lua-tikz3dtools documentation v3.3.0

Jasper Nice

2026-05-24

Contents

1 Introduction

Hi! My name is Jasper, and before I get started with this manual, I wanted to share a bit about where it came from, and where it is going. This is because, like anyone, I make mistakes, and I made quite a few here along the way. I want to acknowledge that openly, so no one is misled, and so that people can have the greatest benefit from this.

Much of the mistakes were the byproduct of a premature release of the software before it was complete. I have now finished what I believe are the core components of the software. I'm sorry for having the incomplete work up for so long.

I'd really recommend taking a look at the packages luadraw, or Asymptore if anyone is interested in a contemporary LaTeX 3D illustration softwares.

This package—lua-tikz3dtools—is geared strongly towards expressing everything in terms of parametric affine/projective algebra. Sounds fancy, but it is really the backbone of a lot of seemingly incoherent elements.

For instance, rotation, translation, perspective, shearing, and reflection are all projective transformations. And they can be composed!

Of course, everything eventually boils down to affine algebra after the transformations, as they map affine space onto itself, resulting in affine-workable

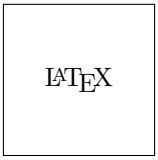

Figure 1: ltdtappendlabel

byproducts. This is a bit abstract, but it is the reason you can look at a perspective picture on your screen, while your screen never changes its basis (orientation, direction, size, and shape).

This package also takes care of occlusion and partitioning, with results akin to those of a BSP tree.

2 Command reference

2.1 ltdtappendlabel

This command appends a label. Its keys are

- v: a homogeneous 3D vector (e.g., `Vector:new{1,2,3,1}`).
- text: a LaTeX label (e.g., `\LaTeX`).
- transformation: A homogeneous 3D transformation matrix.
- filter: A boolean on v.

2.2 ltdtappendlight

This command appends a light direction. I recommend using only one light direction, unless you have a good reason not to.

- v: a homogeneous 3D vector

2.3 ltdtappendcurve

This command appends a curve.

- uparams: A 3-vector of start, stop and step in u
- v: a homogeneous 3D vector
- transformation: A homogeneous 3D transformation matrix.
- draw options: a string of Ti*k*Z options for a path.
- arrow tip: a string of Ti*k*Z options for a path.

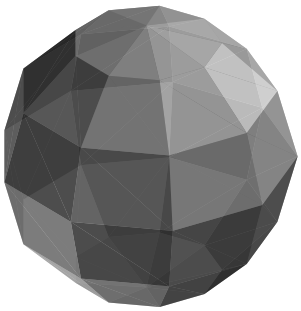

Figure 2: ltdtappendlight

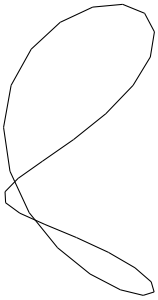

Figure 3: ltdtappendcurve

- arrow scale: a number to change the arrow's size
- filter: A boolean on v.

2.4 ltdtappendsurface

This command appends a surface.

- uparams: A 3-vector of start, stop and step in u
- vparams: A 3-vector of start, stop and step in v
- v: a homogeneous 3D vector
- transformation: A homogeneous 3D transformation matrix.
- fill options: a string of Ti*k*Z options for a path.
- filter: A boolean on v.
- curve: If you know what you're doing, you can embed a table of UV line segments within a surface's simplices. See the example below.

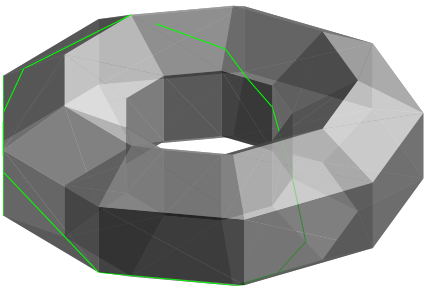

Figure 4: ltdtappendsurface

Figure 5: ltdtappendtriangle

2.5 ltdtappendtriangle

This command appends a triangle.

- m: A 3-vector of start, stop and step in u
- transformation: A homogeneous 3D transformation matrix.
- fill options: a string of Ti*k*Z options for a path.
- filter: A boolean on m.

2.6 ltdtappendsolid

This command appends the surface boundaries of a solid mapped from a cube.

- uparams: A 3-vector of start, stop and step in u
- vparams: A 3-vector of start, stop and step in v
- wparams: A 3-vector of start, stop and step in w
- v: a homogeneous 3D vector
- transformation: A homogeneous 3D transformation matrix.

Figure 6: ltdtappendsurface

- fill options: a string of Ti*k*Z options for a path.
- filter: A boolean on v.

2.7 ltdtdisplaysimplices

This command resolves the appended simplices, and displays the resulting output.

2.8 ltdtsetobject

This command sets a lua object

- name: the lua name of the object.
- object: the lua object (e.g., a function, or a matrix).