Optimality of Greedy Algorithm for Generating Just-Noticeable Difference Surfaces

Jingxi Xu, *Member, IEEE*, and Benjamin W. Wah, *Fellow, IEEE*

***Abstract*—Tuning multimedia applications at run time to achieve high perceptual quality entails the search of nonlinear mappings that determine how control inputs should be set in order to lead to high user-perceived quality. Offline subjective tests are often used for this purpose but they are expensive to conduct because each can only evaluate one mapping at a time and there can be infinitely many such mappings to be evaluated. In this paper, we present a greedy algorithm that uses a small number of subjective test results to accurately approximate this space of mappings. Based on an axiom on monotonicity and the property of just-noticeable differences, we prove its optimality in minimizing the average absolute error between the approximate and the original mappings. We further demonstrate the results using numerical simulations and the application of the mappings found to tune the control of the multimedia game BZFlag.**

***Index Terms*—Greedy algorithm, interactive multimedia, just-noticeable difference (JND), perceptual quality, subjective tests.**

I. Introduction

OPTIMIZING *perceptual quality*, or the quality experienced by humans, is very important in real-time interactive multimedia applications. Because perceptual quality is affected by many hidden factors in a complex fashion, it is not easy to model or quantify. Oftentimes, humans subjects have to grade their perceived quality through *subjective tests*.

Subjective tests can be performed in two ways.

In *absolute rating with single stimulus*, each subject is asked to give an absolute score on the quality of the application under a *control assignment* (or setting of the control parameters). For subjects to give a reliable score, they need to have adequate domain knowledge. Moreover, scoring is difficult when the qualities of two control assignments are not comparable.

In *pairwise comparisons*, subjects are asked a question on two scenarios under different control assignments, one using the original setting (as *reference*), and the other using a modified setting (as *reference + modification*). The two scenarios are presented in a random order. Each subject can choose either alternative or answer undecided. When multiple subjects are asked to do pairwise comparisons, the number of responses from those preferring one over the other is statistical. It is measured by *sample awareness* $\hat{\mathcal{A}}$, or the fraction of subjects who answer correctly to the question posed.

In this paper, we assume pairwise comparisons because even novice subjects without domain knowledge can easily detect the difference between two scenarios.

The three concepts on pairwise comparisons, namely, reference, modification and awareness, can be put together into a continuous 3-D graph [1]. We call it a *Just-Noticeable-Difference (JND) surface* (previously called JND profile and is renamed here to avoid confusion with just-noticeable distortion profiles in video coding [2]). Formally, let awareness $\mathcal{A}$ be the asymptotic value of sample awareness $\hat{\mathcal{A}}$ when the number of subjects is large. Then *JND surface* $\mathcal{A} = p(r, m)$ is a function p that maps reference $r \in [-\infty, +\infty]$ and modification $m \in [-\infty, +\infty]$ to awareness $\mathcal{A} \in [0, 1]$. In this way, a JND surface contains the results of all pairwise subjective comparisons within the range of possible control assignments.

A JND surface is very useful for the run-time control of a multimedia application. Since it captures all subjective-test results conducted offline [1], it can be looked up by the control system at run time with little overhead in order to find the modification to the current state (reference) that will lead to the best perceptual quality.

As an illustration, we use a JND surface to optimize the control of a popular multi-player online tank-battle game BZFlag.[1] In a first-person view, players drive their tanks and shoot each other. The one who shoots the most opponents while being shot less will win the game. To maintain synchronization, an extra buffering delay is needed to compensate for network latency. In our run-time control, we aim to reduce the perceived effects due to this delay. Using the JND surface in Fig. 1(a), we can rearrange the motion of shots in order to conceal their delay effects. The figure shows users' awareness (color) on the delay when the hitting time (x-axis) of a bullet is increased by an extra delay (y-axis). Based on how likely a player will perceive the extra delay under given network latency, the motion of bullets can be adjusted at run time to optimize awareness. (See further details in Section IV-B).

A concept called just noticeable-distortion profile used in video coding [2], [3], [4] has a similar name but is different from the JND surface studied here. A JND profile used in video coding is defined by application-specific properties or closed-form models that can directly be used for optimizing control assignments at run time. Such models and properties were acquired through previous analysis and observations on structural characteristics [5] of signals. However, such application-specific

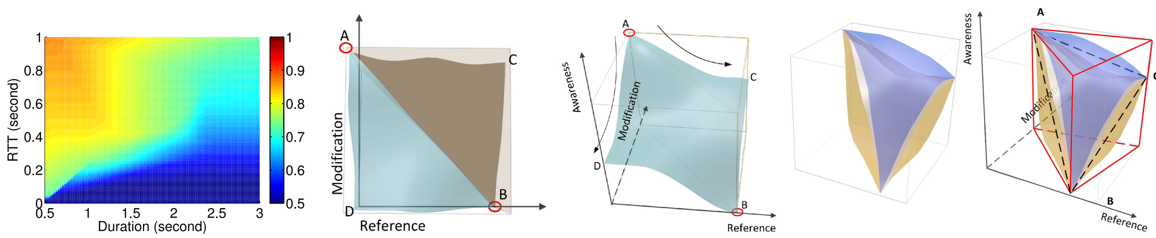
Manuscript received August 05, 2015; revised December 12, 2015 and April 04, 2016; accepted April 06, 2016. Date of publication April 21, 2016; date of current version June 15, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Xiaokang Yang.

The authors are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China (e-mail: jxxu@cse.cuhk.edu.hk; bwah@cuhk.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2016.2557728

[1]"BZFlag 2.4.2," [Online]. Available: http://bzflag.org/



(a) (b) (c)

Fig. 1. JND surface and its various views and explanations. (a) A JND surface with respect to network latency for BZFlag. (b) An example JND surface showing monotonicity, where awareness of A and B defines the range of awareness of points in the prism. (c) Three possible JND surfaces (blue, white, and yellow) passing through A, B, and C whose approximation mesh is denoted by black dashed lines. The awareness of these surfaces is bounded by the height of the triangular prism (in red).

information is not available in many multimedia applications, and offline subjective tests will be required to generate the JND surfaces.

