2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

# Physically-Based Lighting Generation for Robotic Manipulation

Shutong Jin[1*], Lezhong Wang[2*], Ben Temming[1], Florian T. Pokorny[1]

***Abstract*— We propose the first framework that leverages physically-based inverse rendering for novel lighting generation on existing real-world human demonstrations of robotic manipulation tasks. Specifically, inverse rendering decomposes the first frame in each demonstration into geometric (surface normal, depth) and material (albedo, roughness, metallic) properties, which are then used to render appearance changes under different lighting sources. To improve efficiency and maintain consistency across each generated sequence, we fine-tune Stable Video Diffusion on robot execution videos for temporal lighting propagation. We evaluate our framework by measuring the visual quality of the generated sequences, assessing its effectiveness in improving the imitation learning policy performance (38.75%) under six unseen real-world lighting conditions, and conducting ablation studies on individual modules of the proposed framework. We further showcase three downstream applications enabled by the proposed framework: background generation, object texture generation and distractor positioning.**

## I. Introduction

Imitation learning from large-scale human demonstrations has proven to be an effective approach to deploying robotic manipulation tasks [1]. Yet collecting such data is costly, as it often needs to cover both diverse motor skills and varied visual appearances [2], [3]. In a fixed object-environment setup, achieving reliable policy performance on a single skill typically requires a proficient operator to repeat the task about 200 times using specialized teleoperation devices [4]. To ensure broader robustness, dataset construction further incorporates visual variations such as object texture, background, and distractors [5], [6]. Capturing these variations requires repeating the data collection process for each factor, making it especially costly to achieve sufficient visual coverage for every skill in real-world settings.

In response, substantial effort has been devoted to synthesizing object texture and background variations [2], [7], [8], [9], [10]. Inpainting techniques based on generative modeling [8] and physical tools such as green screens [9] are employed, yielding promising results while reducing the need for additional data collection. By comparison, lighting, another pervasive and highly dynamic factor in real-world settings, has received little attention. Even in the relatively controlled indoor environments, lighting can vary significantly with artificial sources such as lamps or daylight through windows. In outdoor environments, the variability is far greater, as field robots may be exposed to natural lighting changes such as those caused by the position of the sun or changes in cloud cover. Such variability points to the need for training data under different lighting. However, given the existing cost of real-world data collection, expanding datasets to cover each skill under lighting variations only compounds the challenge.

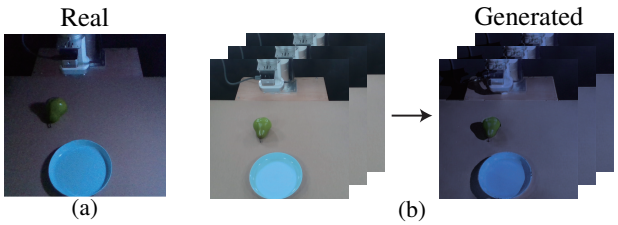



Fig. 1: (a) Example of recursive effect of lighting: when a lamp is turned off, all scene components appear darker (global impact), while nearby components alter one another's appearance by casting shadows (local impact). (b) Left: existing real-world human demonstration. Right: demonstration relit by our method. Our framework generates novel lighting for real-world demonstrations to approximate scenes under unseen lighting conditions.

Meanwhile, lighting poses a unique modeling challenge. As light travels from its source, it scatters and reflects off all scene components before ultimately being captured by a camera. Consequently, variations in lighting alter the appearance of the entire scene and influence the training process that depends on camera observations. This recursive property of lighting makes it extremely difficult to synthesize in robotic scenes. One example is shown in *Fig.* 1a: when a lamp is turned off, all scene components appear darker (global impact), while nearby components alter one another's appearance by casting shadows (local impact). Such behaviors may also help explain why many studies report that trained policies are highly vulnerable to lighting variations [3], [6], [11]. Taken together, these bring us to the question:

*Can we generate lighting that approximates real-world variations to reduce costly data collection for robotic manipulation?*

To tackle this, we propose ***RoboLightGen***, the first framework that leverages physically-based inverse rendering for **robo**tic scene **light**ing **gen**eration. Inverse rendering is introduced for explicit modeling of geometric and material information in existing real-world human demonstrations for simulating accurate light-material interactions. Our contributions are fourfold:

- **Modular Integration.** We adopt an inverse rendering

[1]KTH Royal Institute of Technology, [2]Technical University of Denmark. *Equal contribution. {shutong, temming, fpokorny}@kth.se, lezhong.wang@inria.fr. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by the supercomputing resource Berzelius provided by the National Supercomputer Centre at Linköping University and the Knut and Alice Wallenberg Foundation, Sweden. The code is available at: https://github.com/ShutongJIN/RoboLightGen.

module [12] to decompose a single demonstration frame into geometric and material properties, which are then used by a rendering module to simulate new lighting on the decomposed properties.

- **Domain-Adapted Stable Video Diffusion.** To improve efficiency and maintain consistent generation across consecutive frames of each demonstration, we fine-tune Stable Video Diffusion (SVD) [13] for temporal lighting propagation. Fine-tuning data includes synthetic robot execution videos from Factor World [3] with varied lighting across tasks, and real-world execution videos from RoboNet [14] with visual degradation for radiometric accuracy.
- **Real-world Evaluation.** We validate our framework by assessing structural and temporal consistency in generated demonstrations, and through real-world experiments on a 7-DoF robot with an embodiment unseen during SVD fine-tuning. Under six varied lighting conditions, our method improves the imitation learning policy performance by 38.75% across 1,000 evaluations on two tasks, compared to models trained without lighting generation. Ablation studies are further conducted on individual modules of the proposed framework.
- **Downstream Applications.** We showcase generations on three additional environmental factors using geometric and material properties estimated by our framework.

