Correlated Color Temperature Isotherms

Glenn Davis <gdavis@gluonics.com>

March 2, 2020

Introduction

The *Correlated Color Temperature* of a light source is the temperature of the Planckian (black-body) radiator whose chromaticity is the closest to that of the light source. The chromaticity space is the CIE uv version of 1960. The goal of this **spacesXYZ** vignette is to compare the (slightly different) isotherms of different CCT methods. Featured functions in this vignette are: `planckLocus()` and `CCTfromuv()`.

The *reference Planckian locus* is defined spectrally - from the famous equation for the Planckian radiator and from the tabulated CIE 1931 standard observer color matching functions. It is a C^∞ curve in a 2D chromaticity space, usually either xy (1931) or uv (1960). Since this package does not deal with spectra, we use good approximations of the reference locus instead. The default approximation is a spline (using `stats::splinefun()` with `method="fmm"`) through the 31 uv (1960) locus points in Robertson [Rob68] and Wyszecki and Stiles [WS82] page 228. This spline does not appear in [Rob68], but I think Robertson would approve of it. It has C^2 continuity and is in very good agreement with the reference locus. The maximum RMS error in the uv-plane is about 7.3×10^{-6} over the temperature interval [1667,∞] K. A similar piecewise-linear interpolating path has a maximum RMS error about 10 times larger. The 31 uv values in the table are accurate to the given 5 decimal places. The locus is parameterized directly by reciprocal color temperature ($10^6/T$), and therefore indirectly by T. We call either of these the *native pameterization* of the locus. The lines that are perpendicular to the locus are the *native isotherms*, and we use `stats::splinefun()` with `deriv=1` to calculate them.

A second high-resolution locus is available. It is a quintic spline through 65 points. It is not discussed here; for details see the **Reference manual**. Like the Robertson locus, it also has *native isotherms*.

Two more families of isotherms are available. The Robertson isotherms are tabulated in [Rob68] just like the points on the locus, and linear interpolation is used for intermediate temperatures. The McCamy isotherms in [McC92] and [McC93] are defined by a single cubic rational function in xy, and no interpolation is necessary. Each isotherm family induces a slightly different parameterization of the locus - the temperature at a locus point is the temperature of the isotherm passing through that point. The Robertson parameterization is only continuous - C^0, but the *geometric continuity* class is G^2. The McCamy parameterization is as smooth as the locus, which is C^2. For the Robertson and native isotherms the valid temperature interval is [1667,∞] K. For the McCamy isotherms the valid temperature interval is at most [1621,34530] K, and may be smaller depending on the locus.

```
library( spacesXYZ )
```

