ABSTRACT
Player’s physical experience is a critical factor to consider in designing motion-based games that are played through motion sensor gaming consoles or virtual reality devices. However, adjusting the physical challenge involved in a motion-based game is difficult and tedious, as it is typically done manually by level designers on a trial-and-error basis. In this paper, we propose a novel approach for automatically synthesizing levels for motion-based games that can achieve desired physical movement goals. By formulating the level design problem as a trans-dimensional optimization problem which is solved by a reversible-jump Markov chain Monte Carlo technique, we show that our approach can automatically synthesize a variety of game levels, each carrying the desired physical movement properties. To demonstrate the generality of our approach, we synthesize game levels for two different types of motion-based games and conduct a user study to validate the effectiveness of our approach.

CCS CONCEPTS
• Human-centered computing → User centered design;

KEYWORDS
Level design; optimization; exergaming; generative design

1 INTRODUCTION
Motion-based games, also called exercise games, are a genre of video games that emphasize human-computer interaction through body motion control. With the widespread popularity of household human-computer interaction devices such as depth sensors (e.g., Microsoft Kinect), motion controllers (e.g., Wii Remote) and virtual reality devices (e.g., HTC Vive), many motion-based games are developed, leading to substantial research interests in exploring their applications for improving human fitness.

While the potential of motion-based games for improving fitness is appealing, designing game levels for motion-based games is difficult and tedious. The major difficulty lies in striking the right balance to design an exciting yet not physically overwhelming game level. Unlike traditional games played with a game controller, motion-based games are played by users via body movements. A game level that is too physically challenging could easily cause fatigue on players and prompt them to quit the game, while a game level that is too static may bore the players.

To achieve a good balance, in current practice game level designers often use a trial-and-error approach to manually adjust physical difficulty of game levels [13, 36], largely based on their experience. This routine design process is labor and time intensive.

 Inspired by research on procedural content generation for exergames [56] and physical rehabilitation [12], we propose an optimization-based approach for exergame level design. As depicted in Figure 1, our approach is capable of automatically synthesizing game levels for motion-guided game for achieving desirable physical movement effects specified by a level designer. By formulating the design problem as an optimization problem, a variety of levels can be quickly and automatically synthesized which balance different design considerations. The synthesized levels can be used by level designers as a basis for further refinement.

Body flexibility and balance are important metrics of physical fitness, which researchers attempted to improve through exergaming [7, 52]. We incorporate these metrics into our
approach, by considering the rotation of joints and the movement of the center-of-mass of a player in completing a level synthesized by our approach. As such considerations are explicitly quantified as cost terms in our approach, level designers can easily estimate the physical difficulty posed to the player in completing each synthesized level. The major contributions of this paper include:

- Devising a novel optimization-based approach for synthesizing levels with joint rotation and center-of-mass movement considerations, which can serve as a suggestion engine for designing the content of different pose-based applications.
- Validating the effectiveness of our approach for synthesizing pose-guided games levels via an user evaluation.

2 RELATED WORK

Motion-based Game Design

One of the most important factors in designing motion-based games is physical challenge [40]. Recent research has investigated the relationship between gaming experience satisfaction and game difficulty. Sinclair et al. [48] concluded that the success of exergaming is associated with three factors: player reflexes, gaming experience and player’s physical conditions. Bianchi-Berthouze [4] stated from the ergonomic standpoint that the level of difficulty should be tailored to a player’s fitness and coordination skills. Current practice for designing a motion-based game level is non-trivial and time consuming. Designing an appropriate game level largely depends on the level designers’ game design experience, knowledge of human physical conditions and manual construction. Level designers typically need to go through a tedious trial-and-error process to validate the appropriateness of their design [20]. Mueller et al. [21, 35] summarized the general design guidelines of exertion games. By incorporating physical movement factors as cost terms, our approach allows level designers to automatically analyze such factors and synthesize optimized levels accordingly.

Recent advancement and popularity of virtual and augmented reality devices have created substantial demand for motion-based games and applications. For example, the HTC Vive allows a user to play sports (e.g., Virtual Sports) with his full body in a highly immersive virtual environment. Refer to Gradl et al. [14] for a recent review of virtual reality-based exergames. When it comes to designing a full-body virtual reality experience, motion considerations are especially important as the user relies on his body movement to proceed with the virtual experience; fatigue and frustration can quickly build up if the design involves too much motion, which may prompt the user to quit the experience. Our work facilitates the design of motion-based virtual experiences by automatically optimizing such experiences with respect to the extent of physical movement involved.

