Visual experiments for a subjective-based conditional-replenishment image-sequence coder incorporating texture freeze

Thomas V. Papathomas[1]
David Malah[2]

[1] Laboratory of Vision Research and Department of Biomedical Engineering
Rutgers University, New Brunswick, NJ

[2] Department of Electrical Engineering, Technion-Israel Institute of Technology
Haifa, Israel

ABSTRACT

We report on psychovisual experiments designed to obtain subjective-based thresholds for a novel conditional-replenishment image-sequence coder. This coder attempts to avoid the replenishment of textured blocks for which o subjective change has occurred from the previous to the current frame. Typically, such blocks give rise to a large difference signal with respect to the corresponding block in the previous image, and hence are coded (replenished) in commonly used coders. We designed and conducted extensive visual experiments to study the response of the human visual system to stimuli that are relevant to the coding algorithm.

Three major classes of experiments were conducted with numerous parametric variations for each, in which the observers were asked to discriminate target elements with properties that differed from those of the background: 1) Uniform targets on uniform background of different intensity. 2) Textured targets of varying standard deviation on a uniform background of the same average intensity. 3) Textured targets on a textured background with the same standard deviation, but different average intensity. We report on the results of these experiments and on the improvement in the performance of the coder, as a result of implementing these results in the encoding algorithm.

2. INTRODUCTION

There has been a lot of emphasis in recent years on incorporating relevant characteristics of the human visual system (HVS) into image-coding algorithms because, after all, the end-product of such algorithms is viewed and judged by human observers[1-4]. In particular, such HVS-based algorithms have been developed for image-sequence coders, the input of which can come from a variety of sources[5-11]; see also Yogeshwar [12] for a survey. Applications range from areas in telecommunications (video conferencing, video phone in the ISDN environment) to digital storage and retrieval (in education and training, ntertainment, business and travel guidance).

In parallel with the development activity, there has also been appreciable activity on the standardization front. The CCITT (International Telegraph and Telephone Consultative Committee) standard has been evolving for telecommunications applications, in which the images are typically characterized by limited motion, simple background, stationary video camera, and relatively low temporal and spatial resolutions[6]; data rates are multiples of 64 Kbps and 384 Kbps. At the other extreme, the ISO (International Standards Organization) has been developing standards for more complex image sequences, based on higher rates (typically 1 Mbps), for pictures characterized by fast motion, complex background, moving video camera (zoom and pan), and high resolution[5,8]. Because of the complexity of the sequence in the ISO standard, encoding is usually performed in non-real-time, i.e. slower than video rate. This allows the incorporation of desirable features such as forward and backward normal play, fast reverse, random-image access, etc.

In this paper we present the rationale and results of psychophysical experiments that we designed and conducted in order to obtain relevant HVS characteristics for a subjective-based conditional-replenishment image-sequence coder developed

The work reported here was conducted when T.V. Papathomas was employed at AT&T Bell Laboratories; D. Malah was on sabbatical leave at AT&T Bell Laboratories during this project.

and implemented by Malah[13]. This algorithm is based on the base-line CCITT Reference Model (RM) coder[6]; its present version, however, is an improved modification of the RM coder, which enables it to work well with the stricter requirements of the ISO standard for higher resolution and more complex images. In terms of image complexity, the main advantageous feature of the new algorithm is that it aims to avoid the replenishment (i.e. coding) of textured blocks if no subjective visual change has occurred, in an effort to reduce the transmission bit rate needed for coding image-sequences that contain textured surfaces.

3. OVERVIEW OF THE ENCODING ALGORITHM

3.1 The generic motion-compensated coder

Figure 1 shows a block diagram of a generic motion-compensated conditional-replenishment hybrid inter-frame coder. This figure was adapted from a similar one in Okubo[6] and simplified by not showing the switching between "inter" and "intra" coding. For each incoming block in the current frame, an attempt is made to match it with the corresponding block of the previous reconstructed frame. If the block is detected to have moved between frames, then the displacement components (horizontal, D_x, and vertical, D_y) are estimated to form the displacement vector $D=(D_x, D_y)$, which is transmitted to the decoder as side information. The phrase "motion-compensated", abbreviated *MC*, refers to the block in the previous reconstructed frame that was found to have the best correspondence with the block under consideration in the current frame. Blocks that have such a correspondence are simply "copied", utilizing D; the rest of the blocks are encoded by first transforming each block (using the discrete-cosine transform, DCT, in the special case of Figure 1), and then quantizing the coefficients (by the box Q in Figure 1) and transmitting them as output. The code C, denoting the subclass of each block, is also transmitted as side information. The box marked Q^{-1} performs the inverse function of the quantizer Q and the one marked IDCT performs the inverse discrete-cosine transform. The encoding algorithm that we present below has the same generic structure as the one shown in Figure 1.

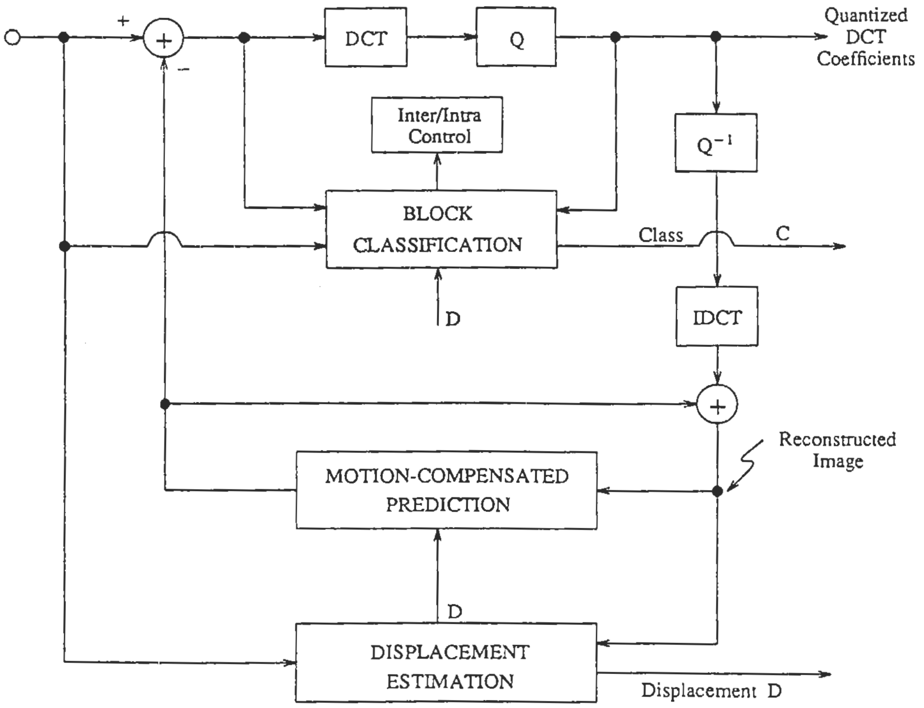

Figure 1: Simplified block diagram for the generic motion-compensated conditional-replenishment hybrid image-sequence coder.

