ARTICLES

The noisy-bit method for digital displays: Converting a 256 luminance resolution into a continuous resolution

RÉMY ALLARD AND JOCELYN FAUBERT
University of Montreal, Montreal, Quebec, Canada

Visual psychophysics often manipulates the contrast of the image on a digital display screen. Therefore, the limitation of the number of different luminance intensities displayable for most computers (typically, 256) is frequently an issue. To avoid this problem, experimenters generally need to purchase special hardware (graphic cards) and/or develop specific computer programs. Here, we describe an easy-to-implement method, consisting of adding noise to the displayed stimulus, that we call the *noisy-bit* method. This random dithering method, generalized to 256 luminance intensities, is equivalent to displaying continuous luminance intensities plus a certain amount of noise. Psychophysical testing using a standard spatiotemporal resolution (60 Hz and 1,024 × 768 pixels) demonstrated that the noise introduced by the noisy-bit method has no significant impact on contrast threshold and is not visible. We conclude that the noisy-bit method, combined with the standard 256 luminance levels, is perceptually equivalent to an analog display with a continuous luminance intensity resolution when the spatiotemporal resolution is high enough that the noise becomes negligible (which is easily attainable with the typical spatiotemporal resolutions of present-day computers).

Visual psychophysics often manipulates the contrast of an image on a digital display screen. A computer screen can display digital images with a relatively high spatiotemporal resolution. Indeed, most computer displays can produce relatively well defined images (typically, at least 1,024 × 768 pixels) at a relatively high temporal resolution (60–200 images/sec). This high spatiotemporal resolution enables computers to display digital images resembling analog images. Indeed, if the spatial and/or temporal resolution is great enough, there will be no significant differences between a digital and an analog image. For instance, a luminance grating, which ideally would vary continuously over space and/or time, varies in a discrete manner when presented on a digital display (see Figure 1) but appears to vary continuously if the spatial and/or temporal resolution is great enough. Hence, high-frequency luminance variations are spatiotemporally summed by the visual system and, therefore, undetected (Watson, Ahumada, & Farrell, 1983).
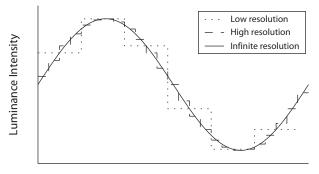
The Grayscale Resolution Problem

Analogously to the spatiotemporal resolution, the luminance intensity of each pixel is also discrete. The luminance of each pixel of a digital image sent to a display is defined by a digital value typically ranging between 0 and 255; these values are called digital-to-analog converter (DAC) values.

For sake of simplicity, we will omit that there are three different color guns and will define each pixel color only by its luminance intensity. In other words, for any given pixel, we will assume that all three guns are set to the same DAC value. The DAC translates each value into a voltage, resulting in a given luminance intensity. Before psychophysical testing, the relation between the DAC value sent to the display (*i* for an integer) and the luminance intensity produced (*d* for a discrete value) is typically made linear (i.e., gamma corrected to be proportional to the DAC value):

$$d = \frac{L_{255}}{255} i, \qquad (1)$$

where L_{255} represents the luminance intensity emitted when the DAC value is 255. The DAC values are integers ranging between 0 and 255, which limits the number of different displayable luminance intensities. However, the mathematical function defining the luminance intensity of each pixel of the stimulus [$L(x, y, t)$ for the luminance intensity of the pixel spatially positioned at (x, y) at time t] is generally continuous. Knowing the relation between the DAC value and the displayed luminance intensity (Equation 1) enables the unit conversion of the luminance intensity of a given pixel [l; for simplicity, we will refer to a given pixel, which enables us to drop the spatiotemporal

R. Allard, remy.allard@umontreal.ca

Figure 1. Luminance variation of two discrete resolutions, as compared with a continuous resolution, as a function of either space or time. If the resolution is high enough, the differences between a discrete and a continuous resolution are not detectable.

position of the pixel (x, y, t), so that $l = L(x, y, t)$] to a continuous DAC value (r for a real value), which is simply the inverse of Equation 1:

$$r = \frac{255}{L_{255}} l. \qquad (2)$$

However, DAC values are not continuous and must be integers. As a result, the DAC values must be rounded to the nearest integer before being sent to the display:

$$i = \lfloor r + 0.5 \rfloor, \qquad (3)$$

where $\lfloor x \rfloor$ represents the floor function (i.e., rounding x to a lower integer) and $\lfloor x + 0.5 \rfloor$ thereby represents rounding x to the nearest integer. By combining these three equations, the relation between the luminance function defining each pixel (l) and the discrete displayed luminance (d) of this pixel is

$$d = \frac{L_{255}}{255} \left\lfloor \frac{255}{L_{255}} l + 0.5 \right\rfloor. \qquad (4)$$

In other words, there is a limited quantity of displayable luminance intensities (for most computers, 256), and the luminance value of each pixel (l) is typically rounded to the nearest DAC value or, which is equivalent, to 1/255 the maximal luminance intensity (L_{255}). This grayscale resolution can often be too low to measure contrast thresholds (Pelli & Zhang, 1991). Indeed, for many spatiotemporal frequencies, the smallest perceptible contrast is less than the smallest luminance intensity difference displayable ($L_{255}/255$). Since contrast thresholds are often measured in psychophysics, the grayscale resolution of most computer displays is frequently an issue.