1 The Planckian Locus, and 3 Families of Isotherms

```
mired = seq( 0, 600, by=10 )
temp  = 1.e6/mired
uv = planckLocus( temp, param='native' )     # a spline curve
par( omi=c(0,0,0,0), mai=c(0.5,0.5,0.1,0.1) )
plot( c(0.15,0.32), c(0.245,0.37), type='n', xlab='', ylab='', las=1, tcl=0,
      lab=c(10,8,7), mgp=c(3,0.25,0), asp=1 )
title( xlab='u', line=1.5 ) ;  title( ylab='v', line=1.5 ) ; grid( lty=1 )
lines( uv[ ,1], uv[ ,2], lwd=0.7 )
text( uv[1,1], uv[1,2], expression(infinity), adj=c(0.6,1.5), cex=1.5 )
#  draw all 3 families of isotherms
temp = c( seq(2000,10000,by=1000), 2500, 20000, 30000, Inf )
param = c('native','Robertson','McCamy') ; color = c('black','red','mediumseagreen')
for( i in 1:3 ) {
delta = 0.05 - (i-1)*0.00275
top = planckLocus( temp, param=param[i], delta=delta )
bot = planckLocus( temp, param=param[i], delta=-delta )
segments( bot[ ,1], bot[,2], top[,1], top[,2], col=color[i], lwd=1 )
if( i == 1 ) text( bot[ ,1], bot[,2], sprintf("%gK",temp), adj=c(0,1.2), cex=0.5 )
}
legend( 'bottomright', param, lwd=3, bty='n', col=color, inset=0.05 )
```
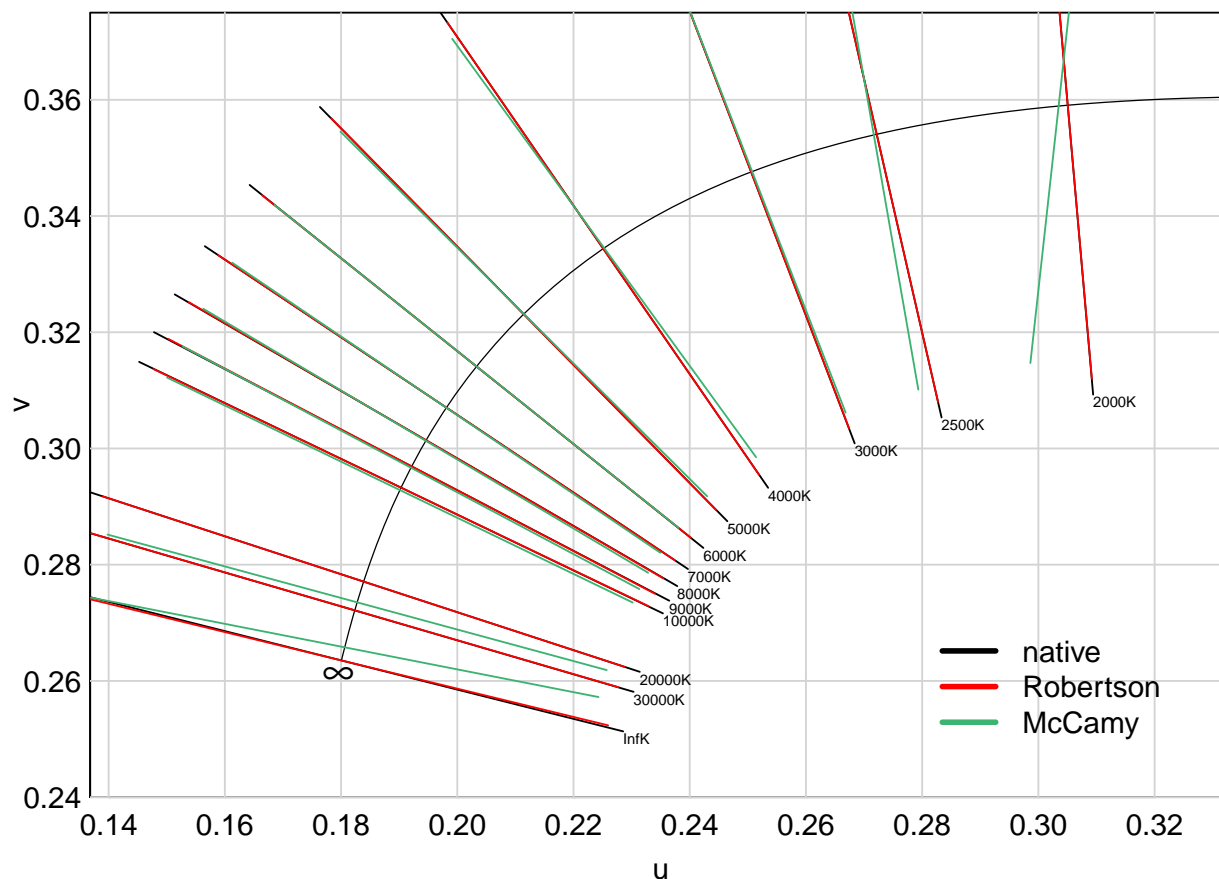

Figure 1: A Comparison of 3 families of isotherms.

In this figure, the 'native' isotherms are drawn with a length of `delta=0.05` on either side. The