Exergaming Research

Exergames allow people to exercise at home while playing games with motion-sensing devices such as Microsoft Kinect and Wii Remote. Research [30] has been conducted on popular commercial exergames such as Just Dance and Wii Sports [11], which validated the positive health effects brought by exergames. Exergames are used for sports training [8, 23], breathing training [41], as well as rehabilitation and therapy purposes such as balance enhancement, weight control and cognitive-motor training. Schoene et al. [46] and Ogawa et al., [38] found that exergaming can potentially improve cognitive functions and dual-task functions. Bohm-Morawitz et al. [3] and Staiano et al. [50] investigated the use of exergames by adolescents and adults to achieve weight loss. On the other hand, Kim et al. [25] proposed the Vizical technique for predicting energy expenditure during exergaming. Padala et al. [39] and Wüest et al. [55] found that exergaming is effective for improving balance and movement related physical performance of elders. In sum, exergaming has been successfully employed for motivating players to do exercise and improve their health conditions, especially for body flexibility and self-balance.

In exercise science, stretching before doing exercises is a practice to enhance performance and reduce the risk of injury [53]. Different exercises require different stretching poses. In our approach, we use joint rotation to evaluate the stretching required in transitioning from one pose to another pose; and we use the center-of-mass movement to evaluate the difficulty of balance control [18]. These metrics are commonly used in exercise science and physiotherapy research [5, 17, 31, 44, 54].

A challenge in designing exergames is on quantifying and evaluating the difficulty of an exergame level which involves body movement. In HCI research, Fitts’s law [32] considers the movement distance and precision in the index of difficulty [33] for a pointing task. Recently, Lee et al. [28, 29] found that duration constraints also impacts the difficulty of a pointing task. Inspired by these findings, we devise our level design framework to also consider the movement distance (in terms of the extent of joint rotations and center-of-mass movement) and duration (Section 6) in evaluating a level. As for the movement precision, to make the synthesized game enjoyable to play, we allow different error tolerances for matching different joints as determined from trial experiments, akin to the settings of popular motion-based games (e.g., Just Dance).

Procedural Level Design

Procedural techniques can be applied to automate the level design of platform games [9]. For example, Smith et al. [49] proposed a rhythm-based approach for automatically designing levels for 2D games. Similarly, rule-based [19] and learning-based [22, 47] approaches have been applied for
We implemented our game to run with a Microsoft Kinect V2 sensor which keeps track of the player’s movement. Yeh et al. [42, 57] considered calories burned as a metric for level design. In contrast, ours considered "emotion" during the gameplay [51]. There are recent interesting efforts by HCI researchers in understanding balancing in exergames [1], such as for a digital table tennis game [2]. Such understanding can facilitate dynamic difficult adjustment of games [10]. Along a similar direction, Dimovska et al. [12] created a real-time skiing game which makes use of the player’s performance for synthesizing the next section of a game level. Similarly, Xie et al. [56] considered calories burned as a metric for level design. Both works have demonstrated that player performance can be tracked and applied for level design. On the other hand, Yeh et al. [58] applied the Metropolis-Hastings algorithm to generate game scenes, where static scene items were placed according to the desired difficulty level. In contrast, ours focuses on the level design of motion-based games, mainly considering the player’s movement during gameplay. Our synthesized levels were validated by extensive evaluation of the player’s body movement.

3 OVERVIEW

Figure 2 shows an overview of our approach. From a pool of different types of poses, our approach assembles a game level by running an optimization. In each iteration, the optimizer evaluates the assembled level with respect to the physical movement goals and other design factors, and updates the level by a move. The synthesis process completes as the synthesized level attains the design goals.

**Just Exercise:** To illustrate our approach, we created a motion-based game called *Just Exercise* as an example to demonstrate and experiment with our approach. The design of *Just Exercise* mimics that of a game called *Just Dance* available on Nintendo Switch, Wii U, PlayStation4 and Xbox One. The logic of our game mimics that of the original game. We implemented our game to run with a Microsoft Kinect V2 sensor which keeps track of the player’s movement.