Current Solutions

Hardware. Various methods are used to solve the grayscale resolution problem. The first obvious solution is to purchase a graphic card able to display more than 256 different luminance intensities. This solution not only requires buying special graphic cards, but also necessitates particular programs (generally homemade) interfacing directly with the graphic card. Indeed, under Windows, very few graphic cards are able to display 1,024 grayscale intensities (i.e., 10 bits), since the majority are limited to 8 bits. To use 10-bit graphic cards, experimenters typically need to develop their own software directly interfacing with the graphic card. On Macintosh computers, today's graphic cards now generally display grayscales with a 10-bit precision, and some special graphic cards can display up to 12 or 14 bits. However, they also require specific programming directly interfacing with the lookup table of the graphic card. Moreover, although they can display more than 256 different luminance intensities, only 256 can be displayed simultaneously. Furthermore, 10-bit resolution may still not be sufficient. As a result, the hardware solution can solve the grayscale resolution problem in some conditions but may not be easily applicable for all experimenters.

Bit-stealing. The bit-stealing method (Faubert, 1991; Tyler, 1997) suggests the use of a chromatic jitter to enhance luminance intensity precision. Instead of having the same DAC value for all three color guns, each gun may have slightly different DAC values. Having different DAC values (e.g., 128, 129, 128) for the three color guns enables the display of luminance intensities with a greater precision than when all three guns have the same DAC values [e.g., (128, 128, 128) or (129, 129, 129)]. This alters the chromaticity of the pixel but also displays luminance intensities with a greater resolution. Since we are less sensitive to chromatic variations than to luminance intensity variation, such chromatic jitter is generally not detectable. The drawback of this method is that it is relatively complex to implement and still limits the number of gray levels that can be displayed.

Dithering. Some printers, faxes, or display devices can produce only two colors (typically, black and white) and are used to display images typically defined by 256 grayscale intensities. Many techniques, called *halftoning* or *dithering* (Ulichney, 1987), have been developed to artificially display grayscale images, using binary output devices. Basically, these techniques consist of using the spatial resolution to give the illusion of presenting grayscale images. For instance, if, within a small spatial region, half of the pixels are black and the others are white, the spatial integration of the visual system will result in a gray percept.

The simpler dithering algorithm is *random dithering* (Ulichney, 1988). This algorithm proposes comparing the luminance intensity of each pixel of the original image with a cutoff criterion that is randomly selected for each pixel from a uniform distribution varying between the two displayable luminance intensities. If the pixel luminance of the original image is greater than the cutoff criterion, the output pixel is white; otherwise, it is black. This method has the advantage of being easy to implement. However, as was stated by Ulichney (1988), "the quality of output of this method does not deserve consideration for practical use" (p. 60). Consequently, this method is generally described only for theoretical purposes.

The difference between the original image and the displayed image corresponds to the noise introduced by the dithering. Many algorithms have been developed to minimize the visual impact of this noise. Ulichney (1988) suggested using high-pass noise (typically referred to as *blue noise*), which is more "pleasant" than the white noise (ran-

dom dithering). The *ordered dithering* algorithm (Bayer, 1973) consists of using a given ordered pattern of the cutoff criteria, instead of randomly selecting each cutoff criterion. The *error diffusion* algorithm (Floyd & Steinberg, 1976) consists of subtracting the noise introduced at each pixel to adjacent pixels. Mulligan and Ahumada (1992) proposed using knowledge about the contrast sensitivity function of the visual system to minimize the noise at the frequencies to which we are more sensitive. All in all, many algorithms have been developed with the goal of proposing a dithering technique that will minimize the visual impact of the noise. All of them have the advantage of giving a better image quality than does random dithering. However, all of them also have the disadvantage of being more complicated to implement and generally requiring more computer resources.

To enhance the luminance resolution of typical displays, Mulligan (1990) proposed a simple way of generalizing ordered dithering used for bilevel displays to higher luminance resolution displays. Basically, the algorithm consists of applying ordered dithering independently to each pixel by selecting between the two nearest luminance intensities displayable, instead of between the only two luminance intensities available. Note that even though Mulligan described this generalization with regard to ordered dithering, it can also be applied to any dithering algorithm.

Daly and Feng (2005) implemented such generalization to ordered dithering in order to enhance apparent luminance intensity resolution, which they named *bit-depth extension*. They developed a sophisticated algorithm with the aim of creating an ordered pattern that would minimize the visibility of the noise. They used the contrast sensitivity function of the visual system to determine at which spatiotemporal frequencies we are less sensitive. They constructed their pattern so that the noise introduced by dithering is concentrated at these frequencies. They found that, in some cases, the noise was invisible, so that the bit-depth of the original image could be reduced without noticeable impact.

As a result, there is no simple solution to the grayscale resolution problem. Most techniques require special hardware and/or relatively complex programming. The purpose of the present article is to propose a technique requiring no special hardware and no complex programming that can display stimuli with an infinite number of gray levels. Combining the high spatiotemporal resolution of computer displays with a simple modification of the stimulus function can solve this problem.