To generate a JND surface, we assume that pairwise subjective tests are conducted in rounds, each involving all subjects on a given pair of scenarios. The awareness values found in the current and previous rounds are then used to determine the new pair of scenarios to be tested in the next round. Hence, the complexity of generating a JND surface depends on both the number of subjects and the number of rounds.

The full generation of a JND surface is expensive because control-assignment values may be continuous, and there can be infinitely many reference-modification pairs to be tested by subjects [6]. The number of tests is even larger when there are combinations of running conditions, say network latency and packet loss rate. Moreover, each offline subjective test may vary from multiple seconds to several minutes. The success of using a JND surface, therefore, depends strongly on how it can be generated efficiently offline.

To predict the duration of subjective tests, previous work studied the mathematical modeling between statistical accuracy and the number of subjects and tests [7], and used the results to guide subjective tests.

We have proposed a greedy algorithm (see Algorithm 1) in our previous paper [1] to sample a JND surface and to interpolate the missing awareness. The algorithm was designed based on two properties on human perception. Firstly, two scenarios in a JND surface need not be compared if the difference is at the 100% JND, since their difference will always be perceived. Secondly, we proposed an axiom on the monotonicity of awareness, which states that awareness is monotonically non-decreasing with respect to modification and is monotonically non-increasing with respect to reference. This property allows certain dominated scenarios in a JND surface to be eliminated in the search.

Unlike traditional methods like PEST [8] or QUEST [9] that rely on strong assumptions on JND, monotonicity is a general property that has been observed in many multimedia systems [1]. By using these two properties, our greedy search focuses on those non-dominated scenarios and selects heuristically the mid-point in an untested region in the next round.

Although the greedy search has performed well experimentally, it is heuristic in nature. In this paper, we prove that the greedy algorithm is optimal in minimizing the average absolute error between the original and the approximate surfaces.

A directly related study is on the approximation of response surfaces in structural safety [10]. Its objective is to accurately construct a response surface with only a few samples, as each sample point is a physics simulation that can take hours to compute. This approach, however, does not consider the unique properties of JND surfaces, namely, monotonicity and binomial distribution of sample data (discussed later). Therefore, its accuracy will be lower than that of our method.

This paper is organized in four sections. Section II presents the basic axioms, assumptions and definitions. In Section III we discuss our method for estimating the error between the original and the approximate JND surfaces. We then prove the optimal test point in each round to be the center point of the region with the largest estimated error. Note that without relying on domain-specific information while using a limited number of subjective tests, the best that our algorithm can achieve in each round is local optimality. Finally, in Section IV, we conduct numerical simulations and apply the JND surface found to tune the controls of BZFlag.

II. Problem Formulation

A. Axiom and Assumptions

Unlike other analytic models studied in psychophysics [11], our study is based on general axioms and assumptions.

Axiom 1: Monotonicity. For given r and m, awareness $\mathcal{A}$ is

1) monotonically non-increasing with respect to r;
2) monotonically non-decreasing with respect to m.

The axiom is true because a given m is less noticeable with a larger r, whereas a larger m is more noticeable for a given r. See Fig. 1(b) for an example.

We also have the following assumptions on subjective tests.

Assumption 1: Unbiased and synchronized tests. Under given r and m, all subjects carry out a pairwise assessment on the same r and m before moving on to another pair.

Asking each subject to carry out only one subjective test on a scenario avoids any bias in repeated tests. Synchronizing the results of all the tests allows the results of one set of tests to guide the selection of r and m for the next set of tests.

Assumption 2: Uniform levels of expertise. All subjects have the same level of expertise, and their ability to perceive the difference between one pair of r and m is independent and identically distributed (IID) with respect to another pair.

This assumption allows responses from multiple subjects to be evaluated statistically. With this assumption, $\hat{\mathcal{A}}$ will converge to $\mathcal{A}$ as more subjects are involved.

Assumption 3: Non-smoothness. The awareness over a JND surface is continuous but not necessarily smooth.

B. Approximating a JND Surface

Definition 1: Approximation mesh. An approximation mesh of a JND surface is a triangular mesh whose vertices are test points corresponding to awareness found by subjective tests.

Let $\mathcal{A} = p(r, m)$ be the JND surface and $f(r, m)$ be the triangular approximation of $p(r, m)$

$$f_{\mathcal{P}}(r, m) = \ell(p(r_x, m_x), p(r_y, m_y), p(r_z, m_z)) \quad \forall\, Tri(x, y, z) \text{ where } x, y \text{ and } z \in \mathcal{P}. \tag{1}$$

Here ℓ is a linear function that interpolates a plane in a triangular region Tri defined by x, y and z belonging to the set of test points $\mathcal{P}$. The set of all $f_{\mathcal{P}}(r, m)$ is the approximation mesh of the JND surface $\mathcal{A}$.

Definition 2: Absolute error $\mathcal{E}$ is the volume between the JND surface and the approximation mesh. Given $f_{\mathcal{P}}(r, m)$ in (1), $\mathcal{E}$ is determined by $\mathcal{P}$ as follows:

$$\mathcal{E} = \int_0^1 \int_0^1 |p(r, m) - f_{\mathcal{P}}(r, m)|\, dr\, dm. \tag{2}$$

To minimize $\mathcal{E}$, we need to optimally place the test points $P_1, P_2, \dots,$ and P_k, where k is the number of tests performed by each subject. In practice, only a finite number of points in a JND surface can be sampled. After sampling the awareness at x, y and z, there are infinitely many JND surfaces passing through them that satisfy monotonicity.