**Game Logic:** A game level of the *Just Exercise* game consists of a sequence of exercise poses. An exercise pose belongs to one of the pose types depicted in Figure 3. We use these poses as they are intuitive for our user study participants to learn, and they cover different extent of joint rotations and center-of-mass movement. They can also be tracked relatively accurately by the Kinect as a participant’s joints do not occlude each other when doing these poses.

During the game, a humanoid model is shown at the center of the screen. As the game starts, the player is asked to follow the pose of the model to move his body accordingly. Different poses requires different player efforts to cope with. For ease of playing, a pose is considered to be completed if each joint angle of the player is within a certain error tolerance of the corresponding joint angle of the target pose. Refer to Section 9 for details of these error tolerance settings in. To motivate the player to follow the poses closely, we display the average angle matching score of all joints on the screen. Depending on the poses used to assemble a level, completing a level requires a different amount of physical movement. The supplementary video shows a gameplay demo.

4 PROBLEM FORMULATION

The goal of our approach is to synthesize levels optimized with respect to a desired extent of joints rotation and center-of-mass movement, as well as other design factors, which are encoded as cost terms.

Let \( l = (p_1, p_2, \ldots, p_n) \) denote a level, which consists of a number of poses \( p_i \in P \) assembled in a sequential order, where \( P \) is the set of all pose types. For example, the game *Just Exercise* has a total of 10 pose types as shown in Figure 3.

The human body, tracked by a Kinect sensor in *Just Exercise*, is represented by 17 joints. We exclude the joints of the neck, hands and feet tracked by Kinect as these joints are insignificant for our purposes. Let \( J = \{j_i\} \) be the set of all joints. Each joint \( j_i = (x_i, \theta_i) \) is represented by a position \( x_i \) and a rotation angle \( \theta_i \). The quality of a level \( l \) is evaluated by a total cost function \( C_{Total}(l) \):

\[
C_{Total}(l) = C_M w_M^T + C_P w_P^T,
\]

where \( C_M = [C_M^R, C_M^{CM}] \) is a vector of movement costs and \( w_M = [w_M^R, w_M^{CM}] \) is a vector of weights corresponding to the costs. \( C_M^R \) and \( C_M^{CM} \) evaluate the movements involved when following the poses in a level: the angle that each joint
We define two costs to evaluate the movements involved in completing a level \( l \).

**Joint Rotation Cost:** The extent of joint rotation is commonly used as a metric for evaluating body flexibility in exercise science and physiotherapy research [5, 17]. As we want our synthesized level to consider body flexibility also, we precompute the movements involved when transitioning between every pose types. Specifically, for a transition from pose \( i \) to pose \( j \), we compute the joint rotation \( R_{i,j,k} \) of each joint \( k \), and the center-of-mass movement \( M_{i,j} \). Figure 5 visualizes example data precomputed for transitioning between several poses. The supplementary material contains full visualization for every pair of poses.

For a transition from pose \( i \) to pose \( j \), we compute the minimum angle rotation of different joints to achieve the transition. Figure 4 shows an illustration. For each joint \( k \), \( R_{i,j,k} \) stores its rotation computed as the absolute angle that the joint has rotated in transitioning from pose \( i \) to pose \( j \). Similarly, the center-of-mass movement from pose \( i \) to pose \( j \) is stored in \( M_{i,j} \). We approximate the center-of-mass \( c = \frac{1}{|J|} \sum_j \kappa_j x_j \) by the average positions of all joints, weighted by the approximate mass \( \kappa_i \) of each joint \( j \) [43].

**Movement Costs**

We define two costs to evaluate the movements involved in completing a level \( l \).

**Joint Rotation Cost:** We approximate the center-of-mass of the body has shifted. \( C_P = [C_P^R, C_P^C] \) is a vector of game-specific prior costs encoding design priors such as the duration of the level and the variation between adjacent poses, and \( w_P = [w_P^R, w_P^C] \) stores the weights of these costs. Section 6 includes further details.

