Personalized Dual-Level Color Grading for 360-degree Images in Virtual Reality

Lin-Ping Yuan, John J. Dudley, Per Ola Kristensson, Huamin Qu

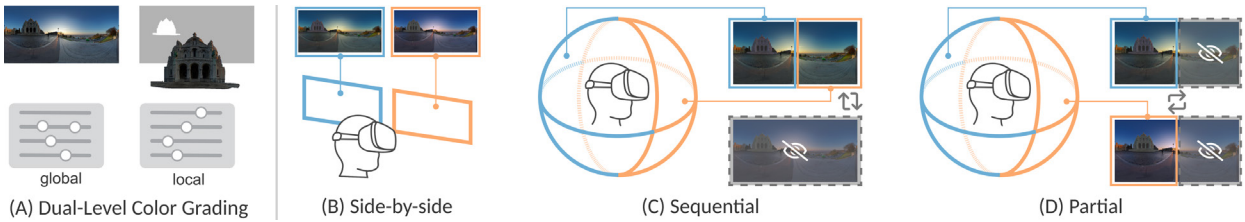

Fig. 1: Illustration of dual-level color grading for 360-degree images in VR. (A) Dual-level grading involves iterative adjustments of global and local color parameters. (B-D) Three VR interfaces are designed to collect creators' preferences for color-graded options at the global and local levels to support personalized color grading. (B) Side-by-side: equirectangular projections of the options are displayed adjacent to each other on a 2D plane. (C) Sequential: only one option is shown at a time, with the ability to switch between different options. (D) Partial: corresponding segments of the two options are displayed simultaneously, with the ability to switch between different segments. Blue and orange indicate different display areas.

Abstract— The rising popularity of 360-degree images and virtual reality (VR) has spurred a growing interest among creators in producing visually appealing content through effective color grading processes. Although existing computational approaches have simplified the global color adjustment for entire images with Preferential Bayesian Optimization (PBO), they neglect local colors for points of interest and are not optimized for the immersive nature of VR. In response, we propose a dual-level PBO framework that integrates global and local color adjustments tailored for VR environments. We design and evaluate a novel context-aware preferential Gaussian Process (GP) to learn contextual preferences for local colors, taking into account the dynamic contexts of previously established global colors. Additionally, recognizing the limitations of desktop-based interfaces for comparing 360-degree images, we design three VR interfaces for color comparison. We conduct a controlled user study to investigate the effectiveness of the three VR interface designs and find that users prefer to be enveloped by one 360-degree image at a time and to compare two rather than four color-graded options.

Index Terms—Preferential Bayesian Optimization, color grading, 360-degree images, virtual reality.

1 Introduction

With the growing access to 360-degree cameras and virtual reality (VR) headsets, more and more creators are engaging in capturing, editing, and sharing 360-degree images in VR. Color grading is a crucial editing process, where creators adjust color parameters (e.g., contrast and white balance) to achieve a desired look or mood, reflecting their personal tastes and preferences. Color grading is important for enhancing visual aesthetics and guiding viewer attention in 360-degree VR experiences [33]. It involves primary grading to alter the global colors across an entire image, and secondary grading to refine the local colors of specific areas (Fig. 1-A). Since the perception of local colors is influenced by their contextual colors, secondary grading needs to consider the established global colors [11].

However, manual color grading is difficult since creators need to experiment in a high-dimensional search space to find parameters that match their preferences [16]. To alleviate these difficulties, several computational frameworks [15, 16, 35] that leverage Preferential Bayesian Optimization (PBO) have been proposed to free creators from this inefficient manual adjustment. These frameworks tackle creators' preferences in a human-in-the-loop manner. Specifically, they iteratively ask creators to compare images with different parameters, learn their latent preferences with a Gaussian process (GP) based on the comparisons, and use efficient sampling strategies to select parameters for the next round of comparison. Although effective, these frameworks only focus on adjusting global colors, neglecting the refinement of local colors that depend on global colors. Moreover, it is necessary to enable creators to perceive and adjust the colors of 360-degree images directly in VR, rather than on a 2D desktop, since human perception of colors differs between VR and desktop environments [14, 40]. However, the user interfaces used in existing frameworks [15, 16, 35] are designed to display general 2D images on desktops in either a side-by-side layout [35] or a grid layout [16], which may not be suitable for effectively comparing 360-degree images in VR.

In line with these frameworks, we aim to facilitate creators with PBO to adjust color parameters globally for an entire 360-degree image and locally for a point of interest (POI) directly in VR environments. This aim introduces two key challenges. First, learning contextual preferences for local colors under dynamic global colors is difficult. As shown in Fig. 2, users' preferences for global colors (red lines) remain constant, while their preferences for local colors (blue and green lines) vary across iterations due to the changing global colors. Second, it is unclear how to design an effective VR interface that facilitates the comparison of 360-degree images at the global and local levels. Specifically, 360-degree images are supposed to fully immerse creators in a spherical space, making it difficult for creators to compare multiple options within the same viewport simultaneously.

- *Lin-Ping Yuan is with the Hong Kong University of Science and Technology. E-mail: yuanlp@cse.ust.hk.*
- *John J. Dudley is with the University of Cambridge. E-mail: jjd50@cam.ac.uk.*
- *Per Ola Kristensson is with the University of Cambridge. E-mail: pok21@cam.ac.uk.*
- *Huamin Qu is with the Hong Kong University of Science and Technology. Email: huamin@cse.ust.hk.*

Received 18 September 2024; revised 13 January 2025; accepted 13 January 2025. Date of publication 12 March 2025; date of current version 31 March 2025. This article has supplementary downloadable material available at https://doi.org/10.1109/TVCG.2025.3549886, provided by the authors Digital Object Identifier no. 10.1109/TVCG.2025.3549886

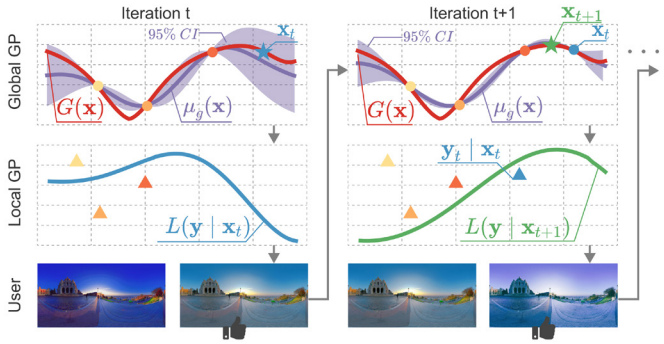

Fig. 2: Illustration of iterative global and local color grading with PBO. The x-axis of the four plots represents 1D color parameter values, and the y-axis represents preference values. In each iteration, the latent global preference $\mathbf{G}(\mathbf{x})$ remains the same (red lines), and the queried points (circles) are used to obtain predicted global preference $\mu_g(\mathbf{x})$ (purple lines and areas). However, local preferences $\mathbf{L}(\cdot)$ vary across iterations (blue and green lines) due to the changing global contexts, and the queried local points (triangles) are associated with previous contexts.

This work proposes a dual-level PBO-based framework (Fig. 3) that integrates global and local parameter adjustments. Our framework uses a classical preferential GP [8] to learn preferences for global colors. To address the first challenge of learning local preferences, we design a novel context-aware preferential GP (Sec. 4) by integrating a multi-task GP [10] into a classical preferential GP. Specifically, to enable the integration of a multi-task GP, we take global colors as contexts or tasks, utilize clustering algorithms to discretize continuous global contexts, and incorporate regression algorithms to obtain latent preference values for unseen contexts to enable the prediction (Fig. 4). Computational experiments demonstrate the effectiveness of the context-aware preferential GP (Sec. 4.4). To address the second challenge, we first design three types (i.e., *side-by-side*, *sequential*, and *partial*) of VR comparison user interfaces (Fig. 1). Then, we conduct a controlled user study to evaluate the three user interfaces with two or four options with different color parameters. Based on the results, we summarize the advantages and disadvantages of the three types of VR interfaces (Sec. 5.2.5) and find that users most preferred the *sequential* interface with two color-graded options.

Our contributions include: (1) a dual-level PBO-based framework integrating local and global color adjustments for personalized color grading in VR; (2) a novel context-aware preferential GP that can learn contextual preferences under dynamic contexts; and (3) an empirical evaluation of comparison interfaces for performing 360-degree image color grading in VR.

2 Related Work

Our research builds upon literature on VR color grading, Preferential Bayesian Optimization, and 360-degree content editing interfaces.

2.1 VR Color Grading

Existing research on VR color grading [9, 22] mainly focuses on the unique characteristics that VR devices offer. For example, a recent study [41] enhances the color contrast in images by utilizing the disparity between the left and right eye views brought by the stereoscopic displays of VR headsets. Despite these advancements, facilitating users to perform active color grading based on their individual needs receives little attention. This oversight is significant given that colors in VR content not only play an important role in conveying aesthetics and mood, but also serve as effective cues to guide viewer attention [19, 26, 42]. In this work, we not only consider the immersive characteristics of VR by investigating the design of VR comparison interfaces, but also fill the gap by exploring personalized color grading.

2.2 Preferential Bayesian Optimization for Color Grading

Previous studies have formulated color grading [16, 35] and other design tasks [4, 28, 36] under human preferences as an optimization problem and demonstrated that PBO is a well-suited algorithm for several reasons. First, the objective functions to present users' preferences are black boxes, with their evaluations considered costly as they rely on subjective user inputs. PBO facilitates finding satisfactory solutions with relatively few iterations of queries to users. Secondly, when querying users, PBO offers a simpler way of gathering feedback. Unlike standard BO, which requires users to express their preferences with absolute scores (e.g., "I rate colors in Image A as 6.5 out of 10"), PBO allows users to provide relative preferences by comparison (e.g., "Image A has better colors than Image B").