## II. Related Work

### A. Data Generation for Robotic Manipulation

To address the robotic data bottleneck, recent efforts [2], [7], [8], [15] have focused on semantic augmentation of real-world images leveraging text-driven generative models [16] to introduce texture variation, visual distractors, etc. For example, ROSIE [2] proposes changing object textures by first segmenting generation regions [17] and then performing text-guided image inpainting [18]. Rendering techniques [19] have also been applied for viewpoint generation [20], [21], [22]. Another line of work breaks long-horizon tasks into object-centric subtasks or manipulation skills and replays transformed demonstrations in simulation to generate new data from a limited number of examples [4], [23], [24]. In this paper, we focus on the underexplored problem of novel lighting generation on real-world human demonstrations for robotic manipulation.

### B. Inverse Rendering and Relighting

Relighting a scene typically requires identifying and altering its properties to produce the intended lighting effect [25]. Inverse rendering facilitates this by providing separate or joint estimations of geometry [26], material [27], and lighting [28] in a scene. Based on input requirements, inverse rendering can be categorized into single-view [29] and multi-view [30] methods. While methods such as DPI [31] and FIPT [32] produce high-fidelity relighting results, they rely on multi-view inputs for scene reconstruction and domain-specific datasets [33], making them incompatible with most existing robotic dataset camera setups. In this work, we present the first integration of single-view inverse rendering into robotic manipulation for explicit modeling of scene geometry and material properties.

### C. Latent Video Diffusion Models

Latent diffusion models [16] generate images by iteratively denoising Gaussian noise to approximate the target distribution. Latent video diffusion models [13], [34] extend this framework to the video domain typically by introducing temporal mixing layers on top of pre-trained image generation architectures. For example, Stable Video Diffusion [13] extends Stable Diffusion [16] by inserting temporal convolution and attention layers after each spatial layer and finetuning on curated video data. The extended temporal coherence combined with flexible conditioning (e.g., text, reference frames) enables applications such as camera-controlled video generation [35], multi-view synthesis [36], and video prediction [37]. In this work, we adopt this structure for video-to-video translation, transferring lighting in real-world human demonstrations to new lighting conditions using physically-based relit frames as reference.

## III. Methodology

### A. Formulation and Overview

Given one episode of real-world human demonstration ($\mathbf{I}$, $\mathbf{P}$, $\mathbf{A}$) recorded from a fixed viewpoint, where $\mathbf{I} \in \mathbb{R}^{H\times W\times 3\times T}$ represents a sequence of $T$ RGB images, $\mathbf{P} \in \mathbb{R}^{p\times T}$ denotes proprioception states, and $\mathbf{A} \in \mathbb{R}^{a\times T}$ corresponds to actions. Our goal is to transform $\mathbf{I}$ into $\mathbf{I}^*$ under new lighting, using physically based rendering (PBR) to ensure accurate light-material interactions.

**Overview.** As shown in *Fig.* 2 the proposed framework consists of three components: (1) we begin by selecting the first frame $I_0$ from $\mathbf{I}$ and applying inverse rendering (*Sec.* III-B) to estimate the geometric and material properties of the scene depicted in $I_0$; (2) the rendering module (*Sec.* III-C) then uses the estimated properties and an environment map $E$ to relight $I_0$, producing $I_0^*$; and (3) the temporal propagation module (*Sec.* III-D) propagates the lighting from $I_0^*$ across the full image sequence $\mathbf{I}$, resulting in the final generated sequence $\mathbf{I}^*$. This forms an generated episode ($\mathbf{I}^*$, $\mathbf{P}$, $\mathbf{A}$), which is then used to train the imitation learning policy. The environment map $E$ has two use cases: (1) approximating the current lighting conditions to train a policy adapted to the current environment (*Sec.* IV-B), and (2) introducing diverse lighting to contribute to a lighting-invariant policy (*Sec.* V).

### B. Single-Frame Inverse Rendering

We use the pre-trained network $\mathcal{P}$ from [12] to predict geometric and material properties of the scene in $I_0$:

$$A_p, R_p, M_p, N_p, D_p = \mathcal{P}(I_0), \tag{1}$$

where $A_p$, $R_p$, and $M_p$ are predicted material properties (albedo, roughness, metallic), and $N_p$ and $D_p$ are predicted geometric properties (surface normal and depth). Examples of predicted properties can be found in *Fig.* 2. All predictions

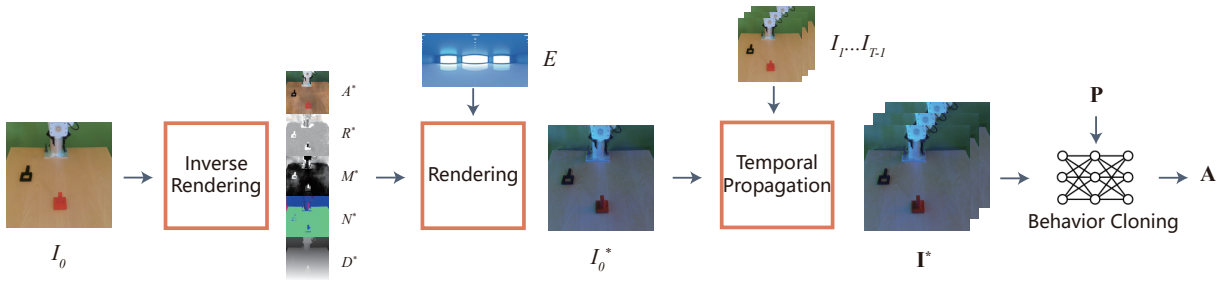



Fig. 2: Given the first frame $I_0$ in a real-world human demonstration, the inverse rendering module estimates material ($A^*$, $R^*$, $M^*$) and geometric ($N^*$, $D^*$) properties. The rendering module uses these estimates and an environment map $E$ providing new lighting to produce a relit frame $I_0^*$. The new lighting is then propagated across the entire sequence $\mathbf{I}$, producing $\mathbf{I}^*$ and forming an generated episode ($\mathbf{I}^*$, $\mathbf{P}$, $\mathbf{A}$) for behavior cloning.

