

## Rendering Data Cubes with Blender

Brian R. Kent, NRAO

<http://www.cv.nrao.edu/~bkent/blender/>

Described here are simple steps to render a FITS data cube with Blender.

Data are obtained from:

<http://www.mpia-hd.mpg.de/THINGS/Data.html>

*Walter, F., Brinks, E., de Blok, W. J. G., et al. 2008, AJ, 136, 2563*

A simple example to start with utilizing a series of TIF files can be found from a CAT scan at the Stanford Volume Data Archive:

<https://graphics.stanford.edu/data/voldata/>

Further examples can be found in the book:

[http://www.amazon.com/dp/1627056114/ref=cm\\_sw\\_su\\_dp](http://www.amazon.com/dp/1627056114/ref=cm_sw_su_dp)

<http://iopscience.iop.org/book/978-1-6270-5612-0>

## FITS file preparation

1. The accompanying Python script can be used to load a FITS data cube.

The user will need PYFITS (or cfitsio) and Matplotlib. These can be obtained from:

- <http://www.scipy.org/>
- [http://www.stsci.edu/institute/software\\_hardware/pyfits](http://www.stsci.edu/institute/software_hardware/pyfits)
- <https://github.com/esheldon/fitsio>

## Set the Material

1. Start Blender (Double click the icon or ./blender from the command line.)
2. Right click to select the default Cube object. This will act as the data container.
3. Scale the data container to match the data cube dimensions in the **Transform** dialog on the right hand side of the GUI.
4. Click on the Material Tab on the far right side of the GUI.
5. Click the “+” button to add a new Material
6. Choose **Volume**.
7. The graphic densities have been set in the accompanying blend file. For different data these will have to be scaled appropriately by the user.
8. Change the **graphic density** to 0.0 and **density scale** to 2.0.
9. Under **Shading** set the **emission** to 0.0, and **scattering** to 1.4.

## Load the material into the Texture

1. Click the Texture Tab.
2. Click the “+ New” button to create a new Texture.
3. From the **Type** drop down box choose “**Voxel Data**”.
4. Check the “**Ramp**” box and set the far right ramp color to that of your choosing.
5. Under the **Voxel Data** section, Open the first file in the image sequence (generated by the Python script)
6. Change the **File Format** and **Source** to **Image Sequence**.
7. Under “**Mapping**” change the **Projection** to **Cube**.
8. Under “**Influence**” select the **Density**, **Emission**, and **Emission Color** and set their values to 1.0

Errata update for October 2015 from YouTube comments:

<https://www.youtube.com/watch?v=3GvTTVEeEmk>

*One detail that wasn't obvious at to me first: for me (on my own dataset), it didn't work until I set "Frames" to however many images were in the TIF stack, under the Voxel Data section of Textures (in this video, at [2:15](#), you can see the value "(59) Frames: 100" near the bottom right).*

*Even though it's a series of TIFs and not a movie, Blender still needed the input of how many images to expect, a value it still considers "Frames." Maybe it auto-detected the number of images on the dataset you use here, but it did not for me, so I hope this comment can be helpful to other people.*

## Cube animation: Simple Rotation

1. Right click to select the Cube mesh object.
2. Press the “I” key to insert a “**Rotation**” keyframe.
3. At the bottom of the GUI, change the frame to halfway between the first and final frame. In the example blend file, this is set to 300 frames. Choose frame 150.
4. Rotate the asteroid mesh by 180 degrees. This can be accomplished in the **Transform** dialog by entering 180 degrees about the axis of rotation chosen by the user.
5. Press the “I” key to insert another Rotation keyframe.
6. Rotate the asteroid mesh to 360 degrees. This can be accomplished in the **Transform** dialog by entering 360 degrees about the axis of rotation chosen by the user.
7. Press the “I” key to insert a final **Rotation** keyframe.

## Animating and Rendering the Sequence

1. To see a preview of the animation, Choose *View->Camera* (last option) and then click the Play button at the bottom of the GUI. Press the square Stop button to halt the animation.

2. Choose the **Render** tab.
3. At the bottom of the Render dialog, Change the output to “*AVI JPEG*”. The “**Stamp**” option is useful as it prints metadata about the animation over the video. Scroll back to the top of the dialog and Click **Animate**. A video file will be generated.