Fig. 1(c) shows that absolute error cannot be uniquely specified because the original nonlinear surface passing through *A*, *B*, and *C* is not unique. (Fig. 1(c) shows 3 of these surfaces.) Further, *C* is not fixed because it could be anywhere along the height of the prism and still satisfies monotonicity. Therefore, we define the average absolute error as follows.

Definition 3: Average absolute error inside a prism is the average of the absolute errors between all possible approximation meshes and the original JND surface, all passing through the highest point A and the lowest point B while satisfying the monotonicity property in Axiom 1.

C. Minimizing the Average Absolute Error

Based on the axiom and assumptions aforementioned, we aim to approximate a JND surface using a triangular mesh generated from a number of test points that are properly placed in order to minimize the sum of the average absolute errors.

Because we do not assume any shape of the JND surface, we will not be able to calculate the average absolute error in a closed form. To this end, we use the volume of the bounding prism as an approximation of the average absolute error (see Section III-A). In addition, because of the non-smoothness of the JND surface (Assumption 3), the surface is approximated by a triangular mesh instead of a global surface function.

Our approach in this paper is to determine test points sequentially after the current state has been known. We take this approach because the awareness of a point is unknown before a subjective test is performed. Moreover, formulating a model to predict future states is difficult because parameters in perceptual modeling are not quantified, and the monotonicity property in Axiom 1 is too weak to identify a unique model.

III. Selecting Test Points to Minimize Average Absolute Error

In this section, we estimate the average absolute error based on the monotonicity property of a JND surface. We then prove the optimality of the greedy method, which sequentially selects test points based on the average absolute error in the last step.

Fig. 1(b) shows the top and the side views of a JND surface with axes normalized to the $[0, 1]$ range. The awareness of a test point (in red) is known after a subjective test. Referring to the points in the top view, points to the top-left of the red point refer to points with smaller reference and larger modification.

A. Properties Due to Monotonicity

Lemma 1: The awareness of a point indicates the lower (*resp.,* upper) bound of the awareness of points to the upper-left (*resp.,* lower-right) of this point.

Proof: From Axiom 1, the awareness of points to the top-left of a test point is larger than its own awareness, and the awareness of points to the bottom-right is smaller. ■

Lemma 2: A pair of top-left and bottom-right test points prescribe the bounding right triangular prism of all possible JND surfaces that pass through these two points.

Proof: Due to monotonicity, all points in the brown right triangular prism in the top view of Fig. 1(b) have awareness between the awareness of A and B. It further shows that the JND surface should be inside this right triangular prism. ■

Next, we show the property of the other corner points.

Lemma 3: The top-right and the bottom-left test points [C and D in Fig. 1(b)] cannot bound the awareness of any JND surface passing through the right triangular prism.

Proof: This is true because monotonicity does not apply in the top-right to the bottom-left direction. ■

Corollary 1: The awareness of A, B, and C in Fig. 1(b) alone is not sufficient for calculating the average absolute error in this region.

Proof: As shown in Fig. 1(c), there can be infinitely many JND surfaces passing through a given set of points A, B, and C. Without a closed form of the various surfaces, we cannot calculate the absolute error between each of these surfaces and the triangular mesh. For the same reason, knowing the awareness of C is not sufficient for computing the average absolute error. ■

With the properties above, we show in the following corollary that the best triangulation of the JND surface is when every triangle is a right-angle triangle when *projected* to the reference-modification plane.

Corollary 2: To conform to monotonicity, the best triangulation of a JND surface, when projected onto the reference-modification plane, should be right-angle triangles.

Proof: With Lemma 2, points A and B prescribe the upper and lower bounds of awareness of points of the corresponding JND surface. Because the directions of monotonicity are from left to right and from top to bottom in the projection, the boundary of the projection should be along these directions. Therefore, each projected region should be a right-angle triangle. ■

Theorem 1: The average absolute error inside a bounding right-angle triangular prism is proportional to the volume of the prism.

Proof: When we resize the bounding right-angle triangular prism in Fig. 1(b) by α, β and θ along the X, Y and Z axes, we also resize the surfaces it bounds. Each surface in this prism that satisfies Axiom 1 is defined by A and B. The absolute error between the original and the approximate surfaces is

$$\begin{aligned}\mathcal{E}_{\text{new}} &= \int_0^\beta \int_0^\alpha \left|\gamma p\left(\frac{r}{\alpha}, \frac{m}{\beta}\right) - \gamma f_{\mathcal{P}}\left(\frac{r}{\alpha}, \frac{m}{\beta}\right)\right| dr\, dm \\ &= \alpha\beta\gamma \int_0^1 \int_0^1 \left|p\left(\frac{r}{\alpha}, \frac{m}{\beta}\right) - f_{\mathcal{P}}\left(\frac{r}{\alpha}, \frac{m}{\beta}\right)\right| d\frac{r}{\alpha}\, d\frac{m}{\beta} \\ &= \alpha\beta\gamma \int_0^1 \int_0^1 |p(r,m) - f_{\mathcal{P}}(r,m)|\, dr\, dm \\ &= \alpha\beta\gamma\mathcal{E}_{\text{original}}. \end{aligned} \tag{3}$$

The absolute error is, therefore, proportionally changed with the volume of the prism. Hence, the average absolute error of all surfaces is proportional to the volume of the prism. ■

B. Selection of the Next Test Point

Based on the average absolute error estimated in Theorem 1, we discuss in this section our strategy of choosing a suitable point in the JND surface to test in the current round that can maximally reduce the average absolute error. We show that the center point of a rectangular region (top view) is the best point to test in that region.

Lemmas 2 and 3 have shown that a top-left and a bottom-right test point can bound a rectangular region. Inside this region, there are infinitely many possible JND surfaces that pass through the two corner points while satisfying Axiom 1. Let $\mathcal{S}$ be the collection of these JND surfaces.