share the spatial resolution of $I_0$. Final property estimates are derived by minimizing the following objective:

$$A^*, R^*, M^*, N^*, D^* = \arg \min_{A_p, R_p, M_p, N_p, D_p} \mathcal{L}_p(I_0, I_p), \quad (2)$$

where $I_p$ denotes the frame rendered using the predicted properties. $\mathcal{L}_p = \mathcal{L}_{re} + \delta \mathcal{L}_{cons}$, where $\mathcal{L}_{re}$ denotes the reconstruction loss between the rendered frame $I_p$ and the original frame $I_0$, and $\mathcal{L}_{cons}$ denotes the $L_1$ consistency loss the optimized and originally predicted properties, scaled by a factor $\delta$ set to 0.005 in most experiments.

### C. Single-Frame Relighting

To perform physically-based relighting, we render new lighting based on incoming radiance directions and modeled scene properties.

*a) Target Light Sampling:* Given an environment map $E \in \mathbb{R}^3$ containing target lighting condition, the incoming radiance $\lambda$ in direction $\omega_i$ is sampled as:

$$\lambda(\omega_i) = E(\omega_i), \quad (3)$$

where $E$ is obtained either from open-source HDRI libraries [38] (*Fig.* 3a), by optimization during inverse rendering [12] (*Fig.* 3b) or by measuring the current lighting condition using graphics techniques [39] (*Fig.* 3c).

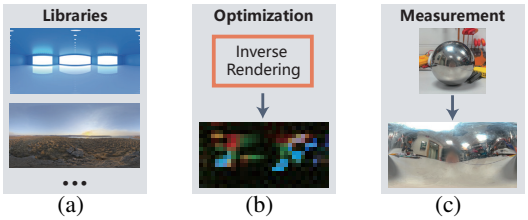



Fig. 3: (a) Environment maps from open-source HDRI libraries retrieved using keywords such as "blue studio" and "sunrise in the field". (b) Environment maps derived via optimization with inverse rendering, where resolution depends on the specific method. (c) Environment maps obtained by photographing a chrome (mirror) ball under multiple exposures and unwrapping the result.

*b) Scene Modeling:* The estimated properties from *Sec.* III-B are used to model the material appearance $\mathcal{M}$ along direction $\omega_i$ of the scene depicted in $I_0$, using the widely adopted Disney BRDF [40]:

$$\mathcal{M}(\omega_i) = (1 - M^*) f_{diffuse}(A^*, R^*, N^*) + f_{specular}(R^*, N^*), \quad (4)$$

where $f_{diffuse}$ and $f_{specular}$ represent the diffuse and specular reflection components of the Disney BRDF, respectively.

*c) Scene Relighting:* Using the simplified rendering equation [41], the generated frame $I_0^*$ under new lighting from environment map $E$ is computed as:

$$I_0^* = \int_\Omega \mathcal{M}(\omega_i) \lambda(\omega_i) (\omega_i \cdot N^*) \, \mathrm{d}\omega_i, \quad (5)$$

where $\Omega$ is the hemisphere centered at surface normal $N^*$, containing all incoming directions $\omega_i$. Note that once the property estimation for $I_0$ from *Sec.* III-B is completed, multiple generations can be applied using the same set of estimated properties.

### D. Temporal Propagation

*a) Domain-Adapted Stable Video Diffusion:* To improve efficiency and maintain consistency in the generated sequences, we adopt Stable Video Diffusion (SVD) [13] for its temporal modeling capabilities to propagate lighting from $I_0^*$ across the entire demonstration. We formulate this as a video-to-video translation task: lighting from the original image sequence $\mathbf{I}$ is transferred to generate the new sequence $\mathbf{I}^*$, guided by the reference frame $I_0^*$. This is defined as:

$$\mathbf{I}^* = \delta(I_0^*, \mathbf{I}_{input}), \quad \mathbf{I}_{input} = \text{Concatenate}(I_0^*, \mathbf{I}_{1:T-1}), \quad (6)$$

where $\delta$ denotes the SVD model.

*b) Robotic Relighting Data Curation:* It's important to note that the vanilla SVD underperforms in this task due to the absence of robotic elements like arms and grippers in its training data, resulting in a severely cartoonish appearance in generated robotic scenes. Following SVD's original fine-tuning paradigm, we construct two datasets for different stages of fine-tuning:

- Synthetic videos with lighting variation ($\mathcal{D}_1$, *Fig.* 4a). We generate domain-specific relighting data for robotic manipulation using the Factor World [3] benchmark, introducing lighting variation under identical task execution trajectories. All 42 built-in scenes covering 19 tasks (e.g., door opening) are used, each captured with

six camera views: one fixed top-down and five randomized within constrained azimuth, inclination, and radius in the range $[-\pi/2, \pi/2]$ to ensure robot visibility. Each scene includes 30 lighting conditions by sampling ambient and diffuse RGB values from $[25, 255]$. Videos are rendered at $512 \times 512$ resolution with 100 frames at 24 fps. For fine-tuning, we randomly select pairs of videos showing the same task trajectories under different lighting, using one as the input $\mathbf{I}$ and the other as the target sequence $\mathbf{I}^*$.

- Real-world videos with visual degradation ($\mathcal{D}_2$, *Fig.* 4b). Since existing real-world robotic datasets lack perfectly paired original and relit videos under identical task execution trajectories, we manually create video pairs by applying visual degradation on original videos from RoboNet [14] with 15 million video frames from 7 different robot platforms. We apply random visual transformations by sampling brightness, contrast, and saturation scaling factors from $[0.2, 1.9]$, and hue shifts from $[-0.5, 0.5]$. The degraded robot execution video serves as the input $\mathbf{I}$, while the original video acts as the output $\mathbf{I}^*$.