THE NOISY-BIT METHOD

As was mentioned above, the luminance intensity for each pixel is typically defined by a continuous value (l) that generally has to be rounded with a precision of 1/255 the maximal luminance intensity displayable (L_{255})—that is, to the nearest DAC value. This procedure can sometimes create sufficiently high artifacts to impair contrast threshold measurement.

Algorithm

Instead of simply rounding to the nearest DAC value (Equation 3), we propose a different algorithm consisting of randomly choosing between the two nearest DAC values. The probability distribution between the two values can be set so that the expected value is equal to the continuous DAC value defined by the stimulus function (r). That is, the probability of choosing the higher DAC value is equal to the remainder of the continuous DAC value. For example, if the continuous DAC value is 123.25, the probability distribution would be .25 for 124 and .75 for 123, resulting in an expected value of 123.25. Consequently, the noisy-bit method proposes to replace Equation 3 by

$$i = \begin{cases} \lceil r \rceil & \text{if } R(r), \\ \lfloor r \rfloor & \text{otherwise}, \end{cases} \quad (5)$$

where $R(x)$ returns *true* with a probability equal to the remainder of x (i.e., $x - \lfloor x \rfloor$) and *false* otherwise. $\lceil x \rceil$ and $\lfloor x \rfloor$ represent the ceiling and floor functions, respectively (i.e., rounding to the upper and lower integers, respectively). Note that this method is equivalent to combining random dithering (Ulichney, 1988) with the generalized application of dithering for 256, instead of 2, luminance intensities (Mulligan, 1990). Indeed, randomly selecting between the two nearest DAC values ($\lfloor r \rfloor$ and $\lceil r \rceil$) with a probability of choosing the highest DAC value ($\lceil r \rceil$) equal to the remainder of the DAC value ($r - \lfloor r \rfloor$) is mathematically equivalent to rounding to one of the two nearest DAC values with a random cutoff criterion selected from a uniform distribution varying between the two nearest DAC values.

The main drawback of the noisy-bit method is that it increases the error between the desired continuous luminance intensity (l) and the displayed luminance intensity (d) (see Figure 2). Indeed, this method will choose between the two nearest DAC values, so that it will occasionally select a DAC value further than the nearest integer. The error will be equal to $1 - x$ or x, depending on whether the continuous DAC value is rounded to the highest or the lowest integer, respectively, where x represents the remainder of the continuous DAC value ($r - \lfloor r \rfloor$). Assuming that the remainders of the continuous DAC values are uniformly distributed, the root-mean square (RMS) error can be calculated using the following equation:

$$\text{RMS error} = \sqrt{\int_0^1 P(x)(1-x)^2 + [1 - P(x)]x^2 \, dx}, \quad (6)$$

where $P(x)$ corresponds to the probability of selecting the highest integer. When simply rounding to the nearest integer, this probability is equal to 1 if $x > 0.5$ and 0 otherwise, resulting in an RMS error of 0.29 DAC values. Using the noisy-bit method, the probability of rounding to the highest integer is equal to x, resulting in an RMS error of 0.41 DAC values. As a result, the RMS error between the continuous DAC value (r) and the DAC value sent to the display (i) will be $\sqrt{2}$ times greater using the noisy-bit method than simply rounding to the nearest integer. However, this random selection will result in an expected value that will be equal to the desired continuous value [$E(i) = r$]. Conversely, rounding to the nearest integer will cause the expected displayed value to be equal to the nearest integer, which is generally not equal to the desired continuous value [$E(i) = \lfloor r + 0.5 \rfloor \neq r$].
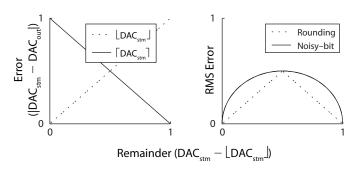
Figure 2. The left graph shows the error (i.e., difference between the continuous digital-to-analog converter [DAC] value [*r*] and the displayed DAC value [*i*]) as a function of the remainder of the continuous DAC value. The two lines show the error for rounding to the lower (dash line) or upper (solid line) DAC values. The right graph shows the root-mean square (RMS) of this error, using two methods of selecting between the lower and the upper DAC value. Rounding to the nearest DAC value always results in the lowest error in the left graph. The noisy-bit method rather proposes to randomly select between the two nearest DAC values with a given probability.

Thus, this method is equivalent to displaying the desired luminance intensity (*l* or *r* in DAC units) plus a certain amount of noise due to the difference between the desired luminance intensity and the luminance displayed (*l* − *d* or *r* − *i*). In other words, the noisy-bit method converts an 8-bit (256) grayscale resolution into a continuous grayscale resolution, with the drawback of adding noise. The noise energy depends directly on the size and duration of presentation of each element (in our case, pixel), and the impact of the noise decreases when the spatiotemporal resolution increases. If the size of the noise elements is small or its duration is brief, the local luminance variation between each pixel will be summed by the visual system, and only the mean value will be perceived. On the other hand, large noise element size and low temporal resolution may result in perceivable noise elements. With a high spatiotemporal resolution, the impact of the noise should be negligible, and the displayed stimulus will be perceptually equivalent to the stimulus defined by continuous luminance values. However, if the spatiotemporal resolution is too poor, the noise introduced can be detected and can affect contrast thresholds.

