The char2path Package*

zongpingding†, Mingyu Xia‡

Released 2025-07-28 v1.0.0

1 Introduction

The char2path package provides a LaTeX package for converting characters into TikZ paths. It supports compilation methods including pdfLaTeX, XeLaTeX, and LuaLaTeX.

2 Usage

To load this package, write the line

```
\usepackage[scale = ⟨choice⟩]{char2path}
```

The supported right listed ⟨*choice*⟩ can scale the converted paths with corresponding factors.

Choice	10pt	11pt	12pt	13pt
Factor	0.82	0.90	0.98	1.07
Choice	14pt	15pt	16pt	...
Factor	1.15	1.23	1.31	...

`\chartopath` `\chartopath [⟨key-vals⟩] {⟨string⟩}`

Updated: 2025-07-27

The `[⟨key-vals⟩]` accepts the listed keys to set how the `{⟨string⟩}` converts to path.

font = ⟨*lmr*|*lms*|*lmm*⟩ choose char's font.

draw, fill = ⟨*color*⟩ set the color of the outline/fill of string (Default: `black`).

outline = ⟨*dim*⟩ set the outline of the string to achieve the "Fake Bold" effect (Default: `0pt`).

scale = ⟨*fp num*⟩ set font's scale factor (Default: `1`).

hfading, vfading = ⟨*color clist*⟩ set the two colors for horizontally/vertically fading the whole string.

rescan = ⟨**true**|*false*⟩ rescan tokens `$`, `&`, and `~` with catcode 12 (other). (Default: `true`).

`\chartoclip` `\chartoclip [⟨key-vals⟩] {⟨char⟩}`

Updated: 2025-07-26

The mandatory argument accepts a single character, and the optional argument accepts the following keys to set the style how the character clip the image.

font = ⟨*lmr*|*lms*|*lmm*⟩ choose char's font.

image = ⟨*image*⟩ set image to clip.

scale = ⟨*fp num*⟩ set font's scale factor (Default: `1`).

anchor = ⟨*tikz anchor*⟩ set image's anchor.

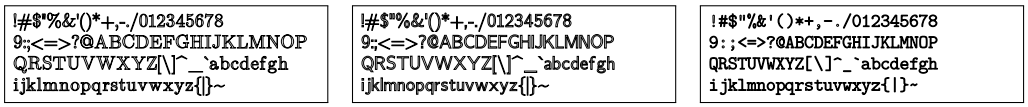
offset = ⟨*(x, y)*⟩ set imgae anchor's offset.

height = ⟨*dim*⟩ image's height.

*https://ctan.org/pkg/char2path, https://github.com/zongpingding/char2path
†zongpingding5@outlook.com
‡xiamingyu@westlake.edu.cn

3 Examples

This package can convert the following characters to path by default: including Latin Modern Roman, Latin Modern Sans, and `Latin Modern Mono`.



3.1 Basic usage

The basic usage only print the outline of the inputted string.

[3.1.1]
```
\def\0{0123456789/ctp}
\chartopath\0
```
0123456789/ctp

Users can specific the color of outline and fill, specific the fill-in pattern, and specific the scale and slant of inputted characters. Just like the keys in Ti*k*Z.

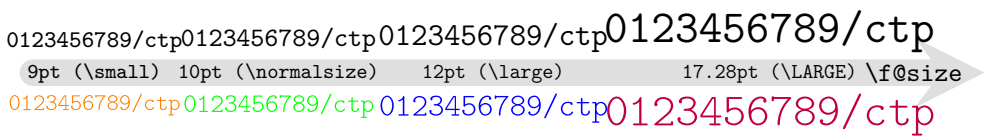
[3.1.2]
```
\chartopath[draw = none, fill = blue]\0
```
0123456789/ctp

[3.1.3]
```
\chartopath[draw = blue, fill = red, scale = 1.2]\0
```
0123456789/ctp

[3.1.4]
```
\chartopath[pattern = north west lines,
            draw, scale = 1.2] \0
```
0123456789/ctp

[3.1.5]
```
\chartopath[draw, fill = green,
            xslant = .15, scale = 1.2] \0
```
0123456789/ctp

Commands like `\small` can be directly applied to `\chartopath`. A comparison of the text and the converted path under 4 different font size is shown in the following figure.