Definition 4: $\mathcal{S}$ is the collection of JND surfaces in a rectangular region (bounded by top-left point A and bottom-right point B) that pass through A, B while satisfying Axiom 1.

Next, we show that the surfaces in $\mathcal{S}$ appear in pairs, and each pair has the same error with respect to the center-point of the normalized rectangular region.

Theorem 2: Symmetry. In the normalized cube of length 1 bounded by A and B in Fig. 2, for any JND surface $s \in \mathcal{S}$, there exists exactly one $s' \in \mathcal{S}$ that is axially symmetric to s. That is, s can be rotated 180° around Line L passing through $(0, 0, 0.5)$ and $(1, 1, 0.5)$ to get to s'.

Proof: For any (x, y, z) in the original surface s, the transformation to get to (x', y', z') can be done [12] by first transforming

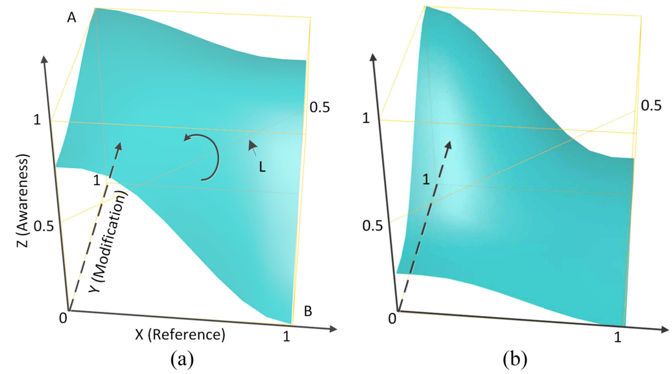


Fig. 2. Any JND surface in the bounding cube has exactly one symmetric JND surface around Line L passing through $(0, 0, 0.5)$ and $(1, 1, 0.5)$ (yellow line in the middle of the cube) when rotated 180°. (a) Original surface s (b) s' rotated 180°.

the whole space (with the surface and L), while keeping the X, Y and Z axes unchanged, so that L is moved to the Z axis. We then rotate the space by 180° about the Z axis. Finally, we transform the whole space so that L goes back to the original place.

1) Transform the space using matrix T_p while keeping the orientation of Line L in such a way that L passes through the origin and lies on the X-Y plane.
2) Next, rotate using T_{XZ} the above space around the Z axis, and let L lie on the X-Z plane. After the rotation, L lies along the X axis.
3) Rotate using T_Z the space about the Y axis and let L lie along the Z axis.
4) Rotate by 180° using R_Z about the Z axis.
5) Apply the inverse transformations of Steps 3, 2, and 1, respectively. That is, rotate using T_Z^{-1} the space about the Y axis and let L lie along the X axis (the state before applying Step 3), and so on.

Without showing the details (due to limited space), the aggregate transformation matrix is

$$T_{\text{comb}} = T_P^{-1} T_{XZ}^{-1} T_Z^{-1} R_Z T_Z T_{XZ} T_P, \tag{4}$$

$$(x', y', z', 1)' = T_{\text{comb}}(x, y, z, 1)' = (y, x, 1 - z, 1)'. \tag{5}$$

Firstly, it is clear that for any $0 \le x, y, z \le 1$, we have $0 \le x', y', z' \le 1$; i.e., the mirror point is still inside the bounding cube. It is also easy to show that each point in s has exactly one corresponding point in s'.

Secondly, the symmetric JND surface s' still satisfies the monotonicity property in Axiom 1 due to (5)

$$\begin{aligned} x'_1 < x'_2 \text{ and } y'_1 > y'_2 &\Rightarrow y_1 < y_2 \text{ and } x_1 > x_2 \\ &\Rightarrow z_1 < z_2 \quad (\text{Axiom 1}) \Rightarrow z'_1 > z'_2. \end{aligned} \tag{6}$$

■

With Theorem 2, the center-point of the cube (from the top view) has the same error to a pair of JND surfaces in the cube. To best approximate a JND surface, choosing the center-point of the region to test is the best choice; otherwise, the test point can be biased towards one of the surfaces that exist in pairs.

Based on Theorem 1, the optimal test point in each round to minimize average absolute error is the center point inside the region with the largest volume.

Theorem 2 also applies to regions subdivided from a larger region. This is true because the continuity stated in Assumption 3 can be satisfied even when the neighboring regions are considered separately. For any JND surface in S_1 of a given region, the boundary curve of this surface is monotonic. Because S_2 of the neighboring region contains all the monotonic surfaces, it would also contain the surface with this boundary curve.

C. Uncertainties due to Limited Subjective Tests

Because each subjective test is a sampling process, the sampled awareness is a random variable. The sum of IID random variables (Assumptions 1 and 2) with Bernoulli distribution follows a binomial distribution. In this section, we estimate the effects on the next point to test due to binomial uncertainties. The process is heuristic because there is no closed-form solution.

In the selection of the best point to test, we need to compare the volume of each region and to find the one with the largest value. We heuristically assume the independence of awareness $\mathcal{A}$. In this case, we calculate the volume of the prisms independently and choose the center point of the largest prism to test. The expectation of the volume is as follows:

$$\mathcal{V}^*_{A,B} = \ell^2 \int_0^1 \int_0^{\mathcal{A}_A} (\mathcal{A}_A - \mathcal{A}_B) Pr(\mathcal{A}_A|\hat{\mathcal{A}}_A) Pr(\mathcal{A}_B|\hat{\mathcal{A}}_B) \times d\mathcal{A}_B \, d\mathcal{A}_A \tag{7}$$

where $\mathcal{A}$ (*resp.*, $\hat{\mathcal{A}}$) is the real (*resp.*, sampled) awareness; $\mathcal{A}_A$ (*resp.*, $\mathcal{A}_B$) is the upper (*resp.*, lower) bound awareness in this prism; and ℓ is the length of the diagonal of the square region.