**Center-of-Mass Movement Cost:** Physiotherapy and biomechanics researchers commonly use the center-of-mass movement sequence as a metric for evaluating body flexibility in exercise science and physiotherapy research [17, 39]. Accordingly, we define a cost to measure self-balancing difficulty based on the extent of center-of-mass movement involved in the level:

\[
C_{CM}^M(l) = 1 - \exp\left(-\frac{\left(\sum_{p,q} M_{p,q} - \rho_{CM}^l\right)^2}{2\sigma_{CM}^2}\right),
\]

where \( \rho_{CM}^l \) is the target sum of center-of-mass movement involved in completing level \( l \). \( \sigma_{CM} \) is set as \( \rho_{CM}^l \).

**Prior Costs**

Prior costs are employed to encode some game-specific level design considerations. Different types of games have their own constraints for assembling a preferable level. In our approach, we use the duration cost to control the length of the gameplay. Also, we define the variation cost to introduce changes to the gameplay experience to discourage the synthesis of monotonous game levels which could be boring.

**Duration Cost:**

\[
C_{D}^P(l) = 1 - \exp\left(-\frac{\left(\sum_{p,q} D(p,q) - \rho_d^l\right)^2}{2\sigma_d^2}\right),
\]

where \( p, q \in l \) refer to a pair of adjacent poses. \( D(p,q) \) is the duration of transitioning from pose \( p \) to pose \( q \). To measure
We employ the reversible-jump Markov chain Monte Carlo (RJMCMC) where the acceptance probability depends on the specific design needs of a game. For an "Add a Pose" move, we include a variation cost to penalize forming a level where the types of a pair of adjacent poses are the same:

$$C_p^V(l) = \frac{1}{|l|-1} \sum_{(p,q)} \Gamma(p, q), \quad (5)$$

where $p$ and $q$ are adjacent poses. $\Gamma(p, q)$ returns 1 if $p$ and $q$ are of the same pose type; it returns 0 otherwise.

Other prior costs can be added to the optimization framework depending on the specific design needs of a game. For example, in synthesizing the levels of a dancing game, a tempo cost which evaluates how well the dancing poses follow the rhythm of the background music can be added. Due to the scope of this paper, we keep our cost definitions simple, focusing on the body flexibility and self-balancing aspects that we want to investigate.

### 7 OPTIMIZATION

Our goal is to synthesize a level assembled by a sequence of poses, optimized with respect to the target costs. As a level can be assembled by an arbitrary number of poses, the solution is searched in a trans-dimensional solution space. We employ the reversible-jump Markov chain Monte Carlo (RJMCMC) method [16] to search for a solution which can cope with changing dimensionality. The method is applied with a Metropolis-Hastings state searching step [6]. First, we define a Boltzmann-like objective function:

$$f(l) = \exp\left(-\frac{1}{t}C_{\text{Total}}(l)\right), \quad (6)$$

where $t$ is the temperature parameter of simulated annealing [26], which decreases gradually throughout the optimization. At each iteration of the optimization, our approach applies a move to the current level $l$ to create a proposed level $l'$. There are three types of moves that can be selected by the optimizer:

- **Add a Pose**: a random pose is selected and added to a random location of the current level $l$ to create a proposed level $l'$;
- **Remove a Pose**: a pose in the current level $l$ is randomly selected and removed to create a proposed level $l'$;
- **Modify a Pose**: a pose in the current level $l$ is randomly selected and changed to another randomly-selected pose, to create a proposed level $l'$.

The selection probabilities of the add, remove and modify moves are $p_a$, $p_r$ and $p_m$. By default, we use $p_a = 0.4$, $p_r = 0.2$ and $p_m = 0.4$, to slightly favor adding and modifying a pose.

To decide whether to accept the proposed level $l'$, our approach compares the total cost value $C_{\text{Total}}(l')$ of the proposed level $l'$ with the total cost value $C_{\text{Total}}(l)$ of the original level $l$. To maintain the detailed balance condition of the RJMCMC method, the acceptance probability $Pr(l'|l)$ is set according to the move type, as follows.

For an **Add a Pose** move,

$$Pr(l'|l) = \min(1, \frac{p_a \eta|\mathcal{P}| - |l|}{p_a \eta|\mathcal{P}| - |l'|} f(l') \bigg/ f(l)), \quad (7)$$

For a **Remove a Pose** move,

$$Pr(l'|l) = \min(1, \frac{p_r \eta|\mathcal{P}| - |l'|}{p_r \eta|\mathcal{P}| - |l|} f(l') \bigg/ f(l)), \quad (8)$$

For a **Modify a Pose** move,

$$Pr(l'|l) = \min(1, \frac{f(l')}{f(l)}). \quad (9)$$

To simplify our formulation, we assume that each pose type can only be selected $\eta$ times rather than an infinite number of times, so that the dimensionality of the solution space has an upper limit. In other words, a level can be assembled by up to $\eta|\mathcal{P}|$ poses. We set $\eta = 15$ for each pose type in our experiments.

