Stage Light is Sequence^2: Multi-Light Control via Imitation Learning

Zijian Zhao[1], **Dian Jin**[2], **Zijing Zhou**[3], **Xiaoyu Zhang**[4] *
[1]The Hong Kong University of Science and Technology [2]The Hong Kong Polytechnic University
[3]The University of Hong Kong [4]City University of Hong Kong

Abstract

Music-inspired Automatic Stage Lighting Control (ASLC) has gained increasing attention in recent years due to the substantial time and financial costs associated with hiring and training professional lighting engineers. However, existing methods suffer from several notable limitations: the low interpretability of rule-based approaches, the restriction to single-primary-light control in music-to-color-space methods, and the limited transferability of music-to-controlling-parameter frameworks. To address these gaps, we propose SeqLight, a hierarchical deep learning framework that maps music to multi-light Hue-Saturation-Value (HSV) space. Our approach first customizes SkipBART, an end-to-end single primary light generation model, to predict the full light color distribution for each frame, followed by hybrid Imitation Learning (IL) techniques to derive an effective decomposition strategy that distributes the global color distribution among individual lights. In this stage, we formulate the light decomposition task as a Goal-Conditioned Markov Decision Process (GCMDP), construct an expert demonstration set inspired by Hindsight Experience Replay (HER), and introduce a three-phase IL training pipeline, achieving strong generalization capability. To validate our IL solution for the proposed GCMDP, we conduct quantitative analysis to compare model performance across different training phases, demonstrating that our design effectively improves performance and generalization capacity. Furthermore, we also conduct a human study to evaluate SeqLight by comparing it with competitive baselines on music-conditioned light generation tasks across different music styles. The results show that SeqLight achieves the best overall preference scores in both in-domain and out-of-domain settings. The code and trained parameters of this paper is provided at the anonymous repository `https://github.com/RS2002/SeqLight`.

1 Introduction

Stage lighting plays a critical role in live music performances, shaping the experience of both performers and audiences. In recent years, Automatic Stage Lighting Control (ASLC) has gained increasing attention for its potential to lower the costs of professional lighting staffing and to empower amateurs in creating expressive stage environments.

Existing ASLC methods can be broadly categorized into two main approaches: rule-based solutions and end-to-end solutions. Rule-based methods [1; 2; 3] typically first segment music pieces into several categories based on attributes such as style, emotion, and chords, and then map each category to predefined light patterns. However, these approaches suffer from limited interpretability [4; 5] and are constrained by the coarseness and accuracy of the underlying classification models (For instance, the emotion classification task involves only four categories, yet a range of methods achieve accuracies below 80%, as reported in [6]). As an alternative, end-to-end solutions frame ASLC as a content

*Corresponding Author: Xiaoyu Zhang (xiaoyu.zhang@cityu.edu.hk)

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

generation task and learn directly from real-world data produced by lighting engineers. For example, Skip-BART [4] is trained on a live video dataset Rock, Punk, Metal, and Core - Livehouse Lighting (RPMC-L^2), mapping music to light hue and value within each frame.[2] However, these methods are limited to single primary light control and overlook the variability of lighting configurations across different venues. More recently, [7] proposed training a model to directly map music to control parameters (DMX) for each individual light. However, this approach lacks transferability across venues with differing setups, and the cost of collecting professional control data for each venue remains prohibitive. More detailed reviews to related works could be found at Appendix A.

To address these issues, we propose SeqLight, the first color-space multi-light generation solution. Our method decomposes the problem into two sub-tasks: first, we customize Skip-BART to predict the full hue and value distribution of all lights within each frame; second, we train a light distribution decomposition model for each venue using Imitation Learning (IL). This decoupled hierarchical design offers two key advantages: (i) The first stage is independent of venue-specific lighting configurations and relies solely on video data, enabling mixed training across different venues, which is a particularly valuable property for the Music Information Retrieval (MIR) field with limited datasets. (ii) The second stage operates independently of music, eliminating the need for professional lighting engineers in data collection. Specifically, we formulate the light decomposition task as a Goal-Conditioned Markov Decision Process (GCMDP) and introduce a Hindsight Experience Replay (HER)-inspired [8] method to collect expert trajectories for IL using only the mixed light data itself. Overall, the proposed method simultaneously addresses the data scarcity and low transferability issues of music-to-controlling-parameter approaches, while achieving multi-light generation, marking a significant advancement in music-to-color-space methods. We evaluate our methodology through both quantitative analysis and human evaluation, and our solution demonstrates consistently promising performance across domains. The main contributions of this paper can be summarized as:

- We propose SeqLight, the first color-space multi-light ASLC method. Specifically, we adapt Skip-BART to predict frame-level hue and value distributions for all lights, and formulate the task of decomposing this global distribution into individual light values and hues as a GCMDP. Additionally, we incorporate each light's previous frame state as a constraint to prevent overshooting and enhance practical control stability.
- We introduce a hybrid IL pipeline that eliminates the need for handcrafted reward functions while learning a competent policy for the proposed GCMDP. Specifically, we pre-train the policy using Behavioral Cloning (BC), then apply Adversarial Inverse Reinforcement Learning (AIRL) to learn a reward model, and finally fine-tune the policy in more complex scenarios. We also propose a HER-inspired method for generating expert trajectories by deriving goals from real light observations, and introduce a model enhancement technique that predicts hue and value distributions from generated lights as an Auxiliary (AUX) loss. Furthermore, we identify a limitation in conventional AIRL with Actor-Critic (AC) policies, namely the difficulty of training the critic due to the constantly evolving reward model, and address this by incorporating Group Relative Policy Optimization (GRPO), which replaces the critic-based advantage with a group-relative advantage.
- We conduct both quantitative analysis and human evaluation to validate our proposed method. The quantitative analysis demonstrates the effectiveness of our proposed three-phase IL framework with GRPO, which improves model performance and generalization capacity. Human evaluation shows that SeqLight achieves the highest overall preference scores in both in-domain and out-of-domain settings. Specifically, our method outperforms the best comparison object 16.4% and 13.5% in the two settings, respectively.

2 Music-Inspired Multi-Light Generation

The overall workflow of SeqLight is illustrated in Fig. 1. First, the modified Skip-BART [4] generates the full hue and value distributions for all lights, conditioned on the music and previously generated results. The resulting light distribution is then decomposed into per-light hue and value controls via a GCMDP For simplicity, we assume in this paper that light positions and the control order within each frame are fixed in advance, and that saturation is fixed at 100%, i.e., we consider only pure colors.

[2]Note that saturation is fixed at 100%, as pure colors are predominantly used in practice, which is also adopted in this paper.

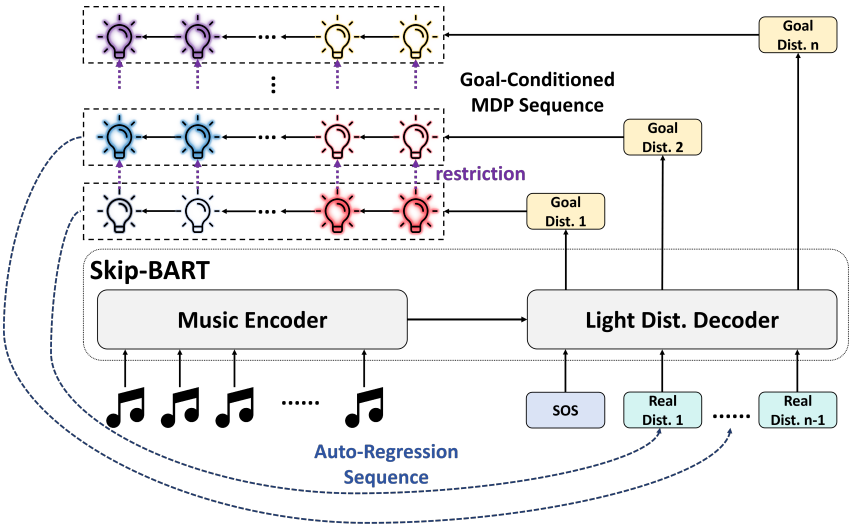


Figure 1: Workflow

To begin, we emphasize the motivation behind this two-stage hierarchical design. First, to ensure high rationality and interpretability [4], we choose to learn from labeled data produced by human lighting engineers rather than relying on rule-based mapping. A naive approach to multi-light control would involve collecting labeled data for each individual light and training an end-to-end model, as in prior music-to-controlling-parameter methods. However, such an approach suffers from low transferability and high data collection costs. Our two-stage solution successfully overcomes these challenges: (i) For the first stage, we observe that in performances involving musicians with professional lighting engineers, the lighting is designed specifically for each piece of music and exhibits similar visual characteristics even across different venues. A real-world example is provided at Appendix G.1.

We now formally define the music-inspired multi-light generation task. For a given music piece consisting of $\mathcal{T}$ frames $X = \{x_1, x_2, \ldots, x_\mathcal{T}\}$, our goal is to find a light control sequence $\mathcal{U} = \{U_1, U_2, \ldots, U_\mathcal{T}\}$ that best approximates the real full-light distributions $Y = \{y_1, y_2, \ldots, y_\mathcal{T}\}$. At the j-th frame, U_j is defined as $U_j = [u_{1,j}, u_{2,j}, \ldots, u_{n,j}]$, where n is the number of lights. For the i-th light, the control action is $u_{i,j} = [h_{i,j}, v_{i,j}]$, with hue $h_{i,j} \in [0, 2\pi)$ and value $v_{i,j} \in [0, 1]$. The ground-truth for frame j, denoted y_j, consists of a hue distribution $H_j \in [0,1]^{b_h}$ and a value distribution $V_j \in [0,1]^{b_v}$, where hue and value are discretized into b_h and b_v bins, respectively, with $\sum_{i=1}^{b_h} H_{j,i} = \sum_{i=1}^{b_v} V_{j,i} = 1$. Our optimization objective is then defined as:

$$\begin{aligned} &\min_{\Theta} \mathbb{E}_{\mathcal{U}}\big[\mathrm{Dist}\big(\mathrm{Mix}(\mathcal{U})||y_j\big)\big], \\ \text{s.t.} \quad &U_j = \mathrm{F}\big(X, U_{:j-1}; \Theta\big), \end{aligned} \tag{1}$$

where $\mathrm{F}(\cdot,\cdot;\Theta)$ is the mapping function parameterized by Θ, $\mathrm{Mix}(\cdot)$ denotes the mixed (aggregated) hue and value distribution induced by the per-light controls U_j, and $\mathrm{Dist}(\cdot||\cdot)$ is a chosen distributional distance (e.g., L1, Wasserstein, KL). Further details are provided in Appendix B.

Although it may appear possible to directly train a network by deriving a loss function from Eq. (1) and applying gradient descent, this approach is infeasible in practice. First, the solution is not unique: many different combinations of per-light controls can yield the same aggregated distribution, making the learning problem more challenging than standard supervised regression. Second, even if the light-mixing process could be simulated, it often involves complex and potentially non-differentiable operations, which hinder end-to-end gradient-based optimization. These challenges further underscore the necessity of our proposed hierarchical solution.

2.1 Training Process

In the first stage, we adapt the original Skip-BART to predict the light distributions for each frame. We employ KL divergence as the distance metric and define the supervised loss as

$$\begin{aligned} L_{\text{sup}}^{\phi} &= \mathbb{E}_{B^t}\big[\mathrm{KL}\big(\hat{H}_j \,\|\, H_j\big) + \mathrm{KL}\big(\hat{V}_j \,\|\, V_j\big)\big], \\ \hat{H}_j, \hat{V}_j &= \text{Skip-BART}\big(X,\ [\hat{H}_{:j-1},\ \hat{V}_{:j-1}]; \phi\big), \end{aligned} \tag{2}$$

where ϕ denotes the network parameters, B^t represents training set, and the direction of the KL divergence is chosen following the principle of Variational Autoencoders (VAE) [9]. Concretely, we modify Skip-BART by replacing its input and output projection layers with new MLPs to match the dimensionality of the full light distributions (as opposed to the single primary light predicted by the original model). For the backbone, we initialize ϕ using the pretrained Skip-BART weights from [4], which reduces training time and improves convergence via transfer learning [10].

In the second stage, we decompose the predicted hue and value distributions into per-light controls within each frame by formulating the task as a GCMDP and solving it with Reinforcement Learning (RL). The objective is to maximize the expected cumulative reward:

$$\max_{\theta} \mathscr{J}(\theta) = \max_{\theta} \mathbb{E}_{\pi_\theta}\Big[\sum_{t=1}^{n} \gamma^{t-1} \mathscr{R}(s_t, a_t, g)\Big], \tag{3}$$

where $\mathscr{R}(\cdot,\cdot,\cdot)$ is the reward function, π_θ is the policy parameterized by θ, s_t and a_t are the state and action at time t, g denotes the goal, and γ is the discount factor. In our task, it is straightforward to evaluate whether a complete trajectory is good by comparing the final aggregated distribution to the goal, but it is difficult to design informative stepwise reward signals. This reward sparsity hinders efficient RL training. To address this, we adopt a hybrid IL approach: we first generate expert trajectories using HER, then pre-train the policy with BC, next employ AIRL [11] to learn a reward model from expert and policy trajectories, and finally fine-tune the policy using the learned reward to enable generalization beyond the expert demonstrations. Further details of our goal-conditioned light decomposition method are provided in Section 3.

2.2 Framework Pipeline

After obtaining the well-trained Skip-BART for full-light distribution generation and the light-decomposition policy, we use SeqLight to generate lights as illustrated in Fig. 1. At each frame j, Skip-BART first predicts the hue and value distributions conditioned on the music and the previously generated frames:

$$\begin{aligned} \hat{H}_j, \hat{V}_j &= \text{Skip-BART}\big(X,\ [\tilde{H}_{:j-1},\ \tilde{V}_{:j-1}]; \phi^*\big), \\ \tilde{H}_k,\ \tilde{V}_k &= \mathrm{Mix}(U_k), \end{aligned} \tag{4}$$

where $\tilde{H}_k, \tilde{V}_k$ denote the aggregated hue and value distributions in frame k resulting from the mixing function $\mathrm{Mix}(\cdot)$.