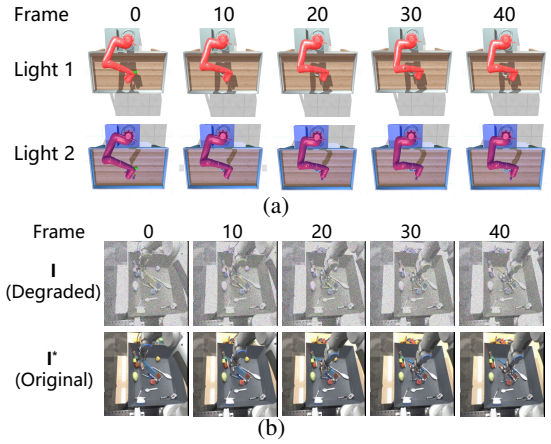


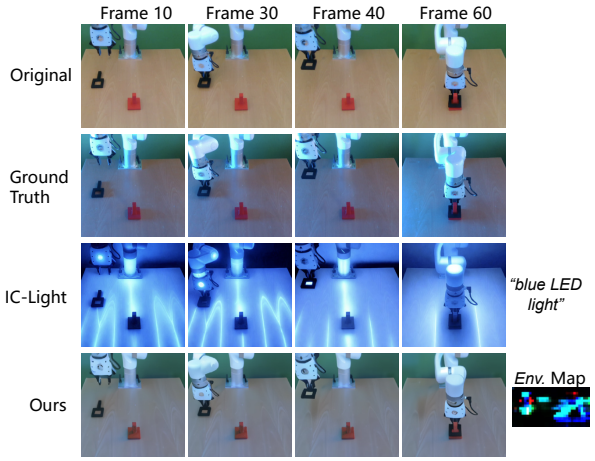

Fig. 4: (a) Examples of synthetic robot execution videos with lighting variations ($\mathcal{D}_1$). (b) Examples of real-world robot execution videos with visual degradation ($\mathcal{D}_2$).

*c) Fine-tuning:* We fine-tune SVD in two stages following [42]. In Stage 1, we use synthetic videos from $\mathcal{D}_1$, which provide the main data for domain adaptation and reduce overfitting via controlled lighting variation. However, due to the non-physically-based lighting in Factor World, this stage lacks radiometric accuracy, motivating fine-tuning on real-world videos from $\mathcal{D}_2$ in Stage 2. Fine-tuning is performed on a pretrained 14-frame SVD model using a single H100 GPU for 8,000 steps for Stage 1 and 1,000 steps for Stage 2. A limited number of steps are applied to Stage 2 due to the absence of ground-truth relit pairs.

## IV. Experiments

### A. Visual Lighting Quality Evaluation

We apply novel lighting generation to 10 randomly selected sequences (600 frames) from the real-world human demonstrations in *Sec.*IV-B using our method and IC-Light [43], a widely recognized text-prompt-based relighting method. Ground-truth episodes are recorded under a side-mounted blue LED panel light under the same task execution trajectories. Two sets of metrics are adopted: (1) *LPIPS* [44] and *SSIM* [45] are computed between the ground-truth and relit image sequences to assess structural consistency and fidelity. (2) *Temporal LPIPS* and *Temporal SSIM* are computed between consecutive frames in the relit sequences to assess temporal consistency. Since IC-Light does not support video relighting, we apply it frame-by-frame using a fixed random seed and the prompt "*blue LED light*" for consistency. We perform relighting using an environment map approximated from the ground truth episodes with [12]. Qualitative and quantitative evaluation results can be found in *Fig.* 5. Our method performs better by preserving structural similarity with the original frames. The use of environment maps for target lighting provides precise control over relighting, making it more effective than textual prompts when approximating real-world data under varying lighting conditions. The temporal propagation module further enhances visual consistency by reducing abrupt changes between frames. Beyond visual quality evaluation, IC-Light is also included in subsequent real-world comparisons in *Sec.* IV-B.




| Method | *LPIPS* (↓) | *SSIM* (↑) | *Temporal LPIPS* (↓) | *Temporal SSIM* (↑) |
|---|---|---|---|---|
| IC-Light | 0.5274 | 0.5359 | 0.121 | 0.898 |
| Ours | **0.3269** | **0.7351** | **0.035** | **0.978** |

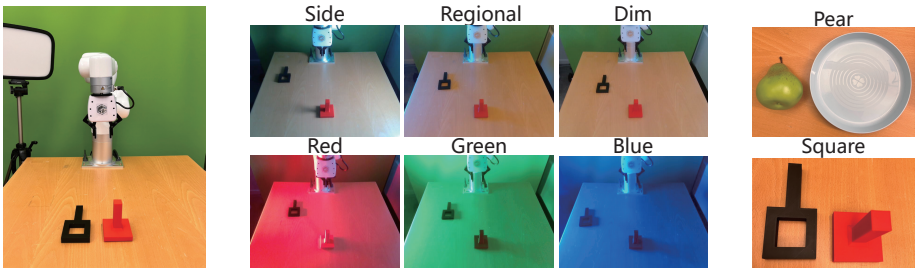
Fig. 5: Qualitative (top) and quantitative (bottom) evaluation of lighting generation. Although exceeding the baseline, the lighting in our generated episodes appears more diffused compared to the ground truth, as indicated by weaker side shadows. This discrepancy mainly results from the rough environment map approximation ($32 \times 16$ resolution) without specialized graphics equipment.

### B. Real-World Evaluation

This experiment examines whether imitation learning policies perform better under unseen lighting when trained on episodes that approximate such conditions.