We use simulated annealing to efficiently explore the solution space containing different level design solutions. Simulated annealing is controlled by the temperature parameter $t$. At the beginning of the optimization, the temperature $t$ is set to be high to prompt the optimizer to aggressively explore possible solutions. The temperature drops over iterations until it reaches a very low value near zero. We empirically use temperature $t = 1.0$ at the beginning of the optimization and decrease it by 0.1 every 500 iterations until it reaches zero. Such setting essentially makes the optimizer more greedy in refining the solution towards the end of the optimization. Our approach terminates the optimization if the total cost change is smaller than 3% over the past 50 iterations.

#### Parameter Settings:

By default, we set the weights of the movement costs as $w^R_M = 1.0$ and $w^CM = 1.0$; the weights of the prior costs as $w^R_p = 1.0$ and $w^CM_p = 0.5$; and the importance value of each joint $k$ for computing the rotation cost as $\lambda^R_k = 1.0$. Figure 6(a) shows a level synthesized with the default parameters. The designer can control these weights and importance values to synthesize different types of levels, which we illustrate in our experiments.

#### Target Settings:

The target sum of rotation $\rho^R_k$ for each joint $k$ and the target sum of center-of-mass movement $\rho^CM$ can be non-trivial to specify without a reference. To allow setting these values intuitively, we manually created several reference levels with different rotation and center-of-mass movement difficulties, and computed the sum of rotation $\rho^R_k$ of each joint $k$ and sum of center-of-mass movement $\rho^CM$ involved in each level, which are taken as reference values.
that a level designer can modify to synthesize a level with desired extents of movement difficulties. The target duration $\rho_d$ of the level is set as the number of seconds that the level should span. By default, we use 60 seconds for a level.

8 EXPERIMENTS AND RESULTS

**Implementation:** We conducted experiments to test our approach on an Alienware PC equipped with an Intel Core i7-5820K CPU and 32GB of memory. The optimization framework was implemented in C# as a plugin for the Unity game engine. The example games were implemented in Unity using the Kinect SDK. We applied our approach to synthesize levels for our illustrative game, *Just Exercise*. We conducted a user evaluation test to validate the synthesized levels.

To demonstrate the general applicability of our approach, we also applied our approach to synthesize pose-guided levels for a classic arcade game called *Speed of Light*, which we describe in our supplementary material.

**Different Movement Goals:** Our approach is capable of generating levels that emphasize different body movements
We conducted a trial experiment with 10 participants to gain early insights about our user evaluation design. They were asked to play the 5 synthesized levels shown in Figure 6. As a result, we set the joint angle error tolerance slightly higher than the mean error for each joint, as follows: 15 degrees for the elbow joints; 10 degrees for the hips and knees joints; and 5 degrees for the spine base joint, based on the participants’ feedback on the matching difficulty and their performances. Refer to the supplementary document for more details about this trial experiment.

Evaluation Experiments

To conduct our user evaluation, we used the game *Just Exercise* and the 5 levels synthesized with default parameters, medium rotation, high rotation, as well as with medium and high center-of-mass movement. Figure 6 shows the levels used. The main goal is to evaluate how well the participants followed the joint rotation and center-of-mass movement targets specified for synthesizing the levels.

Participants: 30 participants were recruited to play the synthesized levels, which were different people from trial experiments. They were university students and staffs, whose average age was $27 \pm 10$ years old and average body mass index (BMI) was $23 \pm 5$kg/m$^2$. Our supplementary material contains more demographic information.

Procedure: Our evaluation procedure was IRB-approved. The participant was briefed about the game control and given a warm-up session to get familiar with the game. Then we asked the participant to play the 5 levels in a randomized order. The participant was asked to match the poses of the levels shown on screen. A level was completed if all of its sequence of poses had been matched. The participant had a 3-minute break after playing each level. In the end, the participant filled out an enjoyment rating questionnaire.