A naive approach to obtain U_j is to sample directly from π_{θ^*} conditioned on the goal $g_j = [\hat{H}_j, \hat{V}_j]$. However, this approach has two drawbacks: (i) it may lead to overshooting of individual lights' previous states; and (ii) it can produce temporally inconsistent control sequences across frames, since many different per-light decompositions can realize the same goal distribution. To mitigate these issues, we adopt a last-frame state constrained sampling strategy:

$$\begin{aligned} h_{i,j}, v_{i,j} &\sim \pi_{\theta^*}(s_{i,j}, g_j, \iota), \\ \text{s.t.} \quad & \mathrm{D}_h\big(h_{i,j} \,\|\, h_{i,j-1}\big) < d^h, \\ & \mathrm{D}_v\big(v_{i,j} \,\|\, v_{i,j-1}\big) < d^v, \end{aligned} \tag{5}$$

where $s_{i,j}$ is the state for light i at step j, ι is a sampling temperature parameter that controls diversity, d^h and d^v are the maximum allowable changes for hue and value to prevent overshooting and ensure temporal smoothness, and $\mathrm{D}_h, \mathrm{D}_v$ are distance metrics (we adopt the same metrics used in [4], detailed in Appendix B).

3 Goal-Conditioned Light Decomposition

3.1 Problem Setup

We formulate the light-decomposition task as a GCMDP, defined by the tuple $\langle S, A, R, P, \gamma, G, \rho_g \rangle$, where S, A, R, P, and γ denote the state space, action space, reward function, transition function, and discount factor, respectively, as in a conventional MDP, while G and ρ_g represent the goal space and the goal distribution.

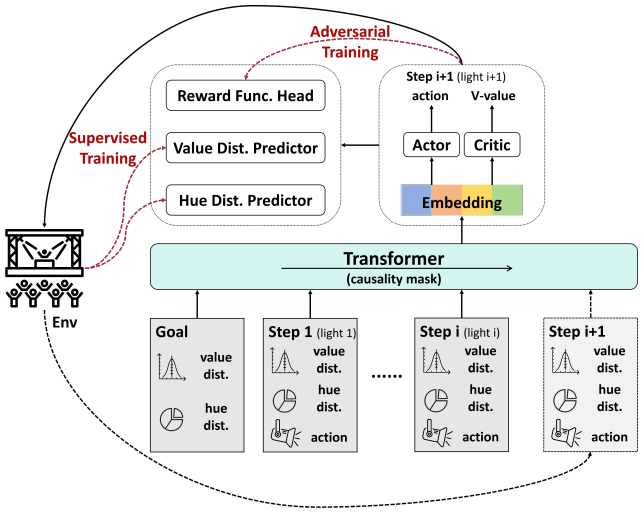


Figure 2: Network Architecture

(i) State: At each step t, the state s_t is defined as $s_t = [s_{:t-1}, a_{t-1}, \mathrm{Mix}(a_{:t-1})]$, where $s_{:t-1}$ and a_{t-1} comprise the history of states and actions up to step $t-1$, and $\mathrm{Mix}(a_{:t-1})$ denotes the aggregated hue and value distribution generated by the first $t-1$ actions.

(ii) Action: The action at step t, denoted a_t, corresponds to the per-light control U introduced in the previous section and consists of hue and value components: $a_t = [a_t^h, a_t^v]$. In this work, we adopt a stochastic policy and sample actions according to $a_t \sim \pi_\theta(s_t, g)$. Since hue is circular with domain $[0, 2\pi)$, we model it using a Von Mises distribution. For value, which lies in $[0, 1]$, we employ a Beta distribution. The detailed formal definition is provided in Appendix B.

(iii) Reward function: As discussed previously, designing a stepwise reward for this task is challenging. Therefore, we parameterize the reward as a neural network $\mathscr{R}_\Phi(s_t, a_t, g) : S \times A \times G \to \mathbb{R}$ with parameters Φ. The reward model is learned via AIRL, which will be described in detail later.

(iv) State transition function: Although we adopt a model-free RL approach in this paper, the environment transition $\mathscr{P}(\cdot \mid s_t, a_t)$ is actually deterministic and known: given an action, the resulting aggregated hue and value distribution can be computed, and the next state is formed accordingly. We exploit this physical regularity by incorporating it as an AUX loss to enhance feature learning in both the policy and reward networks.

(v) Goal and goal distribution: The goal is defined as $G = [H, V]$, i.e., the target full-light hue and value distribution that the aggregated per-light controls are intended to match. We consider two types of goal distributions: ρ_g^e, corresponding to expert trajectories, and ρ_g^a, corresponding to arbitrary goals. Further details are provided in the following sections.

In this paper, we use $\mathscr{V}$, $\mathscr{Q}$, and $\mathscr{A}$ to denote the state-value function, action-value function, and advantage function, respectively, for standard goal-conditioned RL. Their formal definitions are provided in Appendix B.

To solve the proposed GCMDP, we adopt a Transformer-based network [12], as illustrated in Fig. 2. Below, we summarize the network architecture, with further details provided in Appendix C.

$$a_t \sim \pi_\theta(s_t, g), \quad \Upsilon_t = \mathscr{V}_\psi(s_t, g), \quad r_t = \mathscr{R}_\Phi(s_t, a_t, g), \quad (\mathcal{H}_t, \mathcal{V}_t) = \mathrm{P}_\Psi(s_t, a_t), \tag{6}$$

where Υ_t denotes the estimated state value, $\mathcal{H}_t$ and $\mathcal{V}_t$ denote the predicted hue and value distributions after executing the action, and θ, ψ, Φ, Ψ are associated with different network heads but share the same Transformer backbone. Note that although all outputs implicitly depend on (s, a, g), we retain only the relevant arguments in each function definition to avoid notational clutter.

3.2 HER-Inspired Expert Trajectory Generation

In our method, expert trajectories are generated following the HER principle [8]. We first sample random hue and value actions for each light to obtain a trajectory $\langle s_1, a_1, s_2, a_2, \ldots, s_n, a_n \rangle$. The goal for this expert trajectory is then labeled as the aggregated result of all lights, i.e., the optimal goal $g = \mathrm{Mix}(a_{:n})$. During training, we also generate additional expert trajectories by relabeling the goals of trajectories sampled from the current policy. Specifically, we denote the expert dataset as $B^e = \{\tau_1^e, \tau_2^e, \ldots, \tau_N^e\}$, where τ_i^e is a trajectory annotated with its goal and N is the dataset size.

The goals of expert trajectories therefore originate from real mixed lights and follow the distribution ρ_g^e. In contrast, during the light control phase (Section 2.2), the goal is provided by the Skip-BART predictor, which is trained on noisy real-world data. Moreover, prediction errors introduce bias, causing these goals to deviate from ρ_g^e. To account for this discrepancy, we introduce an alternative goal distribution ρ_g^a, in which hue and value are sampled completely at random rather than computed from real lights. After training the policy via IL on expert goals sampled from ρ_g^e, we fine-tune it on arbitrary goals drawn from ρ_g^a to enhance robustness and generalization capacity.

It is worth spelling out why the relabeled trajectories are valid expert demonstrations, since they are the only source of Stage-2 supervision. The core idea of HER is that a trajectory, whether or not it reaches its intended goal, is a perfect demonstration for the outcome it actually achieves. We exploit this directly: we sample random hue and value actions for each light, pass them through the environment, and take the aggregated histogram of the resulting lights as the goal. The goal is thus defined a posteriori as exactly what these actions produce, so the trajectory matches its goal by construction and succeeds with probability one. It is therefore an optimal expert trajectory for that goal. The optimal trajectory for a given goal is not unique, since different per-light assignments can mix to the same aggregate distribution, but the relabeled trajectory is guaranteed to be one of them. This gives an unlimited supply of perfectly labeled expert data with no professional annotation, which is what makes cheap per-venue behavioral cloning possible.

3.3 Training Process

3.3.1 Phase 1: Policy Pre-Training by BC

To initialize the policy π_θ with expert behavior, we pre-train it via BC by maximizing the likelihood of expert actions (equivalently, minimizing the negative log-likelihood). The BC loss is defined as:

$$L_{\mathrm{bc}}^{\theta} = -\mathbb{E}_{B^e}\big[\log \pi_\theta(a \mid s, g)\big]. \tag{7}$$

Additionally, we incorporate an AUX loss from the hue and value predictor to enhance the model's feature extraction and transition modeling capabilities. This AUX loss follows the same supervised formulation used in Skip-BART (Eq. (2)):

$$L_{\mathrm{aux}}^{\Psi} = \mathbb{E}_{B^e}\Big[\mathrm{KL}\big(\mathcal{H}_t \,\|\, \mathrm{Mix}(a_{1:t}).\mathrm{hue}\big) + \mathrm{KL}\big(\mathcal{V}_t \,\|\, \mathrm{Mix}(a_{1:t}).\mathrm{value}\big)\Big], \tag{8}$$

where $\mathcal{H}_t, \mathcal{V}_t$ come from Eq. 6. In summary, the combined loss for Phase 1 is defined as:

$$\mathcal{L}_1^{\theta,\Psi} = L_{\mathrm{bc}}^{\theta} + \eta\, L_{\mathrm{aux}}^{\Psi}, \tag{9}$$

where $\eta \geq 0$ is a weighting coefficient.

3.3.2 Phase 2: Reward Model Training by AIRL

In AIRL, we alternately train a discriminator and a policy. Following the Generative Adversarial Network - Guided Cost Learning (GAN-GCL) framework [13], the discriminator is defined as:

$$\mathscr{D}(\tau) = \frac{\exp(\mathscr{F}(\tau))}{\exp(\mathscr{F}(\tau)) + \pi(\tau)}, \tag{10}$$

where $\mathscr{D}(\tau)$ denotes the probability that trajectory τ originates from the expert, and $\mathscr{F}(\cdot)$ is a learned scoring function (note that τ may be of arbitrary length). The reward induced by the discriminator is given by:

$$\mathscr{R}(\tau) = \log \mathscr{D}(\tau) - \log\big(1 - \mathscr{D}(\tau)\big) = \mathscr{F}(\tau) - \log \pi(\tau), \tag{11}$$

which can be rearranged into the equivalent form:

$$\mathscr{D}(\tau) = \frac{1}{1+\exp(-\mathscr{R}(\tau))} = \text{Sigmoid}\big(\mathscr{R}(\tau)\big). \tag{12}$$

Since the reward function is parameterized by Φ, we write the discriminator as $\mathscr{D}_\Phi(\cdot)$. The discriminator loss follows the standard GAN objective [14]:

$$L^{\Phi}_{\text{dis}} = -\mathbb{E}_{B^e}\big[\log \mathscr{D}_\Phi(s,a,g)\big] - \mathbb{E}_{\pi_\theta}\big[\log\big(1-\mathscr{D}_\Phi(s,a,g)\big)\big], \tag{13}$$

where we use (s,a,g) rather than (s,a,s',g) because our state transitions are deterministic, and s' denotes the next state.

The policy is trained alternately with the discriminator. For our stochastic continuous control problem, we adopt PPO [15] as a common choice. The PPO actor and critic losses are defined using the policy-gradient objective and temporal-difference error, respectively:

$$\begin{aligned}
L^{\theta}_{\text{actor-PPO}} &= \mathbb{E}_{\pi_{\theta^-}}\Big[\min\Big(\frac{\pi_\theta(a|s)}{\pi_{\theta^-}(a|s)}\,\tilde{\mathscr{A}}_{\psi,\Phi}(s,a,g),\ \text{CLIP}\,\big(\frac{\pi_\theta(a|s)}{\pi_{\theta^-}(a|s)}, 1-\epsilon, 1+\epsilon\big)\,\tilde{\mathscr{A}}_{\psi,\Phi}(s,a,g)\Big)\Big],\\
L^{\psi}_{\text{critic-PPO}} &= \mathbb{E}_{\pi_{\theta^-}}\Big[\big(\mathscr{R}_\Phi(s,a,g) + \gamma \mathscr{V}_\psi(s',g) - \mathscr{V}_\psi(s,g)\big)^2\Big],
\end{aligned} \tag{14}$$

where π_{θ^-} is the behavior policy used for data collection, ϵ is the PPO clipping hyper-parameter, and $\tilde{\mathscr{A}}_{\psi,\Phi}$ denotes the Generalized Advantage Estimation (GAE) [16] computed using the value function $\mathscr{V}_\psi$ and the reward model $\mathscr{R}_\Phi$. The overall loss for the PPO-based Phase 2 is therefore:

$$\mathcal{L}^{\Phi,\theta,\psi,\Psi}_{\text{2-PPO}} = L^{\Phi}_{\text{dis}} + \mathbb{E}_{g\sim\rho^e_g}\big[L^{\theta}_{\text{actor-PPO}} + L^{\psi}_{\text{critic-PPO}}\big] + \delta\, L^{\theta}_{\text{bc}} + \eta\, L^{\Psi}_{\text{aux}}, \tag{15}$$

where $\delta, \eta \geq 0$ are weighting coefficients, and L^{θ}_{bc} and L^{Ψ}_{aux} serve as auxiliary losses in this phase. (Note that, except for the policy loss in Phase 3, all other components use goals drawn exclusively from the expert distribution ρ^e_g, which is already incorporated into the respective loss terms.)