*1) Implementation Details:*

*a) Lighting Setup:* White light (*Original*, *Fig.* 6a) used in most existing robotic manipulation datasets is adopted




(a) (b) (c)

Fig. 6: (a) Experimental setup under *Original* lighting. (b) The six lighting conditions used for policy evaluation under unseen lighting. (c) Objects used in the two manipulation tasks.

TABLE I: Success rate with different generation methods under unseen lighting. The best-performing policy is in bold, and the second-best is underscored.

| **Lighting** | *Crop* | | | | *Jitter* | | | | *IC-Light* | | | | *Ours* | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *Pear* | | *Square* | | *Pear* | | *Square* | | *Pear* | | *Square* | | *Pear* | | *Square* | |
| | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* | *R* | *PnP* |
| *Original* | 1 | 0.60 | 1 | 0.85 | / | / | / | / | / | / | / | / | / | / | / | / |
| *Side* | 0 | 0 | 0 | 0 | **0.3** | 0 | 0.15 | 0 | 0.05 | 0 | 0.2 | 0 | 0.05 | 0 | **0.9** | **0.4** |
| *Regional* | 0.05 | 0 | 0 | 0 | 0.45 | 0 | 0.5 | 0 | 0.10 | 0 | 0.15 | 0 | **1.0** | **0.2** | **1.0** | **0.45** |
| *Dim* | **1.0** | **0.5** | 0.95 | 0.55 | 0.9 | 0.1 | 0.95 | 0.45 | 0.20 | 0 | 0.25 | 0 | **1.0** | 0.4 | **1.0** | **0.8** |
| *Red* | 0 | 0 | 0 | 0 | 0.3 | 0 | 0.1 | 0 | 0.45 | 0 | 0.3 | 0 | **0.6** | 0 | **0.9** | 0 |
| *Green* | 0.15 | 0 | 0.1 | 0 | 0 | 0 | 0.05 | 0 | 0.25 | 0 | 0.1 | 0 | **1.0** | 0 | **1.0** | **0.1** |
| *Blue* | 0 | 0 | 0.05 | 0 | 0 | 0 | 0.45 | 0 | 0.4 | 0 | 0.15 | 0 | **0.85** | **0.05** | **0.95** | 0 |

The vanilla BC-MLP with random cropping augmentation (*Crop*) evaluated under *Original* lighting is added as the reference of the task difficulty.

during expert human demonstration collection for policy training. During evaluation, six lighting conditions are designed using a programmable RGB LED panel. This includes two sets with different evaluation purposes: (1) Daily-use conditions—*Side* (strong shadows), *Regional* (blue regional highlight), and *Dim* (low-light); (2) Artificial RGB lighting—*Red*, *Green*, and *Blue*—with illuminance levels of 4,600 lux and RGB values (255, 0, 0), (0, 255, 0), and (0, 0, 255), respectively, designed to evaluate each policy under single-source lighting. These colors are chosen based on the principle of light transport linearity, which allows any colored illumination to be reconstructed as their linear combination. Examples are shown in *Fig.* 6b.

*b) Manipulation Tasks:* (1) *Pear*, grasp a randomly placed plastic pear using a vacuum gripper and place it on a plate. (2) *Square*, grasp a randomly placed black square nut by its handle using a parallel jaw gripper and insert it into a red square peg. To better assess performance under challenging lighting conditions, we report performance separately for the reaching (*R*) subtask, which involves reaching the area of the pear or square nut, and the pick-and-place (*PnP*) outcome, defined as successfully picking up the object and placing it in the corresponding plate or peg.

*c) Baselines:* We perform behavior cloning with ResNet18 [46] and a Multi-Layer Perceptron (BC-MLP) to map features to actions. This architecture is chosen for its widespread use [47], [48] and simplicity, providing a clean testbed to evaluate the impact of generated data. Training is conducted on a 7-DoF UFactory XArm7 robot observed by a single externally mounted RGB camera. We compare four baselines: (a) *Crop*, vanilla BC-MLP with random cropping, trained on 200 human demonstration episodes under *Original* lighting; (b) *Jitter*, random cropping plus color jitter, trained on the same 200 episodes; (c) *IC-Light*, trained on the original 200 episodes plus generated episodes approximating unseen lighting conditions with IC-Light; and (d) *Ours*, trained on the original 200 episodes plus generated episodes from our method. For *IC-Light* and *Ours*, 10 demonstrations are randomly selected and relit with six environment maps matching the six evaluation conditions. *Crop* is used as the default augmentation in all baselines due to its role in improving BC robustness. Color jitter is included as a baseline due to its ability to mitigate color and illumination variation.

*d) Evaluation Protocol:* For each combination of task, lighting condition, and method, we conduct 20 real-world evaluations and report their success rates. *Pear* (*Fig.* 1a) and *Square* (*Fig.* 6a) differ slightly in background, and the wrist camera is included only for automatic evaluation, not for training purposes. All baselines within a task are evaluated in the same environment.

*2) Results: Tab.* I presents the success rates of BC-MLP under six unseen lighting conditions across 1,000 real-world evaluations. Averaged over both subtasks across the two tasks, our method outperforms *Crop* by 38.75%, *Jitter* by 33.13% and *IC-Light* by 41.88%. In the daily-use lighting set, *Crop* shows a major drop in performance, except under *Dim*. While *Jitter* performs reasonably on the reaching subtask, it consistently fails in the final pick-and-place outcome. *IC-Light* exhibits reduced performance on daily-use lighting but better performance on RGB lighting, possibly because it generates episodes that align more closely with RGB lighting. *Ours* improves performance across both subtasks under these conditions. In the artificial RGB lighting set,

both *Crop* and *Jitter* fail in most cases, even on the reaching subtask. Interestingly, *Jitter* displays strong preferences for different RGB lighting in the two tasks, likely due to its use of global color shifts. Our method maintains strong reaching performance but struggles with pick-and-place in these more extreme lighting conditions. Common failure modes for *Crop*, *Jitter* and *IC-Light* involve moving directly to the plate or peg without interacting with the pear or nut. Failures in *Ours* often involve reaching the pear or nut with the gripper open and hovering just above the object, without executing a pick action. This may relate to the altered object appearance under high-intensity lighting (4,600 lux), where high reflectivity could disrupt perception. See the limitations section for further details and the supplementary video for examples of common failure modes across different methods.

