text with real outlines for		SONE
	vector fonts	1	10
2.11			Full outline support for Y&Y
	General: Restricted outline support		DVIPSONE
			1

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in **roman** refer to the code lines where the entry is used.

Symbols		B
\@contour	61, 62	\begingroup . 71, 128,
\@contour@outline 69,		155, 177, 202, 232
<u>126</u> , <u>153</u> , <u>175</u> , <u>199</u>		
\@declaredcolor . . .	14	C
\@empty	94, 110	\CalculateCos
\@ifstar	61	\CalculateSin
\@ifundefined		\chardef . . 233, 238, 243
.	43, 235, 242	\color
\@namedef	44	131, 158, 180, 205
\@tempcnta . . 42–44,		\con@base@length 19, 66
46, 50, 73, 75, 81		
	A	
	\advance . 54, 55, 87, 88	

<code>\con@coloroff</code> 12, 79, 132, 159, 181, 206	E	O
<code>\con@default@copies</code> 21, 62	<code>\endgroup</code> .. 90, 149, 171, 195, 226, 245	<code>\on@line</code> 68, 73
<code>\con@define@copyangles</code> 24, 72	<code>\ExecuteOptions</code> 98, 248, 251, 254	<code>\or</code> 249
<code>\con@driver</code> 94, 97, 99–101, 110, 119, 120, 122	<code>\expandafter</code> 28, 31, 34, 110, 245	P
<code>\con@length</code> .. 23, 66, 84, 85, 129, 130, 139, 156, 157, 165, 178, 179, 187, 203, 204, 214	F	<code>\PackageError</code> . 112, 122
<code>\con@outlinefalse</code> . 103	<code>\f@size</code> 28, 31, 34	<code>\PackageWarning</code> ... 115
<code>\con@outlinetrue</code> .. 102	H	<code>\pdfliteral</code> ... 160, 168
<code>\con@put</code> 16, 83	<code>\hskip</code> 17	<code>\pdfoutput</code> 236
<code>\con@temp@fdim</code> 51–53, 82, 84, 85	I	<code>\ProcessOptions</code> ... 109
<code>\con@tempa</code> .. 25, 27, 58	<code>\ifcase</code> 236, 246	<code>\providecommand</code> ... 94
<code>\con@tempb</code> 26, 27, 29, 32, 35, 37, 42, 59	<code>\ifcon@outline</code> 67, 95, 111	<code>\ProvidesFile</code> 2–6
<code>\contour</code> 2, <u>61</u>	<code>\ifvmode</code> 63	R
<code>\contourlength</code> ... 2, <u>19</u>	<code>\InputIfFileExists</code> 104, 119	<code>\raisebox</code> 17
<code>\contournumber</code> 3, <u>21</u> , 96	L	<code>\repeat</code> 56, 89
D	<code>\leavevmode</code> 64	<code>\rlap</code> 17, 143, 167, 190, 201, 218
<code>\DeclareOption</code> . 96–103	<code>\loop</code> 49, 80	S
<code>\divide</code> 46, 75	M	<code>\special</code> .. 133, 144, 182, 191, 207, 219
	<code>\mbox</code> 91, 150, 172, 196, 227	U
	N	<code>\UseCos</code> 85
	<code>\newif</code> 95	<code>\UseSin</code> 84
	X	
		<code>\x</code> ... 233, 238, 243, 246