3.2 Some general remarks on the encoding algorithm

The new encoding algorithm is described in detail elsewhere[13]. Our objective in this paper is to present the relevant vision psychophysical experiments that were designed to aid in the block-classification stage; this stage is described in detail in the next subsection. In this section we give a brief outline of some of the features of the image sequence coder.

A partial heuristic search is conducted for locating the "best corresponding" block in the previous frame to the block under consideration in the current frame, i.e. in finding the displacement D, for efficiency purposes. The use of multiple previous reconstructed frames makes it possible to attain effectively sub-pixel resolution in some of the motion-compensated blocks. It also enables one to obtain better estimates of the zoom factor and the global pan shift, which can be incorporated in the scheme for increasing the effective search area in motion compensation estimates.

3.3 The block-classification stage of the coder

As we noted earlier, one of the main advantages of the new image-sequence coder is that it makes an attempt to handle differently textured objects and surfaces. This is achieved in the block-classification stage, the algorithm of which is presented in a high-level (C-like) pseudo-language in Table 1 below, where phrases between the symbols "/*" and "*/" are meant to be comments. The meaning of the symbols B'_{dv}, B'_{db} and B_{db} is explained in Equations (1)-(3) below. The names of the other variables are self-explanatory.

Table 1: The block classification algorithm

```
for each incoming block in the current frame
    {
    if ($B'_{dv}$ < $T_2$)    /* is $B'_{dv}$ less than subjective-based threshold $T_2$?
                        Above test is conducted with motion-compensated (MC) blocks.
                        A block that passes the test is classified as a "non-textured" one;
                        a block that fails the test is classified as a "non-smooth" block. */
            {           /* we are dealing with a "non-textured" block */
            if ( $B'_{db}$ < $T_1$ )   /* is $B'_{db}$ less than subjective-based threshold $T_1$?
                                Above test is conducted with MC blocks. */
                        copy block;      /* a "correspondence" was found; no need to code block */
            else                /* i.e., if $B'_{db}$ is larger than threshold $T_1$ */
                        CodeBlock();     /* no correspondence found; call routine to encode block */
            }
    else                /*i.e., if $B'_{vd}$ is larger than or equal to $T_2$ */
            {                   /* we are dealing here with a "non-smooth" block */
            if ( $B_{db}$ < $T_3$ )   /* is $B_{db}$ less than threshold subjective-based $T_3$?
                                Above test conducted with static (not MC) blocks */
                        {       /* block may be a candidate for "copying" */
                        if (block satisfies additional "texture freeze" tests)
                                copy block;      /* "texture freeze" applies; do not code block */
                        else    /* i.e., block fails any of the additional "texture freeze" tests */
                                CodeBlock();     /* call routine to encode block. */
                        }
            else                /* i.e., if $B_{db}$ is greater than or equal to threshold $T_3$ */
                        CodeBlock();     /* first "texture-freeze" test fails; code block */
            }
    }
```

```
CodeBlock()          /* routine to code blocks that fail either the "correspondence" test
                        or the "texture-freeze" test */
compute PredictionGain;
if( PredictionGain < 1)
    use "intra"-frame block encoding;
else
    use "inter"-frame block encoding;
return;
```

Once the best estimate for the displacement D is found, the first test tries to determine whether the quantity B'_{dv} (defined in (1) below), a measure of the variance of the difference between the current block and the previous MC block, is below a threshold T_2:

$$B'_{dv} = (1/N) \sum_{i,j \in B} \left| (s_n(i,j) - \bar{s}_n) - (r_{n-1}(i+D_x, j+D_y) - \bar{r}'_{n-1}) \right| < T_2. \tag{1}$$

Here $s_n(i,j)$ and $r_{n-1}(i,j)$ denote the value of the intensity at pixel (i,j) of the current frame, n, and the previous reconstructed frame, n-1, respectively. The summation is over all the pixels (i,j) that belong to the block B. Each block contains MxM=N pixels. The average values of s_n and of the (MC reconstructed) r'_{n-1} over the block are denoted by $\bar{s}_n$ and $\bar{r}'_{n-1}$, respectively. In what follows, the single-prime symbol, " ' ", is used to denote that we are dealing with motion compensated blocks and their parameters. If relationship (1) holds, then the MC block-difference is almost uniform, meaning that both corresponding blocks are most likely "non-textured", because the intensity does not vary widely within the difference-block. If the inequality is not satisfied, it means that the MC block difference has a high variance; in turn, this is an indication that at least one of the blocks is a "non-smooth" block, i.e. one whose intensity may vary widely from pixel to pixel. It is obvious that the value of the threshold T_2 plays an important role in the performance of the algorithm and, hence, that a good estimate for T_2 from psychophysical experiments would be most useful for obtaining a better performance.

If inequality (1) is satisfied, we apply a second test to the block B that has just been characterized as "non-textured." We next test whether its (nearly uniform) intensity is very close to that of the MC block r'_{n-1} in the previous reconstructed frame. This is done by comparing B'_{db}, the absolute difference between the mean intensities of the two blocks, to a threshold T_1, the value of which is also estimated from psychovisual experiments:

$$B'_{db} = \left|\bar{s}_n - \bar{r}'_{n-1}\right| = (1/N) \left|\sum_{i,j \in B} (s_n(i,j) - r_{n-1}(i+D_x, j+D_y))\right| < T_1. \tag{2}$$

If the mean intensities of the two blocks are nearly the same, then the inequality will be satisfied, and the block B under consideration in the current frame can just be copied from the corresponding MC block of the previous reconstructed frame, thus saving encoding bits. If the test fails, however, the block must be encoded by the routine CodeBlock, which performs "intra-" or "inter-frame" encoding, depending on the value of the prediction gain.

On the other hand, if a "non-smooth" block is involved (this is the case when inequality (1) is not satisfied), then this algorithm incorporates some additional tests that aim at handling textured background. Rather than proceeding to encode the non-smooth blocks, the algorithm attempts to save encoding bits by investigating whether the block belongs to the textured background, which is stationary in most typical cases. The first test checks whether the mean values of the corresponding blocks are sufficiently close and it is very similar to the test in (2) above. However, to account for stationary background, this "texture freeze" test is performed without motion compensation:

$$B_{db} = \left|\bar{s}_n - \bar{r}_{n-1}\right| = (1/N) \left|\sum_{i,j \in B} (s_n(i,j) - r_{n-1}(i,j))\right| < T_3. \tag{3}$$

The value of T_3 was also estimated with the aid of experiments, as were those of T_1 and T_2. The methods and results of these experiments are presented in the next section.