3.2 Fading the path

[3.2.1]
```
\chartopath[hfading = {red, green}]\0
```
0123456789/ctp

[3.2.2]
```
\chartopath[hfading = {blue, -}]\0
```
0123456789/ctp

[3.2.3]
```
\chartopath[hfading = {-, violet}]\0
```
0123456789/ctp

[3.2.4]
```
\chartopath[vfading = {teal, purple}]\0
```
0123456789/ctp

[3.2.5]
```
\chartopath[vfading = {-, orange}]\0
```
0123456789/ctp

[3.2.6]
```
\chartopath[vfading = {violet, -}]\0
```
0123456789/ctp

TEXhackers note: Fading will cost too much time!!!

4 Advanced Usage

4.1 Use in listings

This package can be used within listings or minted, then users could directly copy the source code from .pdf file without worry about including unwanted line numbers.

```
\let\orithelstnumber\thelstnumber
\def\thelstnumber{%
  \chartopath[rescan = false, hfading = {violet, blue}]{\orithelstnumber}%
}
\begin{lstlisting}[language = {[LaTeX]TeX}, gobble = 2]
\documentclass{article}
\begin{document}
  Hello, \LaTeX!
\end{document}
\end{lstlisting}
```

[4.1.1]

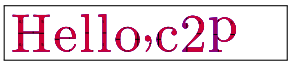
```
1 \documentclass{article}
2 \begin{document}
3   Hello, \LaTeX!
4 \end{document}
```

4.2 String clip

You can also define a command, \strpathclip, that accepts a string instead of a single character:

```
\ExplSyntaxOn
\NewDocumentCommand \strpathclip { O{} m }
  { \int_step_inline:nn { \exp_args:Ne \tl_count:n {#2} }
      { \chartoclip [ #1, offset = {(##1*1, 0pt)} ]
          { \tl_item:nn {#2}{##1}} \kern2pt } }
\ExplSyntaxOff
\strpathclip[scale = 2, font = lmr, image = example-grid-100x100bp]{Hello, c2p}
```

[4.2.1]



4.3 (LA)TEX logo

The logos TEX and LATEX can be recreated using \chartoclip:

```
\def\ctp#1{\chartoclip[font = lmr, image = example-grid-100x100bp]{#1}}
\protected\def\ctpTeX{\ctp T\kern-.1667em \lower.5ex
  \hbox{\ctp E}\kern-.125em \ctp X}
\protected\def\ctpLaTeX{\ctp L\kern-.36em^^A
  {\sbox0 T\vbox to\ht0{\vskip-.021em\hbox{^^A
  \kern.083em\scalebox{.7}{\ctp A}}\vss}}\kern-.15em\ctpTeX}
\scalebox{2}{\ctpTeX} \scalebox{2}{\ctpLaTeX}
```

[4.3.1]

4.4 Get path data

\ctpdata `\ctpdata{⟨font name⟩}{⟨character⟩}`

This function can output the path data (in PGF coordinates). (*Note*[1])

```
\def\0{\ctpdata{lmr}{-}} \scriptsize\ttfamily\detokenize\expandafter{\expanded{\0}}
```

[4.4.1]

```
(7.3025,4.9477)-(7.3025,6.4823)-(0.291,6.4823)-(0.291,4.9477)-cycle
```

[1] [Note]: This command is expandable.

5 Todo List

Warn Resolve the font warning: Font shape `OMS/cmtt/m/n` undefined for `textbackslash`.

Function Support syntax `\chartopath{\usefont{T1}{cmr}{m}{sc} abc}`, `\chartopath{\sffamily abc}`.

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.