Visual appearance of wireframe objects in special relativity

Thomas Müller, Sebastian Boblest

Version: 2014-09-10

1 Description

The visual appearance of a moving object in special relativity can be constructed in a straightforward manner when representing the surface of the object, or at least a wire frame model of it, as a point cloud. The apparent position of each individual point is then found by intersecting its worldline with the observer's backward light cone. In this paper, we present a complete derivation of the apparent position of a point and some more complex geometric objects for general parameter settings (configurations). We implemented our results in python and asymptote and used these tools to generate scripts that create the figures in this paper. These scripts are directly applicable in an undergraduate course to special relativity and can also serve as the basis for student projects with the aim to study more complex sceneries.

Requirements

- asymptote (`http://asymptote.sourceforge.net`)
- python 2.7, matplotlib, numpy, scipy

  `http://matplotlib.org`

  `http://www.numpy.org`

  `http://www.scipy.org`

2 Installation

Both our python and asymptote scripts can be used under Linux and under Windows. However, if you have the choice between these two operating systems, we recommend to use Linux.

2.1 Installations for Windows

2.1.1 Python 2.7

We assume that you do not have installed python 2.7 yet. Please note that you need an archive tool that can handle `.tar.gz` files. E.g. you could install 7-zip from http://www.7-zip.org

1. Download python 2.7.X from https://www.python.org and run the install script. You can follow the standard installation instructions. This installs python in the standard path: `C:\Python27`

   To test that the following steps work, start `Python (command line)`. (Alternatively, double-click on `python.exe` within the `C:\Python27` folder.)

2. Download numpy. The following steps bring you to the sourceforge download page:

   www.numpy.org → Getting Numpy → SourceForge site for NumPy → NumPy → Looking for the latest version

   → `numpy-1.8.1-win32-superpack-python2.7.exe`

   Download and run the exe file.

   Enter: "`import numpy`" in the python command line. If there is no error message, the numpy library should have been installed correctly.

3. Download scipy. The following steps bring you to the sourceforge download page:

   www.numpy.org → Getting Numpy → SourceForge site for SciPy → scipy → Looking for the latest version

   → `scipy-0.14.0-win32-superpack-python2.7.exe`

   Download and run the exe file.

   Enter: "`import scipy`" in the python command line. If there is no error message, the scipy library should have been installed correctly.

4. Additional libs necessary for matplotlib:

   - pyparsing. This lib can be downloaded from https://pypi.python.org/pypi/pyparsing → `pyparsing-2.0.2.win32-py2.7.exe` Download and run the exe. Test it with "`import pyparsing`".
   - dateutil. From https://pypi.python.org/pypi/python-dateutil. Download the file `python-dateutil-2.2.tar.gz`. Open this file with, e.g., 7zip, enter the folder `python-dateutil-2.2`, and copy the folder `dateutil` to `C:\Python27\lib\site-packages\`
   - six. From https://pypi.python.org/pypi/six. Download the file `six-1.7.3.tar.gz`. Open this file with e.g. 7zip and unpack `six-1.7.3` into the `C:\Python27\site-packages\` directory. Open

a command prompt and go to `C:\Python27\Lib\site-packages\six-X.X.X` and run `setup.py install`.

Test it with "`import six`" within the python command line.

5. Download matplotlib. Go to http://matplotlib.org/downloads.html, download and run the latest stable version for `python2.7`, currently: `matplotlib-1.4.0.win32-py2.7.exe`

6. Run the examples from the paper by a double-click on the corresponding `.py` file.

2.1.2 Asymptote / GSview

Asymptote needs a Latex installation, so if you do not have Latex installed, go to http://www.miktex.org → Download → `Basic MiKTeX 2.9.5105 Installer` to install MiKTeX. Then, download the latest asymptote version from http://sourceforge.net/projects/asymptote/files and run the exe.

You also have to install GSview and ghostscript to automatically view the `.eps` files generated by asymptote. Self extracting archives can be found here:

- GSView: http://pages.cs.wisc.edu/~ghost/gsview/get50.htm
- Ghostscript: http://sourceforge.net/projects/ghostscript → Files → GPL Ghostscript

Open a command prompt and change to the directory with the asymptote files. Then, just run e.g. `asy appCircle.asy`

2.2 Installations for Linux

Both, asymptote and the required python libraries are part of the standard linux distributions of Ubuntu and Fedora and can easily be installed with the corresponding package managers `aptitude` or `yum`.

2.2.1 Python 2.7

You can either install python by means of your package manager or from the sources.

Installing via package manager (ubuntu) You only have to run: `sudo apt-get install python-numpy python-scipy python-matplotlib ipython`

Installing from sources (experts only!) Download python and all the necessary packages manually and install them in `/usr/local/python/2.7.X` (X can be any number). To compile `numpy` you need a Fortran 77 compiler (e.g. gfortran). We also assume to work in a bash shell on an ubuntu system.