4. OBTAINING THRESHOLDS FROM HVS CHARACTERISTICS

Three major types of experiments were designed and conducted to determine appropriate values for the threshold parameters T_1, T_2 and T_3, which were defined in section 3 above. Before these experiments are described separately, some general remarks are provided that apply in common to all the experiments.

Almost all of the experiments were performed on two Sony PVM-1271Q color television monitors (30.5-cm (12-inch) diagonal). Some pilot experiments were also conducted on a Sony Trinitron PVM-1910 color monitor (50.8-cm (20-inch) diagonal). Images and image sequences were viewed from a distance of 87.6 cm for the 1271Q model; thus, the picture height subtended a visual angle of 9.53 degrees (a different viewing distance was used for the 1910 model to maintain the same viewing angle of 9.53 degrees). Image size was 720 horizontal by 480 vertical pixels, which translates to 21.9 cm by 14.6 cm on the 1271Q unit. The target blocks were squares, the sides M of which were of three different sizes, depending on the experimental conditions: 8, 16, and 24 pixels.

Images were generated and stored on an Abekas A60 Digital Disk Recorder and were displayed on the monitor at a rate of 30 frames per second (interlaced). The generic form of the stimuli is shown in Figure 2. The observer's task was to report the number of target blocks that were discriminable from the background. They were asked to respond accurately as fast as they could. We decided to ask for a prompt response somewhat arbitrarily, but for a logical reason: if one was to duplicate normal viewing conditions, one must avoid asking observers to scrutinize the image(s), as this is rarely indeed what television viewers do. The discriminability of the targets grows progressively harder from target 1 to target r+G (the role of parameters r and G is explained below). Here is how targets are discriminated from the background in different experiments: In experiments of *type 1* both the background and the targets are of uniform intensity; it is only the difference between these values that makes the blocks visible, assuming this difference is large enough. In experiments of *type 2* the background is uniform and the targets are textured but their average intensity is the same as that of the background; they are discriminated only when their standard deviation is sufficiently large. Finally, in experiments of *type 3*, both the background and the targets are textured and they share the same value of standard deviation; it is only the average intensity of the blocks, provided it is above a certain threshold, that will make them visible against the background.

As seen in Figure 2a, there are r+G target-blocks, where the integer r is a random variable in the range [0,3], typically. These first r targets are easy to detect and we call them "decoys". The decoys were used to discourage the observer from expecting a fixed number of targets; their function was explained to the observers. G is an integer, ranging from 4 to 9, the value of which is fixed for a given type of experiment. The number of visible targets reported by the observer gives us a good

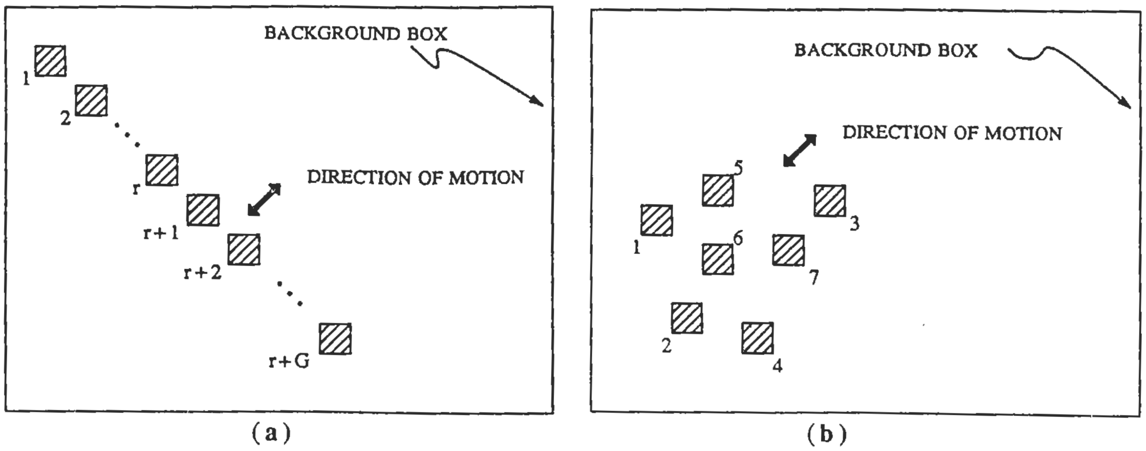

Figure 2: Schematic showing the spatial arrangement of the target blocks in a single frame: (a) Linear arrangement, (b) Random arrangement. The thick black arrows indicate the direction of motion.

estimate for their detectability. Another arrangement, in which the targets were placed at random locations on the screen, as shown in Figure 2b, was also used. As with the linear arrangement, detectability grew progressively harder from target 1 to target r+G; r visible decoys were used with the random arrangement, as well. In addition to static targets, we also created animation sequences, in which the blocks were moving coherently (as a cluster) at pre-determined speeds along the main diagonal, as shown by the arrows of Figure 2.

In a typical experiment, the screen was partitioned into an array of 2 by 2 large rectangular background boxes, each containing 360 horizontal by 240 vertical pixels. This is shown in Figure 3a for the case in which the background is textured.