Measurements: We used a Kinect sensor to capture the participants’ full body motion during the experiments for analysis. Before the evaluation, we calibrated the Kinect sensor with respect to the participant. The captured body motion data includes the participant’s joint positions and rotations at every frame during the gameplay. This motion capture mechanism is non-intrusive and allows the participants to move freely and comfortably, like playing an ordinary motion-based exergame on a home game console.

Analysis Methods: We examined the descriptive statistics of joint rotation and center-of-mass movement of participants in completing different levels. We used the Mauchly’s test statistic to test the assumption of sphericity. One-way repeated measure analysis of variance (ANOVA) was used to compare the body movement results among the three levels synthesized with default, medium and high joint rotation targets, and among the three levels synthesized with default, medium and high center-of-mass movement targets. Paired $t$-test was used to compare the mean difference in each pair of levels synthesized with different targets (e.g., default and medium, default and high, medium and high), and the effect sizes were analyzed. Note that when we evaluated joint rotation, we excluded the shoulder joints due to a Kinect’s
tracking issue we experienced, which we explained in the supplementary material with an experiment.

In addition, we used the physical activity enjoyment questionnaire (PACES) [24] to evaluate the enjoyment and vitality of physical activity perceived by the participants.

Results and Discussion

Joint Rotation: Figure 7(a) shows the average total joint rotations compared with the joint rotation targets used for synthesizing each level. D, M and H refer to the levels with default, medium and high joint rotation targets. The result and target for each joint is shown. (b) Participants’ average total amount of center-of-mass movement compared with the center-of-mass movement targets used for synthesizing each level. Overall, the participants’ results follow similar upward trends as the targets.

We performed paired t-tests to examine where the significant effect lies for each pair of levels (e.g., Default vs. Medium Rotation). Table 2 shows the descriptive statistics. Except for the spine-base and right hip joints in the medium vs. high rotation levels comparison, all joints show a significant increase ($p < 0.05$) in joint rotation results as the targets increase. The effect sizes ($\eta^2$) in default vs. high rotation levels is greater than those in default vs. medium rotation levels. The average percent increase per chunk from default to medium rotation level is 12% and from default to high rotation level is 47%, which are relative close to the corresponding increases in the targets which are (23%) and (64%).

From the results, we observe that the joint rotation results of the participants increase with the joint rotation targets used for synthesizing the levels in general.

Center-of-Mass Movement: Figure 7(b) shows the center-of-mass (COM) movement targets specified for synthesizing the levels with default, medium and high COM movements. It also shows the COM movements attained by the participants, which increase with the targets accordingly.

Similarly, we conducted the one-way repeated measure ANOVA test on the COM movement results of the three levels with default, medium and high COM movement targets. Mauchly’s test, $X^2 = 0.762, p = 0.683$ did not indicate any violation of sphericity. Table 3 shows that there was a significant difference among the results of the three levels ($F(2, 58) = 73.074, p < 0.0001$).

Furthermore, paired t-test and other descriptive statistical results were shown in Table 4, which indicated that there was a significant difference between the result of the default and medium levels, and between the default and high levels. However, the difference in results between the medium and high levels is not statistically significant ($p = 0.060$).

On the other hand, the average percentage increase in COM movement results from the default to medium level is 38.4% and from the default to high level is 45.7%, which are higher than the respective percentage increases in COM movement targets from the default to medium level (18%) and from the default to high level (36%).
Table 2: Paired t-test results for joint rotations. The test was done using the participants’ joint rotation results in levels with default, medium and high joint rotation targets. Most joint rotation results show a significant difference (p<0.05, bolded) between levels.

<table>
<thead>
<tr>
<th></th>
<th>Spine-base</th>
<th>L. Elbow</th>
<th>R. Elbow</th>
<th>L. Hip</th>
<th>L. Knee</th>
<th>R. Hip</th>
<th>R. Knee</th>
</tr>
</thead>
<tbody>
<tr>
<td>two-tail p-value</td>
<td>0.037</td>
<td>0.001</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>0.004</td>
<td>0.003</td>
<td>0.018</td>
</tr>
<tr>
<td>eta squared</td>
<td>0.414</td>
<td>0.773</td>
<td>2.974</td>
<td>0.860</td>
<td>0.439</td>
<td>0.568</td>
<td>0.369</td>
</tr>
<tr>
<td>mean increase %</td>
<td>11%</td>
<td>9%</td>
<td>28%</td>
<td>13%</td>
<td>7%</td>
<td>9%</td>
<td>6%</td>
</tr>
</tbody>
</table>

Table 3: One-way repeated measure ANOVA results. The test was done on the participants’ results among the 3 levels with default, medium and high joint rotation targets; and among the 3 levels with default, medium and high center-of-mass (COM) movements. All joint rotation and COM movement results show a significant difference (p<0.05) among the levels.