1. Download python 2.7.X from https://www.python.org and unpack the `.tar.xz` archive via `tar xJvf Python-2.7.X.tar.xz`

   For the rest of this installation, we use python 2.7.8.

   Change directory into the `Python-2.7.8` directory and run

   ```
   ./configure --prefix=/usr/local/python/2.7.8
   make
   sudo make install
   ```

   Set your `PATH` variable: `export PATH=/usr/local/python/2.7.8/bin:$PATH`

2. Download numpy. The following steps bring you to the sourceforge download page: `www.numpy.org` → Getting Numpy → SourceForge site for NumPy → NumPy → Looking for the latest version → `numpy-1.8.1.tar.gz`

   Unpack the archive file via: `tar xvf numpy-1.8.1.tar.gz`

   run `python setup.py build [--fcompiler=gfortran]`

   `sudo /usr/local/python/2.7.8/bin/python setup.py install`

3. Download SciPy. SciPy can be found next to NumPy: → `scipy-0.14.0.tar.gz`

   Unpack the archive file via: `tar xvf scipy-0.14.0.tar.gz`

   run `python setup.py build`

   `sudo /usr/local/python/2.7.8/bin/python setyp.py install`

4. Download matplotlib from http://matplotlib.org/downloads.html → `matplotlib-1.4.0.tar.gz`

   Unpack the archive file via `tar xvf matplotlib-1.4.0.tar.gz`

   run: `python setup.py build`

   `sudo /usr/local/python/2.7.8/bin/python setyp.py install`

2.2.2 Asymptote

You can either install asymptote by means of your package manager or from the sources.

Installing via package manager (ubuntu) You only have to run

```
sudo apt-get install asymptote
```

Installing from sources (experts only!)

1. Download the sources from http://asymptote.sourceforge.net → Download 2.32 → Files → 2.32 → `asymptote-2.32.src.tgz`

2. Unpack the archive via `tar xvf asymptote-2.32.src.tgz`

   run

   `./configure --prefix=/usr/local/asymptote/2.32/`

   `make`

   `sudo make install`

3. Set your `PATH` variable:

   `export PATH=/usr/local/asymptote/2.32/bin:$PATH`

3 Examples from the paper

If no file ending is given in this list, the corresponding script exists identically in an asymptote and a python version. For reference we also give the corresponding figure in the paper.

- `appDie` (`animDie.py`) [figure 4b)]

  A relativistically moving die over a row of static dice at three different observation times. In the animated version the die moves over the row of static dice. The observer is located at $(8, 12, 2)$. Change this to create figure 4a).

- `appCircle` (`animCircle.py`) [figure 5]

  Two-dimensional sketch of the apparent shape of a circle moving with $\beta = 0.9$ at different observation times compared to circles at rest. In the animated version the circle moves over the row of static circles.

- `appSphere`, `appSphereZ.asy` [figure 6]

  A sphere moving with $\beta = 0.95$ over three static spheres. In the script appSphereZ.asy depth information is also included. This version was used to create the figure in the paper.

- `appRod` (`animRod.py`) [figure 7]

  Illustration of the bending of a rod moving perpendicularly to its orientation and towards the observer, together with the circle of curvature at its centre. The animated version shows how the apparent hyperbola degenerates to a corner when the observer is reached and how the curvature radius increases again afterwards.

- `appRodLight` (`animRodLight.py`) [figure 8]

  Illustration of the light rays originating from the abovementioned rod at different times and how this leads to the apparent shape of a hyperbola. This visual impression is compared to the actual position of the rod at observation time. In the animated version the different light emission times are further emphasized.

- `appRodView` (`animRodView.py`) [figure 9]

  Apparent shape of a rod moving towards the observer with $\beta = 0.9$. In the animated version we observe this situation until the rod moves out of view.

- `appCube` [figure 10b)]

  Apparent distortions of a cube moving by the observer very closely with $\beta = 0.9$. Change the value of β to create figure 10a).

- `appSphereSingle`, `appSphereSingleZ.asy` [figure 11b)]

  Apparent distortions of a sphere moving by the observer very closely with $\beta = 0.9$. In the script appSphereSingleZ.asy depth information is also included. This version was used to create the figure in the paper. Change the value of beta to create figure 11a).

4 Contact

Dr Thomas Müller,
Dr Sebastian Boblest

Visualization Research Center (VISUS)
University of Stuttgart
Allmandring 19
70569 Stuttgart, Germany

Email: `thomas.mueller@visus.uni-stuttgart.de`
`sebastian.boblest@visus.uni-stuttgart.de`

http://go.visus.uni-stuttgart.de/srwireframe