Note that the error between the desired and the displayed DAC values is not constant as a function of the remainder of the desired DAC value. Consider the two extreme cases in which the remainder of desired DAC value is either 0 or 0.5 (e.g., desired DAC values of 128 or 128.5). In the first case, all the output DAC values will be 128, and no noise will therefore be added to the display. In the second case, each pixel will be randomly selected between 128 or 129, and the displayed luminance will be noisy. Consequently, these two desired DAC values will generate extremely different displays: no noise or maximum noise (i.e., the luminance error for each pixel will be ±0.5 DAC values, resulting in an RMS error of 0.5 DAC values). However, in both cases, the mean luminance will be near (or equal to) their desired DAC value. Consequently, if the spatiotemporal resolution is high enough so that the noise is not visible, both

cases will result in similar percepts: two uniform grays with slightly different intensities.

Implementation

This method of randomly selecting between the two nearest DAC values (Equation 5) is mathematically equivalent to rounding to the nearest DAC value after adding a noise value randomly selected from a uniform distribution varying between −0.5 and 0.5 DAC values (*N*). For instance, if the continuous DAC value (*r*) is 123.25, randomly selecting a value between 122.75 and 123.75 and then rounding to the nearest integer results in a probability of selecting the DAC value 123 equal to .75 and a probability of selecting 124 equal to .25. Consequently, the noisy-bit method can be implemented by replacing Equation 5 with

$$i = \lfloor r + N + 0.5 \rfloor. \tag{7}$$

Matching Equations 1, 2, and 7, we obtain

$$d = \frac{L_{255}}{255} \left\lfloor \frac{255}{L_{255}} l + N + 0.5 \right\rfloor. \tag{8}$$

By defining

$$l' = l + \frac{L_{255}}{255} N, \tag{9}$$

we obtain the same function as Equation 4:

$$d = \frac{L_{255}}{255} \left\lfloor \frac{255}{L_{255}} l' + 0.5 \right\rfloor. \tag{10}$$

Consequently, the noisy-bit method can simply be implemented by adding a small amount of noise to the luminance function (Equation 9), rather than by explicitly implementing the random selection between the two nearest DAC values.

As was mentioned above, for sake of simplicity, we referred to a given pixel that enables us to drop the spatiotemporal position of the pixel (*x*, *y*, *t*). Consequently, Equation 9 can be reformulated more generally as

$$L'(x,y,t) = L(x,y,t) + \frac{L_{255}}{255} N(x,y,t). \qquad (11)$$

In other words, the noisy-bit method can be implemented by adding uncorrelated noise $[N(x, y, t)]$ with a given contrast ($L_{255}/255$; i.e., 1 DAC value) to the stimulus function $[L(x, y, t)]$.
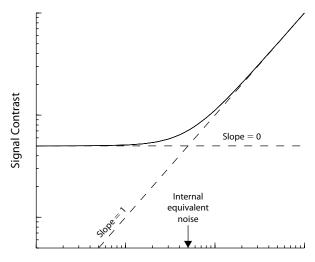
Evaluating the Impact of the Noise

Contrast thresholds in noise have been widely studied. The threshold-versus-contrast (TvC; see Figure 3) function was found to give a reasonably good fit of the contrast threshold as a function of the external noise contrast (n_{ext}; Legge, Kersten, & Burgess, 1987; Pelli, 1981, 1990; Pelli & Farell, 1999):

$$c(n_{int}) = k\sqrt{n_{int}^2 + n_{ext}^2}, \qquad (12)$$

where n_{int} corresponds to the internal equivalent noise—that is, the contrast of the external noise having the same impact as the internal noise—and k is proportional to the smallest signal-to-noise ratio required to detect the signal.

For the purpose of the present study, the important parameter of this function is the internal equivalent noise (n_{int}), which corresponds to the breaking point of the curve. When the external noise is significantly greater than the internal noise ($n_{ext} \gg n_{int}$), the internal noise has no significant impact and the contrast threshold is proportional to the external noise contrast (slope of 1 in log-log units). In other words, for this portion of the curve, if you increase the contrast of the external noise by a given factor, the contrast threshold will increase by the same proportion. However, when the external noise is significantly lower than the internal noise ($n_{ext} \ll n_{int}$), the external noise has no significant impact and the contrast threshold is independent of the external noise contrast (slope of 0).

As was mentioned above, the noisy-bit method is equivalent to presenting a stimulus with a continuous grayscale precision $[L(x, y, t)]$ plus a certain amount of noise. Using the TvC function, it should be possible to determine whether this noise has a significant impact. If the noise introduced by the noisy-bit method is much smaller than the internal noise, it would be insignificant and would have no impact on the contrast threshold. Again, the noisy-bit method is equivalent to presenting the stimulus with the desired continuous value $[L(x, y, t)]$ plus a certain amount of noise. If this noise is not significant, the noisy-bit method is equivalent to presenting a stimulus with continuous gray levels—that is, with an infinite number of luminance intensities.