There are many variations of PBO-based frameworks for color grading. For example, Koyama et al. proposed novel search techniques by constructing proper lines [17] and planes [16] to reduce optimization iterations. Chui et al. [7] further developed a differential subspace search technique for sampling from high-dimensional latent spaces of generative models. Besides improving the computational performance, Yamamoto et al. [35] offered users more freedom to express their preferences by painting in specific areas to guide the parameter search. However, these PBO-based frameworks focus on optimizing a single and constant latent objective function, and thus cannot tackle our problems that involve two levels of objective functions and the local functions are conditioned on the global preference function.

2.3 Interfaces for 360-degree Content Editing

There are several non-immersive and immersive systems for editing 360-degree content. Non-immersive systems, including commercial software such as Adobe Premiere and research prototypes [5, 18, 29, 32], provide functions such as viewing [18, 29], extracting fields of view [32], and adding audio descriptions [5] with desktop-based user interfaces. Although useful, these systems require users to switch between desktop and VR headsets to examine the editing results. To alleviate this tedious switching and provide benefits of *"what you see is what you get"*, several immersive editing tools [12, 24, 25, 37] have been proposed. For example, Hartmann et al. [12] designed a VR authoring interface to allow users to apply view-dependent effects to 360-degree content directly in VR. However, these immersive interfaces lack support for comparing multiple editing results, which is essential for users to determine the most suitable version of their edits. To fill this gap, we investigate the effective VR user interface design for comparing 360-degree images with different colors.

3 Problem Formulation and Overview

In the following, we begin by introducing our problem formulation (Sec. 3.1). Next, we provide an overview of the proposed dual-level PBO-based framework (Sec. 3.2). We then describe the preference learning process for global colors using a classical preferential GP (Sec. 3.3). This serves as preliminary knowledge to understand how GP is used for preference learning. With the preliminary knowledge, we describe our novel context-aware preferential GP for contextual preference learning for local colors (Sec. 4). Finally, we describe three VR interfaces that allow creators to compare colors (Sec. 5).

3.1 Problem Formulation

This work aims to facilitate creators to adjust color parameters (e.g., color balance and contrast) globally for an entire image and locally for individual POIs. We seek to identify the global and local parameters that creators find most preferable. Importantly, the preference for local colors is influenced by global colors [1], which serve as *contexts*, necessitating consideration of this dependency in our approach.

Mathematically, suppose a creator seeks to adjust Q types of color parameters across an entire image and for a specific POI. We define a color parameter space $\Omega = [a,b]^Q$ where global and local parameters reside. Let $\mathbf{x}$ be the global parameters and $\mathbf{y}$ be the local parameters for the POI. Thus, we have $\mathbf{x} = [x_1, x_2, \cdots, x_Q]^T \in \Omega$, and $\mathbf{y} = [y_1, y_2, \cdots, y_Q]^T \in \Omega$. The color parameters are applied to the original images by modifying the RGB values on a pixel basis using specific formulas [16, 23]. For example, let I_1 represent the original image and I_2 the color-graded image. To adjust global brightness using x_{bri}, we multiply the pixel values by a scalar: $I_2 = x_{bri} \cdot I_1$. In dual-level color grading, we first

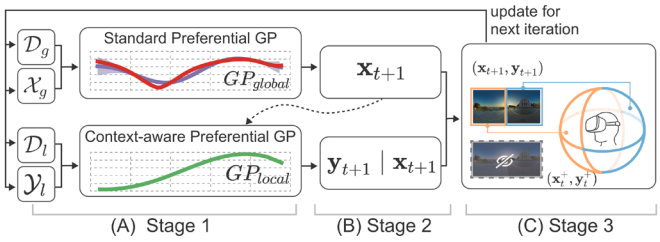

Fig. 3: Proposed dual-level PBO framework with three stages. The local GP takes the output of the global GP as input (the dashed line).

apply x_{bri} to all image pixels for global adjustment, and then apply y_{bri} to pixels within the target POI area for local adjustment.

We formulate personalized dual-level color grading as a dual-level black-box optimization problem. Specifically, the objective is to optimize $\mathbf{x}$ and $\mathbf{y}$ to maximize the aggregate preference across the entire image and the POI:

$$(\mathbf{x}^+, \mathbf{y}^+) = \arg\max_{\mathbf{x},\mathbf{y}\in\Omega} \left(G(\mathbf{x}) + L(\mathbf{y} \mid \mathbf{x})\right) \tag{1}$$

Here, $G : \Omega \to \mathbb{R}$ is the latent preference function for the global parameters that affect the entire image. $L(\mathbf{y} \mid \mathbf{x}) : \Omega \to \mathbb{R}$ is the latent preference function for the local parameters affecting the POI, representing the desirability of $\mathbf{y}$ is conditional on the selected $\mathbf{x}$. $G(\cdot)$ and $L(\cdot)$ map parameter sets to real-valued scores, with higher values indicating better alignment with user-specific preferences.

3.2 Method Overview: a Dual-level PBO Framework

We propose a novel dual-level PBO framework that iteratively optimizes global and local parameters, dynamically conditioning the local parameter selection in response to changes in global parameters each cycle. As shown in Fig. 3, there are three stages in each iteration of the PBO loop.

Stage 1: modeling user preferences with Gaussian processes. PBO utilizes GP as a probabilistic model to capture user preferences. Our PBO framework consists of two GPs: GP_{global} and GP_{local}. We collect two sets of training data with comparisons given by users. With the data, we first train the respective GPs to fit the observed comparisons. Then, the fitted GP is leveraged to predict user preferences at new, unobserved points in the domain Ω. Specifically,

- Global level. GP_{global} is a standard preferential GP [8] to represent the global latent preference function $G(\mathbf{x})$. For GP_{global}, we collect M_g comparisons after t iterations, denoted as $\mathcal{D}_g = \{\mathbf{x}_{i1} \succ \mathbf{x}_{i2} \mid i = 1,\ldots,M_g\}$, where $\mathbf{x}_{i1} \succ \mathbf{x}_{i2}$ indicates a user preference for $\mathbf{x}_{i1}$ over $\mathbf{x}_{i2}$. The M_g comparisons involve N_g queried points $\mathbf{x}_j \in \Omega$, denoted as $\mathcal{X}_g = [\mathbf{x}_j \mid j = 1,\ldots,N_g]$.
- Local Level. For GP_{local}, we propose a context-aware preferential GP to represent the local preference function $L(\mathbf{y} \mid \mathbf{x})$. GP_{local} extends a classical preferential GP with a multi-task GP, explicitly designed to incorporate global colors as conditioning contexts for learning contextual preference. Similarly, for GP_{local}, we gather M_l comparisons, denoted as $\mathcal{D}_l = \{(\mathbf{y}_{i1} \mid \mathbf{x}_{i1}) \succ (\mathbf{y}_{i2} \mid \mathbf{x}_{i2}) \mid i = 1,\ldots,M_l\}$, and N_l queried points, denoted as $\mathcal{Y}_l = [(\mathbf{y}_j \mid \mathbf{x}_j) \mid j = 1,\ldots,N_l]$.

Stage 2: identifying informative query points. The second stage involves selecting the most informative points that will be queried in the current iteration. Points are considered most informative when they offer the greatest potential to help find the global optimum. This selection process is facilitated by an acquisition function $\alpha : \Omega \to \mathbb{R}$, which quantifies the potential of each point based on the fitted GP model's predictions. Following the previous research [35], we select *expected improvement* (EI) as our acquisition functions. Our selection process is also dual-level.

- Global level. We calculate the acquisition function α_{global} for global colors based on the predictions of GP_{global}. We can obtain n points $\mathcal{X}_{t+1} = \{\mathbf{x}_{t+1}^1, \cdots, \mathbf{x}_{t+1}^n\}$ that maximize α_{global}, which will be queried in the current $(t+1)^{th}$ iterations.
- Local level. We select the best set of local parameters for each $\mathbf{x}_{t+1}^i$. To achieve this, we obtain an acquisition function α_{local}^i based on the predictions of GP_{local} and the context $\mathbf{x}_{t+1}^i$, and we find one point $\mathbf{y}_{t+1}^i$ that maximizes α_{local}^i.

In total, we obtain n combinations of global and local parameters, denoted as $\mathcal{Y}_{t+1} = \{(\mathbf{y}_{t+1}^1 \mid \mathbf{x}_{t+1}^1), \cdots, (\mathbf{y}_{t+1}^n \mid \mathbf{x}_{t+1}^n)\}$, with each $\mathbf{y}_{t+1}^i$ the most informative point under the condition of $\mathbf{x}_{t+1}^i$.

Stage 3: engaging users for comparison feedback. In this stage, users need to compare different global and local colors and provide their preference feedback (see Fig. 5). Suppose $\mathbf{x}_t^+$ is the favorite global colors obtained in the last t iterations, and $(\mathbf{y}_t^+ \mid \mathbf{x}_t^+)$ is the favorite local colors under the condition of the favorite global colors.

- Global level. Combining the selected points in Stage 2, we construct options as $C_{global} = \{(\mathbf{y}_t^+, \mathbf{x}_t^+), (\mathbf{y}_{t+1}^1, \mathbf{x}_{t+1}^1), \cdots, (\mathbf{y}_{t+1}^n, \mathbf{x}_{t+1}^n)\}$. We apply these combinations of global and local colors to the target 360-degree image and present the resulting images to users. The users are asked to determine the most preferred option $\mathbf{x}^{chosen}$ regarding global colors. Then, we update the comparison data as $\mathcal{D}_g = \mathcal{D}_g \cup \{\mathbf{x}^{chosen} \succ \mathbf{x}^i\}$, where $\mathbf{x}^i \in \{\mathbf{x}_t^+, \mathbf{x}_{t+1}^1, \cdots, \mathbf{x}_{t+1}^n\} \setminus \mathbf{x}^{chosen}$, and update queried points as $\mathcal{X}_g = \mathcal{X}_g \cup \mathcal{X}_{t+1}$.
- Local level. We update the images by replacing the global parameters with $\mathbf{x}^{chosen}$. In other words, users will compare four options that have the same global colors but different local colors. The local colors are obtained in Stage 2. The options are $C_{local} = \{(\mathbf{y}_t^+ \mid \mathbf{x}^{chosen}), (\mathbf{y}_{t+1}^1 \mid \mathbf{x}^{chosen}), \cdots, (\mathbf{y}_{t+1}^n \mid \mathbf{x}^{chosen})\}$. Suppose the most preferred option regarding local colors is $(\mathbf{y}^{chosen} \mid \mathbf{x}^{chosen})$, we can update comparison data as $\mathcal{D}_l = \mathcal{D}_l \cup \{(\mathbf{y}^{chosen} \mid \mathbf{x}^{chosen}) \succ (\mathbf{y}^i \mid \mathbf{x}^i)\}$, where $(\mathbf{y}^i \mid \mathbf{x}^i) \in \mathcal{Y}_{t+1} \cup C_{local} \setminus (\mathbf{y}^{chosen} \mid \mathbf{x}^{chosen})$.