However, in the PPO-based solution, we observe strong interactions among the reward model, critic, and actor during this phase. If the critic fails to keep pace with changes in the reward model (whose scale may drift during training), actor updates can become misdirected and unstable, which in turn degrades the reward model (since the policy also influences the reward defined in Eq. (11)). To mitigate this issue, we adopt Group Relative Policy Optimization (GRPO) [17] as an alternative. For each goal, GRPO samples multiple trajectories and replaces the advantage with a group-relative reward-to-go. The GRPO-style advantage function, denoted as $\overline{\mathscr{A}}$, is formally defined in Appendix B. Then the GRPO actor loss replaces the PPO advantage with this group-relative advantage:

$$L^{\theta}_{\text{actor-GRPO}} = \mathbb{E}_{\pi_{\theta^-}}\Big[\min\Big(\frac{\pi_\theta(a|s)}{\pi_{\theta^-}(a|s)}\,\overline{\mathscr{A}}_\Phi(s,a,g),\ \text{CLIP}\,\big(\frac{\pi_\theta(a|s)}{\pi_{\theta^-}(a|s)}\big), 1-\epsilon, 1+\epsilon\big)\,\overline{\mathscr{A}}_\Phi(s,a,g)\Big)\Big]. \tag{16}$$

By eliminating the critic and using group-relative returns, GRPO reduces sensitivity to reward scaling and focuses updates on trajectories that perform better within the sampled group. The overall loss for the GRPO-based Phase 2 is therefore:

$$\mathcal{L}^{\Phi,\theta,\Psi}_{\text{2-GRPO}} = L^{\Phi}_{\text{dis}} + \mathbb{E}_{g\sim\rho^e_g}\big[L^{\theta}_{\text{actor-GRPO}}\big] + \delta\, L^{\theta}_{\text{bc}} + \eta\, L^{\Psi}_{\text{aux}}. \tag{17}$$

3.3.3 Phase 3: Policy Fine-Tuning by RL

In Phase 3, we aim to further improve the policy's generalization capacity, particularly for goals outside the expert trajectory distribution. To this end, we freeze the reward model and train only the policy network. Since the policy and reward model share the same Transformer backbone, we must fix the backbone parameters to preserve the reward model unchanged. Let θ' and ψ' denote the parameters of the policy and critic, respectively, that are external to the shared Transformer backbone. The loss functions for this phase are therefore:

$$\begin{aligned}
\mathcal{L}^{\theta',\psi'}_{\text{3-PPO}} &= \mathbb{E}_{g\sim\rho^a_g}\big[L^{\theta'}_{\text{actor-PPO}} + L^{\psi'}_{\text{critic-PPO}}\big] + \delta\, L^{\theta'}_{\text{bc}},\\
\mathcal{L}^{\theta'}_{\text{3-GRPO}} &= \mathbb{E}_{g\sim\rho^a_g}\big[L^{\theta'}_{\text{actor-GRPO}}\big] + \delta\, L^{\theta'}_{\text{bc}}.
\end{aligned} \tag{18}$$

Note that since the backbone parameters are fixed in this phase, adding the AUX loss is neither necessary nor effective.

3.4 Post-Processing

In the light decomposition task, determining the value for each light requires considering both how to achieve the target hue distribution and how scaling a light's value affects its contribution to the overall mixture. Fortunately, scaling the values of all lights by a common factor preserves the relative hue distribution of the full light set. Leveraging this property, we propose a post-processing method that introduces a scaling factor f to adjust the generated light values, aiming to make the resulting mixture's value distribution as close as possible to the goal value distribution. This can be formulated as the following optimization problem:

$$\begin{aligned} &\min_{f\geq 0} \text{Dist}\left(\text{Mix}\left([a_1^h, a_1^v f], [a_2^h, a_2^v f], \ldots, [a_n^h, a_n^v f]\right).\text{value} \parallel g^v\right), \\ \text{s.t.} \quad &0 \leq a_i^v f \leq 1 \quad \forall i = 1, 2, \ldots, n, \end{aligned} \tag{19}$$

where $g = [g^h, g^v]$ denotes the target hue and value distributions.

4 Experiment

To validate our proposed method, we conducted both quantitative experiments and a human study, comparing our approach with several baselines and ablation variants. Specifically, we train the Skip-BART model for full light distribution prediction using the RPMC-L^2 dataset [4]. To avoid information leakage, we adopt the same training and testing set division as the original Skip-BART [4]. For the light decomposition component, we utilize a simulation environment equipped with eight circular point lights. Further details regarding the experimental setup, including the simulation configuration, dataset information, and model parameters, are provided in Appendix D.

4.1 Quantitative Analysis

In this section, we first illustrate the efficiency of our goal-conditioned light decomposition method (Section 3). We consider two different types of goals: the expert-based goal (In-Domain, ID), which involves sampling the hue and value distribution from a mixture of real lights, and the randomly generated goal (Out-Of-Domain, OOD), which involves directly generating random distributions for both hue and value. We compare the model performance under different phases in training using PPO [15] and GRPO [17] as policy strategies, respectively. The evaluation metrics include L1 distance, Wasserstein distance, JS divergence, KL divergence, Bhattacharyya distance, and cosine similarity. Based on the results in Table 1, we observe that our proposed Phase 3 (GRPO) achieves the best performance in the OOD scenario, while Phase 2 (GRPO) performs best in the ID scenario. This suggests that although Phase 3 enhances generalization capacity, it also sacrifices some degree of overfitting capability. A more detailed experimental analysis along with ablation studies can be found in Appendix E.

4.2 Human Evaluation

To better assess the alignment of our results with human preferences, we conducted a human evaluation study. Specifically, we recruited 30 participants to rate six music pieces paired with lighting sequences generated under different styles. Each piece was evaluated across six metrics [18], with scores ranging from 1 to 7 (higher scores indicating better quality). The lighting conditions were generated from four sources: Ground Truth, SeqLight, Skip-BART [4], and a rule-based method. To evaluate the generalization capability of our approach, we selected three music pieces from the test set of RPMC-L^2 [4] (ID group), covering rock, metal, and core genres. Additionally, we included three pieces from the OOD experiment of Skip-BART, which were generated by Suno [19] (OOD group). Note that the OOD group does not have corresponding ground-truth lighting. The results of the human evaluation are presented in Table 2, demonstrating that our proposed method consistently achieves promising performance across different music styles and evaluation dimensions. On the ID group, SeqLight attains a quality level comparable to the lighting designed by professional engineers: its overall score of 4.54 ± 0.88 shows no statistically significant difference from Ground Truth, while it clearly exceeds Skip-BART (3.90 ± 0.84) and the rule-based method (2.70 ± 1.26). It achieves the highest mean score on Emotion, Impact, Rhythm, Atmosphere, Surprise, and Overall, and stays within a few hundredths of Ground Truth on Smoothness. Pairwise comparisons (Table 13) further

Table 1: Model Performance on Goal-Conditioned Light Decomposition Task: The best results are highlighted in **bold**, while the second-best results are indicated with underlining. This notation is consistent throughout the following tables. (Mean ± Standard Deviation (M ± SD))

Model	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
	ID	OOD	ID	OOD	ID	OOD
Hue						
Phase 1	3.58±1.01	3.20±0.76	5.61±5.70	7.99±0.60	3.18±1.25	2.80±0.93
Phase 2 (GRPO)	2.66±0.80	2.99±0.59	4.97±4.96	6.89±0.50	2.17±0.93	2.71±0.93
Phase 3 (GRPO)	2.73±0.09	**2.59±0.89**	5.19±5.58	7.54±5.75	2.23±1.08	**2.19±1.11**
Phase 2 (PPO)	**2.52±0.87**	2.70±0.58	**4.73±5.51**	**6.93±5.22**	**2.03±1.00**	2.39±0.19
Phase 3 (PPO)	2.74±0.84	3.18±0.53	4.80±4.57	7.16±4.34	2.20±1.02	2.87±0.72
Value						
Phase 1	10.21±3.06	11.25±1.46	5.98±3.38	6.87±0.83	1.67±0.67	2.44±0.43
Phase 2 (GRPO)	8.63±3.04	9.40±1.82	5.05±3.32	**5.76±1.30**	1.31±0.65	1.87±0.48
Phase 3 (GRPO)	9.24±3.32	**9.14±2.16**	5.32±3.44	5.78±0.79	1.50±0.71	**1.85±0.53**
Phase 2 (PPO)	**8.07±2.62**	10.73±2.20	**4.74±3.12**	6.33±1.36	**1.21±0.55**	2.31±6.55
Phase 3 (PPO)	9.70±3.26	11.50±2.53	5.60±3.45	7.08±1.13	1.52±0.70	2.57±9.29

Model	KL ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
	ID	OOD	ID	OOD	ID	OOD
Hue						
Phase 1	1.66±1.56	1.15±0.52	5.79±2.88	4.74±2.00	6.64±1.45	6.61±1.33
Phase 2 (GRPO)	3.78±4.64	1.98±0.89	3.49±1.72	4.45±2.13	7.13±1.29	6.23±1.36
Phase 3 (GRPO)	**1.65±2.33**	**1.05±0.66**	3.67±2.06	**3.61±2.17**	7.53±1.24	**7.19±1.40**
Phase 2 (PPO)	2.58±3.46	1.51±5.00	3.19±1.87	3.87±1.74	**7.56±1.21**	7.18±1.19
Phase 3 (PPO)	3.77±4.93	1.93±6.65	**3.05±2.09**	4.69±1.64	7.11±1.27	6.42±1.64
Value						
Phase 1	0.58±0.24	0.86±0.17	2.46±1.14	3.78±0.88	8.00±0.91	6.97±0.60
Phase 2 (GRPO)	0.45±0.23	**0.65±0.19**	1.85±1.07	2.78±0.88	**8.27±0.86**	**7.66±0.66**
Phase 3 (GRPO)	0.52±0.25	0.66±0.21	2.17±1.20	**2.70±0.98**	8.09±0.89	7.45±0.74
Phase 2 (PPO)	**0.44±0.20**	0.82±0.26	**1.64±1.09**	3.60±1.26	8.15±0.75	7.13±0.85
Phase 3 (PPO)	0.52±0.25	0.95±0.45	2.23±1.17	4.28±2.45	8.15±0.91	6.79±1.24

show that SeqLight significantly outperforms Skip-BART in Impact, Rhythm, Surprise, and Overall, and consistently surpasses the rule-based method across all metrics. On the OOD group, SeqLight also achieves the highest scores across all metrics, with an overall score of 3.94 ± 1.32 compared with 3.47 ± 1.01 for Skip-BART and 2.70 ± 1.36 for the rule-based method. These results indicate that SeqLight captures music-lighting correspondence better and maintains stronger generalization ability across unseen music styles. More details regarding the human evaluation setup and complete results can be found in Appendix F.

5 Conclusion

In this paper, we propose SeqLight, the first music-to-color-space multi-light ASLC method. Our approach addresses the challenges of low transferability and dataset scarcity through a hierarchical design. First, we train Skip-BART on mixed-venue live videos to predict the full HV distribution of the lights. Subsequently, we employ IL to derive an effective decomposition strategy that maps the predicted distribution to individual light controls. This decomposition task is formulated as a GCMDP and trained independently for each venue, with expert data collected via simple light mixing and goal labeling using HER. Both quantitative analysis and human evaluation demonstrate the generalization and efficiency of the proposed method across both in-domain and out-of-domain settings. Further discussions can be found in Appendix H.

Acknowledgement

We are grateful to the participants who volunteered for our human study and the anonymous reviewers who helped improve the quality of this paper.

Table 2: Human Evaluation Scores

Method	Emotion	Impact	Rhythm	Smoothness	Atmosphere	Surprise	Overall
ID Evaluation							
Ours	4.27±0.98	**4.83±1.02**	**4.80±1.04**	4.47±1.04	**4.40±0.96**	**4.48±1.10**	**4.54±0.88**
Ground Truth	**4.46±1.03**	4.20±1.03	4.56±0.90	**4.62±0.81**	4.32±0.83	4.13±0.68	4.38±0.74
Skip-BART	4.06±0.98	3.90±0.91	4.01±0.95	3.91±1.13	4.02±0.97	3.51±1.00	3.90±0.84
Rule-based	3.29±1.39	2.82±1.54	2.43±1.37	2.56±1.26	2.77±1.48	2.36±1.44	2.70±1.26
OOD Evaluation							
Ours	**3.72±1.50**	**4.36±1.47**	**3.96±1.32**	**4.08±1.42**	**3.86±1.45**	**3.66±1.44**	**3.94±1.32**
Skip-BART	3.57±1.05	3.38±1.03	3.69±1.15	3.60±1.14	3.38±1.12	3.19±1.11	3.47±1.01
Rule-based	3.06±1.52	2.66±1.52	2.50±1.42	2.47±1.47	2.94±1.61	2.57±1.53	2.70±1.36

References

[1] P. Mabpa, T. Sapaklom, E. Mujjalinvimut, J. Kunthong, and P. N. N. Ayudhya, "Automatic chord recognition technique for a music visualizer application," in *2021 9th International Electrical Engineering Congress (iEECON)*, pp. 416–419, IEEE, 2021.

[2] E. O. Bonde, E. K. Hansen, and G. Triantafyllidis, "Auditory and visual based intelligent lighting design for music concerts," *Eai Endrosed Trasactions on Creative Technologies*, vol. 5, no. 15, p. e2, 2018.

[3] Y.-P. Liao, D.-C. Chen, and B.-H. Chen, "Automatic visual effect adjustment system," in *2023 International Automatic Control Conference (CACS)*, pp. 1–6, IEEE, 2023.

[4] Z. Zhao, D. Jin, Z. Zhou, and X. Zhang, "Automatic stage lighting control: Is it a rule-driven process or generative task?," in *The Fourteenth International Conference on Learning Representations*, 2026.

[5] J. McDonald, S. Canazza, A. Chmiel, G. De Poli, E. Houbert, M. Murari, A. Rodà, E. Schubert, and J. D. Zhang, "Illuminating music: Impact of color hue for background lighting on emotional arousal in piano performance videos," *Frontiers in Psychology*, vol. 13, p. 828699, 2022.

[6] Z. Zhao, "Let network decide what to learn: Symbolic music understanding model based on large-scale adversarial pre-training," in *Proceedings of the 2025 International Conference on Multimedia Retrieval*, pp. 2128–2132, 2025.