EXPERIMENT 1
The Impact of the Noise

The objective of the present experiment was to evaluate whether a spatiotemporal resolution of a typical digital display (60 Hz and 1,024 × 768 pixels) is great enough to measure contrast thresholds, using the noisy-bit method. As was mentioned above, the noisy-bit method may be implemented by adding noise to the luminance function defining the stimulus with a uniform distribution ranging between ±0.5 DAC values or, which is equivalent, ranging between $\pm L_{255}/(255 \times 2)$ luminance intensity. We define the noise contrast, which can be represented in luminance intensity or DAC values, as the range covered by the uniform distribution. Note that Equation 2 can be used to pass from luminance intensity units to DAC units. Using the noisy-bit method, the noise contrast added to the stimulus function is 1 DAC value, or $L_{255}/255$ luminance intensity.

To assess whether the noise introduced within the displayed stimulus by the noisy-bit method affects the contrast threshold, the contrast threshold of a given stimulus was evaluated as a function of the noise contrast. If noise is a limiting factor, increasing the noise contrast should affect the contrast threshold by the same proportion (slope of 1 on the TvC function). Alternatively, if an observer's internal noise is greater than the external noise (i.e., the noise introduced by the noisy-bit method), increasing the external noise will not affect contrast threshold (slope of 0 on the TvC function).

Method

Observers. Two observers participated in the experiment. One of them was one of the authors, and the other was naive as to the purpose of the experiment. Both had normal or corrected-to-normal vision.

Apparatus. The stimuli were presented on a 19-in. ViewSonic E90FB .25 CRT monitor powered by a Pentium 4 computer combined with a Matrox Parhelia512 graphic card. All three color guns were constrained to have the same DAC value. As a result, this setup could display 256 different luminance intensities (8-bit luminance depth). The greatest luminance intensity attainable (L_{255}) was 94 cd/m². The display was gamma corrected using a Minolta CS100 photometer interfaced with a homemade program that produced a linear relationship between the DAC value and the luminance intensity. The refresh rate was set to 60 Hz, which is typically the lowest refresh

Figure 3. Contrast threshold as a function of external noise contrast (threshold-versus-contrast function). The internal equivalent noise corresponds to the breaking point of the curve. When the external noise contrast is significantly lower than the internal equivalent noise, it has no significant impact (slope = 0). When the external noise contrast is significantly greater than the internal equivalent noise, the internal noise has no significant impact and the threshold is proportional to the external noise contrast (slope = 1). Note that the two axes are scaled logarithmically.

Figure 4. Sine wave gratings in noise. The contrast of the signal (*c*) is set to 0.1. From left to right, the noise contrast is 1, 10, and 100 DAC values.

rate for most computers. The screen resolution was set to the most standard screen resolution of 1,024 × 768 pixels, covering an area of 32 × 24 cm. At the viewing distance of 114 cm, the width and height of each pixel were 1/64° of visual angle. In other words, the spatial resolution of the displayed stimulus was 64 pixels/deg. The monitor was the only light source in the room.

Stimuli. To measure contrast thresholds, sine wave gratings are the most widely used stimuli:

$$L(x, y, t) = L_{128}[1 + c\sin(fx + p)], \quad (13)$$

where L_{128} corresponds to the mean luminance value (47 cd/m², which was the luminance intensity emitted when the DAC values were set to 128), *c* corresponds to the stimulus Michelson contrast and was the dependent variable, *f* corresponds to the spatial frequency, which was fixed to 4 cpd (approximately the spatial frequency to which we are the most sensitive), and *p* represents the phase, which was randomized at each presentation. Note that the luminance of the grating depended only on the horizontal position (*x*), and not on the vertical position (*y*) or the time (*t*). Consequently, the grating was vertically orientated and static.

To implement the noisy-bit method, noise must be added to the stimulus function:

$$L'(x, y, t) = L(x, y, t) + n_{ext}N(x, y, t), \quad (14)$$

where n_{ext} represents the noise contrast. As was mentioned above, for the noisy-bit method, the contrast of the noise must be fixed to $L_{255}/255$ luminance intensity (Equation 11) or, which is equivalent, 1 DAC value. However, in the present experiment, we varied the noise contrast so that n_{ext} varied between 1 and 230 DAC values, using seven different noise contrasts. Examples of the stimuli are presented in Figure 4.

For static stimuli, adding dynamic noise implies passing from a static presentation (an image) to a dynamic presentation composed of several images. A dynamic presentation consumes more computer resources (memory, processing time, etc.) than does a static presentation, which requires only the rendering of a single image. Consequently, passing from a static to a dynamic presentation may not always be convenient and may, thereby, limit the application of the noisy-bit method. However, the noisy-bit method may also be applied using static noise. That is, the noise template added to the stimulus will not vary over time [$N(x, y)$ instead of $N(x, y, t)$], so that the exact same image will be presented in all frames. For such an application, only the spatial summation will permit the integration of the different pixels. If the spatial resolution is high enough, the noise introduced by the noisy-bit method should not affect contrast thresholds. To evaluate whether only the spatial resolution could permit the application of the noisy-bit method, we applied the method both spatially (static noise) and spatiotemporally (dynamic noise).