Next, we will describe a classical preferential GP (Sec. 3.3) and our proposed context-aware preferential GP (Sec. 4) used in Stage 1, as well as our proposed three VR interfaces used in Stage 3.

3.3 Preliminaries: Preference Learning for Global Colors

A GP consists of random variables such that every finite collection of them follows a multivariate normal distribution [27]. PBO employs preferential GPs to represent the latent preference function $f(\cdot)$ in Stage 1. Specifically, PBO first trains a GP to fit the observed pairwise comparisons. Then, the fitted GP is leveraged to predict user preferences at new, unobserved points in the domain Ω. We briefly introduce the training and predicting phase of a standard preferential GP [8].

3.3.1 Training

Taking the comparison data $\mathcal{D}_g$ and relevant queried points $\mathcal{X}_g$ as input, GP_{global} does not estimate the latent preference function $G(\cdot)$ directly. Instead, it estimates the latent function values $\mathbf{g} = [G(\mathbf{x}_1), \ldots, G(\mathbf{x}_{N_g})]^T$ at the queried points $\mathcal{X}_g$ via Bayes' theorem [8, 31]:

$$P(\mathbf{g}|\mathcal{D}_g) = \frac{P(\mathbf{g})P(\mathcal{D}_g|\mathbf{g})}{P(\mathcal{D}_g)}. \tag{2}$$

Here, $P(\mathbf{g}|\mathcal{D}_g)$ is the posterior probability, reflecting how likely different values of $\mathbf{g}$ are. The posterior integrates the prior knowledge about the distribution of $\mathbf{g}$ before observing any data (i.e., the prior $P(\mathbf{g})$). The posterior also integrates the new information obtained from the observed comparisons (i.e., the likelihood $P(\mathcal{D}_g|\mathbf{g})$). $P(\mathcal{D}_g)$ is the marginal likelihood, which acts as a scaling factor to ensure that the posterior probabilities across all possible $\mathbf{f}$ sum to one.

Likelihood. The likelihood $P(\mathcal{D}_g|\mathbf{g})$ is calculated as [2, 16, 17, 35]:

$$\begin{aligned} P(\mathcal{D}_g|\mathbf{g}) &= \prod_{i=1}^{M_g} P(\mathbf{x}_{i1} \succ \mathbf{x}_{i2} \mid g_{i1}, g_{i2}) \\ &= \prod_{i=1}^{M_g} \frac{\exp(g_{i1}/s)}{\exp(g_{i1}/s) + \exp(g_{i2}/s)}, \end{aligned} \tag{3}$$

where $g_{i1} = G(\mathbf{x}_{i1})$ and $g_{i2} = G(\mathbf{x}_{i2})$ are the latent function values corresponding to $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$, and s is a scale factor.

Kernel function. The prior $P(\mathbf{g})$ in PBO is commonly calculated from a zero-mean GP, where the correlations between the queried points are captured by a $N_g \times N_g$ covariance matrix. This matrix is constructed using a kernel function $K_G : \Omega \times \Omega \to \mathbb{R}$. The ij^{th} element of the matrix is calculated as $K_G(\mathbf{x}_i, \mathbf{x}_j)$, where $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{X}_g$.

Latent preference values $\mathbf{g}^{MAP}$. Unlike standard BO where the absolute values of queried points are explicitly known, user preference values are latent in PBO, since users give feedback in the form of comparison. PBO uses the Maximum A Posteriori (MAP) estimation to obtain latent values, denoted as $\mathbf{g}^{MAP}$, by minimizing the following function [8, 17]:

$$S(\mathbf{g}) = -\sum_{i=1}^{M_g} P(\mathbf{x}_{i1} \succ \mathbf{x}_{i2} \mid g_{i1}, g_{i2}) + \frac{1}{2}\mathbf{g}^T K_G(\mathcal{X}_g, \mathcal{X}_g)^{-1}\mathbf{g}. \quad (4)$$

The obtained $\mathbf{g}^{MAP}$ is a vector with N_g elements representing user preferences of the N_g points in $\mathcal{X}_g$.

Hyperparameters. There are many types of kernel functions, each consisting of a set of hyperparameters θ. The purpose of the training phase is to find the optimal θ so that the GP_{global} maximizes the marginal likelihood $P(\mathcal{D}_g|\theta)$. The marginal likelihood serves as a loss function, offering a measure of how well the GP model explains the observed pairwise comparisons $\mathcal{D}_g$ under the current set of hyperparameters. More details on how θ is obtained can be found in [8, 17].

3.3.2 Predicting

In the predicting phase, given arbitrary N'_g points denoted as $\mathcal{X}^*_g = [\mathbf{x}^*_i \mid i = 1, \ldots, N'_g]$, the fitted GP_{global} can predict the user preferences $\mathbf{g}^* = [G(\mathbf{x}^*_1), \ldots, G(\mathbf{x}^*_{N'_g})]^T$ in the form of a Gaussian distribution $\mathbf{g}^* \sim \mathcal{N}(\mu^*_g, \Sigma^*_g)$, where Σ^*_g is a $N'_g \times N'_g$ covariance matrix. The mean μ^*_g is a vector with N'_g elements, each representing the expected preference value at a point in $\mathcal{X}^*$:

$$\mu^*_g = K(\mathcal{X}^*_g, \mathcal{X}_g)K(\mathcal{X}_g, \mathcal{X}_g)^{-1}\mathbf{g}^{MAP}. \quad (5)$$

4 Contextual Preference Learning for Local Colors

To model users' contextual preferences over local colors in the first stage, we need to consider the influence of global colors. A straightforward approach would involve collecting user preference feedback on local colors within each global context and training a standard preferential GP for each context. However, this method becomes impractical when users wish to experiment with multiple contexts due to its extensive demand for user feedback. To alleviate this scalability issue, we introduce a novel context-aware preferential GP that integrates a multi-task GP with the standard preferential GP. Previous research [10] has highlighted several benefits of multi-task GPs, including their ability to leverage information sharing between related tasks, improve predictive performance through learned task correlations, and enhance model efficiency by reducing the need for large datasets in each task.

However, integrating a multi-task GP into a preferential GP is non-intuitive for the following three reasons. First, they have different input requirements. Multi-task GPs are traditionally configured to work with absolute input values, while preferential GPs operate primarily on comparisons, lacking mechanisms to incorporate contexts directly. Second, existing multi-task GPs [10, 21] are designed for discrete and static contexts, which are observed in the training data. However, the global color space is continuous, encompassing an entire spectrum of possible values. Third, the global contexts are dynamically updated in each iteration (Fig. 2), necessitating predictive capability over local color preferences within unseen contexts.

Next, we will elaborate on our design of the context-aware preferential GP by tackling the different inputs as well as the continuous and unseen contexts.

4.1 Connecting preferential and multi-task GPs

Similar to standard preferential GP, our context-aware preferential GP takes comparison data $\mathcal{D}_l$ and queried points $\mathcal{Y}_l$ as input. Differently, each queried point in $\mathcal{Y}_l = [(\mathbf{y}_j \mid \mathbf{x}_j) \mid j = 1, \ldots, N_l]$ consisting of local color parameters $\mathbf{y}_j$ and associated global parameters x_j. By considering each unique context $\mathbf{x}_j$ as a related but distinct task, we leverage a multi-task GP [10] to enable cross-contextual information sharing, capturing subtle shifts in local preferences as global contexts vary.

Kernel Function. The kernel function K_L of the context-aware preferential GP incorporates a commonly-used kernel design in multi-task GP called intrinsic co-regionalization model [10,21], which utilizes two kernels to capture the relationships among global colors and local colors. The task kernel, K_T, is responsible for computing the covariance between different global color contexts, measuring how similar or different they are from each other and how changes in one might influence preferences in another. The local input kernel, K_I, on the other hand, computes the covariance between local color parameters. Then, the covariance matrix in the training phase is $K_L(\mathcal{Y}_l, \mathcal{Y}_l)$, and its ij^{th} element is the covariance between $(\mathbf{y}_i \mid \mathbf{x}_i) \in \mathcal{Y}_l$ and $(\mathbf{y}_j \mid \mathbf{x}_j) \in \mathcal{Y}_l$:

$$K_L(\mathbf{y}_i \mid \mathbf{x}_i), (\mathbf{y}_j \mid \mathbf{x}_j)) = K_I(\mathbf{y}_i, \mathbf{y}_j)K_T(\mathbf{x}_i, \mathbf{x}_j), \quad (6)$$

Latent preference values $\mathbf{l}^{MAP}$. We also use MAP estimation to obtain $\mathbf{l}^{MAP}$ with the following function:

$$S(\mathbf{l}) = -\sum_{i=1}^{M_l} P((\mathbf{y}_{i1} \mid \mathbf{x}_{i1}) \succ (\mathbf{y}_{i2} \mid \mathbf{x}_{i2}) \mid l_{i1}, l_{i2}, g_{i1}, g_{i2}) + \frac{1}{2}\mathbf{l}^T K_L(\mathcal{Y}_l, \mathcal{Y}_l)^{-1}\mathbf{l}. \quad (7)$$

The obtained $\mathbf{l}^{MAP}$ is a vector with N_l elements representing user preferences of the N_l points in $\mathcal{Y}_l$.