### C. Ablation Study

We provide ablation study on individual modules of the proposed framework: Inverse Rendering (*Sec.* IV-C.1) and Temporal Propagation (*Sec.* IV-C.2, *Sec.* IV-C.3).

*1) Material Property Ablation:* As shown in *Tab.* II, we ablate material properties by individually masking estimated albedo, roughness, and metallic to 0.5 (originally in [0, 1]) and evaluate the trained policy's success rate (*R*/*PnP*) over 10 trials on task *Square* under *Dim*. Policy trained on the original estimates is included for comparison. Results suggest albedo and roughness are critical for policy performance, while metallic has minimal effect, likely due to the non-metallic object. Masking any property leads to reduced performance.

TABLE II: Material property ablation.

| *Original* | *Albedo* | *Roughness* | *Metallic* |
|---|---|---|---|
| 1.0/0.8 | 0.8/0 | 1.0/0 | 1.0/0.4 |

*2) SVD Fine-tuning Data Ablation:* The fine-tuning data for SVD consist of two sources: synthetic robot execution videos, which provide varied lighting across tasks ($\mathcal{D}_1$), and real-world execution videos, which include visual degradation to improve radiometric accuracy ($\mathcal{D}_2$). In *Fig.* 7, we show an example frame generated by SVD when fine-tuned only on $\mathcal{D}_1$. The result exhibits low radiometric accuracy and a noticeably cartoonish appearance.

Fig. 7: SVD fine-tuning data ablation.

*3) Temporal Propagation Ablation: Tab.* III shows the comparison of relighting time: IC-Light, frame-wise inverse rendering (*Ours (w/o SVD)*), and the proposed first frame inverse rendering + SVD (*Ours (w/ SVD)*). With SVD, after a one-time 10-minute inverse rendering, relighting each episode takes only 20 seconds. The efficiency gain becomes greater when relighting an episode with multiple lighting, e.g. 6 relightings take just 12 minutes: $10 + (20/60) \times 6$.

TABLE III: Time efficiency comparison.

| Methods | *1 Frame* | *1 Episode (61 Frames)* | *6 Episodes* |
|---|---|---|---|
| *IC-Light* | 10 s | 10 min | 1 hour |
| *Ours (w/o SVD)* | 10 min | 10 hours | 60 hours |
| *Ours (w/ SVD)* | 10 min | 10 min + 20 s | **12 min** |

## V. Applications

Through scene decomposition via inverse rendering, we extend our framework to generate three additional environment factors (background, object texture, and distractors) beyond lighting (*Fig.* 8) on existing datasets, as demonstrated on BridgeData v2 [5].

Fig. 8: Examples of lighting generation with different environment maps on existing open-source dataset.

### A. Background Generation

Background appearance is tied to illumination; therefore, we render segmented scene geometry with different environment maps to create new backgrounds with corresponding lighting. Scene geometry is reconstructed by triangulating the depth map $D^*$ from *Sec.* III-B into a mesh $\Lambda$:

$$\Lambda = \text{Triangulate}\left(K^{-1}[x \;\; y \;\; 1]^\top D^*(x,y)\right), \tag{7}$$

where $K$ is a fixed intrinsic matrix of a pinhole camera, and $D^*(x,y)$ is the depth at pixel $(x,y)$. We then project 2D segmentation masks [49] onto the mesh $\Lambda$ from the default viewpoint to isolate and segment the robot arm and task area. The segmented mesh is then rendered with various environment maps to produce new backgrounds and lighting. Examples are shown in *Fig.*9.


Fig. 9: Examples of background generation with different environment maps from [38].

### B. Object Texture Generation

Robotic tasks like grasping and pushing rely on consistent object geometry and frictional properties [50]. We address this by applying object texture generation with preserved visual roughness and geometry. Following the mesh segmentation in *Sec.* V-A, we segment the object mesh and render with altered material properties. Specifically, albedo $A^*$ sets the base color, roughness $R^*$ controls surface scattering, and metallic $M^*$ defines the metallic effect. By adjusting only the albedo $A^*$ while keeping roughness $R^*$ and metallic $M^*$ fixed, and reapplying *Eq.* 4, we achieve object texture generation with consistent visual roughness and geometry. Examples are shown in *Fig.* 10.

Fig. 10: Examples of texture generation via albedo adjustment.

### C. Distractor Placement in Cluttered Environments

Through the meshes of the scene and distractors, we conduct physically plausible distractor placement in Blender [51]. As shown in *Fig.* 11, we first align the surface normal with Blender's default gravity axis (Z-axis). This requires the user to manually select a mesh face whose surface normal is parallel to the gravity direction using the Blender API. A random position within the bounding box of $\Lambda$ (*Eq.* 7) is sampled, and the distractor mesh (assigned a pseudo-mass of 1kg at its geometric center) is dropped to simulate its motion under gravity. The final placement of the distractor is determined by its stable resting state following the gravity-based simulation. The full simulation process is available in the supplementary video.

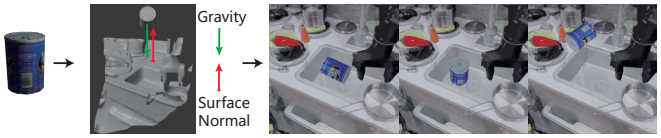



Fig. 11: Examples of gravity-based distractor placement, with distractor mesh sourced from [52].