To minimize contrast thresholds, a relatively large spatiotemporal window was used. The presentation time of the stimulus was 500 msec, and the spatial window was a disk with a diameter of 2° of visual angle with a soft edge defined by a half cosine of 0.5°.

Procedure. A two-alternative-interval forced choice task was used, which consisted in identifying the interval in which the sine wave was present by pressing one of two keys. Both intervals contained the same noise contrast (n_{ext}) but were generated by two distinct noise samplings. The delay between the two intervals was 500 msec. Between stimuli presentations, the screen remained blank at the mean luminance level (L_{128}), and a fixation point was presented.

The contrast (*c*) of the grating in the interval in which the sine wave was presented was manipulated by a two-down one-up staircase procedure (Levitt, 1971). In the other interval, the contrast (*c*) was set to 0. The staircase was interrupted after 10 inversions, and the threshold was evaluated as the geometric mean of the last 4 inversions. The step size was fixed to 0.05 log units, and the initial contrast (*c*) was always set well above threshold.

Overall, there were 14 different noise conditions: There were seven noise contrasts, and the noise was either static or dynamic. These 14 conditions were evaluated three times, each resulting in 42 staircases performed in a pseudorandom order. For each of these 14 conditions, the resulting threshold was estimated as the geometric mean of the 3 staircases.

Results and Discussion

The internal equivalent noise measured was 22 and 16 DAC values in the static noise condition and 71 and 44 DAC values in the dynamic noise condition for observers I.L. and R.A., respectively (see Figure 5). As a result, the
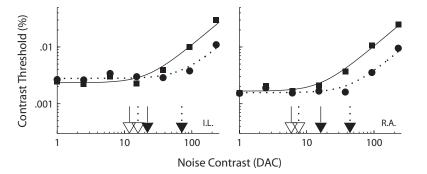

Figure 5. Contrast thresholds as a function of the external noise contrast for the 2 observers. Squares and circles correspond to thresholds when the noise was static and dynamic, respectively. The solid and dashed lines show the best fit of the threshold-versus-contrast (TvC) functions. The filled arrows illustrate the internal equivalent noise corresponding to the breaking point of the TvC functions. Empty arrows illustrate the noise contrast threshold in static (solid) and dynamic (dashed) conditions (results of Experiment 2). DAC, digital-to-analog converter.

detection thresholds using the noisy-bit method (conditions in which the noise contrast was 1 DAC value) were all on the 0-slope portion of the TvC function; that is, the noise introduced by the noisy-bit method was considerably smaller than the observer's internal noise. Hence, it was possible to significantly increase the external noise contrast without affecting contrast threshold. We therefore conclude that the noise introduced by the noisy-bit method (noise contrast of 1 DAC value) did not affect contrast thresholds either in the static or in the dynamic condition.

As was mentioned above, applying the noisy-bit method is equivalent to having a noisy continuous grayscale display. Using this method, the noise corresponds to the luminance variation introduced by randomly selecting between the two nearest DAC values, which correspond to the conditions in which the external noise contrast is 1 DAC value. The present experiment showed that this noise had no significant impact. We therefore conclude that the noisy-bit method enabled a 256 grayscale resolution apparatus to be perceptually equivalent to a continuous (i.e., infinite) grayscale resolution.

EXPERIMENT 2
Noise Detection

The previous experiment showed that the noise introduced by the noisy-bit method did not significantly affect the contrast threshold of a given task. However, this does not imply that the noise was not detectable. A given noise contrast could be perceived without affecting contrast threshold. This would result in a qualitative difference between a continuous grayscale display and a discrete grayscale display combined with the noisy-bit method. The objective of the present experiment was to show that the noise introduced by the noisy-bit method would not be perceived even for relatively low spatiotemporal screen resolutions. If the noise is not perceptible, not only would the noisy-bit method enable contrast threshold measurements equivalent to continuous displays, but also it would be qualitatively (or perceptively) equivalent. Indeed, the difference between a continuous display and 256 grayscale display would not be measurable or perceptible.

Method
The same apparatus as that in the previous experiment was used and the same 2 observers participated. The stimuli were composed of noise:

$$L(x, y, t) = L_{128} + n_{\text{ext}}N(x, y, t). \tag{15}$$

The noise detection task consisted of a two-interval forced choice procedure. One interval was blanked ($n_{\text{ext}} = 0$; i.e., an even gray), and the other contained noise. A two-down one-up staircase procedure, as described in the previous experiment, was used to measure the noise contrast threshold (n_{ext}). Each threshold was evaluated three times in static and dynamic noise conditions, resulting in six staircases.

Results and Discussion
The noise contrast thresholds were 12 and 5.9 DAC values in the static noise condition and 16 and 7.6 DAC values in the dynamic noise condition for observers I.L. and R.A., respectively (see Figure 5). Below these noise con-

trasts, the observers were unable to differentiate between even gray and noise. Consequently, the noise introduced by the noisy-bit method (1 DAC value) was not perceptible. We therefore conclude that there was no qualitative or perceptible difference between a digital 8-bit grayscale display using the noisy-bit method and an analog display able to display an infinite number of grays. Note that this was true even when using a relatively low spatiotemporal resolution (0 Hz [i.e., static] and 64 pixels/deg) for present-day computers.