After conducting the subjective tests in a round, we need to adjust the awareness found (each a random variable) to ensure that they satisfy monotonicity required in Axiom 1. Our heuristic approach is to calculate the average of all JND surfaces that satisfy Axiom 1 by integrating over all possible values of awareness $\mathcal{A}$ that satisfy monotonicity

$$p(r,m) = \int f_P(r,m) Pr(\mathcal{A}|\hat{\mathcal{A}}) d\mathcal{A}, \tag{8}$$

where f is the region-wide linear approximation based on the set of test vectors $\mathcal{P}$ (Definition 1) that satisfy monotonicity. The joint probability of dependent random variables in the vectors $\mathcal{P}$ is calculated by discretizing the value of awareness. After this adjustment, $p(r,m)$ is the final desired output.

Algorithm 1 summarizes the procedure of the greedy algorithm for the case where limited subjective tests give rise to sampling uncertainties that follow the binomial distribution. In each round, we sample the center point inside the region with the largest average absolute error estimated by (7), before further dividing that region. The final result is attained by using the adjustment in (8) to ensure monotonicity.

Fig. 3 illustrates a sequence of subjective tests performed. For simplicity, we assume no binomial errors in subjective tests.

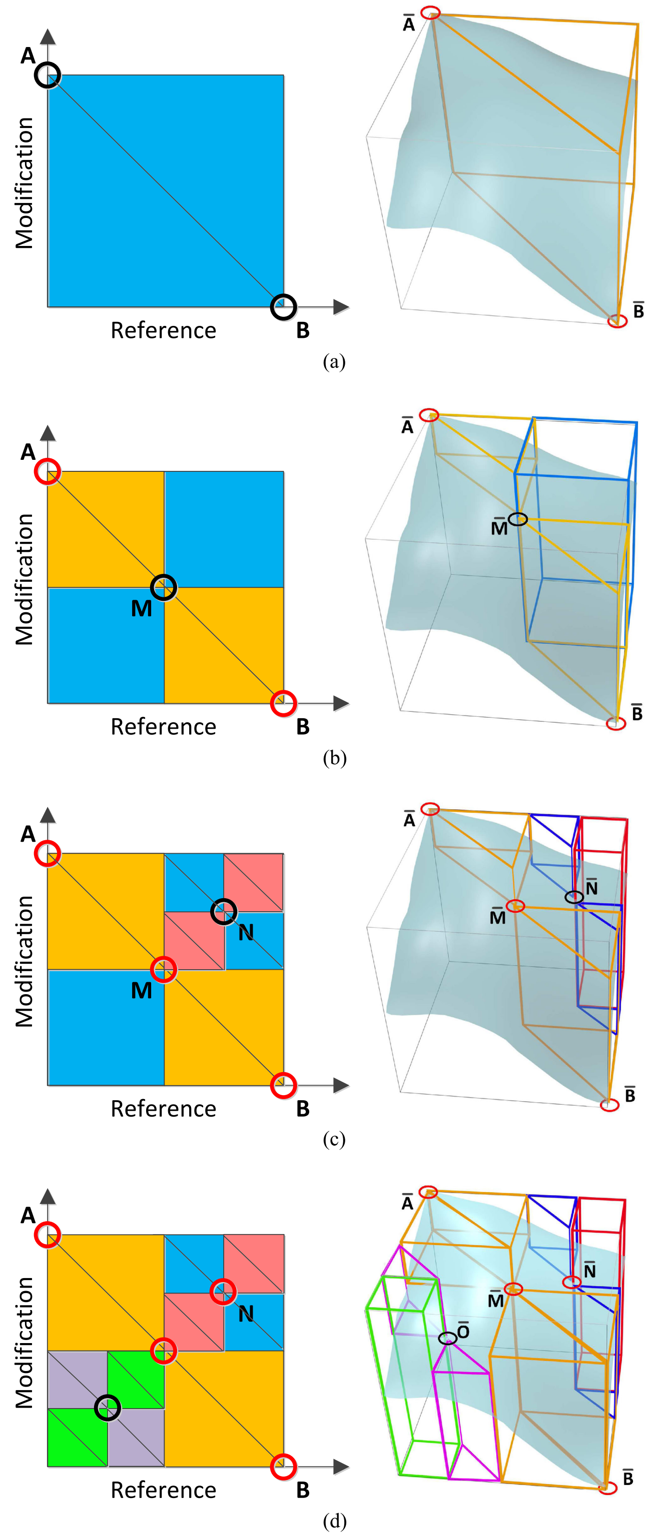


Fig. 3. Illustration of a sequence of subjective tests and the generation of the approximation mesh. The left panels show the test points (say A), and the right panels show how the test results (say awareness $\hat{\mathcal{A}}$) reduce the average absolute error (or volume). The blue half-transparent surface is the real JND surface we want to approximate, the red circles are measured test points, and the black point is to be tested in this step. For clarity, some nearby points are not connected because several prisms as well as points attached to them are hidden. (a) Step 1. $p_A = 1, p_B = 0$ (normalized). (b) Step 2. $p_M = 0.7$. (c) Step 3. $p_N = 0.7$. (d) Step 4. $p_O = 0.6$.

Algorithm 1: Finding JND surface with sampling uncertainties.

Require: $\hat{\mathcal{A}} = \hat{p}(ref, mod)$: fraction of subjects who correctly identify the modified control input $ref + mod$; δ: required error threshold.
Ensure: JND surface $\mathcal{A} = p(ref, mod)$;
Measure $\hat{p}(0, 1)$ and $\hat{p}(1, 0)$; add $(0, 1)$ and $(1, 0)$ to P_{tested};
while $\max \mathcal{V}^*_{i,j} > \delta$ where $i, j \in P_{\text{tested}}$, $\mathcal{V}^*$ is defined in (7), and no mid-point in between was tested **do**;
  Perform subjective tests to measure $\hat{\mathcal{A}}_m$, where m is the mid-point of i and j;
  Add m to P_{tested};
end while
$\forall\, i \in P_{\text{tested}}$, fix $\hat{\mathcal{A}}_i$ that don't satisfy monotonicity by (8);
Interpolate $p(ref, mod)$ with the fixed P_{tested};

In Fig. 3(a), we start testing A and B. The bounding prism is shown in yellow. In the side view, we hide the prism at the opposite side of the diagonal for clarity.