Predicting. We train our GP_{local} in all observed local colors with associated global colors based on $\mathcal{Y}_l$. This allows GP_{local} to fully utilize the observed data to capture the shared information among different contexts. On the contrary, in the predicting phase, we aim to find the best local colors regarding a specific global context rather than all observed contexts. To achieve this, we adopt a Kronecker-structured covariance matrix [21] in the predicting phase, since such structure allows for obtaining predictions for a specific context easily.

Specifically, given a context $\mathbf{x}_i$ and arbitrary N'_l points under this context, denoted as $\mathcal{Y}^*_l = [(\mathbf{y}^*_j \mid \mathbf{x}_i) \mid j = 1, \ldots, N'_l]$, we use the fitted GP_{local} to predict users' contextual preferences $\mathbf{l}^*_i = [L(\mathbf{y}^*_1 \mid \mathbf{x}_i), \ldots, L(\mathbf{y}^*_j \mid \mathbf{x}_i)]^T$. The fitted GP_{local} outputs $\mathbf{l}^*_i$ as a Gaussian distribution with the mean μ^*_{li} as:

$$\mu^*_{li} = \mathbf{M}_1\mathbf{M}_2^{-1}\mathrm{vec}(\mathbf{l}^{MAP}) \quad (8)$$

$$\mathbf{M}_1 = K_I(\mathbf{Y}^*, \mathbf{Y}) \otimes K_T(\mathbf{x}_i, \mathbf{X}) \quad (9)$$

$$\mathbf{M}_2 = K_I(\mathbf{Y}, \mathbf{Y}) \otimes K_T(\mathbf{X}, \mathbf{X}) \quad (10)$$

where $\mathbf{Y}^* = [\mathbf{y}^*_j \mid \mathbf{y}^*_j \in \mathcal{Y}^*_l]$, $\mathbf{Y} = [\mathbf{y}_j \mid \mathbf{y}_j \in \mathcal{Y}_l]$, and $\mathbf{X}$ contains all N_c unique contexts in $\mathcal{Y}_l$. Derived from the Kronecker product $\otimes$ of $K_I(\mathbf{Y}, \mathbf{Y})$ (a $N_l \times N_l$ matrix) and $K_T(\mathbf{X}, \mathbf{X})$ (a $N_c \times N_c$ matrix), $\mathbf{M}_2$ is a $N_lN_c \times N_lN_c$ matrix. Semantically, $\mathbf{M}_2$ is the covariance of all the combinations of N_l local colors and N_c global colors. Similarly, $\mathbf{M}_1$ is a $N'_l \times N_lN_c$ matrix. $\mathrm{vec}(\mathbf{l}^{MAP})$ is a $N_lN_c \times 1$ matrix, generated by reshaping $\mathbf{l}^{MAP}$ to represent the latent preference values of the local colors under all global colors (Fig. 4).

4.2 Discretizing Continuous Global Contexts

To address the continuous nature of global color spaces, we draw inspiration from previous research [38] and discretize these global contexts into a finite set of discrete contexts through four key steps: clustering global colors, selecting representative contexts, transforming local colors, and updating the kernel functions. Initially, a K-means clustering algorithm is applied to the set of queried global colors $\mathcal{X}_g$, yielding k distinct clusters $\{C_1, C_2, \ldots, C_k\}$. Subsequently, for each cluster C_i, we determine a representative point $\mathbf{c}_i$, which is the global color within the cluster that exhibits the highest preference value according to $\mathbf{g}^{MAP}$. These points then serve as the discrete contexts for subsequent modeling. Next, the set $\mathcal{Y}_l = \{(\mathbf{y}_j \mid \mathbf{x}_j)\}$ is transformed into $\mathcal{Y}^c_l = \{(\mathbf{y}_j \mid \mathbf{c}_j)\}$,

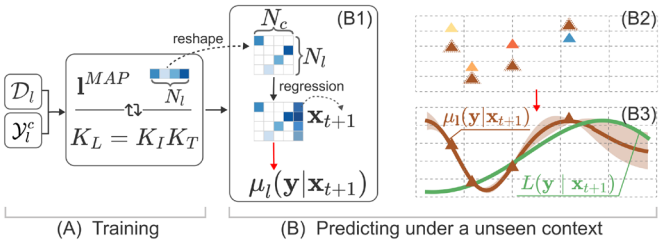

Fig. 4: Illustration of the training and predicting phase of our context-aware preferential GP. The red arrow in (B1) represents that we obtain the contextual preference μ_l based on regression results. (B2) and (B3) show the same step with a more intuitive illustration.

Table 1: Training accuracy under different context numbers and parameter dimensions. Boldface highlights the best results in discrete contexts.

	# Discrete Context				Continuous Context	P-value
	5	10	15	20		
2D	0.831	0.798	**0.856**	0.794	0.784	0.080
6D	0.659	**0.700**	0.683	0.685	0.682	0.964
10D	0.596	**0.607**	0.586	0.568	0.604	0.999
14D	0.653	**0.767**	0.668	0.682	0.813	0.831
18D	0.827	0.856	0.853	**0.958**	0.786	0.019 (∗)

where $\mathbf{c}_j$ corresponds to the representative context of the cluster containing $\mathbf{x}_j$. Finally, the computation of the kernel in Eq. 6 will be updated by replacing $\mathbf{x}_i$ with $\mathbf{c}_i$:

$$K_L((\mathbf{y}_i \mid \mathbf{c}_i),(\mathbf{y}_j \mid \mathbf{c}_j)) = K_I(\mathbf{y}_i,\mathbf{y}_j)K_T(\mathbf{c}_i,\mathbf{c}_j), \tag{11}$$

These updates subsequently influence the computation of latent local preference values in Eq. 7 and predictions in Eq. 8.

4.3 Generalizing to Unseen Contexts

In Stage 2 of each iteration to determine the next queried points at local level, we encounter new global colors $\mathcal{X}_{t+1} = \{\mathbf{x}^1_{t+1}, \cdots, \mathbf{x}^n_{t+1}\}$, which include contexts not previously observed in the training data of GP_{local}. Consequently, these new contexts lack corresponding latent preference values in l^{MAP}, making the calculation of the mean μ^*_l infeasible using Eq. 8. To address this issue, we employ regression models to establish a relationship between the known preferences in $\mathcal{Y}_l$ and l^{MAP}. This model is then used to extrapolate latent preference values for the unseen contexts within $\mathcal{X}_{t+1}$. Once these new values are determined, we expand the vector $\text{vec}(l^{MAP})$ to include these additional preferences, transforming it into a $N_l(N_c+1)\times 1$ matrix. Simultaneously, the matrix $\mathbf{X}$, representing all contexts, is updated to a $(N_c+1)\times(N_c+1)$ matrix to incorporate the new global color contexts. This expansion enables the predictive function to work for these previously unobserved contexts (Fig. 4-B), thereby maintaining the robustness and adaptability of our framework for global colors that dynamically change. Thus, for an unseen global context denoted as $\mathbf{x}_{new}$, we have:

$$\mu^*_{li} = \mathbf{M}_1\mathbf{M}_2^{-1}\text{vec}(\mathbf{l}^{MAP}_{\text{updated}}) \tag{12}$$

$$\mathbf{M}_1 = K_I(\mathbf{Y}^*,\mathbf{Y}) \otimes K_T(\mathbf{x}_i, \mathbf{X}\cup\{x_{\text{new}}\}) \tag{13}$$

$$\mathbf{M}_2 = K_I(\mathbf{Y},\mathbf{Y}) \otimes K_T(\mathbf{X}\cup\{x_{\text{new}}\}, \mathbf{X}\cup\{x_{\text{new}}\}) \tag{14}$$

$$\text{vec}(\mathbf{l}^{MAP}_{\text{updated}}) = \text{vec}(\mathbf{l}^{MAP}) \oplus \mathbf{l}_{\text{new}} \tag{15}$$

where $\oplus$ denotes the concatenation of the original vector with the new latent preference values $\mathbf{l}_{\text{new}}$, calculated or estimated for the new global contexts.

4.4 Ablation Studies

We conduct experiments with synthetic functions to evaluate our proposed local context-aware preferential GP with two goals: (1) to assess the effects of different numbers of contexts (Sec. 4.2), and (2) to investigate the effectiveness of different regressors (Sec. 4.3).

Setup. We first implement our dual-level PBO framework (Sec. 3.2) with BoTorch [3], a widely-used BO library. We use Ackley functions as both global and local latent preference functions, based on which we synthesize users' preference comparisons. Since color grading may involve different numbers of parameters, we conduct multiple experiments on global Ackley functions with varying dimensions (e.g., 2 to 18 dimensions), each dimension ranging from $[0,1]$. The dimension of a local Ackley function is twice the dimension of a global Ackley function, with the second half representing global parameters as a context and the first half representing local parameters under the context.

We perform 20 trials for each combination of context numbers with parameters or regressors. In each trial, we train GP_{global} on $\mathcal{D}_g$ and GP_{local} on $\mathcal{D}_l$ using about 100 random global or local preference comparisons, respectively. Training accuracy is calculated as the percentage of correct comparisons made by the fitted GP_{local} among the 100 local training comparisons. Then, we randomly sample 25 unseen global contexts and for each context, we use the fitted GP_{local} to predict values at grid points obtained by dividing the parameter space Ω into equal-sized intervals. To evaluate how well GP_{local} can capture the variances in local preferences under different unseen contexts, we calculate the pairwise distances between predicted values under adjacent contexts. The distances can be compared with the ground truth, which is obtained from local Ackley functions by fixing the second-half parameters with each context.

Table 2: Predicting results under different context numbers and parameter dimensions with SVR. Boldface highlights the closest results to GT.

	Ground Truth (GT)	# Discrete Context				Continuous Context	P-value
		5	10	15	20		
2D	8.519	**8.960**	0.706	0.035	0.012	0.002	0.002 (∗)
3D	2.560	4.769	**2.436**	0.778	0.378	0.001	0.041 (∗)
4D	4.696	6.303	**5.206**	5.324	3.482	0.035	0.005 (∗)
5D	8.778	13.199	10.877	11.908	**8.654**	1.068	0.001 (∗)
6D	16.608	22.128	21.130	21.692	**17.343**	11.600	0.028 (∗)

Table 3: Predicting results under different context numbers and regressors with 2D parameters. Boldface highlights the closest results to GT.