GENERAL DISCUSSION

The noisy-bit method introduces low-contrast noise to enhance the luminance intensity precision of digital displays. This method is equivalent to displaying gray level with continuous precision and adding noise to the displayed image. The two experiments showed that the low-contrast noise introduced by the noisy-bit method does not affect contrast threshold and is not perceptible. We therefore conclude that when the spatiotemporal resolution is high enough (which is easily attainable with typical computers), a discrete 8-bit display combined with the noisy-bit method is perceptually equivalent to an analog display having a continuous grayscale precision.

Evaluating the Impact of the Noise
Instead of rounding to the nearest DAC value, the noisy-bit method randomly chooses between the two nearest DAC values, so that the expected value is equal to the continuous DAC value. As was shown above, this method can be implemented in two steps. First, add a given amount of noise, and then round to the nearest DAC value. The second step is necessarily already implemented (DAC values must be integers) and is the same as the single step when not applying the noisy-bit method. Often, rounding to the nearest integer is already implicitly implemented by the program sending the image to the graphic card. Consequently, the noisy-bit method can be implemented simply by adding a given amount of noise to the stimulus function (Equation 11).

Both steps actually affect the luminance profile of the displayed stimulus. Indeed, rounding to the nearest integer can also add noise to the displayed stimulus. However, for high noise contrasts, rounding to the nearest integer has no significant impact. Consequently, if adding noise with a contrast significantly greater than 1 DAC value does not significantly affect contrast threshold, the luminance variation introduced by the noisy-bit method (i.e., adding noise with a contrast of 1 DAC value and rounding to the nearest DAC value) certainly has no significant impact. We therefore conclude that simply adding a considerable amount of noise to the luminance function and neglecting the luminance variation introduced by rounding to the nearest integer is an efficient validation to determine whether the noise introduced by the noisy-bit method affects contrast thresholds or not.

Using Less Than 8 Bits
To simulate a display having 7-bit depth, we must add noise to the stimulus function with a contrast of 2 DAC

values and then round to the nearest even integer. More generally, to simulate an N-bit display, we must add noise with a contrast of 2^N DAC values and then round to the nearest integer that is a multiple of 2^{8-N}. We have measured contrast threshold as a function of the number of bits used to display the stimulus (data not shown) and have found that with the current spatiotemporal resolution (64 pixels/deg and 60 Hz), contrast threshold can be measured using only a 5-bit display (noise contrast of 8 DAC values)—that is, using only 32 different luminance intensities. Using a higher spatiotemporal resolution (128 pixels/deg and 120 Hz), we found that contrast threshold can efficiently be measured with a 3-bit display (noise contrast of 32 DAC values)—that is, using only 8 different luminance intensities.

Reducing the Noise of the Noisy-Bit Method

In the present study, we showed that the noisy-bit method can be efficiently implemented using a spatiotemporal resolution that is relatively low for present-day computers (1,024 × 768 pixels at 60 Hz) at a typical viewing distance for psychophysical testing (114 cm). With these parameters, the noise introduced by the noisy-bit method was found to be too low to affect the contrast threshold of a stimulus at which we are highly sensitive (4-cpd sine wave grating with a relatively large spatiotemporal window). Although the noisy-bit method works well using a relatively low spatiotemporal resolution (1,024 × 768 pixels at a viewing distance of 114 cm at 0 Hz), here we describe different ways to reduce the noise introduced by the noisy-bit method. These methods can be used in the eventuality that the noise introduced by the noisy-bit method becomes a limiting factor for a given condition.

The first obvious way to reduce the noise is to enhance the spatiotemporal resolution. This can be achieved by (1) increasing the temporal resolution of the display (many displays can reach 200 Hz), (2) increasing the spatial resolution of the display (many displays can reach 2,048 × 1,536 pixels), and/or (3) increasing the viewing distance.

Although the noisy-bit method was developed for 8-bit displays, it can easily be adapted for displays with more than 8 bits of depth. For instance, the thresholds observed in the current research have been as low as 0.0015 Michelson contrast. A graphic card able to display 1,024 gray levels (10 bits) will not be sufficient to properly evaluate such thresholds. However, the noisy-bit method can be combined to a 10-bit display to enhance luminance precision. Similarly, the bit-stealing method can be combined with the noisy-bit method to enhance luminance intensity precision. The noisy-bit method can randomly choose between two DAC value combinations [e.g., (128, 128, 129) and (128, 129, 128)] so that the expected luminance intensity will be between the luminance intensities produced by these two combinations.

Alternatively, a simple way of reducing the noise introduced by the noisy-bit method is to apply this method independently to each color gun. Combining the luminance noise of the three guns will reduce the luminance noise without requiring any special hardware or sophisticated programming. Hence, each color gun can have its own

noise sampling. When the three color guns are constrained to have the same DAC values, the noise sampling applied to the three guns is perfectly correlated. Without this constraint, the noise sampling is uncorrelated. Consequently, simply applying the noisy-bit method to each gun separately will reduce the luminance noise introduced by the noisy-bit method. Note that since we are less sensitive to chromatic jitter than to luminance noise (especially at high spatiotemporal frequencies), if the luminance noise is not detectable, the chromatic jitter will also not be detectable. Indeed, we found that independently applying the noisy-bit method to the three guns increased the noise contrast threshold (data not shown).