[7] T. Wang, Y. Jiang, W. Jiang, X. Zhou, and X. Guan, "Lightinggen: A dmx based generation method for entertainment stage lighting," *IEEE Transactions on Multimedia*, 2026.

[8] M. Andrychowicz, F. Wolski, A. Ray, J. Schneider, R. Fong, P. Welinder, B. McGrew, J. Tobin, O. Pieter Abbeel, and W. Zaremba, "Hindsight experience replay," *Advances in neural information processing systems*, vol. 30, 2017.

[9] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[10] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2009.

[11] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adverserial inverse reinforcement learning," in *International Conference on Learning Representations*, 2018.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[13] C. Finn, P. Christiano, P. Abbeel, and S. Levine, "A connection between generative adversarial networks, inverse reinforcement learning, and energy-based models," *arXiv preprint arXiv:1611.03852*, 2016.

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[15] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[16] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," *arXiv preprint arXiv:1506.02438*, 2015.

[17] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu, *et al.*, "Deepseekmath: Pushing the limits of mathematical reasoning in open language models," *arXiv preprint arXiv:2402.03300*, 2024.

[18] M. Erdmann, M. von Berg, and J. Steffens, "Development and evaluation of a mixed reality music visualization for a live performance based on music information retrieval," *Frontiers in Virtual Reality*, vol. 6, p. 1552321, 2025.

[19] I. Suno, "Suno."

[20] S.-W. Hsiao, S.-K. Chen, and C.-H. Lee, "Methodology for stage lighting control based on music emotions," *Information sciences*, vol. 412, pp. 14–35, 2017.

[21] C. B. Moon, H. Kim, D. W. Lee, and B. M. Kim, "Mood lighting system reflecting music mood," *Color Research & Application*, vol. 40, no. 2, pp. 201–212, 2015.

[22] I.-D. Stanescu, B.-A. Enache, G.-C. Seritan, S.-D. Grigorescu, F.-C. Argatu, and F.-C. Adochiei, "Automatic control system for stage lights," in *2018 International Symposium on Fundamentals of Electrical Engineering (ISFEE)*, pp. 1–4, IEEE, 2018.

[23] J. Lei, M. Chen, Z. Wang, Y. Wang, and J. Lin, "Music-driven lighting manipulation for stage performance visual design," in *Twelfth International Conference on Graphics and Image Processing (ICGIP 2020)*, vol. 11720, pp. 646–655, SPIE, 2021.

[24] M. Kanno and Y. Fukuhara, "Automatic stage illumination control system by impression of the lyrics and music tune," in *2022 13th International Congress on Advanced Applied Informatics Winter (IIAI-AAI-Winter)*, pp. 219–224, IEEE, 2022.

[25] S. B. Tyroll, D. Overholt, and G. Palamas, "Avai: A tool for expressive music visualization based on autoencoders and constant q transformation," in *17th Sound and Music Computing Conference*, pp. 378–385, Axea sas/SMC Network, 2020.

[26] F. A. Robinson, V. Raj, D. Cooper, F. Du, and D. Gunawan, "Glow with the flow: Ai-assisted creation of ambient lightscapes for music videos," *arXiv preprint arXiv:2602.08838*, 2026.

[27] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," *IEEE Transactions on Cybernetics*, vol. 54, no. 12, pp. 7173–7186, 2024.

[28] Y. Hu and S. Li, "Offline inverse reinforcement learning for joint optimization of energy costs and demand charge in industrial pv-battery load systems," *Applied Energy*, vol. 408, p. 127416, 2026.

[29] T. Oda, "Equilibrium inverse reinforcement learning for ride-hailing vehicle network," in *Proceedings of the Web Conference 2021*, pp. 2281–2290, 2021.

[30] Z. Bing, C. Lemke, L. Cheng, K. Huang, and A. Knoll, "Energy-efficient and damage-recovery slithering gait design for a snake-like robot based on reinforcement learning and inverse reinforcement learning," *Neural Networks*, vol. 129, pp. 323–333, 2020.

[31] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," *Artificial Intelligence*, vol. 297, p. 103500, 2021.

[32] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*, p. 1, 2004.

[33] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey, *et al.*, "Maximum entropy inverse reinforcement learning.," in *Aaai*, vol. 8, pp. 1433–1438, Chicago, IL, USA, 2008.

[34] J. Ho and S. Ermon, "Generative adversarial imitation learning," *Advances in neural information processing systems*, vol. 29, 2016.

[35] C. Finn, S. Levine, and P. Abbeel, "Guided cost learning: Deep inverse optimal control via policy optimization," in *International conference on machine learning*, pp. 49–58, PMLR, 2016.

[36] Z. Zhao and S. Li, "One step is enough: Multi-agent reinforcement learning based on one-step policy optimization for order dispatch on ride-sharing platforms," *arXiv preprint arXiv:2507.15351*, 2025.

[37] L.-Q. Yan, Y. Zhou, K. Xu, and R. Wang, "Accurate translucent material rendering under spherical gaussian lights," in *Computer Graphics Forum*, vol. 31, pp. 2267–2276, Wiley Online Library, 2012.

[38] W. J. Schroeder, L. S. Avila, and W. Hoffman, "Visualizing with vtk: a tutorial," *IEEE Computer graphics and applications*, vol. 20, no. 5, pp. 20–27, 2000.

[39] T. Lei, Y. Wang, Y. Fan, and J. Zhao, "Vector morphological operators in hsv color space," *Science China Information Sciences*, vol. 56, no. 1, pp. 1–12, 2013.

[40] C. of Hunan Television, "I am a singer."

[41] C. of Zhejiang Television, "Sound of my dream."

[42] A. L. Cramer, H.-H. Wu, J. Salamon, and J. P. Bello, "Look, listen, and learn more: Design choices for deep audio embeddings," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3852–3856, IEEE, 2019.

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "Pytorch: An imperative style, high-performance deep learning library. arxiv 2019," *arXiv preprint arXiv:1912.01703*, vol. 10, 1912.

[44] A. Alajanki, Y.-H. Yang, and M. Soleymani, "Benchmarking music emotion recognition systems," *PloS one*, pp. 835–838, 2016.

[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[46] N. A. Nijdam, "Mapping emotion to color," *Book Mapping emotion to color*, pp. 2–9, 2009.

Appendix Contents

A Related Work **14**

A.1 Automatic Stage Light Control 14

A.2 Imitation Learning 14

B Function Definition **15**

C Network Architecture **16**

D Experiment Setup **17**

D.1 Dataset Description 17

D.2 Simulation Setup 17

D.3 Model Configurations 18

D.4 Hardware Configurations 18

E Quantitative Analysis **19**

E.1 Comparison with Alternative Solutions 19

E.2 Ablation Study 19

E.3 RL Training Curves 22

E.4 Error Decomposition of the Two Stages 22

E.5 Robustness across Venue Layouts 24

F Human Evaluation **24**

F.1 Study Setup 24

F.2 Comparative Objects 25

F.3 Results Analysis 25

G Visualizations **26**

G.1 Real-World Livehouse Example 26

G.2 Goal-Conditioned Light Decomposition 26

H Discussions **28**

A Related Work

A.1 Automatic Stage Light Control

Limited by data scarcity, most early-stage ASLC research focused on rule-based approaches. These methods first map music into discrete categories, such as chords [1], emotions [2; 20; 21], or styles [22; 23; 24; 3], and then associate each category with a predefined lighting pattern. However, they suffer from several significant shortcomings: (i) **low interpretability**: predefined lighting patterns often lack empirical justification, and some humanities research questions the strength of the relationship between concepts like emotion and stage lighting [4; 5]; (ii) **coarse granularity**: due to the limitations of MIR datasets, most classification modules support only very coarse categories and overlook finer details such as colorful chord textures and music subgenres; (iii) **low accuracy**: the overall performance of rule-based pipelines is highly sensitive to classification quality, which remains a challenge in MIR. To address some of these issues, prior work used autoencoders [25] to extract music features and then mapped embeddings to lighting patterns, partially mitigating the granularity and mapping problems.

In contrast, Skip-BART [4] introduced a different paradigm by learning lighting control directly from professional light engineers and released the RPMC-L^2 dataset collected from real livehouse performances. Skip-BART is trained end-to-end on this dataset and achieves performance comparable to human light engineers. However, all the above ASLC works concentrated on generating a single primary light, which limits applicability in real livehouse settings with multiple lights at different positions. More recently, [26] proposed an agent-based system that extract features from video and audio and generate editable ambient light objects, but their system differs from stage-lighting applications. LightGen [7] proposed a more direct end-to-end solution that trains a model to map music directly to stage light control parameters (i.e. DMX). However, it introduces a new challenge: different stages have varying numbers of lights and setups, making the trained model difficult to transfer. Moreover, collecting professional data for each stage remains challenging. Even though they collected data in a virtual simulation scenario with expert light engineers, the dataset spans less than three hours, and neither the dataset nor the simulator is publicly available, hindering further research.

In this paper, we propose a novel hierarchical solution: we first train Skip-BART to predict the light hue and value distributions from real-collected stage light videos, and then train a RL framework to decompose the distribution into individual lights. Compared to LightGen, our solution can adaptively transfer to arbitrary stage setups, as the RL process requires no ground-truth annotations and any valid decomposition solution is acceptable.

A.2 Imitation Learning

Imitation Learning (IL) [27] is a fundamental paradigm for enabling agents to learn from expert demonstrations, with widespread applications in energy management [28], ride hailing [29], and robot control [30]. The most straightforward approach within IL is Behavioral Cloning (BC), which frames the problem as supervised learning: the policy is trained to directly predict and replicate expert actions. However, BC suffers from limited generalization; the learned policy can fail catastrophically when encountering states not covered by the offline expert trajectories.

Another major branch of IL is Inverse Reinforcement Learning (IRL) [31], which seeks to infer the underlying reward function (or objective) that expert behaviors are optimizing. By recovering this reward function, IRL can subsequently train a new policy via Reinforcement Learning (RL) that captures the intent behind expert demonstrations, leading to more robust and generalizable performance. A foundational work in this direction is Apprenticeship Learning [32], which assumes the expert's reward function can be represented as a linear combination of known features, and aims to find a policy whose feature expectations match those of the expert. However, this formulation often yields an ambiguous reward function, as multiple reward structures can explain the same expert behavior. This ambiguity arises from the underlying assumption that expert behavior is deterministic and that the demonstrated trajectories are uniquely optimal. To address this limitation, the Maximum Entropy IRL (MaxEnt IRL) framework [33] was proposed, which models expert trajectories as being distributed according to the maximum entropy principle subject to feature-matching constraints. This provides a principled way to handle inherent noise and suboptimality in human demonstrations, recovering a probabilistic model of behavior that is both unique and well-calibrated.

In recent decades, the success of adversarial learning, exemplified by Generative Adversarial Networks (GANs) [14], has also transformed the field of IL. A pivotal development was Generative Adversarial Imitation Learning (GAIL) [34], which directly learns a policy that matches the state-action occupancy measure of the expert. GAIL achieves this by training a discriminator to distinguish between expert and generated trajectories, using its output as a surrogate reward signal. This approach bypasses explicit reward function learning and scales effectively to complex, high-dimensional domains. Building on this foundation, the connection between GAIL and IRL was further strengthened by formulations such as Guided Cost Learning (GCL) [35], which integrates IRL within a maximum entropy inverse optimal control objective using energy-based models and optimizes it via sample-based estimation. This line of work culminated in the GAN-GCL framework [13], which explicitly casts the IRL problem as a generative adversarial game, simultaneously learning both a cost function and a policy. Subsequently, Adversarial Inverse Reinforcement Learning (AIRL) [11] extended these ideas by introducing a more structured and practical reward formulation that is robust to changes in environment dynamics, disentangling the reward function from the dynamics model and thereby enabling better transfer of the learned reward to new tasks. Note that the goal of this paper is not to develop novel IL methods, but rather to adapt existing IL techniques to address our specific task.

B Function Definition

In this section, we formally define the distance and divergence metrics used throughout this paper, including L1 distance, Wasserstein distance, JS divergence, KL divergence, Bhattacharyya distance, and cosine similarity. Their definitions are as follows:

$$\cdot \text{ L1 distance:} \quad D_{\mathrm{L1}}(P \parallel Q) = \sum_i |P_i - Q_i|, \tag{20}$$

$$\cdot \text{ Wasserstein distance:} \quad D_W(P \parallel Q) = \inf_{\gamma \in \Pi(P,Q)} \mathbb{E}(x,y) \sim \gamma[|x-y|], \tag{21}$$

$$\cdot \text{ JS divergence:} \quad D_{\mathrm{JS}}(P \parallel Q) = \frac{1}{2} D_{\mathrm{KL}}(P \parallel \frac{P+Q}{2}) + \frac{1}{2} D_{\mathrm{KL}}(Q \parallel \frac{P+Q}{2}), \tag{22}$$

$$\cdot \text{ KL divergence:} \quad D_{\mathrm{KL}}(P \parallel Q) = \sum_i P_i \log \frac{P_i}{Q_i}, \tag{23}$$

$$\cdot \text{ Bhattacharyya distance:} \quad D_{\mathrm{B}}(P \parallel Q) = -\ln \left(\sum_i \sqrt{P_i Q_i} \right), \tag{24}$$

$$\cdot \text{ Cosine similarity:} \quad \mathrm{CosSim}(P,Q) = \frac{\sum_i P_i Q_i}{\sqrt{\sum_i P_i^2} \sqrt{\sum_i Q_i^2}}, \tag{25}$$

where P, Q are two distributions, $\Pi(P,Q)$ denotes the set of all joint distributions with marginals P and Q. For each distribution, we treat it as a discrete vector representing the probability mass in each bin.

Additionally, for the functions D_h and D_v used in Equation (5) to measure the hue and value distances between two frames, we follow the definitions in [4]:

$$\cdot \text{ Hue distance:} \quad D_h(x \parallel y) = \min\{|x-y|, 2\pi - |x-y|\}, \tag{26}$$

$$\cdot \text{ Value distance:} \quad D_v(x \parallel y) = |x-y|, \tag{27}$$

where x and y are scalar values.