	Ground Truth (GT)	# Discrete Context				Continuous Context	P-value
		5	10	15	20		
Polynomial	8.519	**5.710**	0.460	0.090	0.023	0.005	0.013 (∗)
Random forest	8.519	**6.188**	0.842	0.036	0.012	0.007	0.049 (∗)
Gradient boosting	8.519	**5.769**	0.674	0.026	0.009	0.007	0.024 (∗)
SVR	8.519	**8.960**	0.706	0.035	0.012	0.002	0.002 (∗)

Results. Table 1, Table 2, and Table 3 show average training accuracy or predicting distances of the trials. The P-values are obtained from Tukey's HSD tests comparing the continuous context with the number of discrete contexts that produced the best results. Specifically, there are no significant differences in training accuracy among the various numbers of contexts for most parameter numbers (Table 1), indicating that reducing the number of contexts does not compromise training accuracy. However, selecting a proper number of discrete contexts for a given parameter number can significantly outperform continuous contexts in prediction (Table 2), regardless of the regressor used (Table 3). Besides, regressors can influence the prediction, possibly because they produce different l^{MAP}.

4.5 Qualitative Examples

Figure 5 demonstrates the color grading processes and results of a standard PBO adjusting global colors, and our dual-level PBO adjusting global and local colors, on five color parameters: red, blue, green channels of white balance, contrast, and saturation. We ensure that both the standard PBO and our dual-level PBO operate within the same color parameter spaces (i.e., Ω), but we experiment with initializing these parameter spaces differently. We observe that both the standard PBO and our dual-level PBO can perform color grading either within a specific color hue (e.g., orange in Fig. 5-top) or across broader hue ranges (e.g., green, blue, to purple in Fig. 5-bottom). However, as shown by the results after Iteration 3, our dual-level PBO offers better control granularity, allowing for more nuanced adjustments on POIs, such as varying local hues from the global hues or fine-tuning brightness levels. Thus, our dual-level PBO provides creators with greater creative flexibility to achieve detailed effects and facilitates them to obtain images that can guide user attention to POIs within a 360-degree space.

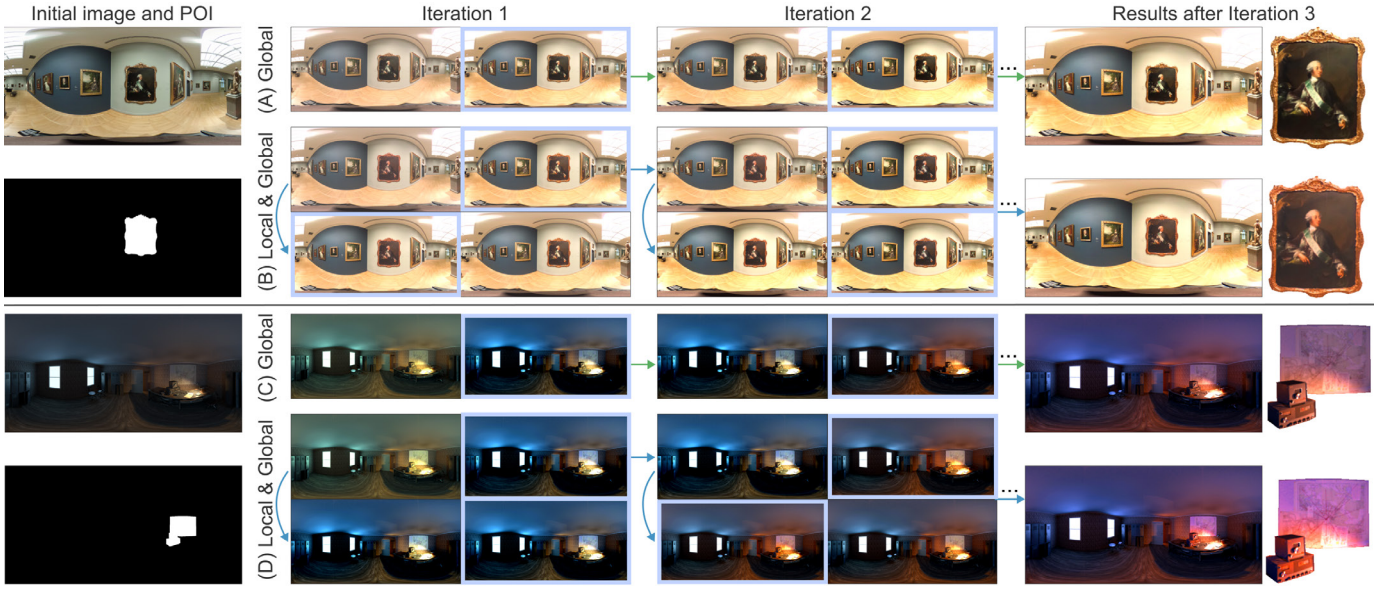

Fig. 5: Comparison of color grading processes and results between a standard PBO and our dual-level PBO. **(A, C) Standard PBO for global-only adjustment**: Only global color parameters are applied to the entire image. The user compares options at the global level and selects their preferred option in each iteration, as indicated by the green arrows. **(B, D) Dual-level PBO for dual-level adjustment**: Global parameters are applied to the entire image and local parameters are applied to the POI. The user provides preference feedback in two steps during each iteration, as indicated by the blue arrows. The first step is the same as the global-only adjustment. In the second step, the global colors of all options are updated to the selected global colors, and the user compares the local colors to select their preferred option. Selected options are highlighted with light blue boxes.

5 VR User Interfaces for Color Comparison

To facilitate creators to compare color-graded options of 360-degree images in VR and provide their preference feedback in the third stage, we design and evaluate three VR comparison interfaces.

5.1 User Interface Design

Previous research [14, 40] has pointed out that the types of displays and viewing environments can affect the appearance and perception of colors. Given that 360-degree images are intended for consumption in VR environments, it is crucial to support visual assessments directly within VR to ensure color fidelity and perceptual accuracy. We consider three VR interfaces (Fig. 1) as a part of our dual-level PBO framework to compare color-graded options and judge the preferred option:

- **Side-by-side** interface adapts typical desktop interfaces for general 2D images [35] by displaying equirectangular projections of color-graded options adjacent to each other on a virtual 2D plane, enabling direct visual contrast.
- **Sequential** interface fully utilizes the VR capabilities by enveloping users within one 360-degree option at a time. Users can switch between different options. This allows for a focused analysis of each option before transitioning to the next.
- **Partial** interface divides the entire VR space into equal areas, with each area displaying the same slice of the target 360-degree image but with different color parameters, enabling simultaneous color comparison for the same slice across multiple options.

5.2 Controlled User Study

We carried out a controlled user study to evaluate different user interfaces for comparing and judging color-graded options for 360-degree images within VR environments.

5.2.1 Study Design and Setup

Participants and Apparatus. With our institutional review board's approval, we recruited 18 participants (P1-P18; eight females; ages 21-30) with normal color vision. On a 7-point Likert scale, their average familiarity with color grading was 5.1 (range: 2-7), and with VR was 4.4 (range: 1-7). Two participants edited photos and adjusted colors daily, seven weekly, seven monthly, and two annually. The study was conducted with Meta Quest 2.

Conditions and tasks. We investigated two independent variables: the type of user interface and the number of options. The types of interfaces are described in Sec. 5.1: *side-by-side*, *sequential*, and *partial*. For the number of options, we assessed two and four, which are typical in existing PBO frameworks [8, 35]. Thus, there were six conditions in total. For each condition, participants were required to perform five iterations, during which they needed to compare options and judge the preferred options for global and local colors in order.

Procedure. The controlled user study followed a within-subjects design, with each participant testing all six conditions. We prepared six 360-degree images and counterbalanced the order of the conditions and images. Each participant joined the study individually. Each session lasted 1-1.5 hours, beginning with tutorials on each user interface. We allowed participants to familiarize themselves with the interfaces using an additional set of 360-degree images in a training session. After that, participants performed the comparison and judgment tasks at both global and local levels for each condition, followed by completing a 7-point Likert scale questionnaire. This questionnaire assessed each condition across five aspects (Fig. 6): overall rating (Q1), effectiveness (Q2–Q5), user experience (Q6–Q8), usability (Q9–Q10), and workload (Q11–Q16). We also recorded their interactions and head movements, conducted semi-structured interviews to gather their opinions on each condition, and asked them to rank the user interface designs after completing all six conditions.

Ethics Statement. Before the user study was conducted, the research procedures were reviewed and approved by the Research Ethics Committee of the Department of Engineering at the University of Cambridge under Application No. 459. Written consent forms were obtained from the participants before they attended the study.

5.2.2 Results and Analysis: Ratings

Figure 6 presents the ratings of the three interfaces across two and four options. To assess statistical significance, we employed the Friedman test on the three interfaces (the statistics are reported below), followed by post-hoc Wilcoxon signed-rank tests for pairwise comparisons among them (the statistics are annotated in Fig. 6).