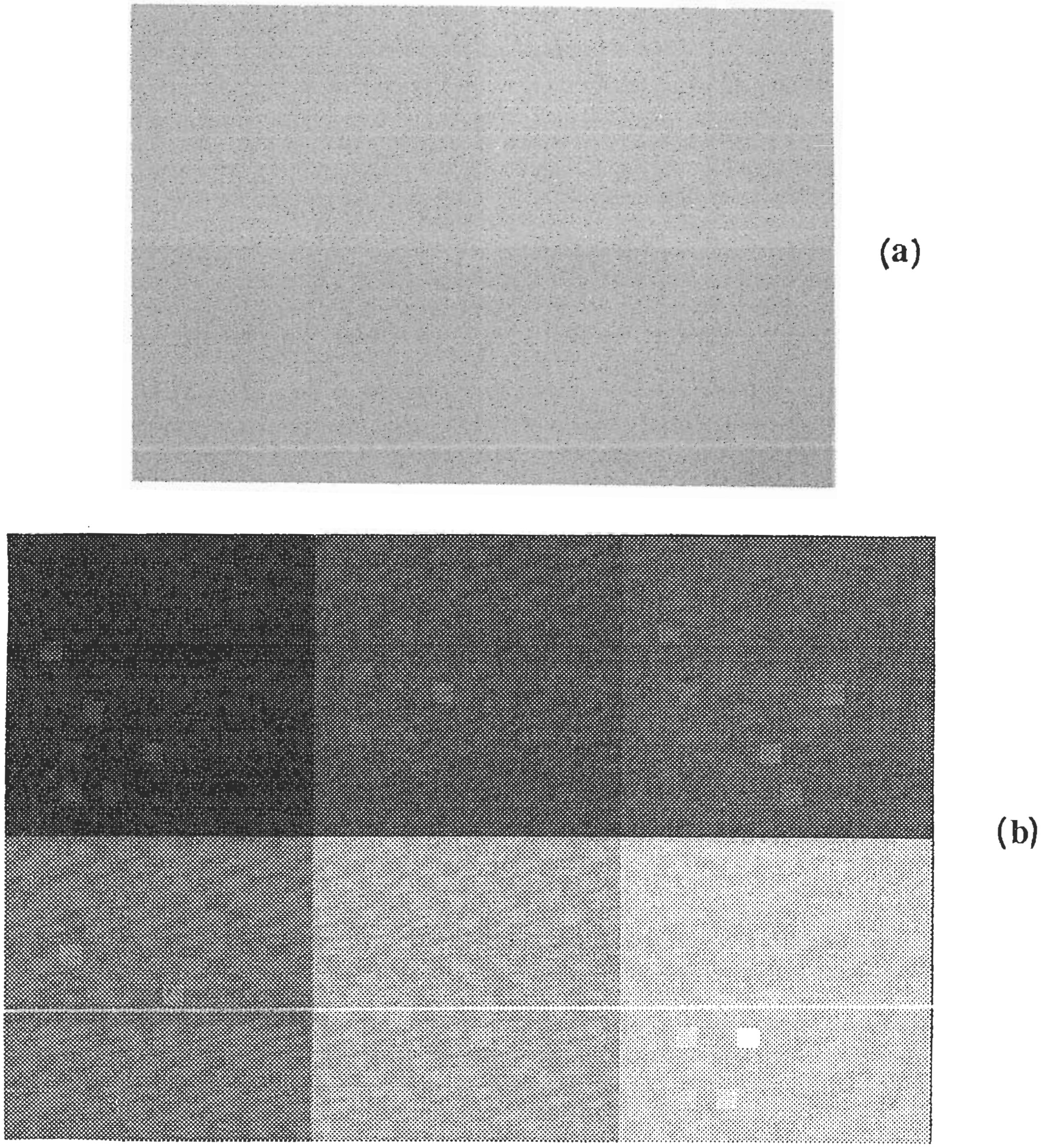

Figure 3: Partitioning of the screen into rectangles with uniform properties: (a) a 2x2 partition with textured-background rectangles, (b) a 2x3 partition, showing the randomly arranged target blocks.

As can be seen in Figure 3a, each rectangle has a different average intensity. This arrangement allowed us to conduct many trials in parallel, instead of requiring a different image for every single value of average intensity, thus speeding up the experimental process. Of course, there is the possibility of interaction among the intensities of the large rectangles, which might affect target discrimination, especially near the borders. However, extensive pilot studies indicated that this was not a problem in our case for the particular conditions used. We also used an arrangement of 2 by 3 large background rectangles, each 240x240 pixels. This is shown in Figure 3b, together with some randomly arranged targets. Let us note that each background rectangle was assigned its own random number of decoys, which varied randomly from rectangle to rectangle and from session to session.

Observers were naive as to the purposes of the experiments and had normal, or corrected to normal, vision. We selected most of the observers from a population that use television mostly as viewers at home (about 90%); the rest were people who work in image processing. We did not observe any systematic differences between the performance of the two groups.

4.1 Experiments of type 1: Uniform background and blocks

4.1.1 Design

These experiments were designed to obtain a good estimate for the value of T_1 of expression (2). Since the test in the algorithm compares the intensities of two nearly uniform blocks for determining when the two are nearly indistinguishable, it is natural to ask how well a human observer can accomplish the same task. Accordingly, we displayed uniform targets against a uniform background to determine how discriminability varied as a function of the intensity difference, and how it was affected by the value of the background intensity. We varied the following parameters in order to study their effect on discriminability: block size, background intensity y, and speed of the targets.

4.1.2 Stimuli-Procedure.

A given background rectangle (see Figure 3) was assigned a uniform intensity y. Target q was assigned intensity value y+(r+G+1)-q, for q=1,2,...,r+G; thus target intensities ranged from y+1 for q=r+G (least discriminable) to y+r+G for q=1 (most discriminable). Each observer was shown the stimuli and was asked to report the total number, w, of discriminable targets.

The difference Δy = (r+G+1)-w is a reasonable estimate for the threshold T_1. Indeed, Δy is a good measure of the *just-noticeable difference (jnd)* for which a target is barely distinguishable from the uniform background y. The dependence of T_1 on block size was studied with blocks of size 8, 16 and 24 pixels on a side (the number of observers that participated in these sub-experiments of type 1 were 10, 10 and 8, respectively). The dependence of T_1 on the background intensity y was investigated by trying several (at least 12) values of y, spanning the range 1 to 254 (8-bit images, values 0 and 255 being reserved for timing signals). Finally, we also studied the effect of target speed on T_1 by generating sequences in which the targets (sizes 16 and 24) moved in the direction of the arrows shown in Figure 2. Three values of speed were used: 50.6, 101.2 and 202.4 minutes of arc per second; the first one corresponds to a velocity vector with one pixel/frame of displacement along the horizontal and vertical dimensions.

4.1.3 Results

The value of $\Delta y = T_1$ is plotted in Figure 4a as a function of background intensity y with block size M as a parameter. There were no systematic differences in the performances of the observers. This is why we obtained each point on the graph as the mean value of Δy = (r+G+1)-w (see above subsection), averaged across all observers. This graph plots the psychophysical results with stationary targets on the 1271Q monitor and it makes it clear that the value of T_1 decreases monotonically with block size for the conditions studied.

The effect of speed on discriminability is shown in Figure 4b, which plots the ratio $\Delta L/L$ versus L with the speed v as a parameter. L is the background luminance corresponding to the intensity y, and ΔL is the jnd luminance corresponding to Δy

(as y varied in the range 10 to 240, luminance varied between 0.9 and 357 cd/m^{2}). As expected, discriminability improves when the targets move rather than when they are stationary. These results allow the algorithm designer to use appropriate values of T_1 in expression (2), depending on the local environment of the block under examination.