For the action output in our IL framework, we employ the Von Mises and Beta distributions, whose probability density functions are given by:

$$\mathrm{VonMises}(x \mid \mu, \kappa) \;=\; \frac{\exp\big(\kappa \cos(x-\mu)\big)}{2\pi \mathrm{I}_0(\kappa)} \quad (x \in [0, 2\pi)), \tag{28}$$

where μ and $\kappa > 0$ (mean direction and concentration) are predicted by the policy network, and I_0 is the modified Bessel function of the first kind of order zero, defined as:

$$\mathrm{I}_0(\kappa) = \frac{1}{\pi} \int_0^{\pi} \exp(\kappa \cos\theta)\, d\theta. \tag{29}$$

$$\mathrm{Beta}(x \mid \alpha, \beta) \;=\; \frac{x^{\alpha-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha,\beta)} \quad (x \in [0,1]), \tag{30}$$

where $\alpha > 0, \beta > 0$ are shape parameters predicted by the policy, and $\mathrm{B}(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$ is the Beta function, where

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t}\, dt. \tag{31}$$

We further define the standard goal-conditioned RL value functions as follows:

$$\begin{aligned}
\mathscr{V}_\pi(s_t, g) &= \mathbb{E}_\pi \left[\sum_{i=t}^{n} \gamma^{i-t} \mathscr{R}(s_t, a_t, g) \mid s_t \right], \\
\mathscr{Q}_\pi(s_t, a_t, g) &= \mathbb{E}_\pi \left[\sum_{i=t}^{n} \gamma^{i-t} \mathscr{R}(s_t, a_t, g) \mid s_t, a_t \right], \\
\mathscr{A}_\pi(s_t, a_t, g) &= \mathscr{Q}_\pi(s_t, a_t, g) - \mathscr{V}_\pi(s_t, g),
\end{aligned} \tag{32}$$

where $\mathscr{V}$, $\mathscr{Q}$, and $\mathscr{A}$ denote the state-value, action-value, and advantage functions, respectively. Specifically, the advantage function in GRPO is defined as:

$$\begin{aligned}
\overline{\mathscr{A}}_\Phi(s_t, a_t, g) &= \frac{1}{\sigma_\Phi^{t:}(g)} \Big(\sum_{i=t}^{n} \gamma^{\,i-t} \mathscr{R}_\Phi(s_i, a_i, g) - \mu_\Phi^{t:}(g) \Big), \\
\mu_\Phi^{t:}(g) &= \mathbb{E}_{\pi_{\theta^-}} \Big[\sum_{i=t}^{n} \gamma^{\,i-t} \mathscr{R}_\Phi(s_i, a_i, g) \Big], \\
\sigma_\Phi^{t:}(g) &= \sqrt{ \mathbb{E}_{\pi_{\theta^-}} \Big[\Big(\sum_{i=t}^{n} \gamma^{\,i-t} \mathscr{R}_\Phi(s_i, a_i, g) - \mu_\Phi^{t:}(g) \Big)^2 \Big] },
\end{aligned} \tag{33}$$

where the states and actions in the first line are computed for a single collected trajectory, while $\mu_\Phi^{t:}$ and $\sigma_\Phi^{t:}$ are estimated as expectations over multiple trajectories under the behavior policy. (Note that this formulation is a generalized version adapted from the original GRPO used in language modeling [36].)

C Network Architecture

The network architecture for the IL stage is illustrated in Fig. 2. At time step t, the input is the sequence $\mathcal{X}_t = [\langle [\mathrm{SOS}], g\rangle, \langle a_1, \mathrm{Mix}(a_{:1})\rangle, \langle a_2, \mathrm{Mix}(a_{:2})\rangle, \ldots, \langle a_{t-1}, \mathrm{Mix}(a_{:t-1})\rangle]$, where each position i contains the hue-and-value action executed at time $i-1$, i.e., a_{i-1}, together with the light distribution aggregated from the actions taken before time i, i.e., $\mathrm{Mix}(a_{:i-1})$. For the first position, the distribution is set to the goal g, and the action slot is filled with an $[\mathrm{SOS}]$ token to ensure that all positions share the same dimensionality. The sequence $\mathcal{X}_t$ thus serves as a compact representation of $[s_t, g]$. This sequence is fed into a causal Transformer:

$$\mathcal{E}_t = \mathrm{Transformer}(\mathcal{X}_t), \tag{34}$$

where $\mathcal{E}_t$ is the embedding produced at the final position, which compresses information about the goal and the entire history up to time t.

In this work, we adopt an AC framework for the RL component. The embedding $\mathcal{E}_t$ is passed to an actor head and a critic head, each implemented as an MLP. The actor outputs the action parameters (specifically the parameters of the Von Mises distribution μ, κ and the Beta distribution α, β defined in the previous subsection), and the critic outputs the state value estimate. To accommodate a learned reward model and the auxiliary objective of predicting state transitions, we introduce three additional MLP heads that take as input the concatenation of $\mathcal{E}_t$ and the chosen action a_t. This concatenation serves as a compressed representation of the full tuple $[s_t, a_t, g]$. Concretely, one head predicts the scalar reward r_t, while the other two heads predict the resulting hue and value distributions after applying a_t.

D Experiment Setup

D.1 Dataset Description

We utilize the publicly available RPMC-L^2 dataset [4], which comprises recordings from 35 live performances across various commercial venues between December 2020 and July 2024. After data cleaning, videos shorter than 20 seconds were discarded, yielding a total of 699 valid samples. The dataset primarily covers music genres including generalized rock, punk, metal, and core.

For each timestep t, we align each music segment with its corresponding video frame j to extract a hue histogram h_j and a value histogram v_j. The histogram counts are normalized so that each frame-wise distribution sums to 1. For the value histogram, we set the lowest-intensity bin to zero ($v_j^0 \leftarrow 0$), thereby reducing the contribution of near-black pixels and sensor noise. Through this pre-processing pipeline, we obtain synchronized per-frame distributions h_j (360 bins) and v_j (100 bins) that serve as the ground truth for our modeling task.

D.2 Simulation Setup

In this paper, we conduct experiments using a simulated multi-light environment rather than real venues, which we leave for future work. *Nevertheless, our method is not limited to simulation and inherently supports real-world scenarios, where the following mixing process could be implemented by collecting data directly from cameras.*

Specifically, we adopt a circular stage lighting setup, which is commonly used in many television productions, as illustrated in Fig. 3. We consider N lights, each modeled as a point light source, as shown in Fig. 3c. The following describes how we compute the mixed light distribution via simulation, under several simplifying assumptions. Let the image be discretized into a grid of size $H \times W$ (height and width in pixels). For each light $i = 1, \ldots, N$, its position in pixel coordinates is (x_i, y_i), where $x_i \in [1, W], y_i \in [1, H]$. The hue $h_i \in [0, 2\pi)$ is expressed in radians and the value $v_i \in [0, 1]$ is a scalar. *Note that the notation in this section is self-contained and independent of the rest of the paper for clarity.*

Distance and Weighting [37]: For a given pixel at integer coordinates (u, v) with $u = 1, \ldots, W$, $v = 1, \ldots, H$, the squared Euclidean distance from light i is

$$d_{i,u,v}^2 = (u - x_i)^2 + (v - y_i)^2. \tag{35}$$

The contribution weight of light i to pixel (u, v) is modeled by a Gaussian decay function:

$$\begin{aligned} w_{i,u,v} &= \exp\left(-\frac{d_{i,u,v}^2}{2\sigma_{\text{pix}}^2}\right), \\ \sigma_{\text{pix}} &= \sigma\sqrt{W^2 + H^2}, \end{aligned} \tag{36}$$

where σ is a relative spread parameter controlling the spatial influence of each light.

Per-Pixel Value [38]: The raw value at pixel (u, v) is the weighted sum of the light values:

$$V_{u,v}^{\text{raw}} = \sum_{i=1}^{N} v_i w_{i,u,v}. \tag{37}$$

To prevent excessive brightness, an optional soft clipping may be applied:

$$V_{u,v} = 1 - \exp\big(-cV_{u,v}^{\text{raw}}\big), \tag{38}$$

where $c > 0$ is a clipping factor. The final value map is $\{V_{u,v} | u = 1, \ldots, W; v = 1, \ldots, H\}$.

Per-Pixel Hue [39]: The mixed hue at pixel (u, v) is obtained by combining the individual light hues in a vector average. Define the weight for hue mixing as

$$\tilde{w}_{i,u,v} = \frac{v_i w_{i,u,v}}{V_{u,v}^{\text{raw}} + \varepsilon}, \tag{39}$$

which ensures that pixels with very low intensity have a negligible hue contribution. The averaged sine and cosine components are

$$S_{u,v} = \sum_{i=1}^{N} \tilde{w}_{i,u,v} \sin h_i, \tag{40}$$

$$C_{u,v} = \sum_{i=1}^{N} \tilde{w}_{i,u,v} \cos h_i. \tag{41}$$

Then the mixed hue angle (in radians) is

$$H_{u,v} = \operatorname{atan2}(S_{u,v}, C_{u,v}) \mod 2\pi, \tag{42}$$

with the convention $H_{u,v} = 0$ when $V_{u,v}^{\text{raw}} < \varepsilon$.

Histograms: To obtain the overall hue and value distributions, we compute normalized histograms over all pixels. Hue is discretized into B_h bins of equal width $\Delta_h = \frac{2\pi}{B_h}$:

$$\widehat{H}[k] = \frac{1}{HW} \sum_{u,v} \mathbf{1}\left(\left\lfloor \frac{H_{u,v}}{\Delta_h} \right\rfloor = k \right), \quad k = 0, \ldots, B_h - 1, \tag{43}$$

where $\mathbf{1}(\cdot)$ is the indicator function. Value is discretized into B_v bins of width 0.01:

$$\widehat{V}[\ell] = \frac{1}{HW} \sum_{u,v} \mathbf{1}\left(\lfloor B_v V_{u,v} \rfloor = \ell \right), \quad \ell = 0, \ldots, B_v - 1. \tag{44}$$

These histograms serve as the aggregated light distribution $y_j = [\widehat{H}, \widehat{V}]$ in our framework.



(a) TV "I Am a Singer" [40] (b) TV "Sound of My Dream" [41] (c) Simulation Scenario

Figure 3: Examples of Circle Stage Lights: The first two sub-figures are screenshots from TV shows [40; 41]. In our simulator (the last sub-figure), lights are indexed and actions are generated in clockwise order.

D.3 Model Configurations

For Skip-BART, we adopt the same configuration as the original paper [4], with the exception of modifying the input and output layers for hue and value. Specifically, we replace these layers with MLPs of input and output dimensions 360 and 100, respectively, to accommodate the distributional input and output required by our task. Detailed model configurations are provided in Table 3.

For the IL stage, the configurations are summarized in Table 4. Given the well-known difficulty of training large models in RL settings, we adopt a relatively small Transformer architecture compared to Skip-BART.

D.4 Hardware Configurations

The experiments were conducted using the PyTorch framework [43]. The Skip-BART fine-tuning was performed on a server running Ubuntu 22.04.5 LTS, equipped with an Intel(R) Xeon(R) Gold

Table 3: Model Configurations of Skip-BART

Configuration	Our Setting
Bin Amount for Hue and Value	[360, 100]
Music Embedding	OpenL3 [42]
Embedding Dimension	512
Input Length	1024
Number of Network Layers	8
Hidden Size	2048
Inner Linear Size	2048
Attention Heads	8
Dropout Rate	0.1
Total Number of Parameters	231M
Trainable Parameters	9 M
Optimizer	AdamW
Learning Rate	0.0001
Batch Size	16
Hyper-parameters $[d^h, d^v]$	$[\frac{\pi}{2}$,0.3]
Training Iterations	200

Table 4: Model Configurations of IL

Configuration	Our Setting
Bin Amount for Hue and Value	[360, 100]
Embedding Dimension	64
Input Length (Light Amount)	8
Number of Network Layers	3
Hidden Size	64
Inner Linear Size	256
Attention Heads	4
Dropout Rate	0.0
Total Number of Parameters	393K
Optimizer	AdamW
Learning Rate	0.0003
Batch Size	64
Hyper-parameters $[\eta, \delta, \epsilon]$	[0.1,0.1,0.2]
Phase 1 Iterations	300
Phase 2 Iterations	200
Phase 3 Iterations	500

6133 CPU @ 2.50 GHz, along with two NVIDIA 4090 GPUs and one NVIDIA A100 GPU. The light decomposition policy was trained concurrently on a separate workstation running Windows 11, equipped with an Intel(R) Core(TM) i7-14700KF processor and an NVIDIA RTX 4080 graphics card.

E Quantitative Analysis

E.1 Comparison with Alternative Solutions

The decomposition problem admits no closed-form solution, since the mixing operator is non-linear and non-differentiable and its inverse is not unique. In settings with more complex optics it may only be available as a black box. We therefore compare against the strongest tractable alternatives rather than against a prior multi-light method, which does not exist. Supervised Synthetic Regression trains the same network to regress per-light actions directly from the goal by behavior cloning, and coincides with the Phase 1 variant of our pipeline without reward learning or RL fine-tuning. Heuristic Distribution Matching greedily assigns to each light the hue and value of the dominant peak in the goal histogram. Non-IL RL replaces the learned reward with a terminal KL divergence between the achieved and target distributions and optimizes it with textbook PPO. The remaining configuration is our full IL pipeline at Phase 3 (GRPO).

Table 5 and Table 6 report the results. Our IL solution attains the lowest L1, JS, and Bhattacharyya errors and the highest cosine similarity in nearly every setting, and its advantage over behavior cloning shows that the reward learning and RL fine-tuning stages contribute measurably beyond pure imitation of the synthetic expert data. The greedy heuristic occasionally reaches a lower Wasserstein distance, which is expected because it is built around matching the histogram mode, the statistic that Wasserstein is most sensitive to, at the cost of a much worse JS and Bhattacharyya error and a lower cosine similarity. Non-IL RL is competitive on a single metric in the value setting but degrades sharply elsewhere, with its Wasserstein error an order of magnitude larger than ours under hue. This pattern is consistent with the analysis in Section E: a terminal distributional reward supplies no intermediate credit and gives an unstable learning signal when the action space is high-dimensional.