Overall ratings (Q1). Significant effects of interfaces were found for both the two ($\chi^2 = 9.7, p < 0.01$) and four ($\chi^2 = 20.6, p < 0.001$)

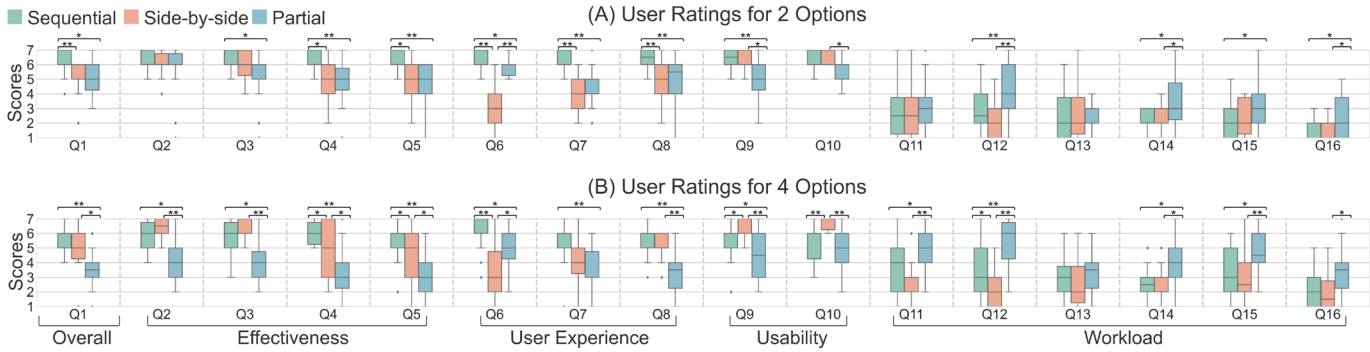

Fig. 6: User ratings on three user interfaces with two and four options. The scores for Q1–Q10 are the higher the better, for Q11–Q16 the lower the better. The questions cover overall preferences (Q1), effectiveness in comparing global (Q2) and local (Q4) color differences, effectiveness in judging preferred global (Q3) and local (Q5) colors, immersion (Q6), enjoyment (Q7), consistency (Q8), ease of use (Q9), ease of learning (Q10), mental demand (Q11), physical demand (Q12), temporal demand (Q13), performance (Q14), effort (Q15), and frustration (Q16). * represents $p < 0.05$ and ** represents $p < 0.01$.

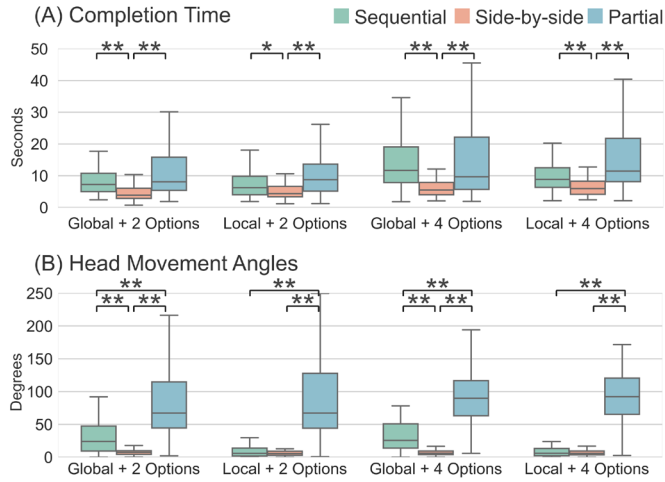

Fig. 7: Completion time for comparison and adjustment (A) and average head movement angle per unit of time (B).

options. For two options, participants significantly liked *sequential* over *side-by-side* and *partial*. For four options, participants significantly liked *sequential* over *partial*, *side-by-side* over *partial*.

Effectiveness (Q2–Q5). At the global level, significant effects of interfaces were found for perceiving global color differences (Q2) among four ($\chi^2 = 15.7, p < 0.001$) options, and judging preferred global colors (Q3) among two ($\chi^2 = 8.8, p < 0.05$) and four ($\chi^2 = 11.0, p < 0.01$) options. For two options, the pairwise comparisons show that no significance existed between *sequential* and *side-by-side*, *side-by-side* and *partial*. This was because the three interfaces had their unique benefits for comparing two options at the global level. Specifically, *"the sequential mode allows me to observe the color changes by switching options and experiencing how each one feels in the entire space"* (P4), *"I can make decisions quickly with the side-by-side mode since it provides thumbnails within my viewport"* (P15), and *"the difference between the two options around the dividing line in the partial mode is easy to perceive"* (P7). However, for four options, participants found *sequential* and *side-by-side* were significantly more effective than *partial*, since *"dividing into four segments in partial mode makes it difficult to imagine and assess the overall aesthetics of each option"* (P9).

At the local level, the three interfaces showed significantly different effectiveness in perceiving local color differences (Q4; two: $\chi^2 = 12.7, p < 0.05$; four: $\chi^2 = 21.8, p < 0.001$), and judging preferred local colors (Q5; two: $\chi^2 = 13.0, p < 0.05$; four: $\chi^2 = 20.3, p < 0.001$). The pairwise comparisons show that *sequential* was overall better than both *Side-by-side* and *partial* for local colors, regardless of the number of options. *side-by-side* and *partial* were inefficient mainly because *"they require me to shift my eyes and even turn my body between different options"* (P3), *"they cost extra effort for me to locate where the POI is"* (P17), and *"the colors not belonging to POIs disturb me during eye shifting"* (P10). On the contrary, the sequential mode can avoid the above issues since *"I only need to focus on the POI and switch options with the controller without moving my head"* (P3). Besides, *side-by-side* received complaints that *"the POI is too small to observe the local colors, but I don't want bigger displaying areas, which then require more eye shifting"* (P12).

User Experience (Q6–Q8). Significant effects of interfaces were found for both the two and four options regarding immersion (Q6; two: $\chi^2 = 28.5, p < 0.001$; four: $\chi^2 = 16.4, p < 0.001$), enjoyment (Q7; two: $\chi^2 = 17.0, p < 0.001$; four: $\chi^2 = 13.8, p < 0.001$), and consistency (Q8; two: $\chi^2 = 17.4, p < 0.001$; four: $\chi^2 = 18.6, p < 0.001$). Specifically, *sequential* and *partial* were significantly more immersive (Q6) than *side-by-side*, regardless of the number of options. P12 stated, *"The sequential and partial modes wrap me entirely and allow me to feel more details of each option, such as its light, shadow, and overall atmosphere." Sequential* were significantly more enjoyable (Q7) and consistent (Q8) than *partial* regardless of the number of options. *Partial* was regarded as *"strange"* (P4), *"unnatural, and even scary"* (P15) because *"repeated segments stitched together wouldn't appear in the real world"* (P7). Although *sequential* were significantly more enjoyable (Q7) and consistent (Q8) than *side-by-side* under two options, no significance existed under four options. This is because *"the sequential mode with four options contains too many details and makes me feel overwhelmed"* (P14).

Usability (Q9–Q10). Significant effects were found for both the two and four options regarding ease of use (Q9; two: $\chi^2 = 13.8, p < 0.001$; four: $\chi^2 = 22.3, p < 0.001$) and ease of learning (Q10; two: $\chi^2 = 8.0, p < 0.05$; four: $\chi^2 = 20.4, p < 0.001$). No significance existed between *sequential* and *side-by-side* under two options, since *"both have similar practices in 2D photo editing software"* (P1). However, *side-by-side* was significantly easier to use (Q9) and learn (Q10) than *sequential* with four options, which was overwhelming; and *partial*, regardless of the number of options, due to its relatively novel design.

Workload (Q11–Q16). Significant effects were found on physical demand (Q12; two: $\chi^2 = 22.7, p < 0.001$; four: $\chi^2 = 31.1, p < 0.001$), performance (Q14; two: $\chi^2 = 11.8, p < 0.01$; four: $\chi^2 = 12.1, p < 0.01$), effort (Q15; two: $\chi^2 = 8.5, p < 0.05$; four: $\chi^2 = 18.6, p < 0.001$), and frustration (Q16; two: $\chi^2 = 10.6, p < 0.01$; four: $\chi^2 = 14.4, p < 0.001$). No significant difference existed between *sequential* and *side-by-side*, except that *sequential* was significantly more physically demanding than *side-by-side* for four options due to body movement. *Partial* required significantly more workload than *sequential* and *side-by-side*, since *"the options are distributed in different directions, forcing me to glance around, while with the sequential mode, I can choose whether to turn my body or focus solely on the POI"* (P2).

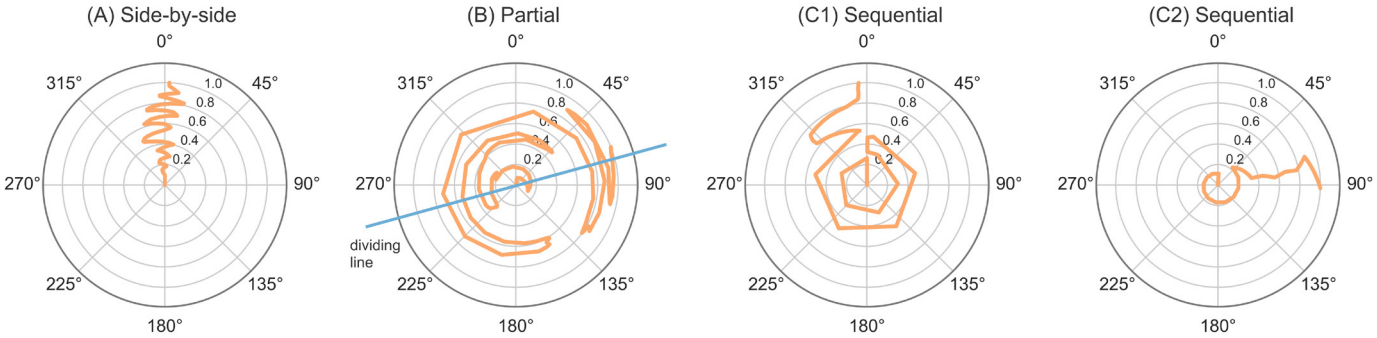

Fig. 8: Examples of how the participants' facing direction changed throughout the elapsed time with the three interfaces. The radius represents the normalized time elapsed since the iteration started. The angular coordinate represents the direction that the participant faced.

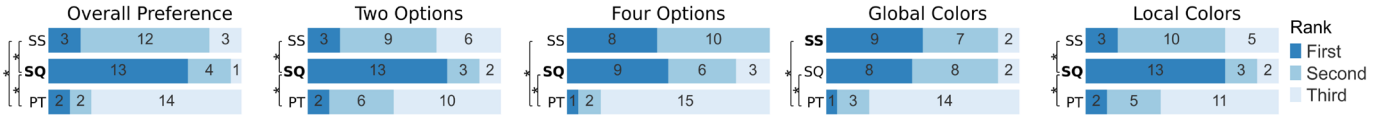

Fig. 9: Participants' rankings of the three user interfaces, including overall preference and their breakdown of preferences when comparing two options, four options, global colors, and local colors. SS = *Side-by-side*, SQ = *Sequential*, and PT = *Partial*.