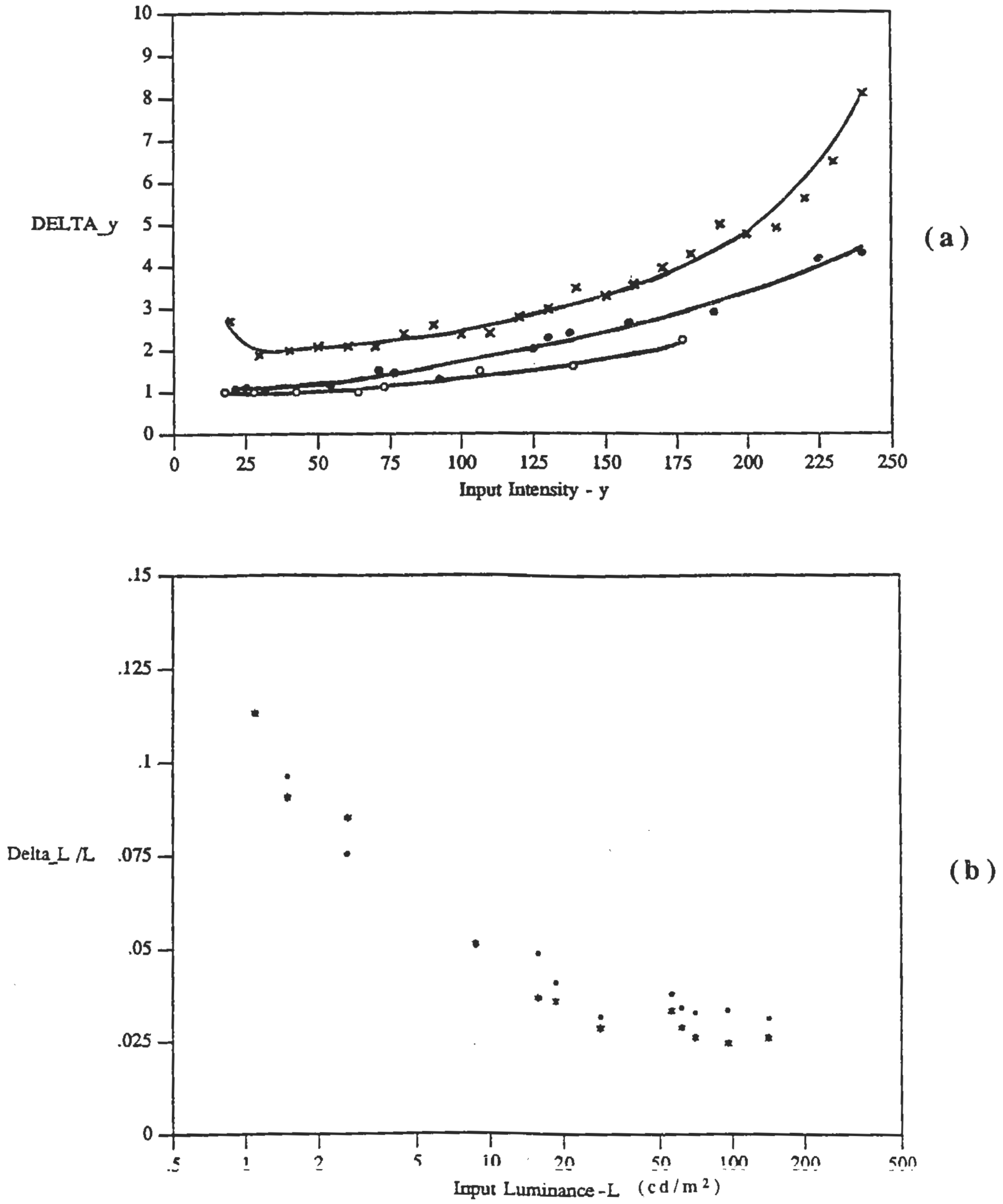

Figure 4: Results from experiments of type 1: (a) The value of $\Delta y=T_1$ is plotted versus the background intensity, y, with block size, M, as a parameter. Δy is the just-noticeable difference (jnd) from y for which a target can be barely discriminated from the background. Data for M=8, 16 and 24 are shown by x's, solid circles and open circles, respectively. The continuous curves have been drawn by hand. (b) The ratio $\Delta L/L$ versus L with target speed, v, as a parameter. L denotes the value of the intensity of the uniform background. Data for v=0 and v=101.2 minutes of arc per second are indicated by circles and asterisks, respectively.

4.2 Experiments of type 2: Textured blocks on uniform background

4.2.1 Design

The objective of this set of experiments was to obtain reasonable estimates for the threshold parameter T_2 of expression (1). This test attempts to classify each block into two major classes: *non-smooth* ones, the intensity of which varies widely across the block's pixels, and *non-textured* blocks which are most often almost uniform in intensity. The question becomes: How would the human visual system (HVS) classify a block? How widely must intensity vary across the pixels for the HVS to characterize the block as non-smooth? To answer this question quantitatively, we displayed several textured blocks on a uniform background. The intensity of the pixels in each block was assigned randomly with a uniform distribution whose average value was the same as that of the background. Thus, targets were indistinguishable from the background on the basis of mean intensity. However, the standard deviation σ varied from block to block. As a result, blocks with high value of σ were easier to see than blocks with low σ. The value of σ for which the block is barely distinguishable provides a good estimate for calculating the threshold T_2. We also studied the effect of target speed and of the spatial-frequency content of the targets on T_2.

4.2.2 Stimuli-Procedure

Targets were created by assigning to each of their pixels a random intensity, distributed uniformly in the interval [y-c, y+c], where y is the intensity of the uniform background. The value of c was different from target to target: target 1 was assigned the largest value of c and each subsequent target, q, was assigned progressively smaller values of c as q increased. To be precise, target q was assigned a value of $c=2(r+G+2-q)$, for $q=1, 2, ..., r+G$. Thus, the values of c varied form $2(r+G+1)$ for the most visible element ($q=1$) to 4 for the least visible one ($q=r+G$). Of course, the standard deviation σ of a uniform distribution in [y-c, y+c] is $c/\sqrt{3}$. The values of M and G in this experiment were 24 pixels and 7, respectively; r varied from 0 to 2 (a maximum of 2 decoys). As with experiments of type 1, we used stationary as well as moving targets with speeds v of 50.6, 101.2 and 202.4 minutes of arc per second.

We also experimented with filtered blocks to study the effect of spatial-frequency content (i.e., visual spatial detail) on discriminability. Thus, in addition to working with the images just described in the last paragraph (what we term *unfiltered* stimuli), we also filtered the blocks with low-pass, two-dimensional, finite-impulse response (FIR) spatial filters with two different cutoff frequencies f_0: a) f_0 = 12.56 cycles/degree, corresponding to 1/4 of the sampling frequency f_s = 50.27 pixels/degree, which we term *half-band* filtered stimuli, since f_0 is half the effective signal band; b) f_0 = 6.28 cycles/degree, which we call *quarter-band* filtered stimuli.