other two are shrunk slightly to make it easier to see the differences. Zoom in and see that the agreement between native and Robertson is very good; it is only noticeable at ∞K. The McCamy isotherms agree well near 6000K but diverge somewhat further away. We now examine the differences in intersection with the locus, but not the differences in slopes, more closely.

```
temp = sort( c( seq(2000,8000,by=100), 6667, 5714, 4444, 3636, 3333, 3077, 2857 ) )
uv = planckLocus( temp, param='native' )
diffRob = CCTfromuv( uv, isotherms='Robertson' ) - temp
diffMcC = CCTfromuv( uv, isotherms='McCamy' ) - temp
par( omi=c(0,0,0,0), mai=c(0.5,0.6,0.1,0.1) )
plot( range(temp), range(diffRob,diffMcC), type='n', las=1, tcl=0, xlab='', ylab='',
        lab=c(10,5,7), mgp=c(3,0.25,0) )
title( xlab='native temperature K', line=1.5 )
title( ylab='temperature delta K', line=1.8 )
grid( lty=1 ) ; abline( h=0 )
lines( temp, diffRob, col=color[2] )
lines( temp, diffMcC, col=color[3] )
legend = sprintf( "%s - native", param[2:3] )
legend( 'bottom', legend, lwd=3, bty='n', col=color[2:3], inset=0.05 )
```
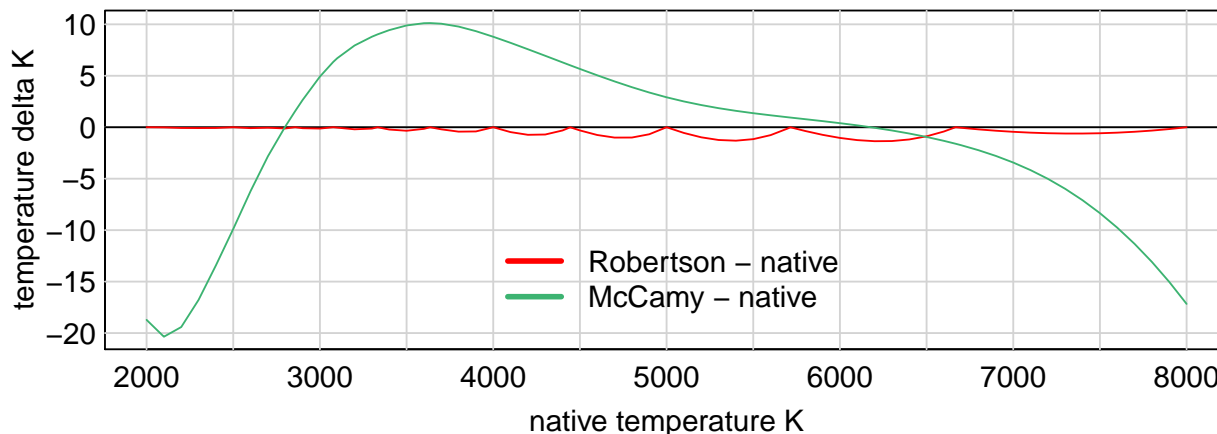

Figure 2: A Comparing the Robertson and McCamy parameterizations to the native parameterization. This comparison is only along the Planckian locus.

The temperatures in the Robertson lookup table (5000K, 5714K, 6667K, 8000K, etc.) are apparent here, and also the fact that the Robertson parameterization of the locus is not C^1 at those points. It is also apparent that the McCamy isotherms are optimized for the range 5000K to 7000K, and that McCamy and Robertson agree best (the curves intersect) near 6500K. We emphasize that these differences are only for points on the locus. For points off the locus the difference can be greater, as seen in Figure 1 near 2000K.

2   The McCamy Correlated Color Temperature Function

The McCamy function in CIE 1931 (x, y) chromaticity coordinates is

$$CCT(x, y) := p\left(\frac{x - 0.3320}{y - 0.1858}\right) \quad \text{where} \quad p(t) := -449t^3 + 3525t^2 - 6823.3t + 5520.33$$