<table>
<thead>
<tr>
<th></th>
<th>Spine-base</th>
<th>R. Elbow</th>
<th>L. Elbow</th>
<th>R. Hip</th>
<th>L. Hip</th>
<th>R. Knee</th>
<th>L. Knee</th>
<th>R. Knee</th>
<th>L. Knee</th>
</tr>
</thead>
<tbody>
<tr>
<td>two-tail p-value</td>
<td>0.285</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>0.220</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>eta-squared</td>
<td>N/A</td>
<td>6.62</td>
<td>4.27</td>
<td>N/A</td>
<td>1.15</td>
<td>2.14</td>
<td>2.26</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>mean increase %</td>
<td>6.87%</td>
<td>74.10%</td>
<td>33.88%</td>
<td>3.23%</td>
<td>20.34%</td>
<td>50.24%</td>
<td>53.93%</td>
<td>N/A</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 4: Paired t-test results for center-of-mass movement. The test was done using the participants’ COM movement results in levels with default (D), medium (M) and high (H) COM movement targets. Significant difference (p<0.05) was found between default and medium levels, and between default and high levels.

<table>
<thead>
<tr>
<th></th>
<th>D-M</th>
<th>D-H</th>
<th>M-H</th>
</tr>
</thead>
<tbody>
<tr>
<td>two-tail p-value</td>
<td>&lt;0.001</td>
<td>&lt;0.001</td>
<td>0.060</td>
</tr>
<tr>
<td>eta squared</td>
<td>1.876</td>
<td>1.96</td>
<td>N/A</td>
</tr>
<tr>
<td>mean increase %</td>
<td>38.4%</td>
<td>45.7%</td>
<td>N/A</td>
</tr>
</tbody>
</table>

Table 5: Physical enjoyment rating results. Scores range from 1 (strongly disagree) to 7 (strongly agree).

<table>
<thead>
<tr>
<th>Question</th>
<th>Mean</th>
<th>S.D.</th>
</tr>
</thead>
<tbody>
<tr>
<td>I enjoy it</td>
<td>5.8</td>
<td>1.4</td>
</tr>
<tr>
<td>I like it</td>
<td>5.4</td>
<td>1.6</td>
</tr>
<tr>
<td>I feel good physically</td>
<td>5.8</td>
<td>1.3</td>
</tr>
<tr>
<td>It’s a lot of fun</td>
<td>5.4</td>
<td>1.3</td>
</tr>
<tr>
<td>I am not at all frustrated</td>
<td>5.4</td>
<td>1.6</td>
</tr>
</tbody>
</table>

Physical Activity Enjoyment Rating

Physical activity enjoyment scale questionnaire (PACES) is frequently used in exercise science as a quantitative measure of perceived enjoyment level for an exercise activity. It consists of 18 7-point Likert scale questions validated by Kendzierski and DeCarlo [24] on young adults for evaluating enjoyment. Table 5 showed some of the results for Just Exercise rated by our user evaluation participants. We include full results in the supplementary material. Overall, participants rated about 5.6 out of 7 for how much they enjoyed the game, and our average PACES percentage score was 79%.

Grave et al. used the average PACES percentage scores to compare the enjoyment of exergames on Wii Fit with aerobic exercises [15]. Comparing with their results, the PACES percentage score of Just Exercise (79%) is higher than that of Wii Yoga (67%) and Wii Muscle (74%), but lower than that of Wii Balance (80%) and Wii Aerobic (85%). It is also higher than that of regular exercises such as brisk treadmill walking (69%) and treadmill jogging (77%). We note that the comparison may only be taken as a general reference due to different groups of subjects. We believe the enjoyment of Just Exercise is comparable to common exergames.