Eight observers took part in this set of experiments. The task of the observers was to report the number, w, of targets that they could discriminate from the background. To study the effect of the background intensity on T_2, we used 16 different values of y in the range 10 to 240. Each observer viewed each of the 16 background intensities, four different speeds (0 to 202.4 minutes per second), and three types of blocks (unfiltered, 1/2- and 1/4-band filtered) for a total of 192 combinations of conditions per observer.

4.2.3 Results

The observer's response w gave us a good estimate of the threshold value c_t of c, for which a textured block was barely segregated from the uniform background of the same intensity y: this is the value of c corresponding to the block numbered (r+G-w+1). The corresponding value of just-noticeable standard deviation is denoted by σ_t. The ratio σ_t/L, where L is the uniform background luminance, is plotted as a function of L in Figure 5a, with speed as a parameter, averaged across observers; as with uniform targets, textured targets are also easier to detect when moving at reasonable speeds, rather than when they are stationary.

Finally, the effect of filtering on discriminability is shown in Figure 5b, which plots σ_t/L as a function of L. Results with unfiltered and quarter-band filtered targets are indicated by solid dots and open circles, respectively.

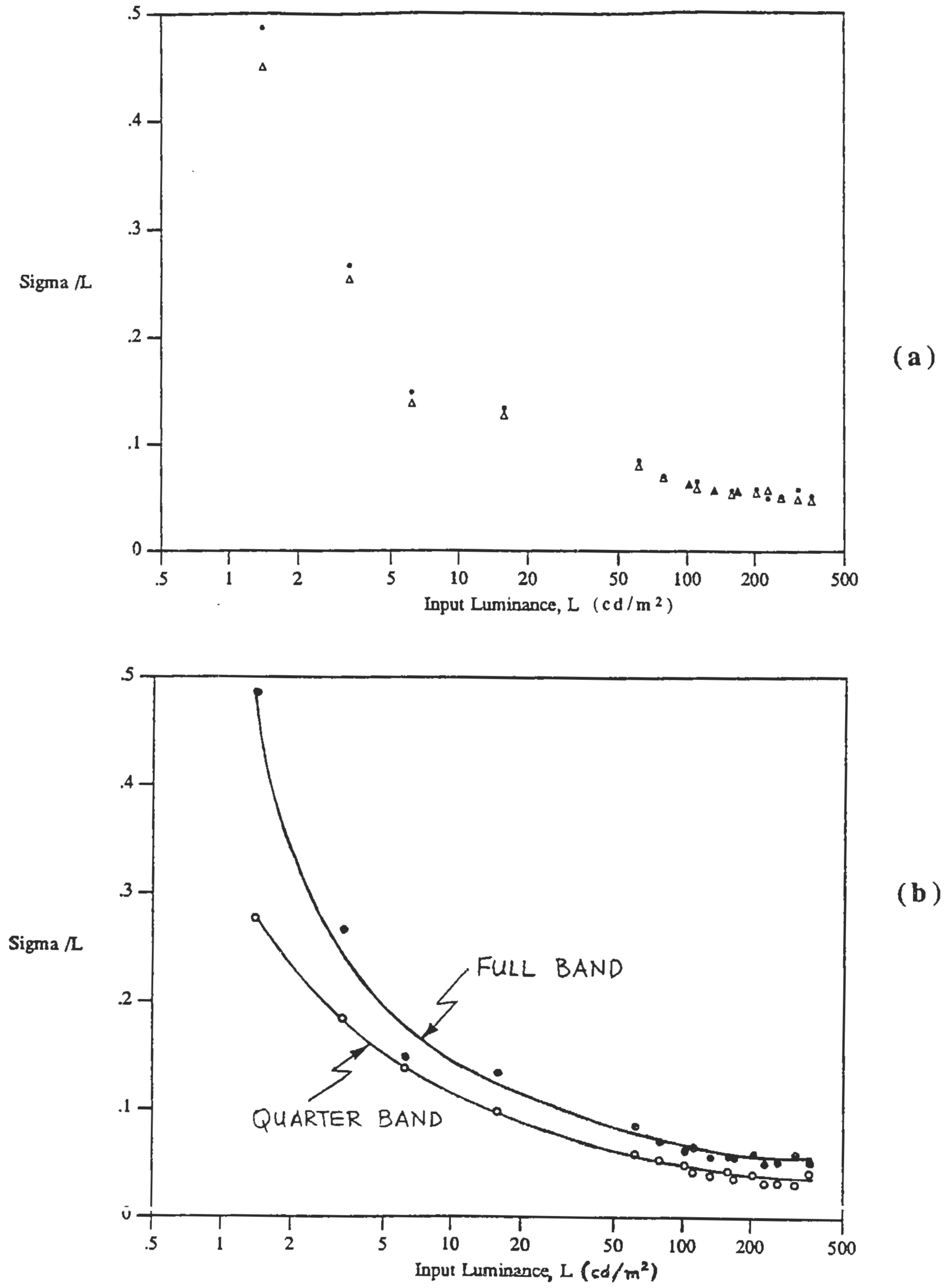

Figure 5: Results from experiments of type 2: (a) The ratio σ_l/L is plotted versus background luminance, L, with target speed, v, as a parameter. Results for v=0 and v=101.2 minutes of arc per second are shown by solid circles and open triangles, respectively. (b) The graph of σ_l/L versus input luminance, L , for data with unfiltered images (solid circles) and with quarter-band filtered images (open circles). The continuous curves have been drawn by hand.

4.3 Experiments of type 3: Textured background and blocks

4.3.1 Design

Let us recall the algorithm's strategy behind the test of expression (3). If the block B under consideration in the present frame is classified as a non-smooth block then, rather than encoding it right away, the algorithm first examines to see whether one can save encoding bits by copying it, provided that it belongs to the stationary background (what is termed as *texture-freeze*). The first test applied to establish texture-freeze conditions is the one in expression (3), involving the threshold T_3, which tests whether the mean values of the current block s_n and the corresponding non-MC block r_{n-1} in the previous frame are sufficiently close. Accordingly, experiments of type 3 were designed to study the conditions under which a textured block can be detected against a background of a similar texture, based on the difference of mean intensities between the two. The objective was to find what is the just-noticeable difference (jnd) between the intensities for human observers so as to get accurate estimates for T_3.