This is the "second equation" in [McC92]. The form implies that in the (x, y) plane all isotherms are lines through the point $(x, y) = (0.3320, 0.1858)$. Since (u, v) is a projective function of (x, y), the isotherms in (u, v) are lines through a point too. Let's verify this with another plot. First we compute the point of intersection in (u, v) and draw a circle (in red) centered at the point and passing through the locus at $T = 6500$K. Then we plot the locus and isotherms as before.

```
uv0 = uvfromxy( c(0.3320,0.1858), 1960 ) ; uv1 = planckLocus(6500,param='native')
rad = uv1 - uv0 ; rad = sqrt(sum(rad*rad))
par( omi=c(0,0,0,0), mai=c(0.5,0.5,0.1,0.1) )
len = 0.13
mired = seq( 0, 600, by=10 ) ; temp  = 1.e6/mired
uv = planckLocus( temp, param='native' )
plot( c(0.16,0.32), c(0.23,0.36), type='n', xlab='', ylab='', las=1, tcl=0,
        lab=c(10,8,7), mgp=c(3,0.25,0), asp=1 )
title( xlab='u', line=1.5 ) ;  title( ylab='v', line=1.5 ) ; grid( lty=1, lwd=0.75 )
symbols( rep(uv0[1],2), rep(uv0[2],2), circles=c(rad,0.001), fg='red', inch=FALSE, add=TRUE )
points( uv1[1], uv1[2], pch=20, col='red' )
lines( uv[ ,1], uv[ ,2], lwd=0.7 )  ;  points( uv[1,1], uv[1,2], pch=20, cex=0.5 )
text( uv[1,1], uv[1,2], expression(infinity), adj=c(0.6,1.6), cex=1.5 )
temp = c( seq(2000,10000,by=1000), 20000, 30000, Inf )
top = planckLocus( temp, param=param[3], delta=0 )
bot = planckLocus( temp, param=param[3], delta=-len )
segments( bot[ ,1], bot[,2], top[,1], top[,2], col=color[3], lwd=1 )
text( top[ ,1], top[,2], sprintf("%gK",temp), adj=c(1,-0.2), cex=0.5 )
legend( 'topleft', param[3], lwd=3, bty='n', col=color[3], inset=0.05 )
```
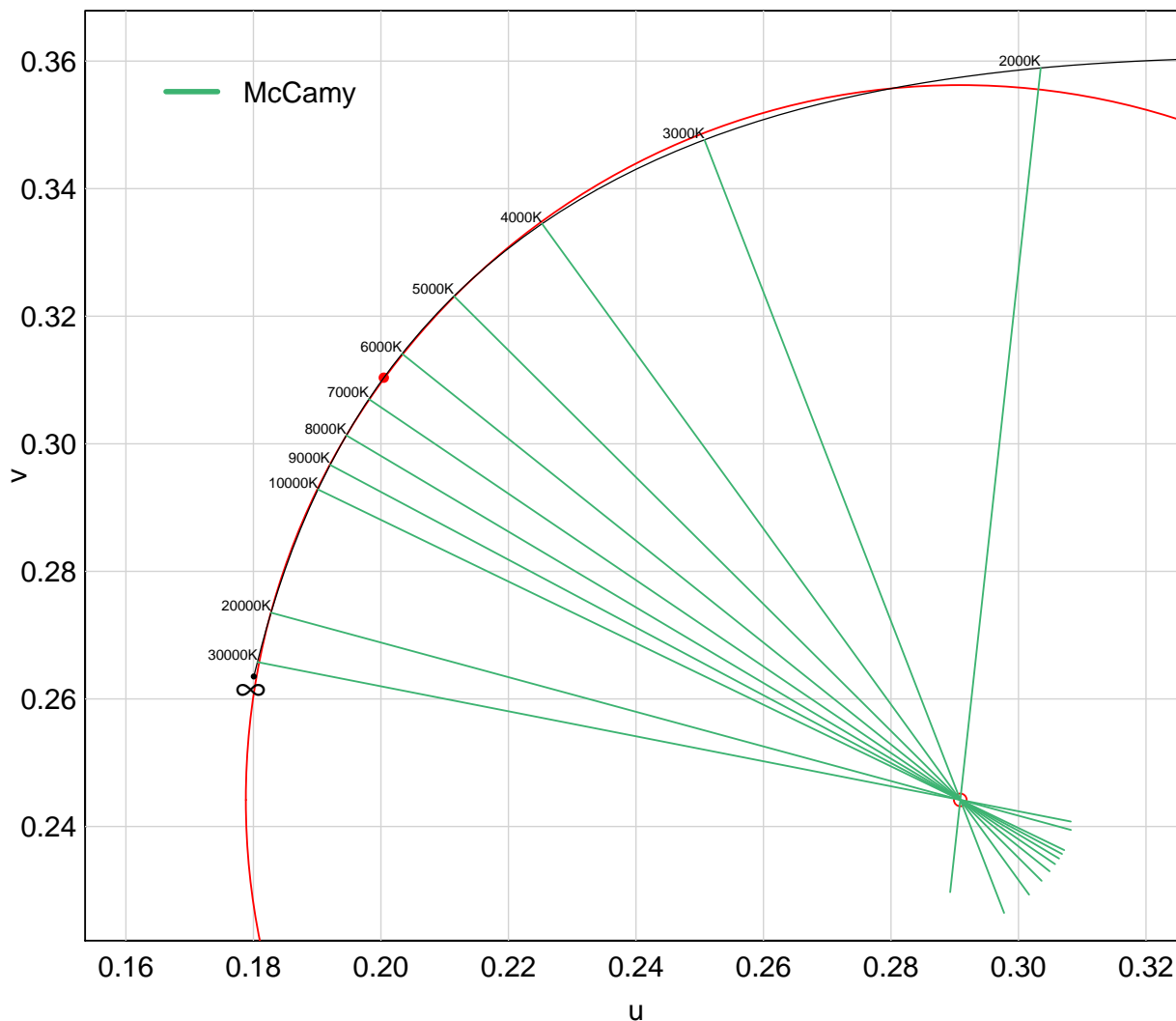
Figure 3: McCamy isotherms below the Planckian locus, with length 0.13.
They all intersect at the same point, which is the center of the red circle.