Moreover, instead of simply having an uncorrelated error between the three guns, they can be negatively correlated, as has been suggested by Mulligan (1990), using order dithering. For the noisy-bit method, negatively correlating the error can be implemented by inverting (i.e., subtracting instead of adding) the noise added to the stimulus (Equation 11) of one of the three color guns. In other words, the same noise matrix will be used for two guns, and the inverted matrix will be used for the other gun. Since the green gun generally produces the highest luminance intensity, we suggest inverting the noise matrix of this gun. As a result, the perfectly correlated luminance errors of the red and blue guns will be partially canceled by the luminance error of the green guns.

If the noise introduced by the noisy-bit method affects the contrast threshold, another modification can be applied to limit its impact. The noise can be filtered to keep only the high spatial and/or temporal frequencies. Indeed, for small details or for high-frequency flicker, low-contrast stimuli (in our case, noise) become undetectable and are, therefore, spatially and/or temporally summed by the visual system. Note that to add this modification to the noisy-bit method, the contrast of the noise added to the stimulus function will have to be of at least 1 DAC value once filtered.

CONCLUSION

Although the spatiotemporal resolution of today's computers is relatively high, the luminance intensity resolution is often too low (256 luminance intensities) for many tasks involving contrast manipulation. The noisy-bit method uses the high spatiotemporal resolution of computers to improve the luminance intensity resolution. By randomly selecting between the two nearest DAC values, instead of rounding to the nearest DAC value, the noisy-bit method is a powerful tool with which to bypass the luminance intensity resolution problem. This method can be simply implemented by adding low-contrast noise to the luminance function defining the stimulus. By testing the effect of adding higher contrast noise, one can assert that the noise added to the displayed stimulus has no significant impact on a given task. By evaluating the noise contrast detection threshold, one can also assert that the noise is not visible. As a result, the noisy-bit method successfully makes a typical digital display perceptually equivalent to a continuous luminance intensity resolution system.

REFERENCES

Bayer, B. E. (1973). An optimum method for two-level rendition of continuous-tone pictures. In *Proceedings of the IEEE International Conference on Communications* (Vol. 1, pp. 26.11-26.15). Piscataway, NJ: IEEE Press.

Daly, S., & Feng, X.-F. (2005). Bit-depth extension: Overcoming LCD-driver limitations by using models of the equivalent input noise of the visual system. *Journal of the Society for Information Display*, **13**, 51-66.

Faubert, J. (1991). Effect of target size, temporal frequency and luminance on temporal modulation visual fields. In R. P. Mills & A. Heijl (Eds.), *Perimetry update 1990/1991* (pp. 381-390). Amsterdam: Kugler.

Floyd, R. W., & Steinberg, L. (1976). An adaptive algorithm for spatial greyscale. *Proceedings of the Society for Information Display*, **17**, 75-77.

Legge, G. E., Kersten, D., & Burgess, A. E. (1987). Contrast discrimination in noise. *Journal of the Optical Society of America A*, **4**, 391-404.

Levitt, H. (1971). Transformed up–down methods in psychoacoustics. *Journal of the Acoustical Society of America*, **49**(Suppl. 2), 467-477.

Mulligan, J. B. (1990). Digital halftoning methods for selectively partitioning error into achromatic and chromatic channels. In J. P. Allebach & B. E. Rogowitz (Eds.), *Human vision and electronic imaging: Models, methods, and applications* (Vol. 1249, pp. 261-270). Santa Clara, CA: SPIE.

Mulligan, J. B., & Ahumada, A. J., Jr. (1992). Principled halftoning based on human vision models. In *Human vision, visual processing, and digital display III* (Vol. 1666, pp. 109-121). San Jose, CA: SPIE.

Pelli, D. G. (1981). *The effects of visual noise*. Unpublished doctoral dissertation, Cambridge University.

Pelli, D. G. (1990). The quantum efficiency of vision. In C. Blakemore (Ed.), *Vision: Coding and efficiency* (pp. 3-24). Cambridge: Cambridge University Press.

Pelli, D. G., & Farell, B. (1999). Why use noise? *Journal of the Optical Society of America A*, **16**, 647-653.

Pelli, D. G., & Zhang, L. (1991). Accurate control of contrast on microcomputer displays. *Vision Research*, **31**, 1337-1350.

Tyler, C. W. (1997). Colour bit-stealing to enhance the luminance resolution of digital displays on a single pixel basis. *Spatial Vision*, **10**, 369-377.

Ulichney, R. A. (1987). *Digital halftoning*. Cambridge, MA: MIT Press.

Ulichney, R. A. (1988). Dithering with blue noise. *Proceedings of the IEEE*, **76**, 56-79.

Watson, A. B., Ahumada, A. J., Jr., & Farell, J. E. (1983). *The window of visibility: A psychophysical theory of fidelity in time-sampled visual motion displays* (NASA Tech. Paper 2211). Springfield, VA: National Technical Information Service.