4.3.2 Stimuli-Procedure

Pixels in the background were assigned random intensities, using a Gaussian distribution with mean intensity y and a standard deviation σ_0. The target blocks were also composed of random pixels, using a Gaussian distribution with the same value of standard deviation σ_0. What differentiated them from the background was their mean intensity: Target q was assigned mean intensity $y_q = (r+G+1)-q$ for q=1, 2, ..., r+G. The observer's task was to report the number, w, of visible targets and, just as in Experiment 1, the difference $\Delta y = (r+G+1)-w$ is a good estimate for the threshold T_3 to be used in the coder. Target size was uniform at M=24 for all experiments. Five different values were used for σ_0 (0, 4, 8, 12 and 16) to study the effect of the standard deviation. Eight values were used for the mean intensity of the background in the range 20 to 200. Two speeds were employed: 0 (stationary targets) and 101.2 minutes of arc per second. Finally, we used both unfiltered as well as quarter-band filtered images, obtained as in the case of experiments of type 2.

As before, the task of the observer was to report the number of visible target blocks. Two classes of experiments were conducted. In the first class, in which eight observers participated, all five values of σ_0 were used for each of the eight different mean background intensities, and each experiment was conducted with both, unfiltered and quarter-band filtered images for a total of 5x8x2 = 80 conditions per observer. In the second class, conducted also with eight observers, σ_0 was fixed at 12. All eight mean background intensities were used and, for each of them, we employed stationary and moving targets as well as unfiltered and quarter-band filtered images in all combinations, resulting in 8x2x2 = 32 conditions per observer.

4.3.3 Results

Results were averaged across observers, since there were no systematic differences in their data. The value of Δy as a function of y from the first class of experiments (see previous subsection) is shown in Figure 6a, using the standard deviation σ_0 as a parameter (only three values of σ_0 are shown for clarity). It is obvious that, as σ_0 increases, $\Delta y = T_3$ also increases; this is to be expected, since it becomes more difficult for the HVS to detect the targets on the basis of differences in mean intensity as the texture becomes rougher with increasing σ_0. This trend is also observed in Figure 6b, which plots Δy as a function of the standard deviation, with the input intensity y as a parameter.

The results of the second class of experiments are shown in Figure 6c, which plots the Weber ratio $\Delta L/L$ as a function of L, using data from both unfiltered and filtered images; the value of σ_0 was fixed at 12, as noted in the previous subsection. Notice that the value estimated for T_3 obtained in our experiments is much larger than the value estimated for T_1, even though the tests are very similar. Again, this is to be expected, since it is much easier to detect uniform blocks against a uniform background than to detect textured blocks against a textured background for the same difference in mean intensities. We also found that moving targets are easier to detect than stationary ones.

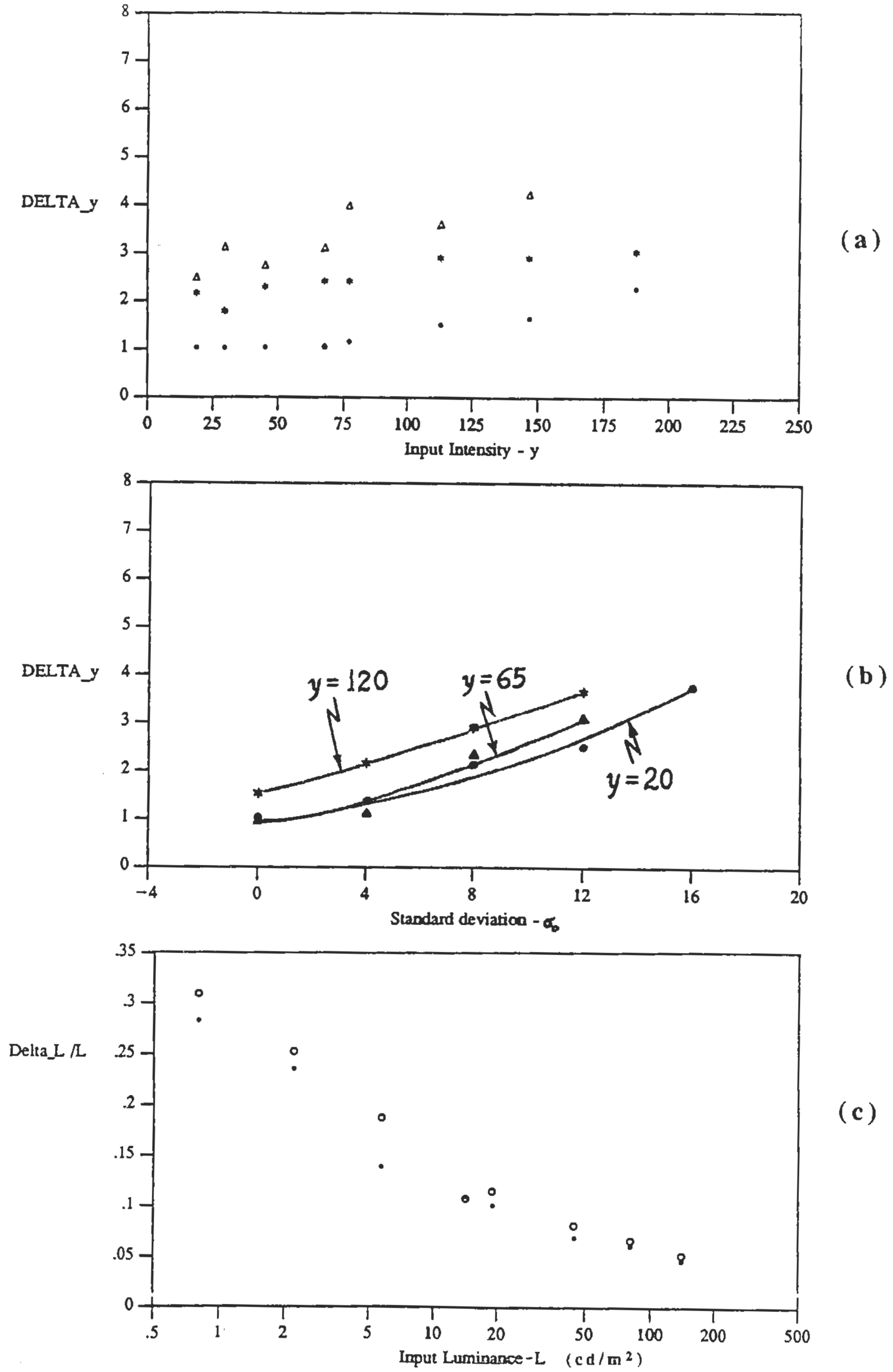

Figure 6: Results from experiments of type 3: (a) The jnd $\Delta y=T_3$ is plotted versus the average background intensity, y, with the standard deviation of the noise texture, σ_0, as a parameter. Data for $\sigma_0=0$, 8 and 12 are shown by solid circles, asterisks and triangles, respectively. Unfiltered images with stationary targets of size M=24 were used. (b) The graph of $\Delta y=T_3$ versus the standard deviation σ_0, with input intensity y as a parameter; the continuous curves have been drawn by hand. (c) The ratio $\Delta L/L$, plotted versus L, for data with unfiltered images (solid circles) and with quarter-band filtered images (open circles).