It is apparent that the curvature of the locus near $T = 6500K$ is fairly constant, and that is what makes the good approximation possible in that temperature range. Another way to say it is that the *osculating circle* to the locus at each T in this range is nearly constant, and the red circle is a good approximation to all of them.

3   The Robertson Isotherms

Each pair of adjacent Robertson isotherms intersect at a point below the locus. The interpolation at intermediate temperatures is defined so that all the intermediate isotherms intersect at that same "pivot point", see Figure 2(3.11) in [WS82]. Let's calculate and plot these 30 points.

```
mired = seq( 0, 600, by=10 )
temp  = 1.e6/mired
uv = planckLocus( temp, param='native' )    # a spline curve
par( omi=c(0,0,0,0), mai=c(0.5,0.5,0.1,0.1) )
plot( c(0.18,0.40), c(0.18,0.36), type='n', xlab='', ylab='', las=1, tcl=0,
      lab=c(10,8,7), mgp=c(3,0.25,0), asp=1 )
```

```
title( xlab='u', line=1.5 ) ;  title( ylab='v', line=1.5 ) ; grid( lty=1, lwd=0.5 )
lines( uv[ ,1], uv[ ,2], lwd=0.7 )
text( uv[1,1], uv[1,2], expression(infinity), adj=c(0.6,1.5), cex=1.5 )
mired = c( seq(0,90,by=10), seq(100,600,by=25) )   # these are from Robertson's table
temp = 1.e6 / mired
top = planckLocus( temp, param='robertson', delta=0.1 )
bot = planckLocus( temp, param='robertson', delta=-0.05 ) ;  off = bot - top
lines( bot[ ,1], bot[ ,2], lty=2 ) ; center = matrix( 0, length(temp)-1, 2 )
for( i in 1:(length(temp)-1) )  {
    A  = cbind( off[i,], off[i+1,] ) ;    s = solve( A, top[i+1,] - top[i,] )
    center[i,] = top[i,] + s[1]*off[i, ]
    lines( c(top[i,1],center[i,1]), c(top[i,2],center[i,2]), col='red', lwd=0.5 )
    lines( c(top[i+1,1],center[i,1]), c(top[i+1,2],center[i,2]), col='red', lwd=0.5 )
}
points( center[ ,1], center[ ,2], pch=21, cex=0.5, bg='white' )
```
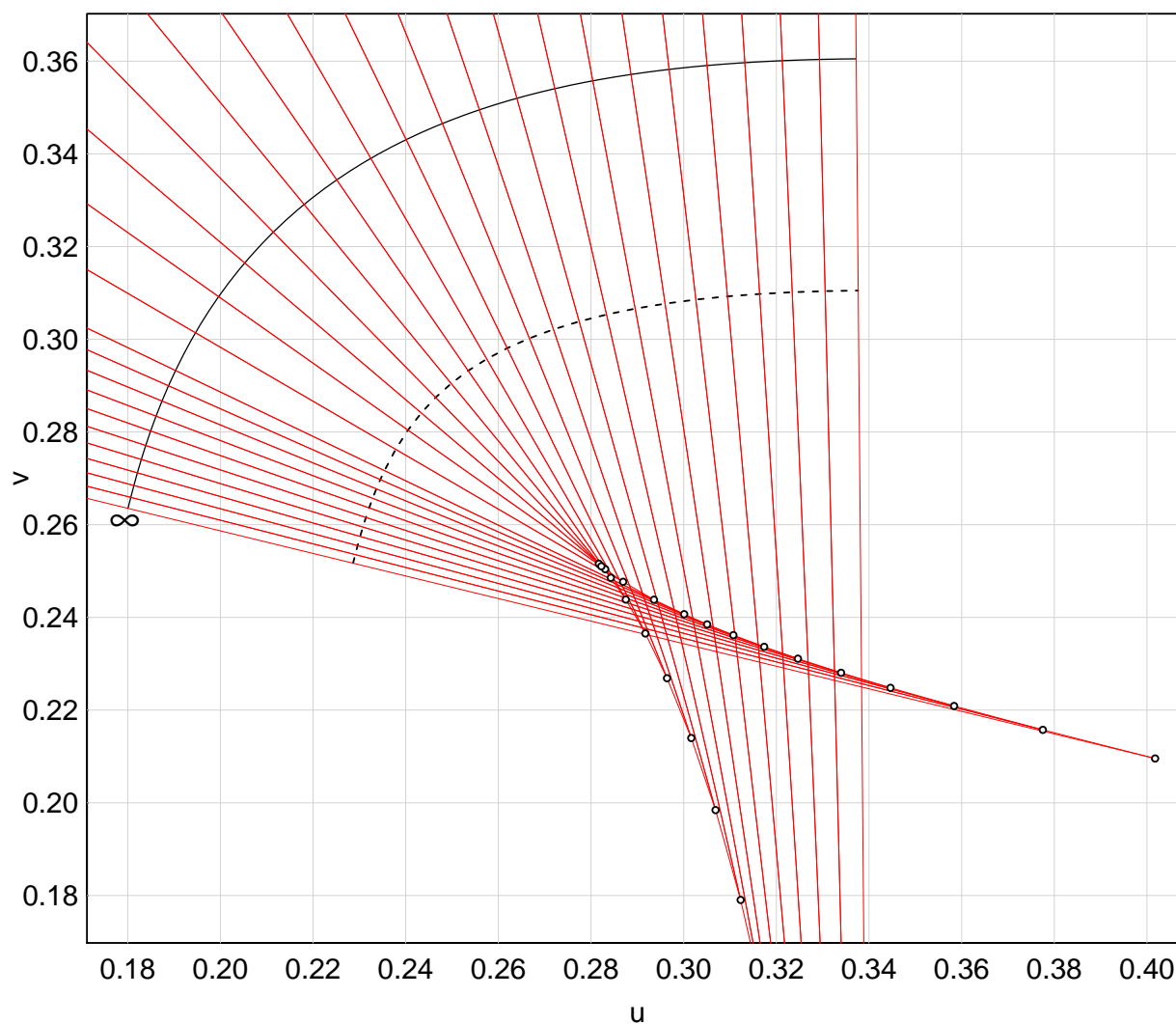