In Fig. 3(b), we test the mid-point M at $a = 0.5, b = 0.5$, as there is only one prism (the opposite one is symmetric) and it has largest volume. The test result is shown in the right panel, where the awareness of M is shown as its height. The two yellow prisms are shrunk after M has been added. Note the height of the blue prisms. Because M does not bound the awareness of points to its top-right, the awareness of points in the blue prisms (merged into a blue cuboid) is still bounded by the awareness of A and B. Similarly we test mid-point N in Fig. 3(c) and O in Fig. 3(d).

Finally, we interpolate points not tested in order to approximate the entire surface.

IV. Experimental Results

In this section we present experimental results using synthetic data and a real application. The goals of these experiments are to demonstrate that our proposed greedy algorithm can attain the desired accuracy with a limited number of subjective tests and to show that the JND surface found can be used in a real application to improve perceptual quality.

A. Monte Carlo Simulations

1) Synthetic JND Surfaces: We generate synthetic JND surfaces with $N = 6 \times 6$ grid points evenly placed on the X-Y plane. Their awareness is randomly generated by a uniform distribution but satisfies the monotonicity requirement in Axiom 1. We then use cubic interpolations to expand them into a smooth surface with 101×101 grid points. We generate a new JND surface for each application of Algorithm 1.

2) Algorithms Tested: We evaluate Algorithm 1 by selecting $k \ll N$ test points in a step-by-step fashion using the information collected. After selecting the points, the surface is interpolated linearly to expand the awareness to a 101×101 discrete-point set, which is compared with the synthetic surface. We measure quality by the average approximation error.

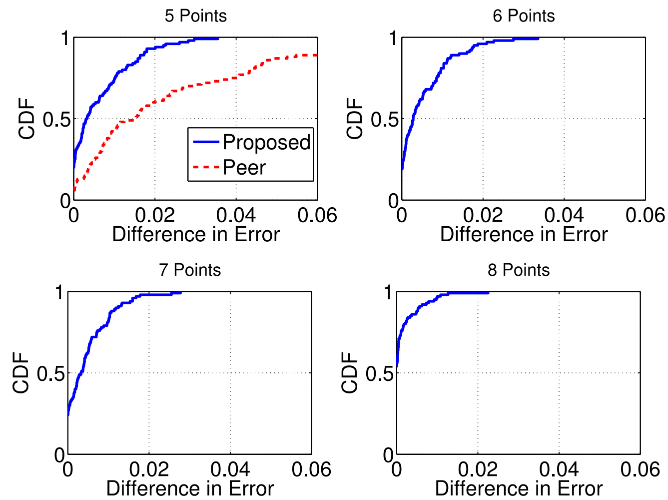


Fig. 4. Empirical CDFs showing the difference in approximation errors between the benchmark and our proposed method, as well as between the benchmark and a peer method [10] with five points, with no binomial uncertainties in subjective tests (averaged over 300 different synthetic JND surfaces).

Next, we find the upper-bound performance by approximating the synthetic surface using $k \ll N$ points as best as possible. This is done by a brute-force method that tries all possible ways of choosing k among N points and finding the placement that minimizes the average approximation error.

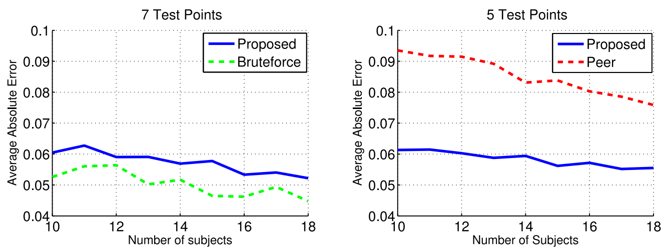
We also compare the performance against a peer method [10] described in Section I. Because the method only sampled patterns with 3, 5, 9, 13, and more test points, considering the complexity, we only tested 5 points in our experiments.

3) Experimental Results Without Binomial Sampling Uncertainties: We assume that the exact awareness is returned when testing a point on the surface. The purpose of the experiment is to evaluate the approximation quality of the algorithm without sampling uncertainties.

Fig. 4 shows the results. When 5 points were generated, the error between the benchmark and our method is small, whereas the peer method performed much worse. When 7 points were generated, 90% of the surface has error smaller than 0.015, which is adequate as the range of awareness is between 0.5 and 1.0. The results show that our method is competitive with respect to the brute-force method.

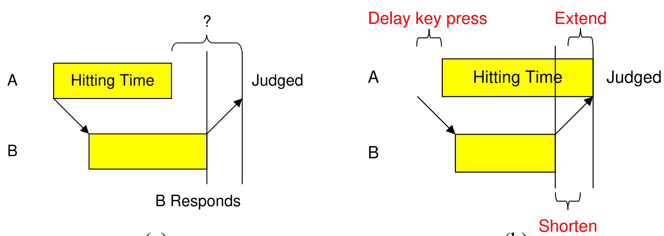
4) Experimental Results With Binomial Sampling Uncertainties: This experiment is used to evaluate the performance when the number of tests is finite. In this case, a random error is introduced in the sample awareness, which follows the binomial distribution $B(n, p)$. For the benchmark method, we use the brute-force method to find the optimal placement. Because this tries all possible combinations, it will have smaller errors when compared to the greedy method.