E.2 Ablation Study

In this section, we conduct an ablation study to test model performance when eliminating the BC and AUX losses. The detailed results for hue and value are presented in Tables 7 and 8. According to the results, we notice:

Table 5: Comparison with alternative decomposition solutions on hue. The best results are highlighted in **bold**. This notation is consistent throughout the following tables. (Mean ± Standard Deviation (M ± SD))

Method	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
IL (Phase 3, GRPO)	**2.73±0.09**	**2.59±0.89**	5.19±5.58	7.54±5.75	**2.23±1.08**	**2.19±1.11**
Alternatives						
Supervised Synthetic	3.58±1.01	3.20±0.76	5.61±5.70	7.99±0.60	3.18±1.25	2.80±0.93
Heuristic Matching	3.88±0.84	3.45±1.34	**3.94±1.75**	**2.87±2.90**	3.88±1.00	3.20±1.64
Non-IL RL	3.72±0.97	3.87±1.24	15.97±6.16	12.20±2.46	3.52±1.32	3.32±1.40

Method	KL ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
IL (Phase 3, GRPO)	**1.65±2.33**	**1.05±0.66**	**3.67±2.06**	**3.61±2.17**	**7.53±1.24**	**7.19±1.40**
Alternatives						
Supervised Synthetic	1.66±1.56	1.15±0.52	5.79±2.88	4.74±2.00	6.64±1.45	6.61±1.33
Heuristic Matching	2.06±0.39	1.30±0.76	7.37±2.76	6.15±4.01	5.49±1.17	6.23±1.56
Non-IL RL	14.18±1.30	2.32±0.41	6.85±3.73	4.83±1.34	6.12±1.60	5.53±0.54

Table 6: Comparison with alternative decomposition solutions on value. The best results are highlighted in **bold**. This notation is consistent throughout the following tables. (Mean ± Standard Deviation (M ± SD))

Method	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
IL (Phase 3, GRPO)	**9.24±3.32**	**9.14±2.16**	**5.32±3.44**	**5.78±0.79**	1.50±0.71	**1.85±0.53**
Alternatives						
Supervised Synthetic	10.21±3.06	11.25±1.46	5.98±3.38	6.87±0.83	1.67±0.67	2.44±0.43
Heuristic Matching	18.33±1.85	17.29±3.00	11.85±5.87	11.14±5.27	2.82±0.93	2.28±0.90
Non-IL RL	12.70±5.02	9.66±5.37	6.01±0.29	6.46±1.96	**0.93±0.63**	1.00±0.49

Method	KL ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
IL (Phase 3, GRPO)	**0.52±0.25**	**0.66±0.21**	2.17±1.20	**2.70±0.98**	8.09±0.89	7.45±0.74
Alternatives						
Supervised Synthetic	0.58±0.24	0.86±0.17	2.46±1.14	3.78±0.88	8.00±0.91	6.97±0.60
Heuristic Matching	1.06±0.41	0.98±0.71	4.64±1.87	3.56±1.74	6.31±1.30	6.95±1.03
Non-IL RL	3.04±1.65	0.87±0.35	**1.36±0.98**	1.29±0.65	**8.32±0.86**	**8.78±0.65**

- Without BC or AUX loss, model performance worsens in most circumstances, illustrating that they indeed help improve policy performance and enhance the model's capacity to extract features and relationships among input observation sequences. Additionally, when eliminating the BC or AUX loss, performance in Phase 3 decreases compared to Phase 2. This is because the accurate learning of the reward function in AIRL [11] relies on the policy achieving optimal performance or being very close to it. When the policy is not sufficiently good, the reward function can be biased, leading the model update in the wrong direction during the Phase 3 fine-tuning.

Table 7: Model Performance on Hue

Model	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
Phase 1	3.58±1.01	3.20±0.76	5.61±5.70	7.99±0.60	3.18±1.25	2.80±0.93
Phase 2 (GRPO)	2.66±0.80	2.99±0.59	4.97±4.96	6.89±0.50	2.17±0.93	2.71±0.93
Phase 3 (GRPO)	2.73±0.09	**2.59±0.89**	5.19±5.58	7.54±5.75	2.23±1.08	**2.19±1.11**
Phase 2 (PPO)	**2.52±0.87**	2.70±0.58	**4.73±5.51**	6.93±5.22	**2.03±1.00**	2.39±0.19
Phase 3 (PPO)	2.74±0.84	3.18±0.53	4.80±4.57	7.16±4.34	2.20±1.02	2.87±0.72
w/o BC						
Phase 2 (GRPO)	2.89±0.85	3.58±0.40	5.06±4.11	6.68±4.27	2.43±1.02	3.29±0.60
Phase 3 (GRPO)	3.24±1.06	3.93±0.36	5.75±2.52	**6.64±2.93**	2.80±1.32	3.19±0.57
Phase 2 (PPO)	2.79±0.61	3.25±0.30	5.37±3.07	6.75±3.37	2.31±0.81	2.88±0.54
Phase 3 (PPO)	2.75±0.92	2.77±0.45	5.90±7.15	8.75±6.69	2.84±1.14	2.37±0.86
w/o AUX						
Phase 1	2.98±1.03	2.84±0.78	4.94±5.12	7.49±5.53	2.56±1.23	2.53±0.99
Phase 2 (GRPO)	2.70±0.86	2.97±0.83	5.32±5.08	7.03±0.51	2.25±0.97	2.60±1.00
Phase 3 (GRPO)	2.75±0.81	2.78±0.91	5.21±5.22	8.90±0.82	2.27±0.92	2.36±1.03
Phase 2 (PPO)	2.80±0.94	2.82±0.79	5.77±5.17	8.60±1.11	2.30±1.16	2.42±1.04
Phase 3 (PPO)	2.98±1.15	2.89±0.69	6.76±5.22	7.38±0.52	2.57±1.39	2.63±8.15

Model	KL ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
Phase 1	1.66±1.56	1.15±0.52	5.79±2.88	4.74±2.00	6.64±1.45	6.61±1.33
Phase 2 (GRPO)	3.78±4.64	1.98±0.89	3.49±1.72	4.45±2.13	7.13±1.29	6.23±1.36
Phase 3 (GRPO)	1.65±2.33	1.05±0.66	3.67±2.06	**3.61±2.17**	7.53±1.24	**7.19±1.40**
Phase 2 (PPO)	2.58±3.46	1.51±5.00	3.19±1.87	3.87±1.74	**7.56±1.21**	7.18±1.19
Phase 3 (PPO)	3.77±4.93	1.93±6.65	**3.05±2.09**	4.69±1.64	7.11±1.27	6.42±1.64
w/o BC						
Phase 2 (GRPO)	7.04±0.58	2.57±0.48	4.01±2.12	5.42±1.60	6.57±1.09	5.46±0.81
Phase 3 (GRPO)	9.22±1.58	2.72±0.36	4.89±2.95	5.80±1.32	6.05±1.22	5.85±1.79
Phase 2 (PPO)	6.53±5.71	2.13±0.39	3.68±1.66	4.53±1.38	7.02±0.73	6.30±0.97
Phase 3 (PPO)	1.89±2.95	1.08±3.84	3.55±2.23	3.75±1.82	7.18±1.02	6.27±1.04
w/o AUX						
Phase 1	1.28±1.17	1.16±0.69	4.39±2.29	4.24±2.07	7.18±1.47	6.85±1.45
Phase 2 (GRPO)	4.40±4.84	1.67±0.86	3.62±1.89	4.21±1.95	7.22±1.12	6.59±1.47
Phase 3 (GRPO)	4.02±4.74	**0.90±0.39**	3.68±1.78	3.89±1.98	7.24±1.06	6.91±1.33
Phase 2 (PPO)	2.13±2.63	1.06±0.50	3.84±2.40	4.01±2.11	7.36±1.25	6.79±1.29
Phase 3 (PPO)	**1.15±0.73**	1.67±0.78	4.51±2.91	4.25±1.65	7.20±1.52	6.69±1.21

- In our task, we observe that PPO does not achieve performance improvement in Phase 3, even with the BC and AUX loss. This is due to the fact that in Phase 2, the critic, actor, and reward model are trained together, which may interfere with each other, making it difficult for the actor to converge and resulting in an incorrect learned reward function. In contrast, GRPO addresses this issue by replacing the advantage estimated by the critic with the group relative reward-to-go.
- Although GRPO achieves the best performance, we still notice that model performance in the ID scenario decreases in Phase 3 compared to Phase 2. This is likely because the model gains high generalization capacity at the expense of overfitting to the original data domain.

Table 8: Model Performance on Value.

Model	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
Phase 1	10.21±3.06	11.25±1.46	5.98±3.38	6.87±0.83	1.67±0.67	2.44±0.43
Phase 2 (GRPO)	8.63±3.04	9.40±1.82	5.05±3.32	**5.76±1.30**	1.31±0.65	1.87±0.48
Phase 3 (GRPO)	9.24±3.32	**9.14±2.16**	5.32±3.44	5.78±0.79	1.50±0.71	**1.85±0.53**
Phase 2 (PPO)	**8.07±2.62**	10.73±2.20	**4.74±3.12**	6.33±1.36	**1.21±0.55**	2.31±6.55
Phase 3 (PPO)	9.70±3.26	11.50±2.53	5.60±3.45	7.08±1.13	1.52±0.70	2.57±9.29
w/o BC						
Phase 2 (GRPO)	9.25±3.25	10.86±0.60	5.44±3.30	6.58±0.94	1.38±0.66	2.20±0.51
Phase 3 (GRPO)	9.56±3.31	11.36±1.88	5.39±3.30	6.60±0.94	1.40±0.65	2.30±0.57
Phase 2 (PPO)	9.47±3.07	11.27±1.83	5.66±3.02	6.87±0.87	1.56±0.73	2.45±0.57
Phase 3 (PPO)	10.25±2.98	12.28±1.71	6.42±3.34	7.23±1.08	1.84±0.68	2.74±0.45
w/o AUX						
Phase 1	9.84±3.15	11.07±1.67	5.71±3.39	6.59±1.03	1.62±0.65	2.40±0.45
Phase 2 (GRPO)	8.96±3.14	10.54±2.28	5.27±3.18	6.45±0.92	1.40±0.64	2.26±0.71
Phase 3 (GRPO)	9.60±2.91	11.21±2.13	5.37±3.25	6.56±0.83	1.54±6.36	2.47±0.69
Phase 2 (PPO)	8.97±3.13	10.54±2.09	5.23±3.32	6.39±0.90	1.41±0.65	2.24±0.60
Phase 3 (PPO)	10.47±3.09	12.19±1.80	6.12±3.36	7.24±0.96	1.78±0.68	2.81±0.62

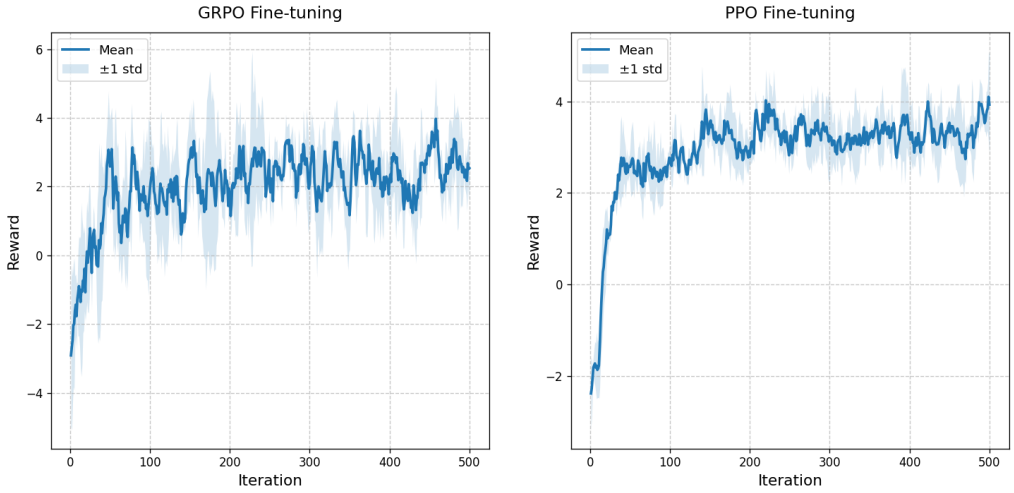
Model	KL ($\times 10^{-1}$) ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
	ID	OOD	ID	OOD	ID	OOD
Proposed						
Phase 1	5.76±2.41	8.60±1.71	2.46±1.14	3.78±0.88	8.00±0.91	6.97±0.60
Phase 2 (GRPO)	4.54±2.30	**6.46±1.85**	1.85±1.07	2.78±0.88	8.27±0.86	**7.66±0.66**
Phase 3 (GRPO)	5.22±2.49	6.55±2.07	2.17±1.20	**2.70±0.98**	8.09±0.89	7.45±0.74
Phase 2 (PPO)	**4.44±1.99**	8.16±2.57	**1.64±1.09**	3.60±1.26	8.15±0.75	7.13±0.85
Phase 3 (PPO)	5.24±2.49	9.53±4.49	2.23±1.17	4.28±2.45	8.15±0.91	6.79±1.24
w/o BC						
Phase 2 (GRPO)	4.81±2.41	7.68±2.07	1.94±1.06	3.32±1.23	8.26±0.92	7.31±0.71
Phase 3 (GRPO)	4.90±2.42	8.14±2.37	1.96±1.05	3.48±1.16	8.26±0.88	7.17±0.78
Phase 2 (PPO)	5.41±2.44	8.75±2.31	2.26±1.12	3.85±1.19	8.06±0.87	6.91±0.75
Phase 3 (PPO)	6.37±52.52	9.87±1.91	2.79±1.20	4.44±098	**8.85±0.99**	6.53±0.54
w/o AUX						
Phase 1	5.58±2.34	8.49±1.82	2.31±1.10	3.72±0.88	8.04±0.93	6.97±0.65
Phase 2 (GRPO)	4.85±2.28	8.04±2.96	2.02±1.05	3.52±1.41	8.18±0.87	7.15±1.06
Phase 3 (GRPO)	5.31±2.24	8.81±2.96	2.24±1.07	3.92±1.45	8.09±0.83	6.96±0.94
Phase 2 (PPO)	4.89±2.36	7.88±2.45	2.02±1.07	3.47±1.17	8.20±0.88	7.23±0.85
Phase 3 (PPO)	6.11±2.47	10.25±2.77	2.67±1.17	4.61±1.37	7.95±0.97	6.51±0.85

E.3 RL Training Curves

The convergence curve for phase 3 is shown in Fig. 4. Both GRPO and PPO converge in roughly 100 iterations and achieve positive rewards, indicating that they successfully fool the discriminator. Note that because the reward models (discriminators) are trained independently in phase 2 for each method, direct comparison of their absolute reward values is not meaningful.

