EditEPS Helper

J.C. Martín

October 15, 2023

Introduction

EditEPS Helper is a PyQt5 application whose purpose is to help to use the package editeps. Editeps is a LaTeX package whose purpose is to add LaTeX elements in an EPS figure through LaTeX. It is a package that I created many years ago.

To use editeps with an EPS file, we need to know the bounding box of the EPS figure and give a scaling factor that determines the size of the LaTeX elements that we want to introduce. If we want the LaTeX elements introduced in the EPS file to have the same size as in the text, it is necessary to calculate the scaling factor that editeps uses.

EditEPS Helper reads the bounding box of the selected EPS file and calculates the scaling factor, that way the LaTeX elements inside the EPS file and in the text are the same size. It also gives us an idea of the page, text, and EPS image relations.

Installation and uninstallation

You need Python 3.9. Download the project folder `EditEPSHelper`, create a virtual environment with `pip` and `requirements.txt`, and run the program running the file `EditEPSHelper.pyw` with Python in that virtual environment.

To uninstall the application, delete the folders of the project and the virtual environment. If you want and if it exists, you can also delete the Registry key

`HKET_CURRENT_USER/SOFTWARE/ItacaSoftware/EditEPSHelper`

Or delete the key

`HKET_CURRENT_USER/SOFTWARE/ItacaSoftware`

if `EditEPSHelper` is the only subkey of `ItacaSoftware`.

For the users of Windows 10 and newer, the application is packaged with PyInstaller in the folder `EditEPSHelper`. To install the application, copy the `EditEPSHelper` folder anywhere, and to run the application, double-click on `EditEPSHelper.exe`. To uninstall the application, delete the folder `EditEPSHelper` and then delete the Registry key as above.

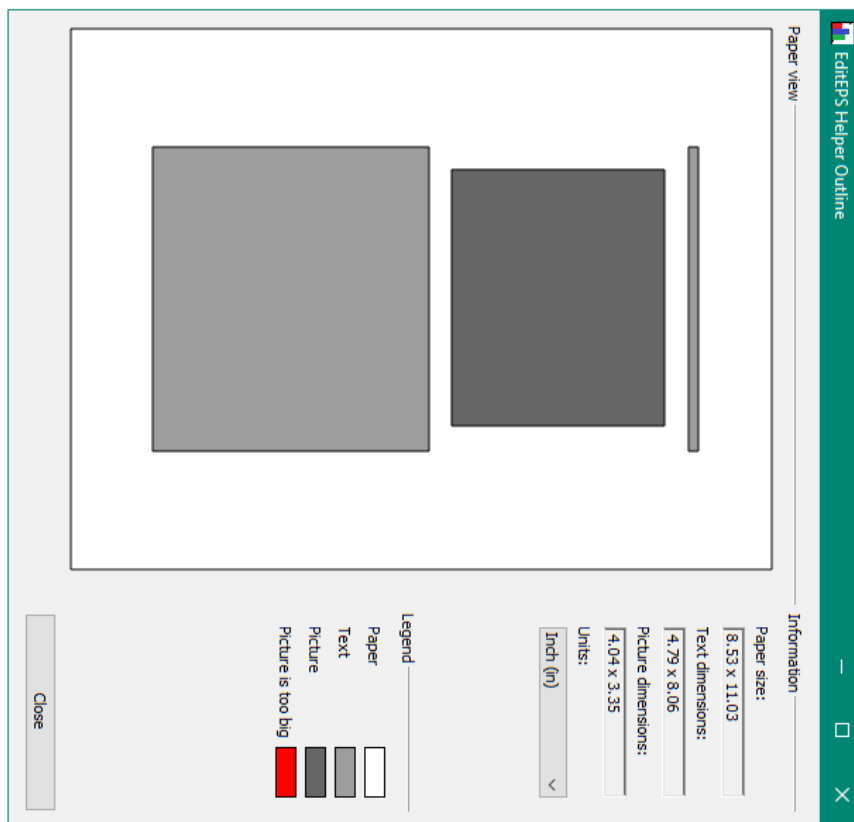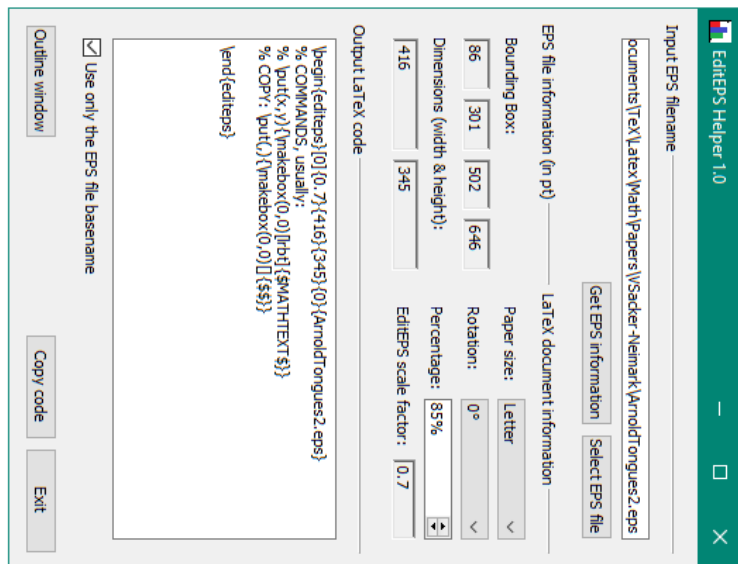In Figure 1 we can see the main and outline windows of the EditEPSHelper application.