User Feedback

Participants gave us additional feedback after the evaluation. Most commented that the game was entertaining and motivating for exercising. Some thought that the intensity of our difficult levels was comparable to that of a regular work-out.
session if they played it for a longer duration. A few participants who gave low enjoyment ratings commented that some poses were demanding and difficult to match in terms of stretching and self-balancing requirements. Some said that this kind of exergames, added with more game elements such as dynamic sound and visual effects, would be their choices for replacing routine work-out exercises.

10 SUMMARY
We demonstrated that our optimization-based level design framework can take into account the player’s the joint rotations and center-of-mass movement, as well as common level design factors such as duration and variation which can be added as prior costs. Other specific game design factors (e.g., tempo consideration for a dancing game) can be incorporated similarly. Our framework can be generally applied to optimize body flexibility and balance requirements of other motion-based games. In the supplementary material, we detail how it can be applied to synthesize levels for Speed of Light, a classic arcade game. We also show how designers can generate a variety of levels with specific needs by adjusting the weights and constraints of our optimization framework. Besides, our approach can be applied for synthesizing levels for motion-based games (e.g., Reflex Ridge of Kinect Adventures! and Climbeby of Steam VR) where the required player actions (e.g., climb, squat, jump, dodge) are optimized against the joint rotation and center-of-mass movement targets.

Mueller et al. [34, 37] envisioned that future bodily games will allow players to experience their bodies as digital play, where the players’ emotions, feelings, stimulation and perception will be part of the gameplay. Towards this endeavor, our approach contributes by synthesizing gameplay that takes the player’s physical movements into account in a quantifiable manner.

Limitations and Future Work
We measured poses through a Kinect sensor. To make it easier for tracking, we used poses which are simple to learn and whose joints do not occlude each other from the sensor. In future work, it would be interesting to investigate the possibility of replacing the body tracking mechanism with other devices, such as motion-capture suits (e.g., Rokoko), which allow tracking and using more complex poses like those in daily exercises for creating more varieties of exergames. In our preliminary experiments, we tested with the Enflux motion-capture suit but found that the tracking results were unstable and too noisy for analyzing, so we resorted to using a Kinect sensor for tracking.

In precomputation, we assume that when transitioning from one pose to another pose, players follow the joint movement trajectory that involves the least amount of joint rotation for each joint. In reality, the joint movement trajectories may vary depending on the players’ joint flexibility, self-balancing capability and movement style. It is possible that players make some extra movements during a transition. We estimate the joint rotation with the above assumption for simplicity, considering that players should not do a lot of extra movements during each transition which only lasts for about one to two seconds.

While our approach mainly focuses on body movement factors, there are other factors in game level design that need to be taken into account. Cognitive considerations (e.g., player’s attention control) are not directly incorporated due to the scope of our paper. However, our body movement (joint-rotation, center-of-mass movement) and duration considerations were inspired by Fitts’ Law, which was found to be related to cognitive factors [45]. For future work, we would like to extend our framework to consider cognitive factors associated with different movement difficulties.

Also, the synthesized level may lack aesthetic considerations, our optimization framework allows the designer to incorporate additional considerations (e.g., rhythm) according to the specific needs of a game. We synthesized levels for user evaluation purposes. The levels are short compared to a typical workout which usually lasts for 30 to 45 minutes. In practice, levels should be synthesized with a longer duration and with more types of poses. While we showed in our evaluation that our synthesized levels can guide users to achieve the specified joint rotation and center-of-mass movement goals, it would be helpful to conduct a more long-term and large-scale evaluation to study the possible body movement training effects brought about by practicing with the synthesized levels regularly for a prolonged period.

Previous research [13, 27, 39, 54] has shown that exergames can effectively improve the body flexibility and self-balancing capability of older adults. While our user evaluation was conducted mainly with young adults, the flexibility of our level design framework would allow synthesizing appropriate levels for aged players, for example, by adjusting the joint rotation target $\rho^R$, center-of-mass movement target $\rho^{CM}$ and duration target $\rho_d$; and also by adjusting the importance values $\lambda^R_j$ of each joint to impose different amounts of exercise on different body regions (refer to supplementary material for examples) depending on the player’s body condition. In future work, we would like to investigate the training effects of our synthesized levels on aged populations.

ACKNOWLEDGMENTS
This Research is supported by the Oracle Undergraduate Fellowship, the McNair Program and the National Science Foundation under award number 1565978.
REFERENCES


