Hookah Color Variation: no

j.p.lewis

The sand vortex thing at the beginning *Mission to Mars* was done using a program called Hookah, written by Jim Callahan with additions and maintenance by Lewis Siegel and myself. During the production of Mission to Mars, it was proposed that color variation be obtained by a scheme that was equivalent to adding *independent* random numbers to each vortex particle, rather than the scheme that produced the beautiful color variation (from the same software) in *Armageddon*. This suggestion seemed a bit silly to anyone who understands probability, since it guarantees that there will not be any color variation. Nevertheless (and despite the argument below), this was the method that was used, and the vortex in the finished *Mission to Mars* was pure monochrome brown.

Suppose the randomly sampled color ramp ranges over 200 pixel values on an 8bit scale, e.g. from 25 to 224 – excluding near-black and near white colors (why: it's plausible, but this choice also makes the variance come out to nice easy numbers below.)

Variance of a uniform RV:

$$\text{Var} = E[(x-\bar{x})^2] = \int (x-\bar{x})^2 p(x) dx$$

If uniform in $-c \ldots c$ then

$$= \frac{1}{2c}\int_{-c}^{c} x^2 dx$$

$$= \frac{1}{2c}\frac{c^3}{3}|_{-c\ldots c} = \frac{c^2}{3}$$

So our variance is that of a uniform RV centered at 125; $\frac{c^2}{3} = (100)^2/3 = 3333.3 \approx 3000$.

Thousands of particles are blended together in a pixel using density values such that the result is still in visible pixel range – the result is an average (in a general sense) of the pixel colors. Suppose that the particles are in a thin layer (true to Mission) so that we can ignore exponential falloff effects (all particles end up in one or few camera voxels) and suppose that the particles are the same size (we can come back with different sizes later). Then the pixel color will be modeled by an average (strict sense) of the particle colors.

The variance of an average is the variance of the individual elements divided by the number of particles:

$$E[(\frac{\sum x_k}{n} - Ex)^2] = E\frac{1}{n^2}((x_1 - Ex) + (x_2 - Ex) + \ldots)^2$$

Since the random numbers are independent, variance of sum is sum of variances:

$$E[a - \bar{a} + b - \bar{b}]^2 = E[(a-\bar{a})^2] + E[(b - \bar{b}^2] + 2E[(a-\bar{a})(b-\bar{b})]$$

but the cross term is zero since if independent, expectations interchange with multiply:

$$E[(a-\bar{a})(b-\bar{b})] = E[a-\bar{a}]E[b-\bar{b}] = 0 \cdot 0$$

So all the cross terms cancel, and

$$E[(\frac{\sum x_k}{n} - Ex)^2] = \frac{n\sigma^2}{n^2} = \frac{\sigma^2}{n}$$

The Tchebychev inequality gives the probability that a random variable with particular variance will be a specified amount different than the mean:

$$P(|x - \bar{x}| > \epsilon) \leq \frac{\sigma^2}{\epsilon^2}$$

Now plug in some more numbers. Lets say there are about 3000 particles falling in one pixel (realistic number according to Jim) - this particular choice makes our math nice: the variance (of the average) is then 1. The JND for images is on the order of $1/2^6$ or larger (6 bits are enough to make a continuous ramp in a carefully chosen color space, fewer than six bits can quantize a discontinuous monochrome image with no perceived alteration). So on an 8bit scale we need at least 4 code values to make any perceived difference. The inequality says that the probability of a pixel being even on the order of 1 JND different than the mean (i.e., just *barely* noticeable) is $\frac{1}{\epsilon^2} = 1/16$. It's worse though - people consider that the resolution of 1k or 1280 digital projectors (the ones that were used to show Star Wars) is not too different from that of film, in part because of film transport jitter. Also consider that a barely noticeable difference in one pixel will not be visible at all if it happens only on one frame and disappears the next.

To get a noticeable color shift in some small area, it needs to be more than one pixel in area or last for more than one frame. These two factors presumably trade off (tens of pixels being different in one frame might be noticeable; likewise a single pixel being different from the mean but across tens of frames is probably noticeable).

The joint probability of independent events is the probability of the individual events multiplied. What is the probability that 4 pixels will be *just-barely* darker than the average on a single frame? This is $\frac{1}{16}^4$, one chance in 65,000! What is the probability that this teeny weeny color variation will last for THREE WHOLE FRAMES? One chance in 2^{48}, i.e. something like 1/281,474,976,711,000.

The Tchebychev bound is not at all tight: the central limit theorem says that the distribution of the sum of many identically distributed variables converges rapidly to a Gaussian. The probabilities are above are much smaller than described above because of this. For example, let's try a different calculation: say that there are only 100 particles in a pixel, but model the distribution as Gaussian rather than the loose Tchebychev bound. Let's drop the multiple-pixel / multiple frame argument above and just ask that a single pixel be noticeably different than its surround. What is the probability that a pixel will be more than 20 code values different from the mean? This is the integral of the tails of a Gaussian RV with variance 33.33 (= 3333 / n=100 particles). Make the substitution $y = x/\sqrt{33.33}$ to convert to a unit-variance RV as found in a statistical table. This divides the desired deviation (20 pixels) by $\sqrt{33.33} = 5.773$. Look up $20/5.773 = 3.488$ in a Gaussian table gives the probability that the deviation will be less than this (but greater than zero); one minus twice this gives the probability than the pixel will be more than 20 values different from the mean in either direction. Unfortunately 3.488 deviations do not appear in any statistical table – the probability of this happening is vanishingly small. Oh well.

In other words, the proposed scheme will achieve nothing more than a fine scale dithering that will only be barely visible in still frames – the color of the moving vortex will simply be the mean of the color ramp as sampled by the random generator, as expected. Why don't they just random dither noise in Shake? They would at least have the control then to dial up the dithering so that it is visible!