E.4 Error Decomposition of the Two Stages

The pipeline predicts a global distribution in Stage 1 and realizes it per light in Stage 2, so the end-to-end error has two contributions. Table 9 separates them under the in-domain goal: the Stage-1 error measures the distance between the predicted distribution and the ground-truth distribution



(a) GRPO (b) PPO

Figure 4: RL Training Curve in Phase 3.

Table 9: Error decomposition of the two-stage generation. Stage 1 compares the predicted distribution with the ground truth, and Stage 2 compares the achieved distribution with the goal that was given to the policy. The two stages were compared under the in-domain goal, and the Stage-2 measurement does not depend on the goal type because the policy is trained to reach an arbitrary distribution. The best results are highlighted in **bold**. This notation is consistent throughout the following tables. (Mean ± Standard Deviation (M ± SD))

Channel	**Stage**	**L1** ($\times 10^{-3}$) ↓		**Wasserstein** ($\times 10^{-2}$) ↓		**JS** ($\times 10^{-1}$) ↓	
		ID	**OOD**	**ID**	**OOD**	**ID**	**OOD**
Hue	Stage 1	3.66±0.73	–	4.70±2.51	–	1.11±0.39	–
	Stage 2	**2.59±0.89**	**2.59±0.89**	7.54±5.75	**7.54±5.75**	2.19±1.11	2.19±1.11
Value	Stage 1	**5.71±1.47**	11.25±1.46	9.81±3.76	6.87±0.83	**0.83±0.42**	2.44±0.43
	Stage 2	9.14±2.16	**9.14±2.16**	**5.78±0.79**	**5.78±0.79**	1.85±0.53	**1.85±0.53**

Channel	**Stage**	**KL** ↓		**Bhattacharyya** ($\times 10^{-1}$) ↓		**Cosine** ($\times 10^{-1}$) ↑	
		ID	**OOD**	**ID**	**OOD**	**ID**	**OOD**
Hue	Stage 1	**0.81±0.37**	–	**1.93±2.12**	–	**8.37±0.56**	–
	Stage 2	1.05±0.66	**1.05±0.66**	3.61±2.17	**3.61±2.17**	7.19±1.40	7.19±1.40
Value	Stage 1	**0.56±0.40**	**0.86±0.17**	**1.09±0.82**	3.78±0.88	**7.63±1.03**	6.97±0.60
	Stage 2	0.66±0.21	0.66±0.21	2.70±0.98	**2.70±0.98**	7.45±0.74	**7.45±0.74**

of the real lights, while the Stage-2 error measures the distance between the distribution actually achieved by the decomposed controls and the goal that was given to the policy. Stage 1 cannot be scored against a ground truth in the out-of-domain setting, because a randomly generated goal has no corresponding real lighting, so those cells are left empty. The Stage-2 measurement is reported on the same goal in both columns: the policy is trained to reach an arbitrary distribution regardless of where that distribution came from, and evaluating the same goal under a random target therefore adds no information. The two stages are of the same order of magnitude, and neither dominates the total error in either the hue or the value channel. This indicates that the two stages perform comparably and that neither one constitutes a bottleneck for the current framework.

Table 10: Model performance under different light layouts. The best results are highlighted in **bold**. This notation is consistent throughout the following tables. (Mean ± Standard Deviation (M ± SD))

Channel	Layout	L1 ($\times 10^{-3}$) ↓		Wasserstein ($\times 10^{-2}$) ↓		JS ($\times 10^{-1}$) ↓	
		ID	OOD	ID	OOD	ID	OOD
Hue	Rectangle (6 lights)	3.15±0.81	3.14±0.83	**3.16±3.62**	8.66±5.04	2.90±1.97	2.97±1.22
	Circle (8 lights)	**2.73±0.09**	**2.59±0.89**	5.19±5.58	**7.54±5.75**	**2.23±1.08**	**2.19±1.11**
Value	Rectangle (6 lights)	**3.64±1.60**	**5.49±2.23**	**2.54±1.49**	**2.32±0.79**	**0.29±0.19**	**0.64±0.44**
	Circle (8 lights)	9.24±3.32	9.14±2.16	5.32±3.44	5.78±0.79	1.50±0.71	1.85±0.53

Channel	Layout	KL ↓		Bhattacharyya ($\times 10^{-1}$) ↓		Cosine ($\times 10^{-1}$) ↑	
		ID	OOD	ID	OOD	ID	OOD
Hue	Rectangle (6 lights)	2.67±3.38	1.52±0.52	4.51±2.05	8.73±0.72	6.66±1.50	6.08±1.14
	Circle (8 lights)	**1.65±2.33**	**1.05±0.66**	**3.67±2.06**	**3.61±2.17**	**7.53±1.24**	**7.19±1.40**
Value	Rectangle (6 lights)	**0.10±0.06**	**0.25±0.19**	**0.31±0.36**	**0.87±0.72**	**9.64±0.24**	**9.09±0.43**
	Circle (8 lights)	0.52±0.25	0.66±0.21	2.17±1.20	2.70±0.98	8.09±0.89	7.45±0.74

E.5 Robustness across Venue Layouts

The main experiments use a circular arrangement of eight point lights, which is common in television productions. Since the spatial arrangement is a property of the venue rather than of the method, we repeat the Stage-2 experiment on a second venue with a rectangular arrangement of six lights in two rows of three. Stage 1 is shared between the two venues because it predicts the global distribution and never sees the layout, while Stage 2 is trained from scratch for the new venue using only mixed-light data. Table 10 compares the two. Under the rectangular layout the hue errors stay within the same range as under the circular one, and the value errors drop substantially, with the L1 error falling from 9.24×10^{-3} to 3.64×10^{-3}, the KL divergence from 0.52 to 0.10, the Bhattacharyya distance from 2.17×10^{-1} to 0.31×10^{-1}, and the cosine similarity rising from 8.09×10^{-1} to 9.64×10^{-1} in the ID setting. A smaller number of lights leaves fewer degrees of freedom to match the same aggregate distribution, which plausibly explains the improvement in the value channel. The result supports the intended deployment pattern: one general Stage-1 model is trained once, and a cheap venue-specific Stage-2 policy is fit for each new stage.

F Human Evaluation

F.1 Study Setup

To better assess how our results align with human preferences, we designed a questionnaire to collect participants' feedback on the stage lighting effects created by each method. We recruited 31 respondents through social media platforms and live music venues, with the human evaluation spanning half a month. After excluding one invalid response (outlier), 30 valid questionnaires were retained for analysis. The participants (based on these 30 valid responses) included 19 males and 11 females aged 18–55 (predominantly 18–30), among whom 7 had professional experience in lighting design, music production, or stage art. All participants reported normal hearing, normal or corrected-to-normal vision, and no history of color blindness or color weakness. All participants are required to evaluate three music pieces alongside the lights across six metrics [18], with each metric scored from 1 to 7 (the higher the score, the better the evaluation). The lighting conditions were derived from four sources: Ours, Ground Truth, Skip-BART, Rule-based.

Regarding the human evaluation, we follow the same paradigm as Skip-BART [4]. The questionnaire was structured as follows:

- **6 music pieces**: We select six music pieces of different styles to evaluate the generalization capacity of the methods. Specifically, three of them (the ID group) are taken from the testing set of RPMC-L^2 and belong to the same domain as the training set, including the styles of rock, metal, and core. The other three are generated by Suno [19] (the OOD group), covering the styles of folk, R&B, and jazz. Consequently, for the OOD group, there is no ground truth.
- **4 objects per group**: Human Light Engineer (HLE), Rule-based, Skip-BART, and proposed SeqLight

- **6 evaluation dimensions**: Emotional Match Between Lighting and Music, Visual Impact, Rhythmic Synchronization Accuracy, Smoothness of Lighting Transitions, Immersive Atmosphere Intensity, and Innovative Surprise [18; 4].

Each music piece formed one group, which included four videos corresponding to the four objects, with the music serving as background sound and a dynamic color block representing the light changes. Participants were asked to rate each video within each music group across six dimensions using a 7-point Likert scale (1 = very dissatisfied, 7 = very satisfied). At the end of each group, participants were also asked to select their favorite video along with a reason (optional), shown as Fig. 5. More details about the questionnaire can be found in [6].

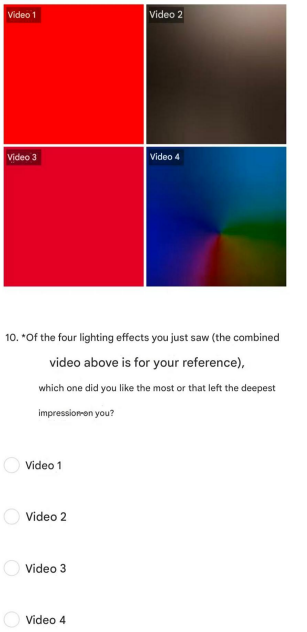


Figure 5: The Screenshot of Questionnaire

Every participant rated all six pieces, and each piece contributed multiple questions per dimension, so the analysis rests on 30 valid participants and six stimulus pieces rather than on a larger stimulus pool. Expanding the pool is bounded by data availability more than by study effort. Skip-BART [4] is currently the only public source of paired music and multi-light recordings with ground-truth lighting, which fixes the ceiling on the in-domain side, and music outside that domain has to be generated rather than collected in order to avoid copyright issues, which is why the out-of-domain pieces come from Suno and have no ground truth by construction.

F.2 Comparative Objects

The experimental subjects can be summarized as follows:

- **Human Light Engineer (HLE)**: The HLE is extracted from the original videos in the RPMC-L^2 dataset, which consist of live performance footage with lights controlled by professional lighting engineers. For each sample, the corresponding video segment is temporally aligned, resized, and processed with large Gaussian blurring to retain the illumination pattern while suppressing scene details.
- **Rule-Based Method**: The rule-based method first maps music pieces into emotion categories and then maps each category to a predefined light pattern. Following the core idea of [20], we adopt the approach of [44] to train a bidirectional Long Short-Term Memory (LSTM) model [45] on the DEAM dataset. The color mapping, which considers both hue and value, is defined according to [46].
- **Skip-BART**: Skip-BART [4] is the first end-to-end stage lighting generation method. It directly maps the music sequence to a single primary color at each frame and is trained on the RPMC-L^2 dataset.
- **SeqLight**: This refers to our proposed method.

F.3 Results Analysis

Table 11 summarizes the median and plurality scores for ID and OOD evaluations. Our method performs strongly under both criteria, achieving the highest median Overall score in both settings and remaining close to Ground Truth in ID. The plurality results show a similar trend, with our method competitive across most dimensions. These results suggest robust perceptual quality and better generalization than the baselines.

Table 12 reports direct preference percentages. In ID, our method receives the highest preference rate (42.22%), slightly above Ground Truth (40.00%) and clearly above Skip-BART (16.67%) and Rule-based (1.11%). In OOD, our advantage is larger, with 51.11% preferences compared to 35.55% for Skip-BART and 13.33% for Rule-based.

Table 13 presents the statistical significance analysis. In ID, our method is not significantly different from Ground Truth, while significantly outperforming Skip-BART on several key dimensions and Rule-based across all dimensions. In OOD, our method significantly improves over Skip-BART on Impact and over Rule-based on most dimensions.

Table 11: Median and Plurality Scores for both In-domain and Out-of-domain evaluations.

Method	ID Evaluation							OOD Evaluation						
	Emo	Imp	Rhy	Smo	Atm	Sur	Ovr	Emo	Imp	Rhy	Smo	Atm	Sur	Ovr
Median														
Ours	4.00	**4.67**	**4.83**	4.33	**4.33**	**4.33**	**4.39**	3.50	**4.50**	**4.00**	**4.00**	**3.83**	**4.00**	**4.08**
Ground Truth	**4.50**	4.00	4.50	**4.50**	**4.33**	4.00	4.31	–	–	–	–	–	–	–
Skip-BART	4.17	4.00	4.00	3.67	4.00	3.67	3.97	**3.67**	3.33	3.67	3.67	3.50	3.33	3.47
Rule-based	3.83	3.00	2.33	2.67	3.00	1.83	3.14	3.50	2.50	2.33	2.50	3.33	2.67	3.06
Plurality														
Ours	4.00	**4.33**	**5.33**	3.33	4.00	**4.00**	3.94	**3.33**	**5.00**	3.00	3.00	2.67	**5.00**	**2.44**
Ground Truth	3.33	3.67	3.67	**4.33**	**4.33**	**4.00**	3.44	–	–	–	–	–	–	–
Skip-BART	**4.67**	**4.33**	4.00	2.67	4.00	2.67	**4.78**	**3.33**	3.33	**4.33**	**3.67**	**3.33**	4.00	1.89
Rule-based	4.00	1.00	1.00	1.00	1.00	1.00	1.00	1.33	1.00	1.00	1.00	1.00	1.00	1.06

Table 12: Which video participants preferred within each group.