Figure 4: Adjacent pairs of Robertson isotherms intersect below the locus.
The dashed curve is the set of points at distance 0.05 from the locus.

The points are drawn as small white disks. As one moves along the locus, the isotherm "pivot point" jumps from one of these points to the next. Compare with Figure 3. These 30 points are approximately on the *evolute* [Wik18] of the locus. The *evolute* is the path of the center of the osculating circle (also called the *center of curvature*). Since the center of curvature depends only on the locus and its first and second derivatives, and since the locus is C^2, the evolute of the spline locus is a continuous curve.

References

[McC92] C. S. McCamy. Correlated color temperature as an explicit function of chromaticity coordinates. *Color Research & Application*, 17(2):142–144, 1992.

[McC93] C. S. McCamy. Correlated color temperature as an explicit function of chromaticity coordinates. erratum. *Color Research & Application*, 18(2):150, 1993.

[Rob68] A. R. Robertson. Computation of correlated color temperature and distribution temperature. *J. Opt. Soc. Am.*, 58(11):1528–1535, Nov 1968.

[Wik18] Wikipedia contributors. Evolute — Wikipedia, the free encyclopedia. `https://en.wikipedia.org/w/index.php?title=Evolute`, 2018. [Online; accessed 26-November-2018].

[WS82] G. Wyszecki and W.S. Stiles. *Color Science: Concepts and Methods, Quantitative Data and Formulae, Second Edition*. Wiley Classics Library. John Wiley, 1982.

Session Information

This document was prepared March 2, 2020   with the following configuration:

- R version 3.6.3 (2020-02-29), `i386-w64-mingw32`

- Running under: `Windows 7 (build 7601) Service Pack 1`

- Matrix products: default

- Base packages: base, datasets, grDevices, graphics, methods, stats, utils

- Other packages: knitr 1.28, spacesXYZ 1.1-1

- Loaded via a namespace (and not attached): Rcpp 1.0.3, compiler 3.6.3, digest 0.6.25, evaluate 0.14, highr 0.8, htmltools 0.4.0, magrittr 1.5, microbenchmark 1.4-7, rlang 0.4.4, rmarkdown 2.1, stringi 1.4.6, stringr 1.4.0, tools 3.6.3, xfun 0.12, yaml 2.2.1