## VI. Limitations and Conclusion

### A. Limitations

Although we provide physically-based lighting generation by simulating light–material interactions, the absence of an environment map that fully reconstructs the observed lighting leads to reduced performance. As shown in *Fig.* 12, under high-intensity red LED lighting (4600 lux), the plastic pear and plate exhibit strong reflections not captured by our generation framework. This discrepancy arises because the side-mounted LED produces strong directional lighting, whereas the approximated environment map ($32 \times 16$ resolution) provides mostly diffuse illumination with a shifted color tone, which may also explain the limited success rates under artificial lighting. Capturing lighting using graphical equipment [39] to generate accurate environment maps could mitigate this issue. For compatibility with existing datasets, we employ a single-frame inverse rendering framework, though incomplete geometry reconstruction can lead to shadow artifacts under certain lighting directions. Comparable artifacts may also occur in SVD-propagated episodes, since geometry is not estimated during propagation and artifacts may appear when frames differ substantially from the initial frame. This choice represents a compromise driven by the costly estimation process described in *Sec.* IV-C.3.

### B. Conclusion

In this paper, we propose the first framework that leverages physically-based inverse rendering for novel lighting generation on existing real-world human demonstrations. By decomposing robotic scenes into geometric and material properties, we perform lighting generation on a single frame and propagate it across the demonstration using finetuned Stable Video Diffusion. We validate our framework through qualitative and quantitative visual quality evaluations, 1,000 real-world trials under six varied lighting conditions using a 7-DoF robot, and ablation studies on individual modules. We further showcase generations on three additional visual variations enabled by our framework. Our framework improves performance under unseen lighting by generating approximated episodes, and future work will investigate strategies for policies that operate consistently across diverse illumination settings.

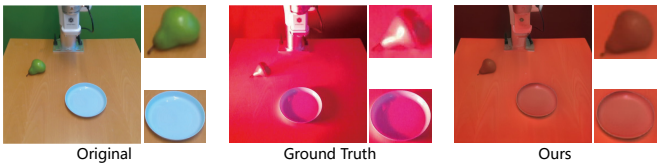



Fig. 12: Failure cases observed under high-intensity lighting.

## References


[1] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter *et al.*, "$\pi_0$: A vision-language-action flow model for general robot control," *arXiv preprint arXiv:2410.24164*, 2024.

[2] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, J. Peralta, B. Ichter *et al.*, "Scaling robot learning with semantically imagined experience," *arXiv preprint arXiv:2302.11550*, 2023.

[3] A. Xie, L. Lee, T. Xiao, and C. Finn, "Decomposing the generalization gap in imitation learning for visual robotic manipulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 3153–3160.

[4] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox, "Mimicgen: A data generation system for scalable robot learning using human demonstrations," *arXiv preprint arXiv:2310.17596*, 2023.

[5] H. Walke, K. Black, A. Lee, M. J. Kim, M. Du, C. Zheng, T. Zhao, P. Hansen-Estruch, Q. Vuong, A. He, V. Myers, K. Fang, C. Finn, and S. Levine, "Bridgedata v2: A dataset for robot learning at scale," in *Conference on Robot Learning (CoRL)*, 2023.

[6] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao, "Data scaling laws in imitation learning for robotic manipulation," *arXiv preprint arXiv:2410.18647*, 2024.

[7] Z. Mandi, H. Bharadhwaj, V. Moens, S. Song, A. Rajeswaran, and V. Kumar, "Cacti: A 578 framework for scalable multi-task multi-scene visual imitation learning. arxiv preprint 579," *arXiv preprint arXiv:2212.05711*, vol. 580, 2022.

[8] Z. Chen, S. Kiami, A. Gupta, and V. Kumar, "Genaug: Retargeting behaviors to unseen situations via generative augmentation," *arXiv preprint arXiv:2302.06671*, 2023.

[9] E. Teoh, S. Patidar, X. Ma, and S. James, "Green screen augmentation enables scene generalisation in robotic manipulation," *arXiv preprint arXiv:2407.07868*, 2024.

[10] C. Yuan, S. Joshi, S. Zhu, H. Su, H. Zhao, and Y. Gao, "Roboengine: Plug-and-play robot data augmentation with semantic robot segmentation and background generation," *arXiv preprint arXiv:2503.18738*, 2025.

[11] E. Xing, A. Gupta, S. Powers, and V. Dean, "Kitchenshift: Evaluating zero-shot generalization of imitation-based policy learning under domain shifts," in *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

[12] L. Wang, D. M. Tran, R. Cui, T. TG, M. Chandraker, and J. R. Frisvad, "Materialist: Physically based editing using single-image inverse rendering," *arXiv preprint arXiv:2501.03717*, 2025.

[13] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts *et al.*, "Stable video diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023.

[14] S. Dasari, F. Ebert, S. Tian, S. Nair, B. Bucher, K. Schmeckpeper, S. Singh, S. Levine, and C. Finn, "Robonet: Large-scale multi-robot learning," *arXiv preprint arXiv:1910.11215*, 2019.

[15] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar, "Roboagent: Generalization and efficiency in robot manipulation via semantic augmentations and action chunking," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 4788–4795.

[16] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[17] M. Minderer, A. Gritsenko, A. Stone, M. Neumann, D. Weissenborn, A. Dosovitskiy, A. Mahendran, A. Arnab, M. Dehghani, Z. Shen *et al.*, "Simple open-vocabulary object detection," in *European Conference on Computer Vision*. Springer, 2022, pp. 728–755.

[18] S. Wang, C. Saharia, C. Montgomery, J. Pont-Tuset, S. Noy, S. Pellegrini, Y. Onoe, S. Laszlo, D. J. Fleet, R. Soricut *et al.*, "Imagen editor and editbench: Advancing and evaluating text-guided image inpainting," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 18 359–18 369.

