Embedding some `misc3d` figures in a PDF as interactive 3D objects

Luke Tierney[1] and Dai Feng[2]
[1]Department of Statistics and Actuarial Science
University of Iowa
[2]Biometrics Research
Merck & Co., Inc.

April 2, 2012

Recent versions of PDF and of the Acrobat reader support inclusion of 3D formats for representing interactive 3D models. There are two file formats supported called U3D and PRC. Both are complex binary formats, so it is useful to find simple textual formats for representing 3D objects along with software for converting these to U3D or PRC. This vignette describes how this support can be used to include interactive versions of 3D contour plots and other triangular mesh scenes produced by the `misc3d` package. So far there are three formats available in `misc3d` through the function `exportScene`, one that captures only the geometry and the other two that captures geometry along with color and transparency.

1 Geometry-only images using MeshLab and U3D

MeshLab is an open source system for the processing and editing of unstructured 3D triangular meshes. It can read and write several formats representing such meshes. An easy format to write that MeshLab can read is the OFF format. For OFF format, the function does not attempt to write out color, transparency or material properties. Color and transparency are supported in OFF format but MeshLab doesn't seem to write them properly into the U3D, though some versions are capable of reading OFF files with color and transparency (COFF). The OFF file produced can then be read in by MeshLab and then exported in U3D format. The `movie15` LaTeX package can then be used to include the image in a PDF document created with `pdflatex`.

Figures 1 and 2 show two images created by this approach, one shows nested contours of the density of a mixture of three-dimensional normals, the other shows the volcano surface from the R distribution. The normal mixture contours were created with a variant of code from the `contour3d` help page:

```
R> nmix3 <- function(x, y, z, m, s) {
+     0.4 * dnorm(x, m, s) * dnorm(y, m, s) * dnorm(z, m, s) +
+     0.3 * dnorm(x, -m, s) * dnorm(y, -m, s) * dnorm(z, -m, s) +
+     0.3 * dnorm(x, m, s) * dnorm(y, -1.5 * m, s) * dnorm(z, m, s)
+ }
R> f <- function(x,y,z) nmix3(x,y,z,.5,.5)
R> gs1 <- function(n = 40, k = 5, cmap = heat.colors, ...) {
+     th <- seq(0.05, 0.2, len = k)
+     col <- rev(cmap(length(th)))
+     x <- seq(-2, 2, len=n)
+     m <- function(x,y,z) x > .25 | y < -.3
+     contour3d(f,th,x,x,x,color=col, mask = m, engine = "none",
+               scale = FALSE, ...)
+ }
```

(nmix.u3d)

Figure 1: Embedded U3D image showing nested contours of the density of a mixture of three tri-variate normal distributions.

.

```
R> conts <- gs1(40, 5, screen=list(z = 130, x = -80),
+              color2 = "lightgray", cmap=rainbow)
R> exportScene(conts, "nmix", "OFF")
```

The volcano example was created with

```
R> z <- 2 * volcano
R> x <- 10 * (1:nrow(z))
R> y <- 10 * (1:ncol(z))
R> vtri <- surfaceTriangles(x, y, z, color="green3")
R> exportScene(vtri, "volcano", "OFF")
```

When using the `movie15` package, to vary the view of embedded 3D objects the options in `includemovie` need to be adjusted by specification of different entries or use a predefined view file with extension `.vws`. See the manual of `movie15` for more details and the accompanied source file `misc3d-pdf.Rnw` for examples.

2 Images including color and transparency using IDTF or Asymptote

Compared with the OFF format, the *Intermediate Data Text Format* (IDTF) and the Asymptote support not only geometry but color and transparency as well. When every triangular mesh surface has only one color and transparency level, we suggest using the IDTF since the size of 3D objects it produces is much smaller; otherwise, use the Asymptote format.

(nmix.u3d)

Figure 2: Embedded U3D image of the R volcano surface data.

.

The IDTF is another textual format that can be transformed into the U3D format. As an example, to use color to distinguish multiple contour surfaces as shown in Figure 3, we need to export the contour surfaces

```
R> exportScene(conts, "nmix-color", "IDTF")
```

and then convert it to U3d file using the IDTF converter.

Transparency can also be used to show multiple contour surfaces. This is illustrated in Figure 4. The additional code to produce this figure is

```
R> g1 <- function(n = 40, k = 5, alo = 0.1, ahi = 0.5, cmap = heat.colors) {
+     th <- seq(0.05, 0.2, len = k)
+     col <- rev(cmap(length(th)))
+     al <- seq(alo, ahi, len = length(th))
+     x <- seq(-2, 2, len=n)
+     contour3d(f,th,x,x,x,color=col,alpha=al, engine = "none")
+ }
R> aconts <- g1(40,5)
R> exportScene(aconts, "nmix-alpha", "IDTF")
```

As another example, Figure 5 shows the contour of a brain from an MR image along with several contours of the activation level measures in a PET experiment.

Asymptote is a powerful vector graphics language for 2D and 3D graphics. It is inspired by MetaPost and generalizes MetaPost path construction algorithms to three dimensions. It is possible to write out an asymptote program to represent a triangular mesh scene produced by `misc3d`. As an example, the code

(nmix-color.u3d)

Figure 3: Color version of nested density contours.

```
R> vtri <- surfaceTriangles(x, y, z, color="green3")
R> cvtri <- updateTriangles(vtri, color = function(x,y,z) {
+     cols <- terrain.colors(diff(range(z)))
+     cols[z - min(z) + 1]
+ })
R> exportScene(cvtri, "volcano", "ASY")
```

adds coloring to the volcano surface. The command

```
./asy -inlineimage -tex pdflatex volcano.asy
```

then produces a PRC file that can be included in a LaTeX document. The result is shown in Figure 6. The code produced seems to require a fairly recent version of Asymptote; we used version 1.86. Embedding in LaTeX also requires a PCR-aware version of the `movie15` package; such a version is included in the Asymptote distribution.

The PCR files produced by this approach are huge. We are looking into possible ways to reduce them. This is not an easy task possibly because of the file sizes the images do not immediately load when a page is shown in the reader. Also we have observed a few crashes of the reader when the file includes a number of large embedded 3D images.

The current available document on the IDTF and U3D seems to be limited. For the IDTF format, we found a way to produce a single mesh surface with different colors like in the volcano example in Figure 6. However, since the size of IDTF files and corresponding U3D files are huge, we didn't incorporate that function into the package. For both the IDTF and Asymptote, questions remained to be solved are whether it is possible to have different colors on the two sides of a facet and how to specify other material properties.

(nmix-alpha.u3d)

Figure 4: Nested density contours with colors and transparency.

3 Other approaches

One other option using Open Source tools is to write out the objects in JVX format (an XML variant) and convert with jReality, but that also doesn't seem to write out colors for U3D properly.

The Adobe 3D software available as part of Acrobat Pro Extended may be able to read OFF files and/or produce smaller PCR files. We are looking into this.

(brain.u3d)

Figure 5: MRI contour of a brain along with several contours of the activation level in a PET experiment.

(vol.prc)

Figure 6: Color version of the volcano surface.