Figure 1: EditEPS Helper Main Window and Outline Window

Use

To use the application, select an EPS file using the *Select EPS file* button or write the complete path of the file in the text line of the *Input EPS filename* section. If the EPS file information is not in the *EPS file information* section, click on the *Get EPS information* button. Then the application gives the output LaTeX code in the *Output LaTeX code* section. Verify that the data in the *LaTeX document information* section is the desired. Otherwise, select another paper size, rotation, or percentage. The *EditEPS scale factor* data is calculated by the application, you cannot select it.

The percentage means that the EPS figure width and height are not greater than the percentage of the text width and height, respectively. Also, the EPS figure width is equal to the percentage text width or the EPS figure height is equal to the percentage text height. If the rotation is 90° or 270°, then the figure width and height are interchanged.

With the *Outline window* button you can display a new window that gives you an idea of the relations between paper size, text width and height, and picture width and height. You can see also these dimensions in this window. You can select also the units for these dimensions.

If the EPS file is in the same directory as the LaTeX TEX file, we can check the *Use only the EPS file basename* checkbox.

When you are satisfied, you can click the *Copy code* button that copies the content of the text edit control. This code must be pasted in a LaTeX figure environment like

```
\begin{figure}[t]
\centering

\end{figure}
```

An example

Suppose that we have the EPS file `ArnoldTongues2.eps` with the picture that we can see in Figure 2. We use EditEPSHelper with that EPS file to obtain the EditEPS environment command

```
\begin{editeps}[0]{0.7}{416}{345}{0}{ArnoldTongues2.eps}
% COMMANDS, usually:
% \put(x,y){\makebox(0,0)[lrbt]{$MATHTEXT$}}
% COPY: \put(,){\makebox(0,0)[]{$$}}

\end{editeps}
```