5.2.3 Results and Analysis: Participants' Behaviors

To have a more granular understanding of how the participants used the three interfaces, we also analyzed participants' behaviors based on logged user interactions and head movements.

Completion time. In each iteration, the participants were required to select the options with their most preferred colors at the global and local levels, respectively. Figure. 7-A shows the time they used for comparison and judgment. We employed the Friedman test, followed by post-hoc Wilcoxon signed-rank tests. Significant effects of the interfaces were found in all the following combinations: global colors of two options ($\chi^2 = 35.0, p < 0.001$), local colors of two options ($\chi^2 = 25.8, p < 0.001$), global colors of four options ($\chi^2 = 40.4, p < 0.001$), and local colors of four options ($\chi^2 = 42.2, p < 0.001$). The pairwise comparisons show that there were no significant differences between *sequential* and *partial* across the four combinations. However, *sequential* and *partial* required significantly more time than *side-by-side* across the four combinations. This is because *"the sequential and partial mode exposure more information and details to me, and consuming the details takes time"* (P16). Although more time-consuming, participants appreciated the detail and immersion provided by *sequential*, as selecting appropriate colors to enhance image appearance in the VR environment was more important to our participants. An extreme example was P13, who stated, *"When adjusting colors for my 2D photos, I often check whether the colors look good on others' smartphones. The same applies to 360-degree images. Examining the images in the environments where my audience will consume them is important."*

Head movement. Figure 7-B illustrates the participants' average head movement angle per unit of time across the four combinations. Significant effects were found: global colors of two options ($\chi^2 = 83.3, p < 0.001$), local colors of two options ($\chi^2 = 74.0, p < 0.001$), global colors of four options ($\chi^2 = 105.2, p < 0.001$), and local colors of four options ($\chi^2 = 96.4, p < 0.001$).

Figure 8 illustrates some representative examples of how our participants moved their heads when using the three interfaces. Specifically, in the side-by-side mode, users made small head movements to compare options within their field of view (Fig. 8-A). Similarly, in the partial mode, since all options were displayed at once but distributed in the entire space, users moved their heads more widely to compare them. Interestingly, we found many participants relied on the dividing line, which allowed them to distinguish between options with minimal head movement. They often started at the dividing line and gradually explored both sides. For example, as shown in Fig. 8-B, a person might start facing one direction, shift left and right, and then gradually turn to face the opposite direction to repeat the left and right shifting. In the sequential mode, we found two opposite representative user behaviors. As shown in Fig. 8-C1, some users made a complete turn for each option since only one option was displayed at a time. However, it was tiring to make several complete turns to view all options. Thus, after familiarizing themselves with the entire scene, some other users only focused on a specific area of interest and limited their head movement to that section (Fig. 8-C2).

5.2.4 Results and Analysis: Rankings

We asked the participants to rank the interfaces and option numbers after they experienced all six conditions.

Rankings of the interfaces. Figure 9 shows our participants' rankings of the three interfaces. The Friedman tests revealed significant differences in their preferences, including overall preference ($\chi^2 = 16.0, p < 0.001$) and breakdowns when comparing two options ($\chi^2 = 10.8, p < 0.01$), four options ($\chi^2 = 16.4, p < 0.001$), global colors (($\chi^2 = 14.1, p < 0.001$)), and local colors ($\chi^2 = 11.4, p < 0.01$). Specifically, users significantly preferred *sequential* overall the most, as well as for two options and local colors. However, there were no significant differences between *sequential* and *side-by-side* for four options and global colors. These ranking results align with the detailed ratings shown in Fig. 6.

Rankings of numbers of options. Five out of eighteen participants preferred four options, which allowed them to indicate their color preferences more efficiently by comparing larger batches at once. The other thirteen participants preferred comparing and judging the colors of two options in VR for three reasons. First, the 360-degree images carried a lot of information, especially when they wrapped the users and took up the entire space, so comparing four options was overwhelming for most participants. Second, some participants mentioned that afterimages interfered with their ability to judge the four options in the sequential mode. For example, P3 said, *"When switching options, the previous one stayed in my mind, affecting my judgment of the next color. This caused interference when the four options were displayed sequentially. However, with only two options in the sequential mode, this issue didn't occur. Similarly, there was no interference in the side-by-side mode because I could compare them at the same time."* Third, for the sequential and partial modes with four options, participants often struggled to remember the associations between the options and the joystick or displayed directions. This difficulty led them to find and check the options back and forth by pushing the joystick or turning their bodies in different directions.

5.2.5 Summary

Our results indicate that creators showed the highest preference for the sequential interface with two color-graded options. In the following, we summarize our findings regarding the strengths and limitations of the three interfaces.

Sequential. The sequential mode with two options is the preferred condition for comparison and judgment at both global and local levels. It offers high immersion for participants while maintaining a manageable workload, enabling them to observe the details of a 360-degree image effectively. In contrast, four options are less effective due to the afterimage phenomenon, which interferes with users' ability to make comparisons and judgments. Moreover, four options place a greater burden on users to associate the options with their display locations.

Side-by-side. The side-by-side mode is more effective for comparing global colors than local colors, regardless of the number of options. It provides thumbnails of all options in the form of 2D projections within a user's field of view, minimizing physical workload. However, it thereby suffers from a small display area for POIs and makes it challenging to imagine how a scene looks and feels in VR.

Partial. The partial mode with four options is the least preferred condition for both global and local color comparisons. It is physically demanding and creates a strange visual experience with the whole space divided into four segments. While the partial mode with two options is also considered unrealistic, it retains relatively high immersion and is learnable for users. Additionally, the mode is effective for comparing colors along the dividing line.

6 Discussion

This section discusses our proposed algorithm and interfaces from various aspects, limitations, and possible solutions.

Generalizing to different parameters and other images. The user study in our work focused on common parameters such as color balance and brightness, following the practices of previous research [16]. However, our framework can be generalized to handle other aspects of image editing, such as lens shading and color curves, as long as they can be quantified using scalar values. The optimized parameters in our method are highly coupled with individual images. It aligns with the common practice of general users, who often edit images individually and adapt their manual parameter adjustment according to the content and style of each image. In this regard, our method can facilitate these user scenarios by automating the parameter optimization process for individual images. However, the learned user preferences for a specific image cannot be easily generalized to other different images. To increase the generalizability, a potential extension could be to capture the relationships between user feedback and image features [30]. This allows the framework to initialize the parameter values for new images based on the learned relationships, before further optimizing the parameters based on user feedback.

Considering the unique characteristics of 360-degree images. The core difference between adjusting 360-degree image colors in VR and desktop environments, which the proposed method addresses, lies in how we present color-graded options to users. 360-degree images are closely associated with VR because they are best experienced in this medium. Investigating suitable VR interfaces is as important as proposing an effective optimizer (i.e., GP in our work), because interfaces can allow users to express their implicit color preferences accurately, thereby laying the foundation for the optimizer to model correct color preferences. As to our context-aware preferential GP, it has generalizability across both 2D and 360-degree images. However, there are opportunities to further tailor the algorithm to the unique characteristics of 360-degree images. This could involve developing specialized techniques to address the spherical distortion [34] inherent in 360-degree images and ensure color consistency across the entire panoramic view. Additionally, considering the crucial role of colors in directing viewer focus in VR, future work could incorporate saliency models [20, 23] into our dual-level PBO-based framework.

Implications for VR comparison interface design. Although the sequential mode with two options is generally preferred, each interface for comparing 360-degree image colors in VR has strengths and weaknesses. Future VR color-grading systems can combine the three types of interfaces to maximize their benefits. For example, a system could primarily utilize the sequential mode for comparison. To reduce users' cognitive load in associating options with their display locations, the side-by-side mode can be incorporated to display small thumbnails. This would be particularly helpful when navigating through more than two options in VR. Additionally, to address the limitations of both the sequential (e.g., inability to compare options simultaneously) and side-by-side (e.g., limited POI display area) modes, the partial mode could be employed. By placing the POIs of two options near the dividing line, users can examine more details with minimal head movement. Regarding the number of options, the sequential and partial modes can overwhelm users when dealing with more than two options. This suggests that the methods commonly employed on 2D screens to reduce the number of iteration rounds, such as comparing multiple options using sequential lines [17] or galleries [16], may not be applicable in VR environments. Future research should explore alternative approaches, such as developing more efficient acquisition functions, to accelerate the optimization process in VR.

Limitations. While our proposed dual-level color grading method can streamline the image editing process, we acknowledge that the method has some limitations in unleashing creators' creativity. First, the dual-level PBO currently falls short of fostering unexpected results that could inspire highly original and creative work. This limitation arises because the acquisition functions currently used in Stage 2 lean towards exploitation (i.e., refining results based on known preferences) over exploration (i.e., searching for diverse and unexplored results). To increase exploratory divergence, future research could develop acquisition functions tailored to the dual-level PBO. Additionally, creators could be given control over this balance based on their goals: those seeking efficiency could prioritize exploitation to achieve satisfactory results quickly, while those valuing creative diversity could emphasize exploration to uncover more unconventional outcomes. Second, as the dual-level PBO directly manipulates raw color parameters such as contrast and color balance, some sampled parameters may lead to disharmonious colors that fail to meet creators' expectations. Future work could investigate how to integrate our personalized algorithms with color recommendation models [6, 13, 39] to ensure parameter adjustments toward aesthetically pleasing outcomes. Third, the current VR interfaces, while intuitive, only support option selection and limit creators' autonomy. Future work could explore ways to enable users to express creative intent more freely, such as interactively constraining parameter spaces (e.g., defining hue ranges in Fig. 5), leveraging domain knowledge [15], or selecting arbitrary local regions [35]. This requires advancements in VR interaction design and integration of user inputs into the dual-level PBO algorithm.

