

Working with figure environments in texor

by Abhishek Ulayil

Abstract This is a small sample article to demonstrate usage of `texor` to convert figure environments.

1 Introduction

Images are an essential component of any article, however, due to the differences in support for various graphic formats between LaTeX and markdown/HTML we need to fall back on raster graphics. The support for different image formats across markup languages is summarized in Table 1.

Graphics Format	LaTeX	Markdown	Rmarkdown	HTML
PNG	Yes	Yes	Yes	Yes
JPG	Yes	Yes	Yes	Yes
PDF	Yes	No	No	No
SVG	No	Yes	Yes	Yes
Tikz	Yes	No	Yes	No
Algorithm	Yes	No	No	No

Table 1: Image Format support in various Markup/Typesetting Languages

2 Image with width parameters

The following code includes an image with width parameters, producing the output in Figure 1.

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.35\textwidth]{Rlogo-5.png}
  \caption{The logo of R.}
  \label{figure:rlogo}
\end{figure}
```



Figure 1: The logo of R.

This is the most basic example of figure. Similarly PDF figures will be included as rasterized images.

3 Multiple images

Pandoc v3 and above now support a new Figure object (Krewinkel, Lucero, 2023) which supports multiple images side by side or in a grid format. The following subsections demonstrate these capabilities.

Two or more Images side by side

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.45\textwidth]{Rlogo-5.png}\includegraphics[width=0.45\textwidth]{normal}
  \caption{Images side by side}
  \label{fig:twoimages}
\end{figure}
```

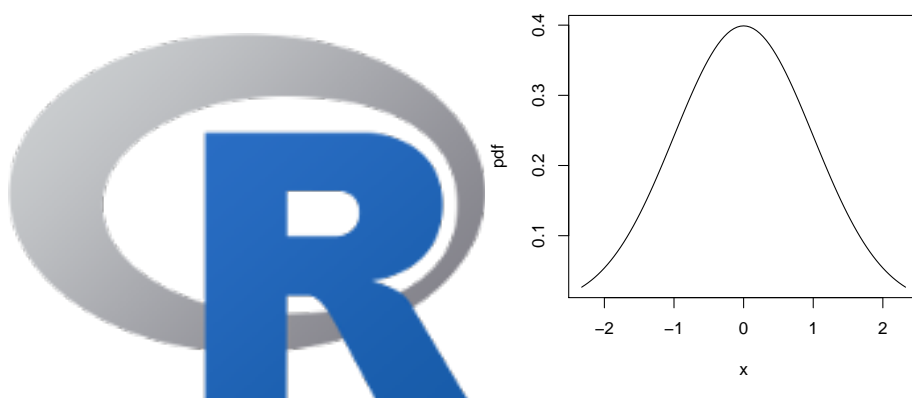


Figure 2: Images side by side

Four Images in a grid

```
\begin{figure}[htbp]
  \centering
  \includegraphics[width=0.45\textwidth]{Rlogo-5.png}\includegraphics[width=0.45\textwidth]{normal}
  \includegraphics[width=0.45\textwidth]{normal}\includegraphics[width=0.45\textwidth]{Rlogo-5.png}
  \caption{Multiple images in a grid}
  \label{fig:fourimages}
\end{figure}
```

4 Tikz images

The **texor** package supports tikz images by rasterizing it and making it web friendly. Figure 4 shows a tikz image adapted from (Cassidy, 2013).

Tikz Code:

The image in Figure 4 is a graphical representation of how **texor** handles tikz images:

- Figures containing tikz images are isolated.
- Tikz libraries are fetched from the wrapper file.
- The Tikz code section is isolated into a standalone LaTeX file and compiled.
- The compiled LaTeX file generates a PDF Image.
- This is converted to PNG format for embedding in the HTML output.
- `\includegraphics{tikz/somefigure.png}` is added to the figure environment.
- A lua filter removes redundant text the figure environment.

If you use **texor** to convert your articles using `texor::latex_to_web()` with `temp_mode=TRUE` (it is `TRUE` by default). The resultant Rmarkdown/HTML file will not modify the contents of your LaTeX file. In this case you can keep reloading the article after making changes to the tikz images, without having to do the above steps manually in case you are converting the article by hand.

