This Mathematica Notebook contains computations and derivations used to create plots in Figs. 8, 9 and 10 of our paper Ohzawa, DeAngelis, Freeman, Encoding of Binocular Disparity by Complex Cells in the Cat’s Visual Cortex, J. Neurophysiol. 77: 2879-2909, (1997).

Copyright 1996,1997, All rights reserved, Izumi Ohzawa, izumi@pinoko.berkeley.edu

Monocular response (Left eye) of Cx cell based on the energy model is

\[
L_{\text{resp}}(x_L) = (\exp[-k x_L^2] \cos[2 \pi x_L])^2 + (\exp[-k x_L^2] \sin[2 \pi x_L])^2
\]

\[
= (\exp[-k x_L^2])^2
\]

\[
= \exp[-2 k x_L^2]
\]

k is proportional to the inverse of subunit RF width, and pD is the phase difference of L and R subunit RF fields (same for all 4 subunits).

Monocular response for the Left eye is given by,
Sum of Left and Right monocular responses (which may be a good model for a non-phase specific cells) is given by:
This is the derivation of Equation 7 from Equation 5 in our paper Ohzawa, DeAngelis, Freeman, Encoding of Binocular Disparity by Complex Cells in the Cat’s Visual Cortex, J. Neurophysiol. 77: 2879-2909, (1997).

Binocular response for matched contrast conditions (BB and DD) may be simplified as follows:
Simplify[
(Exp[-k xL^2] Cos[2 Pi F xL] + Exp[-k xR^2] Cos[2 Pi F xR + pD])^2
+ (Exp[-k xL^2] Sin[2 Pi F xL] + Exp[-k xR^2] Sin[2 Pi F xR + pD])^2
]

\[\begin{align*}
E^{-2 k xL^2} + E^{-2 k xR^2} + \\
\frac{2 \cos[pD - 2 F Pi xL + 2 F Pi xR]}{E^k (xL^2 + xR^2)}
\end{align*}\]

Just to be general, try the case where spatial freq are different between L and R eyes (use FL and FR for each eye).

Simplify[
(Exp[-k xL^2] Cos[2 Pi FL xL]
+ Exp[-k xR^2] Cos[2 Pi FR xR + pD])^2
+ (Exp[-k xL^2] Sin[2 Pi FL xL]
+ Exp[-k xR^2] Sin[2 Pi FR xR + pD])^2
]

\[\begin{align*}
E^{-2 k xL^2} + E^{-2 k xR^2} + \\
\frac{2 \cos[pD - 2 FL Pi xL + 2 FR Pi xR]}{E^k (xL^2 + xR^2)}
\end{align*}\]

As shown above, the first two terms are monocular responses. The last term is the binocular interaction component, which is a Gabor function (Is this really a Gabor?) that is oriented at 45 degs in (xL, xR) domain (see below).

The binocular component alone may be obtained without separate monocular measurements by computing BB+DD-BD-DB. Since BB == DD and BD == DB, because of squaring which makes the function independent of the inversion of sign,

\[BB+DD-BD-DB = 2 (BB-BD)\]

BB-BD is given by below which is reduced the same expression as above.
\[ k = .; \]
\[ pD = .; \]
\[
\text{Simplify[}
\begin{align*}
&\left(\exp[-k \, xL^2] \, \cos[2 \pi \, xL] + \exp[-k \, (xR)^2] \, \cos[2 \pi \, (xR) + pD]\right)^2 \\
+&\left(\exp[-k \, xL^2] \, \sin[2 \pi \, xL] + \exp[-k \, (xR)^2] \, \sin[2 \pi \, (xR) + pD]\right)^2 \\
-&(\exp[-k \, xL^2] \, \cos[2 \pi \, xL] - \exp[-k \, (xR)^2] \, \cos[2 \pi \, (xR) + pD])^2 \\
+&(\exp[-k \, xL^2] \, \sin[2 \pi \, xL] - \exp[-k \, (xR)^2] \, \sin[2 \pi \, (xR) + pD])^2
\end{align*}
\] \\
\[
\frac{4 \, \cos[pD - 2 \pi \, xL + 2 \pi \, xR]}{e^{k \, (xL^2 + xR^2)}}
\]

This is a Gabor function oriented at 45 degs in the (xL, xR) domain as shown below: (ModelDecomp/GaborLR-0.eps)
z[x_] := GrayLevel[1 - 2*Abs[x - 0.5]];
pD := 0;
k := 5.5;

ContourPlot[
  Exp[-k (xL^2 + xR^2)] Cos[2 Pi (xL - xR) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {-1.1, 1.1},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False]
\textbf{k:=5.5;}
Plot[
    \text{Exp}[-k \times (xL^2)] \times \text{Cos}[2 \ \text{Pi} \ \times xL],
    \{xL, -1, 1\},
    \text{PlotPoints} \rightarrow 40]

\textit{-Graphics-}

Phase Difference = 45 degs (ModelDecomp/GaborLR-45.eps)
z[x_] := GrayLevel[1 - 2*Abs[x - 0.5]];
pD := Pi/4;
k := 5.5;
ContourPlot[
  Exp[-k (xL^2 + xR^2)] Cos[2 Pi (xL - xR) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {-1.1, 1.1},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False]

Phase Difference = 90 degs (ModelDecomp/GaborLR-90.eps)
z[x_] := GrayLevel[1 - 2*Abs[x - 0.5]];
pD := Pi/2;
k := 5.5;

ContourPlot[
  Exp[-k (xL^2 + xR^2)] Cos[2 Pi (xL - xR) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {-1.1, 1.1},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False]

Phase Difference = 180 degs (ModelDecomp/GaborLR-180.eps)
Zero disparity detector for both phase and position models is the same:
Non-Zero disparity detector for the phase model
z[x_] := GrayLevel[1-x];
pD := Pi/2;
k := 5.5;

ContourPlot[
  Exp[-2 k xL^2] + Exp[-2 k xR^2] + 2 Exp[-k (xL^2 + xR^2)] Cos[2 Pi (xL - xR) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {0, 4.4},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False]

Non-Zero disparity dector for the Position model
z[x_] := GrayLevel[1 - x];

oR := 0.25;
pD := 0;
k := 5.5;

ContourPlot[
  Exp[-2 k xL^2] + Exp[-2 k (xR + oR)^2]
  + 2 Exp[-k (xL^2 + (xR + oR)^2)] Cos[2 Pi (xL - (xR + oR)) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {0, 4.4},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False]
z[x_]:=GrayLevel[1-x];
pD:=Pi/2;
k:=5.5;

ContourPlot[
  Exp[-k (xL^2 + xR^2)] Cos[2 Pi (xL - xR) - pD],
  {xL, -1, 1}, {xR, -1, 1},
  Contours -> 12, PlotRange -> {-1.1, 1.1},
  ColorFunction -> z,
  PlotPoints -> 40, Axes -> None, Frame -> False ]