Effect of different parameters on model performance.

The following figure reproduces the last panel of Fig 4, viewing angle = 20°, for different choices of the “sensitivity” parameters r and s, and the effective “noise” parameter sigma. In these plots, r and s are always equal. For the blue curve in each panel, A=SNR (signal to noise – as noted, we set the noise level to 1 without loss of generality) and B=0, i.e. “perfect compensation”. For the red curve, A=0 and B=SNR, i.e. “no compensation”.

![Graph showing the effect of different parameters on model performance.](image-url)

The curves get wider as the effective noise level increases (reading down each column from top to bottom). This is basically because with more noise, you need more of a difference in signal to get a result.

The noise level also affects the peak performance achieved by each model. For the “perfect compensation” model (blue curves), at large rendering angles, the normal-rendered cube should be preferred, but noise can obscure this. Mathematically, for the perfect-compensation model with large render angles (θrend>>r), ΔV=A and so the probability of picking the normal-rendered cube is
P=0.5+0.5 erf(A/2). For the “no compensation” model (red curves), the obliquely-rendered cube should be preferred when $\theta_{\text{rend}}=\theta_{\text{view}}$ but again noise pushes performance closer to chance.

Unsurprisingly, the curves get wider as the sensitivity parameter $r$ increases (reading across each row from left to right). This is because with larger $r$, the “perceived veridicality” declines less steeply with mismatches in render angle. The sensitivity parameter also has an interesting effect on the asymptotic performance of the “no compensation” model (red curves). For a positive viewing angle, as chosen here ($\theta_{\text{view}}=+20^\circ$), and in the absence of compensation, the normal-rendered cube should be chosen whenever $\theta_{\text{rend}}$ is negative (or $>40^\circ$). However, as $r$ is decreased, the model gets so sensitive to angles in render angle that both the normal and obliquely-rendered cubes appear equally bad and performance declines to chance. This effect does not happen in the “perfect compensation” model because the normal-rendered cube is always ideal for this model.

This graph can be reproduced by running EffectOfParameters.m.