[19] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[20] L. Y. Chen, C. Xu, K. Dharmarajan, M. Z. Irshad, R. Cheng, K. Keutzer, M. Tomizuka, Q. Vuong, and K. Goldberg, "Roviaug: Robot and viewpoint augmentation for cross-embodiment robot learning," *arXiv preprint arXiv:2409.03403*, 2024.

[21] X. Zhang, M. Chang, P. Kumar, and S. Gupta, "Diffusion meets dagger: Supercharging eye-in-hand imitation learning," *arXiv preprint arXiv:2402.17768*, 2024.

[22] A. Zhou, M. J. Kim, L. Wang, P. Florence, and C. Finn, "Nerf in the palm of your hand: Corrective augmentation for robotics via novel-view synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 17 907–17 917.

[23] Z. Jiang, Y. Xie, K. Lin, Z. Xu, W. Wan, A. Mandlekar, L. Fan, and Y. Zhu, "Dexmimicgen: Automated data generation for bimanual dexterous manipulation via imitation learning," *arXiv preprint arXiv:2410.24185*, 2024.

[24] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu, "Robocasa: Large-scale simulation of everyday tasks for generalist robots," *arXiv preprint arXiv:2406.02523*, 2024.

[25] D. Azinovic, T.-M. Li, A. Kaplanyan, and M. Nießner, "Inverse path tracing for joint material and lighting estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 2447–2456.

[26] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2650–2658.

[27] Z. Li and N. Snavely, "Cgintrinsics: Better intrinsic image decomposition through physically-based rendering," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 371–387.

[28] M.-A. Gardner, K. Sunkavalli, E. Yumer, X. Shen, E. Gambaretto, C. Gagné, and J.-F. Lalonde, "Learning to predict indoor illumination from a single image," *arXiv preprint arXiv:1704.00090*, 2017.

[29] Z. Li, J. Shi, S. Bi, R. Zhu, K. Sunkavalli, M. Hašan, Z. Xu, R. Ramamoorthi, and M. Chandraker, "Physically-based editing of indoor scene lighting from a single image," in *European Conference on Computer Vision*. Springer, 2022, pp. 555–572.

[30] Y. Yao, J. Zhang, J. Liu, Y. Qu, T. Fang, D. McKinnon, Y. Tsin, and L. Quan, "Neilf: Neural incident light field for physically-based material estimation," in *European Conference on Computer Vision*. Springer, 2022, pp. 700–716.

[31] L. Lyu, A. Tewari, M. Habermann, S. Saito, M. Zollhöfer, T. Leimkühler, and C. Theobalt, "Diffusion posterior illumination for ambiguity-aware inverse rendering," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 6, pp. 1–14, 2023.

[32] L. Wu, R. Zhu, M. B. Yaldiz, Y. Zhu, H. Cai, J. Matai, F. Porikli, T.-M. Li, M. Chandraker, and R. Ramamoorthi, "Factorized inverse path tracing for efficient and accurate material-lighting estimation," in *Proceedings of International Conference on Computer Vision (ICCV)*. IEEE, 2023, pp. 3848–3858.

[33] J. J. Park, A. Holynski, and S. M. Seitz, "Seeing the world in a bag of chips," in *Proceedings of Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020, pp. 1417–1427.

[34] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis, "Align your latents: High-resolution video synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 22 563–22 575.

[35] H. He, Y. Xu, Y. Guo, G. Wetzstein, B. Dai, H. Li, and C. Yang, "Cameractrl: Enabling camera control for text-to-video generation," *arXiv preprint arXiv:2404.02101*, 2024.

[36] V. Voleti, C.-H. Yao, M. Boss, A. Letts, D. Pankratz, D. Tochilkin, C. Laforte, R. Rombach, and V. Jampani, "Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion," in *European Conference on Computer Vision*. Springer, 2024, pp. 439–457.

[37] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 14 749–14 759.

[38] Poly Haven, "Poly haven hdris," 2025, accessed: 2025-03-19. [Online]. Available: https://polyhaven.com/hdris

[39] P. Debevec, "Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *Acm siggraph 2008 classes*, 2008, pp. 1–10.

[40] B. Burley and W. D. A. Studios, "Physically-based shading at disney," in *Acm Siggraph*, vol. 2012, no. 2012. vol. 2012, 2012, pp. 1–7.

[41] J. T. Kajiya, "The rendering equation," in *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, 1986, pp. 143–150.

[42] L. Wang, S. Jin, R. Cui, A. B. Dahl, J. R. Frisvad, and S. Bigdeli, "Relumix: Extending image relighting to video via video diffusion models," *arXiv preprint arXiv:2509.23769*, 2025.

[43] L. Zhang, A. Rao, and M. Agrawala, "Scaling in-the-wild training for diffusion-based illumination harmonization and editing by imposing consistent light transport," in *The Thirteenth International Conference on Learning Representations*.

[44] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[47] S. Dasari, M. K. Srirama, U. Jain, and A. Gupta, "An unbiased look at datasets for visuo-motor pre-training," in *Conference on Robot Learning*. PMLR, 2023, pp. 1183–1198.

[48] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, "What matters in learning from offline human demonstrations for robot manipulation," *arXiv preprint arXiv:2108.03298*, 2021.

[49] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[50] J. Mahler, F. T. Pokorny, B. Hou, M. Roderick, M. Laskey, M. Aubry, K. Kohlhoff, T. Kröger, J. Kuffner, and K. Goldberg, "Dex-net 1.0: A cloud-based network of 3d objects for robust grasp planning using a multi-armed bandit model with correlated rewards," in *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 1957–1964.

[51] Blender Online Community, "Blender - a 3d modelling and rendering package," https://www.blender.org, 2018.

[52] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: The ycb object and model set and benchmarking protocols," *arXiv preprint arXiv:1502.03143*, 2015.