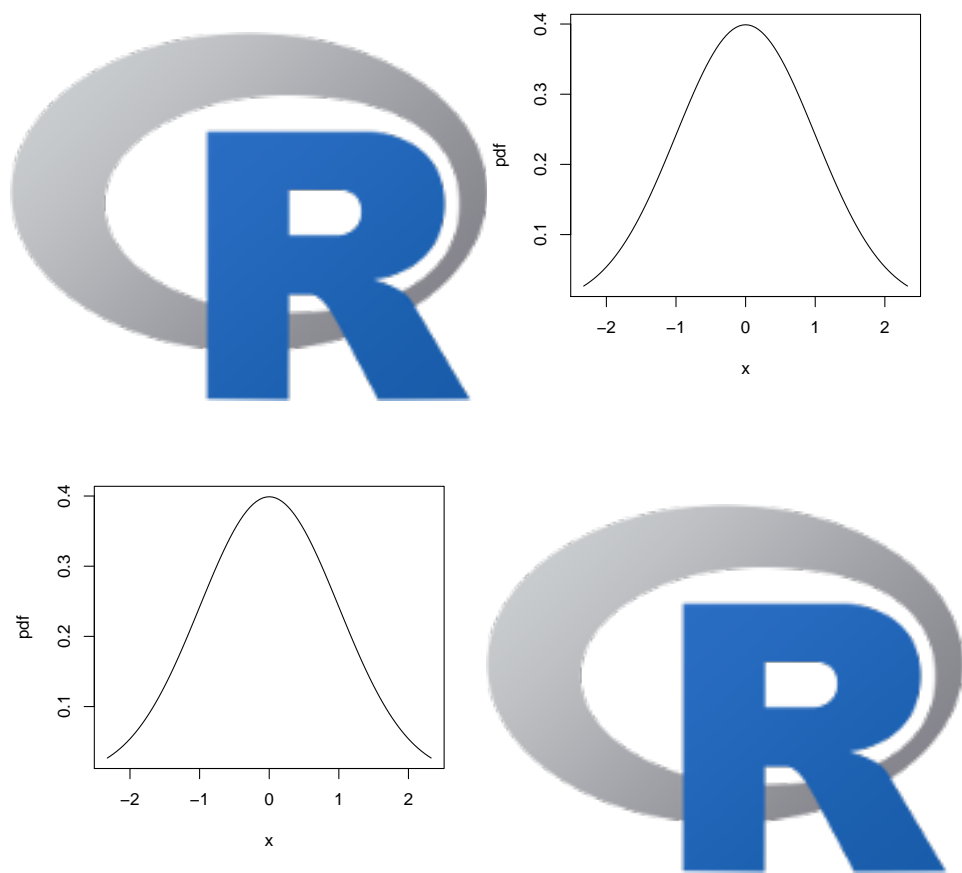


Figure 3: Multiple images in a grid

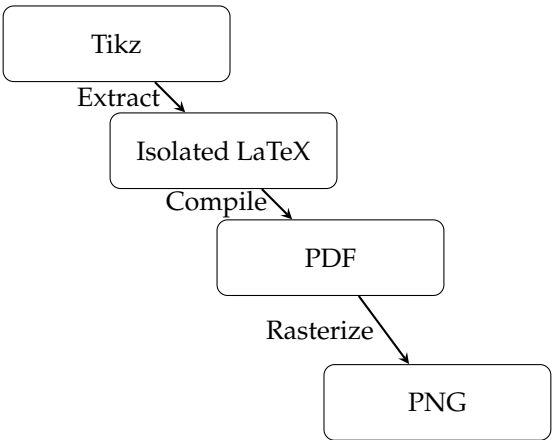


Figure 4: Tikz Image example

5 Algorithm2e diagrams

Diagrams and images using the `algorithm2e` environment are supported, these will be numbered differently. We strongly suggest to use "alg:" in labels for best results. As a part of the filtering requires "alg:" to number algorithm references separately, otherwise they will share references with normal figures.

Algorithm 1 is an example from the `algorithm2e` vignette [Fiorio \(2017\)](#).

Data: this text

Result: how to write algorithm with $\text{\LaTeX}2\text{e}$ initialization;

```
while not at end of this document do
  read current;
  if understand then
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end
```

Algorithm 1: How to write algorithms

6 Other elements in figure objects

Figures can also house non-image environments like code blocks. Code blocks in figure environments would share numbering with normal figures, similar to LaTeX. Here Figure 5 refers to a code block.

```
code_in_figure <- function() {
  if (pandoc_version >= 3) {
    print("code in figure supported")
  }
  else {
    print("code in figure not supported")
  }
}
```

Figure 5: Example Code inside Figure environment

7 Limitations

The limitations of `texor` package in figure handling are:

- Animations using the `animation` package in LaTeX wont be supported. The authors can include GiF equivalents of the same in the Rmarkdown/web article.
- Multi-page PDF files are currently not supported.
- Usage of `subfigure`, `sclaebox`, `wrapfigure` environments will not work as intended.
- Currently only `[width=X.X\textwidth]` format is supported for defining the width in web article (default width would be 100% in other cases).
- Other `\includegraphics[...]` options will be ignored.

8 Summary

In summary the `texor` package supports:

- Multiple images in grid, side-by-side configuration.
- Image Captions with Numbering and Labelling.
- Algorithm2e and tikz figures.

Bibliography

Josh Cassidy LaTeX Graphics using TikZ: A Tutorial for Beginners (Part 3)—Creating Flowcharts
Overleaf tutorials 2013 URL <https://www.overleaf.com/learn/latex/> [p2]

Christophe Fiorio algorithm2e.sty — package for algorithms, release 5.2 CTAN, 2017 URL <https://mirror.kku.ac.th/CTAN/macros/latex/contrib/algorithm2e/doc/algorithm2e.pdf> [p4]

Albert Krewinkel and Aner Lucero pandoc 3.0 Release notes *pandoc* 2023 URL <https://pandoc.org/releases.html> [p1]

Abhishek Ulayil
Student, Institute of Actuaries of India
Mumbai, India
ORCID: 0009-0000-6935-8690