Fig. 5(a) shows the average difference between the benchmark and our greedy method. The result shows two properties. 1) Our method has comparable precision in approximating the JND surface even though each point can only be sampled once,



(a) (b)

Fig. 5. Comparison of approximation errors when the results of subjective tests are with binomial errors (averaged over 300 different synthetic JND surfaces). The errors of our proposed algorithm in (a) decrease faster than those in (b) because more test points are used.



(a) (b)

Fig. 6. (a) Due to network latency there is a blank period in the schedule in which we do not know the outcome of the shot. (b) To conceal the blank period, we propose to delay the start of the bullet, extend the hitting time in the attacker's view, and shorten the hitting time in the opponent's view, in a way that the combined effect is less noticeable..

whereas the benchmark method can re-sample each point multiple times and find the one with the smallest error. 2) The difference in average approximation error is generally reduced as the number of subjects is increased, although there are some fluctuations due to binomial sampling uncertainties.

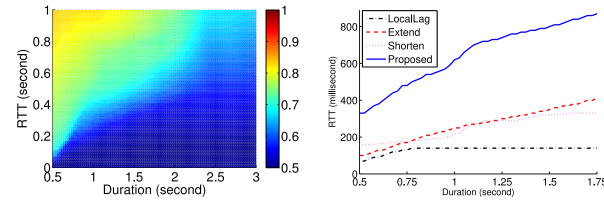
Fig. 5(b) compares the performance between our method and the peer method [10]. It shows that our method outperforms the peer method, independent of the number of subjects.

B. Experiments on Multimedia Game BZFlag

Finally, we demonstrate the effectiveness of using JND surfaces in optimizing a popular multi-player online tank-battle game BZFlag. In a first-person view, players drive their tanks and shoot at each other. The one who shoots the most opponents while being shot less will win the game.

The delay effects of BZFlag can be noticed when the outcome of a bullet shooting an opponent is delayed. Due to network latency, the message of shooting a bullet will arrive late in the defender's view, and whether she can dodge the bullet is also returned late in the attacker's view. This leads to a noticeable artifact because a blank period exists in the attacker's view after shooting the bullet [see Fig. 6(a)]. However, this blank period is needed for ensuring synchronization in the outcomes. It is used to ensure the consistency of a bullet hitting an opponent in the attacker's view, while dodging the shot in the opponent's view.

To reduce the delay effects, we propose to rearrange a bullet's motion by three delay-concealment strategies: delaying the shot in the attacker's view (i.e. the local lag strategy), extending the



(a) (b)

Fig. 7. (a) Due to network latency there is a blank period of the schedule where we do not know the judgment of the bullet. (b) Our proposed strategy can attain the same 75% awareness in a significantly more delayed network.

hitting time (or duration of the flight) of the bullet, and shortening its hitting time in the opponent's view [13] [see Fig. 6(b)].

We use a JND surface to determine the extent of modifying the hitting time so that players will not perceive a change in their game-playing experience and the resulting artifacts.

1) *Subjects:* We recruited 14 paid students to conduct the subjective tests who had experience in First-Person-Shooting games, and were expected to be familiar with game control and have sharp awareness on delay effects.

2) *Experiment Setup:* A computer lab with 16 computers was used for the experiments. One computer was used to schedule the experiments and to analyze the results. Another was set up as the game server and the troll router, which emulated network latencies by adding delays to forwarded packets. The remaining 14 computers were set up as game clients that ran the original and the modified BZFlag.

Players were placed uniformly on a 200×200 unit^2 empty map. We had 2 maps with 5 players each, and 1 map with 4 players. The speed of a bullet was 100 units/sec., which allowed the game to move at a fast pace.

3) *Test Process:* We conducted 2 sets of tests in 2 h.

The first set was for measuring JND surfaces with our proposed algorithm. During the tests, subjects played two 1-min game scenarios, each with a unique set of control assignments. The subjects then compared the speed of the bullets in the 2 scenarios and reported which was faster. Responses were collected immediately for calculating awareness.

The second set was for collecting subjects' opinion on the game experience. We first let them play the game in groups of 4 to 5 people, with the default setting under two network conditions with 0 ms and 1000 ms RTT. We explicitly told them the latency value. We then let them play the modified version of the game using our proposed delay-concealment strategy. In all tests, we collected their comments regarding the playing experience. They were encouraged to report artifacts immediately by saying aloud or marking them down on paper.

4) *Results:* We collected the JND surfaces of extending the hitting time in the attacker's view, shortening the hitting time in the defender's view, and delaying the start of the shot. In collecting the test points of each of these JND surfaces, we tested 7 rounds for each of the 3 surfaces with 14 subjects (*i.e.*, 21 tests per subject).

We then performed simulations with BZFlag controlled by the JND surfaces obtained. Fig. 7(a) shows the combined JND surface based on the JND surfaces we have measured with the proposed greedy algorithm. Comparing it with the surface of merely using the extended-duration strategy shown in Fig. 1(a), our new strategy significantly reduces the awareness of delay artifacts. Fig. 7(b) further presents the network RTT under which 75% awareness (75% of the subjects noticing the difference) is achieved. Our strategy has clearly increased the tolerable RTT by more than 250 ms, even when the referenced hitting time is as short as 500 ms. Note that more improvements can be gained when the hitting time increases.

To further support the analysis, we let subjects play a complete BZFlag game (lasting 5 min) in a network in which delays can be controlled. When subjects played the original BZFlag in a network with 1000-ms RTT, they could easily notice the difference when compared to the no-delay case. Subjects complained that bullets could not shoot an opponent correctly, and that an opponent could dodge a bullet even after the bullet had shot her. These happened because the judgment of the bullet arrived late from the opponent's view. In short, delay artifacts can significantly affect the game experience in a fast-paced shooting game.