7 Conclusion

This work presents a novel dual-level PBO-based framework aimed at facilitating personalized color grading for 360-degree images directly in VR environments. By integrating both global and local color adjustments, our framework advances the capabilities of existing PBO algorithms, which have traditionally focused solely on global color grading. To address the challenge of learning contextual preferences for local colors under dynamic global contexts, we introduce a context-aware preferential GP that incorporates a multi-task GP, enabling the efficient prediction of local preferences across varying global conditions. Additionally, our work addresses the need for effective VR interfaces that allow creators to compare the colors of 360-degree images in an immersive environment. Through a comprehensive evaluation of three types of VR interfaces (i.e., side-by-side, sequential, and partial), we identify the sequential interface as the most effective in facilitating preference feedback, particularly when comparing two options.

Acknowledgments

We would like to thank Yue Jiang and the anonymous reviewers for their valuable comments. This work was partially supported by RGC GRF Grant 16210722. John J. Dudley and Per Ola Kristensson were supported by EPSRC (Grants EP/S027432/1 and EP/W02456X/1).

REFERENCES

[1] T. D. Albright and G. R. Stoner. Contextual influences on visual processing. *Annual Review of Neuroscience*, 25(1):339–379, 2002. 2

[2] R. Astudillo, Z. J. Lin, E. Bakshy, and P. Frazier. qEUBO: A decision-theoretic acquisition function for preferential Bayesian optimization. In *International Conference on Artificial Intelligence and Statistics*, pages 1093–1114. PMLR, 2023. 3

[3] M. Balandat, B. Karrer, D. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. *Advances in Neural Information Processing Systems*, 33:21524–21538, 2020. 5

[4] L. Chan, Y.-C. Liao, G. B. Mo, J. J. Dudley, C.-L. Cheng, P. O. Kristensson, and A. Oulasvirta. Investigating positive and negative qualities of human-in-the-loop optimization for designing interaction techniques. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2022. 2

[5] R.-C. Chang, C.-H. Ting, C.-S. Hung, W.-C. Lee, L.-J. Chen, Y.-T. Chao, B.-Y. Chen, and A. Guo. Omniscribe: Authoring immersive audio descriptions for 360 videos. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 1–14, 2022. 2

[6] C.-K. T. Chao, J. Klein, J. Tan, J. Echevarria, and Y. Gingold. ColorfulCurves: Palette-aware lightness control and color editing via sparse optimization. *ACM Transactions on Graphics*, 42(4):1–12, 2023. 9

[7] C.-H. Chiu, Y. Koyama, Y.-C. Lai, T. Igarashi, and Y. Yue. Human-in-the-loop differential subspace search in high-dimensional latent space. *ACM Transactions on Graphics*, 39(4):85–1, 2020. 2

[8] W. Chu and Z. Ghahramani. Preference learning with Gaussian processes. In *Proceedings of the International Conference on Machine Learning*, pages 137–144, 2005. 2, 3, 4, 6

[9] B. Duinkharjav, K. Chen, A. Tyagi, J. He, Y. Zhu, and Q. Sun. Color-perception-guided display power reduction for virtual reality. *ACM Transactions on Graphics*, 41(6):1–16, 2022. 2

[10] Q. Feng, B. Letham, H. Mao, and E. Bakshy. High-dimensional contextual policy search with unknown context rewards using Bayesian optimization. *Advances in Neural Information Processing Systems*, 33:22032–22044, 2020. 2, 4

[11] C. Haine. *Color Grading 101: Getting started color grading for editors, cinematographers, directors, and aspiring colorists*. Routledge, 2019. 1

[12] J. Hartmann, S. DiVerdi, C. Nguyen, and D. Vogel. View-dependent effects for 360 virtual reality video. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 354–364, 2020. 2

[13] Z. Huang, N. Zhao, and J. Liao. Unicolor: A unified framework for multimodal colorization with transformer. *ACM Transactions on Graphics*, 41(6):1–16, 2022. 9

[14] T. Kaminokado, Y. Hiroi, and Y. Itoh. StainedView: Variable-intensity light-attenuation display with cascaded spatial color filtering for improved color fidelity. *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3576–3586, 2020. 1, 6

[15] Y. Koyama and M. Goto. Bo as Assistant: Using Bayesian optimization for asynchronously generating design suggestions. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 1–14, 2022. 1, 9

[16] Y. Koyama, I. Sato, and M. Goto. Sequential gallery for interactive visual design optimization. *ACM Transactions on Graphics*, 39(4):88–1, 2020. 1, 2, 3, 9

[17] Y. Koyama, I. Sato, D. Sakamoto, and T. Igarashi. Sequential line search for efficient visual design optimization by crowds. *ACM Transactions on Graphics*, 36(4):1–11, 2017. 2, 3, 4, 9

[18] J. Li, J. Lyu, M. Sousa, R. Balakrishnan, A. Tang, and T. Grossman. Route tapestries: Navigating 360 virtual tour videos using slit-scan visualizations. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 223–238, 2021. 2

[19] Z. Li, Y. Cui, T. Zhou, Y. Jiang, Y. Wang, Y. Yan, M. Nebeling, and Y. Shi. Color-to-depth mappings as depth cues in virtual reality. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 1–14, 2022. 2

[20] G. Ma, S. Li, C. Chen, A. Hao, and H. Qin. Stage-wise salient object detection in 360 omnidirectional image via object-level semantical saliency ranking. *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3535–3545, 2020. 9

[21] W. J. Maddox, M. Balandat, A. G. Wilson, and E. Bakshy. Bayesian optimization with high-dimensional outputs. *Advances in Neural Information Processing Systems*, 34:19274–19287, 2021. 4

[22] E. Markley, N. Matsuda, F. Schiffers, O. Cossairt, and G. Kuo. Simultaneous color computer generated holography. In *SIGGRAPH Asia Conference Papers*, pages 1–11, 2023. 2

[23] S. M. H. Miangoleh, Z. Bylinskii, E. Kee, E. Shechtman, and Y. Aksoy. Realistic saliency guided image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 186–194, 2023. 2, 9

[24] C. Nguyen, S. DiVerdi, A. Hertzmann, and F. Liu. CollaVR: Collaborative in-headset review for VR video. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 267–277, 2017. 2

[25] C. Nguyen, S. DiVerdi, A. Hertzmann, and F. Liu. Vremiere: In-headset virtual reality video editing. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 5428–5438, 2017. 2

[26] S. Rothe, D. Buschek, and H. Hußmann. Guidance in cinematic virtual reality-taxonomy, research status and challenges. *Multimodal Technologies and Interaction*, 3(1):19, 2019. 2

[27] M. Seeger. Gaussian processes for machine learning. *International Journal of Neural Systems*, 14(02):69–106, 2004. 3

[28] J. Shen, J. Hu, J. J. Dudley, and P. O. Kristensson. Personalization of a mid-air gesture keyboard using multi-objective Bayesian optimization. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 702–710. IEEE, 2022. 2

[29] Y. Shen, Y. Shen, J. Cheng, C. Jiang, M. Fan, and Z. Wang. Neural canvas: Supporting scenic design prototyping by integrating 3D sketching and generative AI. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2024. 2

[30] X. Shi, M. Liu, Z. Zhou, A. Neshati, R. Rossi, and J. Zhao. Exploring interactive color palettes for abstraction-driven exploratory image colorization. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–16, 2024. 9

[31] S. Takeno, M. Nomura, and M. Karasuyama. Towards practical preferential Bayesian optimization with skew Gaussian processes. In *Proceedings of the International Conference on Machine Learning*, volume 202, pages 33516–33533. PMLR, 2023. 3

[32] A. Truong, S. Chen, E. Yumer, D. Salesin, and W. Li. Extracting regular fov shots from 360 event footage. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–11, 2018. 2

[33] E. Williams, C. Love, and M. Love. *Virtual reality cinema: narrative tips and techniques*. Routledge, 2021. 1

[34] Y. Xu, Z. Zhang, and S. Gao. Spherical DNNs and their applications in 360-degree images and videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):7235–7252, 2021. 9

[35] K. Yamamoto, Y. Koyama, and Y. Ochiai. Photographic lighting design with photographer-in-the-loop Bayesian optimization. In *Proceedings of the ACM Symposium on User Interface Software and Technology*, pages 1–11, 2022. 1, 2, 3, 6, 9

[36] Z. Yin, Z. Yang, M. Van De Panne, and K. Yin. Discovering diverse athletic jumping strategies. *ACM Transactions on Graphics*, 40(4):1–17, 2021. 2

[37] E. Yu, F. Chevalier, K. Singh, and A. Bousseau. 3D-Layers: Bringing layer-based color editing to VR painting. *ACM Transactions on Graphics (TOG)*, 43(4):1–15, 2024. 2

[38] L.-P. Yuan, B. Li, J. Wang, H. Qu, and W. Zeng. Generating virtual reality stroke gesture data from out-of-distribution desktop stroke gesture data. In *IEEE Conference Virtual Reality and 3D User Interfaces*, pages 732–742. IEEE, 2024. 4

[39] L.-P. Yuan, Z. Zhou, J. Zhao, Y. Guo, F. Du, and H. Qu. InfoColorizer: Interactive recommendation of color palettes for infographics. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):4252–4266, 2021. 9

[40] Y. Zhang, R. Wang, Y. Peng, W. Hua, and H. Bao. Color contrast enhanced rendering for optical see-through head-mounted displays. *IEEE Transactions on Visualization and Computer Graphics*, 28(12):4490–4502, 2021. 1, 6

[41] F. Zhong, G. A. Koulieris, G. Drettakis, M. S. Banks, M. Chambe, F. Durand, and R. K. Mantiuk. DiCE: Dichoptic contrast enhancement for VR and stereo displays. *ACM Transactions on Graphics*, 38(6):1–13, 2019. 2

[42] Q. Zhu, L. Yuan, Z. Xu, L. Yang, M. Xia, Z. Wang, H.-N. Liang, and X. Ma. From reader to experiencer: Design and evaluation of a VR data story for promoting the situation awareness of public health threats. *International Journal of Human-Computer Studies*, 181:103137, 2024. 2