Then we add the following LaTeX commands

```
\begin{figure}[t]
\centering
\begin{editeps}[0]{0.7}{416}{345}{0}{ArnoldTongues2.eps}
% COMMANDS, usually:
% \put(x,y){\makebox(0,0)[lrbt]{$MATHTEXT$}}
```
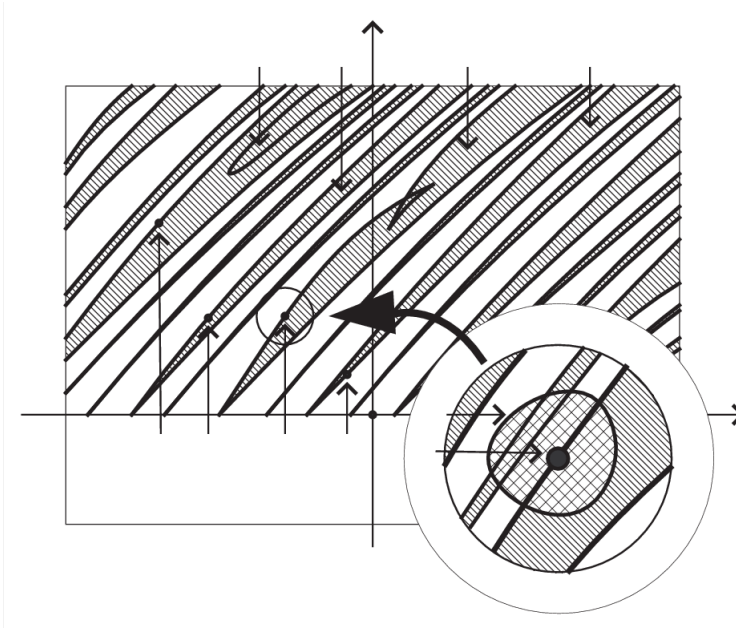
Figure 2: The original EPS picture

```
% COPY: \put(,){\makebox(0,0)[]{$$}}

\put(7.3,4.1){\makebox(0,0)[tl]{$0$}}
\put(8.7,4.3){\makebox(0,0)[r]{$N^n$}}
\put(8.5,3.6){\makebox(0,0)[r]{$\mu^n$}}
\put(12.5,5){\makebox(0,0)[bl]{$L_n$}}
\put(14,4.5){\makebox(0,0)[bl]{$\mu_{k-1}$}}
\put(7.5,11.5){\makebox(0,0)[l]{$\mu_k$}}
\put(5,11.2){\makebox(0,0)[b]{$L_{n-2}$}}
\put(6.6,11.2){\makebox(0,0)[b]{$L_{n-1}$}}
\put(9.1,11.2){\makebox(0,0)[b]{$L_n$}}
\put(11.4,11.2){\makebox(0,0)[b]{$L_{n+1}$}}
\put(3.0,3.8){\makebox(0,0)[t]{$\mu^{n-2}$}}
\put(4.1,3.8){\makebox(0,0)[t]{$\mu^{n-1}$}}
\put(5.5,3.8){\makebox(0,0)[t]{$\mu^n$}}
\put(6.6,3.8){\makebox(0,0)[t]{$\mu^{n+1}$}}
\end{editeps}
\end{figure}
```
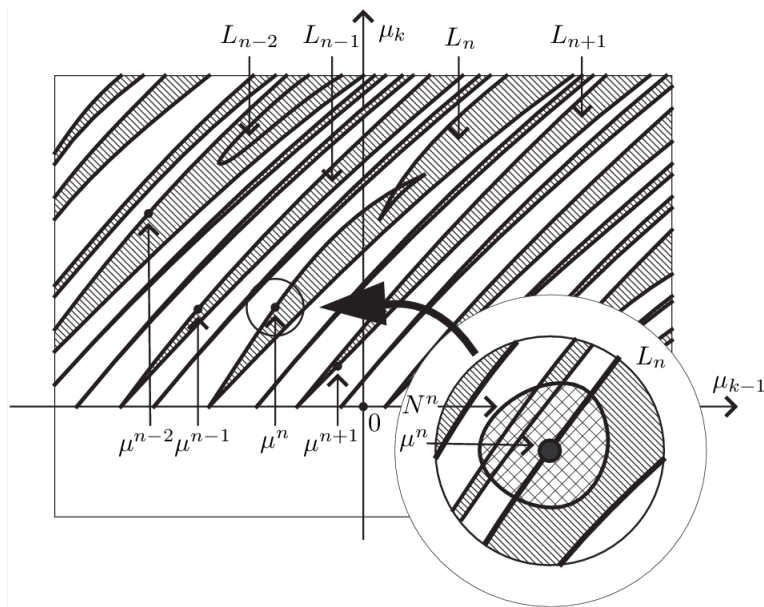
When we run LaTeX we obtain the EPS figure shown in Figure 3.

4

Figure 3: The EPS picture produced by EditEPS