We then let subjects play the modified BZFlag controlled by the JND surfaces found. During the game, no one complained about synchronization issues. Instead, 10 out of 14 players reported better playing experience, as if the game were run on a network with much smaller RTT, even though the RTT was still 1000 ms; 2 reported the same experience, and 2 reported poorer experience.

V. Conclusion

In this paper, we have presented a general framework for optimizing perceptual quality in multimedia applications. We have proved the optimality of the greedy algorithm for finding JND surfaces using subjective tests. We have shown that the accuracy of approximation can be achieved by a limited number of subjective tests using both Monte-Carlo simulations and a real application BZFlag. Experimental results have shown that the JND surfaces found can provide significant improvements in perceptual quality in BZFlag.

Acknowledgment

The authors cordially acknowledge the comments from the three reviewers and Dr. Z. Chen.

References

[1] J. Xu and B. Wah, "Optimizing the perceptual quality of real-time multimetric multimedia applications using JND profiles," *IEEE Multimedia Mag.*, vol. 22, no. 4, pp. 14–28, Oct.–Dec. 2015.

[2] N. Jayant, "Signal compression: Technology targets and research directions," *IEEE J. Sel. Areas Commun.*, vol. 10, no. 5, pp. 796–818, Jun. 1992.

[3] X. Yang, W. Ling, Z. Lu, E. P. Ong, and S. Yao, "Just noticeable distortion model and its applications in video coding," *Signal Process., Image Commun.*, vol. 20, no. 7, pp. 662–680, 2005.

[4] C.-H. Chou and Y.-C. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 467–476, Dec. 1995.

[5] L. Ma, K. N. Ngan, F. Zhang, and S. Li, "Adaptive block-size transform based just-noticeable difference model for images/videos," *Signal Process., Image Commun.*, vol. 26, no. 3, pp. 162–174, 2011.

[6] B. Sat and B. Wah, "Statistical scheduling of offline comparative subjective evaluations for real-time multimedia," *IEEE Trans. Multimedia*, vol. 11, no. 6, pp. 114–1130, Oct. 2009.

[7] J.-S. Lee, "On designing paired comparison experiments for subjective multimedia quality assessment," *IEEE Trans. Multimedia*, vol. 16, no. 2, pp. 564–571, Feb. 2014.

[8] M. Taylor and D. Creelman, "PEST: Efficient estimates on probability functions," *J. Acoust. Soc. Amer.*, vol. 41, no. 4A, pp. 782–787, 1967.

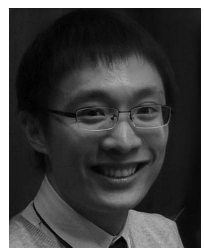
[9] A. Watson and D. Pelli, "QUEST: A Bayesian adaptive psychometric method," *Percept. Psychophys.*, vol. 33, no. 2, pp. 113–120, 1983.

[10] V. Romero, L. Swiler, and A. Giunta, "Construction of response surfaces based on progressive-lattice-sampling experimental designs with application to uncertainty propagation," *Structural Safety*, vol. 26, no. 2, pp. 201–219, 2004.

[11] G. Fechner, *Elements of Psychophysics*, vol. I. New York, NY, USA: Holt, Rinehart and Winston, 1966.

[12] G. Murray, "Rotation about an arbitrary axis in 3 dimensions," Jun. 6, 2013. [Online]. Available: http://inside.mines.edu/fs_home/gmurray/ArbitraryAxisRotation/

[13] J. Xu and B. Wah, "Concealing network delays in delay-sensitive online interactive games based on just-noticeable differences," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2013, pp. 1–6.



Jingxi Xu (S'13–M'14) received the B.S. degree in computer science from Sun Yat-Sen University, Guangzhou, China, in 2009, and is currently working toward the Ph.D. degree in computer science and engineering at The Chinese University of Hong Kong, Hong Kong, China.

His research interests include strategies for the optimization of multimedia systems, especially the video conferencing and online interactive games.

Benjamin W. Wah (S'74–M'77–SM'85–F'91) received the Ph.D. degree in computer science from the University of California at Berkeley, Berkeley, CA, USA, in 1979.

He is currently the Provost and Wei Lun Professor of Computer Science and Engineering at The Chinese University of Hong Kong, Hong Kong, China, as well as the Chair of Research Grants Council of Hong Kong. Before this, he served as the Director of the Advanced Digital Sciences Center (a research center of UIUC) in Singapore, as well as the Franklin W. Woeltge Endowed Professor of Electrical and Computer Engineering, and as a Professor of the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA. He has served on the faculty of Purdue University, West Lafayette, IN, USA. His research interests include the areas of nonlinear search and optimization, multimedia signal processing, and computer networks.

Prof. Wah is a Fellow of the AAAS and ACM. He cofounded the IEEE Transactions on Knowledge and Data Engineering in 1988 and served as the Editor-in-Chief between 1993 and 1996, and is the Honorary Editor-in-Chief of the *Knowledge and Information Systems*. He currently serves on the Editorial Board of *Information Sciences*, the *International Journal on Artificial Intelligence Tools*, the *Journal of VLSI Signal Processing*, the *World Wide Web*, and the *Journal of Computer Science and Technology*. He has served the IEEE Computer Society in various capacities, including the Vice President for Publications (1998 and 1999) and the President (2001). He has been the recipient of number of awards for his research contributions, including the IEEE CS Technical Achievement Award (1998), the IEEE Millennium Medal (2000), the Society for Design and Process Science Raymond T. Yeh Lifetime Achievement Award (2003), the IEEE-CS W. Wallace-McDowell Award (2006), the Pan Wen-Yuan Outstanding Research Award (2006), the IEEE-CS Richard E. Merwin Award (2007), the IEEE-CS Technical Committee on Distributed Processing Outstanding Achievement Award (2007), the IEEE-CS Tsutomu Kanai Award (2009), and the Distinguished Alumni Award in Computer Science of the University of California, Berkeley (2011).