5. CODING RESULTS - DISCUSSION

The values of threshold parameters T_1, T_2 and T_3 are very important for the encoding algorithm's performance. Of course, T_2 determines which blocks are classified as non-smooth versus non-textured and experiments of type 2 gave us a good feeling on how the HVS would perform such a classification.

The values of T_1 and T_3 have a more direct bearing on the bit rate and the quality of the resulting decoded images (following encoding and transmission): Lower values of T_1 and T_3 result in a smaller number of blocks passing the corresponding tests, hence fewer blocks will be copied, resulting in more encoding bits per pixel per frame. Simultaneously, however, lower values for T_1 and T_3 also result in better *objective* quality for the decoded images. Our goal was to find values that were as large as possible to satisfy the requirement of low bit rate, but not too large, since large values tend to affect image quality adversely. The idea was to find the largest possible value which results in a just-below-the-noticeable image degradation, i.e. an image whose *subjectively* perceived quality is hard to tell apart from another image which took a lot more bits to encode. That was precisely what we strived to achieve with our psychophysical experiments.

A few of our experiments have already been conducted by other researchers in the past, and their results reported in the literature. In particular, data on the Weber ratio DL/L versus L are included in many standard texts on vision[14,15] or on picture coding[16]. There are two reasons why we duplicated such experiments: first, since we were going to study detectability of moving targets, data for which are not as easy to find in the literature, it was convenient to conduct the same experiments with stationary targets with little extra effort; second, it was easy to express our data in terms of the intensity values, y, for the particular display system that was used in the actual encoding experiments, rather than the universal variable of luminance, L (the calculation of y in terms of L is not straightforward).

We emphasize that the appropriate values for the thresholds T_1, T_2 and T_3 were found to vary with the intra-frame (background intensity, standard deviation, spatial-frequency content) and inter-frame (speed) parameters; in short, they vary with the local spatial and temporal environment of the block under consideration. Fortunately, as seen from the plots in Figures 4, 5 and 6, this variation is well-behaved and smooth. Thus, piece-wise linear or quadratic functions can be used to adjust the values of the thresholds according to the local conditions, resulting in an *adaptive HVS-based* threshold modification technique.

The algorithm was implemented by Malah[13] on an Alliant FX/8 parallel computer, capable of a peak performance at 188 MFLOPS (million floating-point operations per second). The coder was tried on several standard and locally available benchmarks: "Miss America", a video of a sitting woman who speaks in a newscaster's style; "Beach and Flowers", which depicts a rotating bouquet of flowers in front of a poster of a beach, with panning; "Table Tennis", a video segment of a ping-pong game with panning, zooming and a textured background; and "Flower Garden", showing a sloped flowery field with a windmill, shot by a camera in a moving vehicle. The last two ones are ISO test sequences[5]. The encoded-decoded sequences were comparable in quality to the original ones, with few noticeable artifacts. In a typical case, the rate was reduced from approximately 100 Mbits/second to roughly 1 Mbit/second (which is equivalent to about 33 Kilobits/frame for a video rate of 30 frames/second) for the encoded ISO sequences.

6. ACKNOWLEDGEMENTS

We thank Mr. K. Mackay for generating stimuli and administering some experiments of type 1 and 2. We also thank Dr. N.S. Jayant, Head of the Signal Processing Research Department at AT&T Bell Laboratories, for providing a stimulating working environment; and Drs. T. Pappas, R. J. Safranek and C. Podilchuk, from the same department, for valuable interactions and comments.

7. REFERENCES

1. P. Pirsch, "Design of DPCM quantizers for video signals using subjective tests," *IEEE Transactions on Communications*, COM-29, pp. 990-1000, 1981.
2. J.J. Szarek, S.A. Rajala, and S.T. Alexander, "An approach to transform domain DPCM coding incorporating the human psychovisual response," *Proceedings of the IEEE ICC Conference*, pp. 443-447, 1983.

3. N. Nill, "A visual model weighted cosine transform for image compression and quality assessment," *IEEE Transactions on Communications*, COM-33, pp. 551-557, 1985.

4. B. Girod, H. Almer, L. Bengtsson, B. Christenson, P. and Weiss, "A subjective evaluation of noise shaping quantization for adaptive intra/interframe DPCM of color television signals," *IEEE Transactions on Communications*, COM-36, pp. 332-346, 1988.

5. L. Chiariglione, "Standardization of moving picture coding for interactive applications," *IEEE Global Communications Conference*, Nov. 1989, pp. 559-563, 1989.

6. S. Okubo, "Video codec standardization in CCITT study group XV," *Signal Processing: Image Communication* 1(1), June 1989, pp. 45-54, 1989.

7. R.J. Safranek, and J.D. Johnston, "A perceptually tuned sub-band image coder with image dependent quantization and post-quantization data compression," *Proceedings of the 1989 IEEE ASSP Conference*, pp. 1945-1948, 1989.

8. H.. Yasuda, "Standardization activities on multimedia coding in ISO," *Signal Processing: Image Communication* 1(1), June 1989, pp. 3-16, 1989.

9. T.R. Reed, T. Ebrahimi, G. Giunta, P. Willemin, F. Marques, T.G. Campbell, and M. Kunt, "Image sequence coding using concepts in visual perception," *Human Vision and Electronic Imaging: Models, Methods and Applications,* B.E. Rogowitz, and J.P. Allebach, eds, Proceedings SPIE 1249, pp. 272-283, 1990.

10. R.J. Safranek, J.D. Johnston, and R.E. Rosenholtz, "A perceptually tuned sub-band image coder," *Human Vision and Electronic Imaging: Models, Methods and Applications,* B.E. Rogowitz, and J.P. Allebach, eds, Proceedings SPIE 1249, pp. 284-289, 1990.

11. R.J. Safranek, J.D. Johnston, N.S. Jayant, and C. Podilchuk, "Perceptual coding of image signals," *Proceedings of the 1990 Asilomar Conference*, Asilomar, California, 1990.

12. J. Yogeshwar, "A new perceptual model for video sequence encoding," Ph.D. Thesis, Rutgers University, New Brunswick, N.J, 1990.

13. D. Malah, "Improvements in hybrid image-sequence coders for storage applications," AT&T Bell Laboratories Technical Memorandum, in preparation, 1991.

14. T.N. Cornsweet, *Visual Perception*, New York: Academic Press1970.

15. H.B. Barlow, and J.D. Mollon, eds. *The Senses*, Cambridge, England: Cambridge University Press, 1982.

16. A.N. Netravali, and B.G. Haskell, *Digital Pictures: Representation and Compression*, New York: Plenum Press, 1988.