Method	ID Evaluation	OOD Evaluation
Ours	**42.22%**	**51.11%**
Ground Truth	40.00%	–
Skip-BART	16.67%	35.55%
Rule-based	1.11%	13.33%

G Visualizations

G.1 Real-World Livehouse Example

Figure 6 shows recordings of an intro by the Chinese rock band Mekader from two different performances, in different years and venues. Despite differing lighting setups and recording viewpoints, the designed light patterns remain similar, and the resulting HV distribution shapes are largely consistent. This empirical evidence supports our use of recorded videos from multiple venues to predict the full light distribution, thereby helping to address the data scarcity challenge that has long plagued many MIR fields. (ii) In the second stage, we aim to find an effective way to decompose the distribution obtained from stage 1 into individual light controls. To this end, we formulate the light decomposition task as a GCMDP, decoupling it from any music-related information. Consequently, this stage requires no professional lighting engineers for data collection. Specifically, we propose to solve the GCMDP using IL, where expert trajectories can be readily collected within each venue using only mixed light data (whether from simulation or real image capture). This also means that each venue can independently train its own light decomposition model and combine it with the light distribution prediction model from stage 1. In this way, our method achieves high transferability and practical applicability.

G.2 Goal-Conditioned Light Decomposition

In this section, we present visualizations of our method applied to the goal-conditioned light decomposition task. Goals are first constructed using our HER-based labeling method, and then decomposed into individual light controls via the proposed imitation learning framework. Representative results are shown in Fig. 7, where each goal comprises either a single color or a combination of multiple colors.

As illustrated in the histogram plots, the generated distributions closely match the target goal distributions. However, due to the inherent ambiguity of the decomposition task, where multiple

Table 13: Statistical comparisons for In-domain and Out-of-domain evaluations. Mean difference (ΔM), standard deviation (SD), and p-values. Significance: *** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$. Missing comparisons (due to no Ground Truth in OOD) are marked with '–'.

Metrics	**Comparison**	**ID Evaluation**			**OOD Evaluation**		
		ΔM	**SD**	p	**ΔM**	**SD**	p
Emotion	ours vs GT	-0.19	0.23	1.000	–	–	–
	ours vs SB	0.21	0.21	1.000	0.16	0.30	1.000
	ours vs RB	0.98	0.29	0.013^{*}	0.67	0.37	0.239
	GT vs SB	0.40	0.17	0.144	–	–	–
	GT vs RB	1.17	0.26	0.001^{**}	–	–	–
	SB vs RB	0.77	0.21	0.006^{**}	0.51	0.24	0.130
Impact	ours vs GT	0.63	0.23	0.053	–	–	–
	ours vs SB	0.93	0.23	0.002^{**}	0.98	0.28	0.005^{**}
	ours vs RB	2.01	0.32	$< 0.001^{***}$	1.70	0.37	$< 0.001^{***}$
	GT vs SB	0.30	0.17	0.531	–	–	–
	GT vs RB	1.38	0.29	$< 0.001^{***}$	–	–	–
	SB vs RB	1.08	0.22	$< 0.001^{***}$	0.72	0.23	0.011^{*}
Rhythm	ours vs GT	0.24	0.22	1.000	–	–	–
	ours vs SB	0.79	0.22	0.007^{**}	0.27	0.27	0.989
	ours vs RB	2.37	0.30	$< 0.001^{***}$	1.46	0.31	$< 0.001^{***}$
	GT vs SB	0.54	0.20	0.058	–	–	–
	GT vs RB	2.12	0.31	$< 0.001^{***}$	–	–	–
	SB vs RB	1.58	0.24	$< 0.001^{***}$	1.19	0.24	$< 0.001^{***}$
Smoothness	ours vs GT	-0.16	0.22	1.000	–	–	–
	ours vs SB	0.56	0.22	0.103	0.48	0.26	0.218
	ours vs RB	1.91	0.29	$< 0.001^{***}$	1.61	0.30	$< 0.001^{***}$
	GT vs SB	0.71	0.23	0.030^{*}	–	–	–
	GT vs RB	2.07	0.27	$< 0.001^{***}$	–	–	–
	SB vs RB	1.36	0.21	$< 0.001^{***}$	1.13	0.22	$< 0.001^{***}$
Atmosphere	ours vs GT	0.08	0.20	1.000	–	–	–
	ours vs SB	0.38	0.20	0.424	0.48	0.28	0.284
	ours vs RB	1.63	0.32	$< 0.001^{***}$	0.91	0.40	0.091
	GT vs SB	0.30	0.17	0.571	–	–	–
	GT vs RB	1.56	0.27	$< 0.001^{***}$	–	–	–
	SB vs RB	1.26	0.25	$< 0.001^{***}$	0.43	0.26	0.299
Surprise	ours vs GT	0.34	0.21	0.699	–	–	–
	ours vs SB	0.97	0.24	0.002^{**}	0.47	0.29	0.346
	ours vs RB	2.12	0.31	$< 0.001^{***}$	1.09	0.35	0.012^{*}
	GT vs SB	0.62	0.17	0.006^{**}	–	–	–
	GT vs RB	1.78	0.28	$< 0.001^{***}$	–	–	–
	SB vs RB	1.16	0.27	0.001^{**}	0.62	0.18	0.004^{**}
Overall	ours vs GT	0.16	0.19	1.000	–	–	–
	ours vs SB	0.64	0.18	0.008^{**}	0.47	0.26	0.226
	ours vs RB	1.84	0.27	$< 0.001^{***}$	1.24	0.31	0.001^{**}
	GT vs SB	0.48	0.15	0.022^{*}	–	–	–
	GT vs RB	1.68	0.24	$< 0.001^{***}$	–	–	–
	SB vs RB	1.20	0.20	$< 0.001^{***}$	0.77	0.18	0.001^{**}

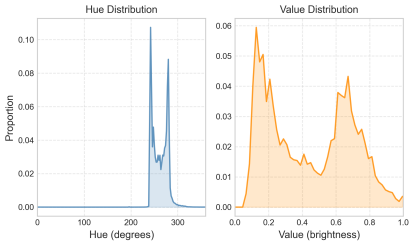
per-light configurations can yield the same aggregated distribution, the spatial arrangement of the generated lights may differ from that of the ground-truth goal. This positional discrepancy does not adversely affect performance, as our method incorporates each light's previous frame state as a constraint to ensure temporal smoothness and practical control stability. Consequently, even when the generated light positions deviate from those in the goal, the resulting control sequence remains both valid and coherent.



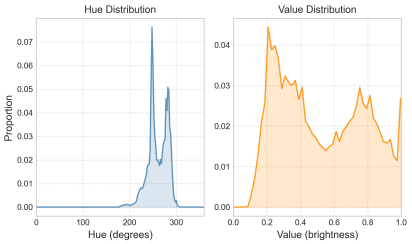
(a) IMPACT MOTION Livehouse, Chongqing, China, 2021



(b) ALSOLIVE Livehouse, Foshan, Guangdong, China, 2025



(c) HV Distribution (computed from a)



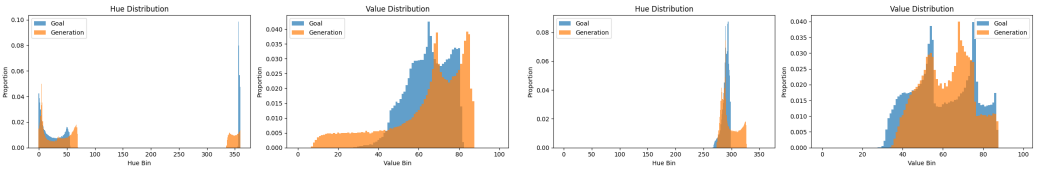
(d) HV Distribution (computed from b)

Figure 6: The Live Performance of Chinese Rock Band Mekader.

H Discussions

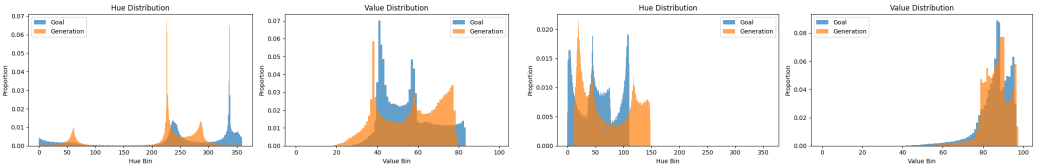
In this section, we discuss the potential limitations of this work and outline promising directions for future research. Although this paper presents the first color-space multi-light ASLC method, several simplifying assumptions are made. For instance, we assume all lights are point sources, thereby ignoring their directional properties. Moreover, in our experiments, we only consider a simple simulation setup with eight point lights, which simplifies the spatial relationships among lights. Future work could aim to realize this approach in real livehouse venues by utilizing cameras for data collection. A further simplification concerns color representation: we fix saturation at 100% following Skip-BART [4], since the dataset is extracted from real performance videos where the colors observed under normal exposure are close to their fully saturated versions, whereas strong lighting can shift the extracted values toward gray and distort the hue. Saturation is not a structural part of the framework, and supporting it would amount to adding one more output dimension to both Stage 1 and the Stage-2 policy, handled in the same way as hue and value. We keep it fixed because the expected gain on stages where pure colors dominate is limited relative to the added control complexity.From a technical perspective, several avenues for future research emerge. First, the light direction and cross-frame temporal relationships could be modeled more explicitly. For example, incorporating light direction into the action space could enhance realism, although this would significantly increase the complexity of the problem. Second, we currently perform light decomposition independently for each frame. While we enforce temporal consistency through our proposed constrained sampling strategy, there remains room for improvement. One promising direction is to formulate the task as a Multi-Agent Reinforcement Learning (MARL) problem, treating each light as an independent agent. However, this introduces new challenges, such as achieving effective cooperation among agents, and may require incorporating music information into the state space. Although such an approach could potentially improve the alignment between music and lighting effects, it would also substantially increase training difficulty and computational cost.

Additionally, most existing ASLC methods, including ours, do not support online control due to high computational demands and the requirement of processing the entire music sequence as input. While online control remains an important direction for future investigation, we argue that offline control still holds significant practical value. For instance, many live performances rely on pre-programmed lighting and VJ setups, where a manual trigger or click track is necessary to maintain synchronism between the artists' live performance and the pre-computed elements. In such contexts, pre-computed lighting control is entirely acceptable.



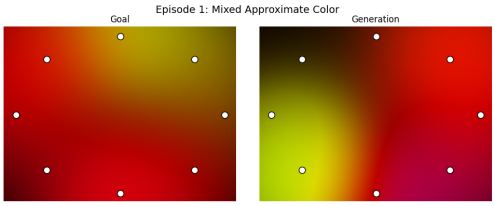
(a) Case 1 Histogram

(b) Case 2 Histogram



(c) Case 3 Histogram

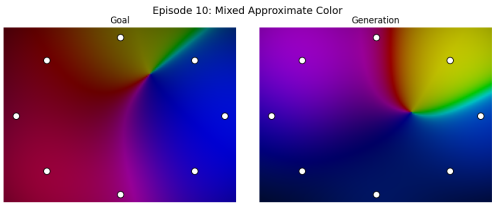
(d) Case 4 Histogram



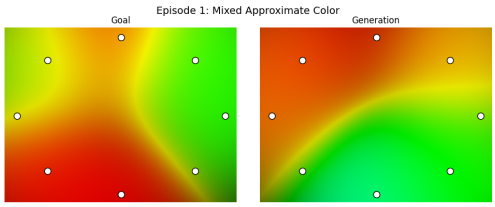
(e) Case 1 Light



(f) Case 2 Light



(g) Case 3 Light



(h) Case 4 Light

Figure 7: Visualization results of goal-conditioned light decomposition. In the histogram plots, blue regions represent the target goal distribution, while orange regions show the distribution produced by our method. For each light case, the left image depicts the ground-truth goal and the right image shows the generated result.

Finally, although our method outputs lighting control in a color space (HV) that can adapt to various venues, a human operator is still required to convert these per-light color values into low-level lighting control parameters (e.g., DMX signals). Future research could explore fully automating this conversion process, thereby further reducing manual intervention.

High-end lighting design is an art that depends on creative judgment, and the intended role of SeqLight is to augment that work rather than to stand in for it. The clearest value lies in settings where commissioning a professional lighting engineer is not an option at all, such as small and independent venues, amateur and student performers, rehearsals, and the pre-visualization stage of a production; there the method widens access to expressive lighting instead of displacing existing labor. For professional engineers the system functions as a drafting aid, handling the routine decomposition of a target color distribution into individual light settings so that attention can stay on artistic intent. The corresponding risk is a reduction in demand for some manual lighting design work, which we note as a social consideration of the automation this method contributes to.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No]" or "[N/A]" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:The abstract and introduction accurately reflect the paper's contributions by proposing the first color-space music-to-multi-light generation framework via imitation learning.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Appendix H.

   Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: As mentioned in Section 2, this paper assumes that all lights are point light sources.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides the code, data, and implementation details needed to reproduce the main experimental results in Appendix D and the anonymous repository `https://anonymous.4open.science/r/SeqLight-23EE`.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We have provided our code at anonymous repository `https://anonymous.4open.science/r/SeqLight-23EE`.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: The paper reports the mentioned evaluation protocols needed to understand the experimental results in Appendix D.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: This paper provided standard deviations in tables.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We have proposed these computation resource in Appendix D.4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: [N/A]

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: This paper provides discussion about broader impacts in Appendix H.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: [N/A]

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have cited the original paper [4] which proposed the dataset utilized in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: The paper provides the full participant instructions, questionnaire details and compensation details in Appendix F.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: The human evaluation was low-risk; participants were informed of the procedure and potential visual discomfort risks. The study was exempt from formal IRB approval under applicable institutional guidelines.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: This paper uses LLMs solely to check grammar and spelling.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.