Improved Cinematic-Guided Camera Language Transfer in 3D Scene

Yuan Wang, Zhuoling Jiang, Bailin Deng, Yipeng Qin
Cardiff University, UK
{WangY559, JiangZ30, DengB3, QinY16}@cardiff.ac.uk

Abstract

Directors and cinematographers often recreate iconic scenes by replicating the underlying camera language to evoke shared aesthetic and narrative meaning. In this work, we refer to this as the task of Cinematic-Guided Camera Language Transfer, where the goal is to reproduce the cinematic camera language of a reference video clip in a new 3D scene. The pioneer work, Jaws [62], tackles this problem by adapting generic computer vision methods but fails to model the essential principles of cinematography, often leading to inaccurate framing, motion mismatches, and loss of expressive intent. To overcome these limitations, we systematically define the objectives of camera language transfer, grounding them in professional cinematography literature. Specifically, we conduct an in-depth review of cinematography literature to identify eight key cinematic features and encode them into five novel camera language losses. These losses not only guide optimization of camera parameters for effective transfer, but also serve as quantitative metrics for evaluating cinematographic fidelity. Extensive experiments demonstrate the superiority of our method.

1. Introduction

Throughout film history, directors and cinematographers have frequently paid visual homage to iconic scenes by recreating the key *camera language* such as signature camera trajectories, compositions, and framings to evoke shared aesthetic or narrative meaning (*e.g.*, the dolly zoom from *Vertigo* [19] or the suspenseful tracking shots in *Jaws* [55]). With the rise of virtual production tools, it has become common to replicate such cinematic effects in simulated 3D environments (*e.g.*, NeRF [45], 3DGS [26], Unity [58]) before principal photography [2, 8], and to train robots [12, 42, 49]. In this work, we refer to this as the task of **Cinematic-Guided Camera Language Transfer**: given a reference video clip and a new 3D scene, the objective is to reproduce the cinematic camera language of the reference clip within the new scene, such that the re-rendered video conveys a consistent cinematic visual style.

Jaws [62], a pioneering effort in this direction, address this task by formulating it as a camera parameters (both extrinsics and intrinsics) optimization problem. Specifically, they define camera language losses (*i.e.*, the objective function) as an on-screen loss(full-body pose matching) and an inter-frame loss(optical-flow matching). While promising, their approach largely relies on a naive adaptation of existing computer vision techniques, rather than adhering to principles of cinematic camera language. As a result, Jaws [62] easily breaks down, leaving a critical gap in capturing the expressive cinematographic intent. For example, naive human pose matching using all skeleton joints is highly sensitive to pose variation, causing mismatched shot size and framing; likewise, global optical flow ignores motion parallax, conflating near and far motions and weakening supervision on camera-induced depth-dependent dynamics. Moreover, Jaws overlooks key cinematic features, such as filmic space and camera angle, thereby limiting its ability to reproduce authentic cinematic visual styles.

In this work, we address the above-mentioned limitations by systematically defining the *objectives* of cinematic-guided camera language transfer, explicitly grounding them in professional cinematography literature [9, 44] that prior approaches have overlooked. Specifically, we first review the cinematography literature [9, 44] and identify 8 key cinematic features for camera language, including (i) shot size (how much of the frame the subject occupies); (ii) framing (the subject's screen position); (iii) camera angle (relative orientation to subjects); (iv) camera movement (frame-to-frame motion cues); (v) lens choice (perceived depth and spatial compression); (vi) camera position (camera-to-subject location); (vii) zooming (cynamically focal) and (viii) focus (the effect of depth of field). Then, we carefully examine them and capture these 8 features using 5 novel camera language losses, including (i) shot size loss; (ii) framing loss; (iii) filmic space loss; (iv) camera movement loss; (v) camera angle loss; using computer vision techniques. Similar to Jaws [62], we formulate the task as optimizing camera parameters under our novel camera language losses, which enables more effective and consistent camera language transfer. Notably, our losses can also be

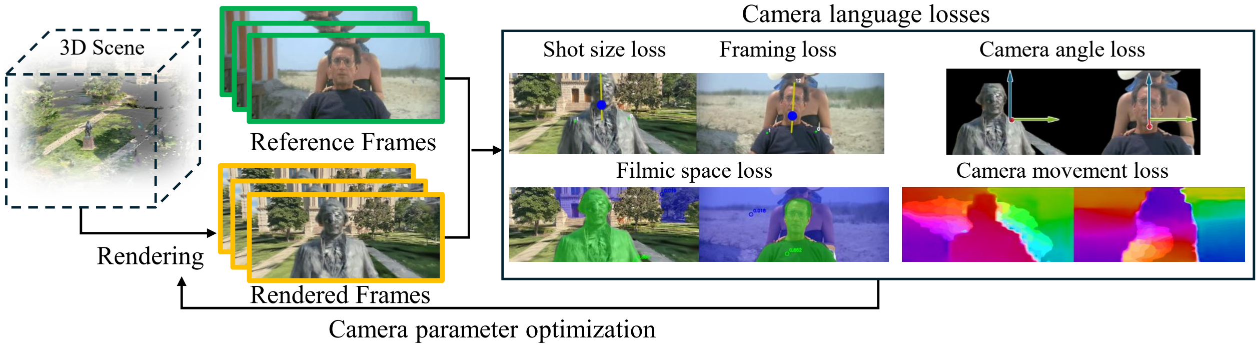

Figure 1. Our camera language transfer framework using five camera-language losses: shot size (yellow chain), framing (blue dot), camera angle, filmic space (green:character; blue:background), and camera movement (depth-layered optical flow).

used as quantitative metrics to evaluate how well the camera language of generated shots matches that of the reference video. Our experiments demonstrate that the proposed method generates videos of effective and consistent camera language, better preserving both the narrative intent and the cinematic visual characteristics of the reference video clips. Our main contributions include:

- We are the first to systematically define the *objectives* of cinematic-guided camera language transfer in 3D scenes, which are explicitly grounded in professional cinematography literature [9, 44].
- To achieve it, we first conduct an in-depth review of professional cinematography literature and identify 8 key cinematic features for camera language, including (i) shot size; (ii) framing; (iii) camera angle; (iv) camera movement; (v) lens choice; (vi) camera position; (vii) zooming and (viii) focus.
- We then carefully analyze the 8 features and encode them with 5 novel camera language losses: (i) shot size loss, (ii) framing loss, (iii) filmic space loss, (iv) camera movement loss, and (v) camera angle loss, and implement them using computer vision techniques. Notably, our losses can also be used as cinematic metrics for the task.
- Experimental results show that our method outperforms state-of-the-art approaches, generating videos with more effective and consistent camera language that better preserve both the narrative intent and the cinematic visual characteristics of reference clips.

2. Related Work

2.1. 3D Scene Representation

Scene representation has long been central in computer vision and graphics, with traditional methods relying on explicit forms such as meshes [40], voxels [39], point clouds [30], and light fields [1]. However, they often required dense sampling, manual reconstruction, or heavy computational resources, limiting their accessibility. The advent of neural scene representations, NeRF [5, 32, 45, 46] offers photorealistic rendering and greatly lowers the barrier for high-quality 3D scene construction, albeit at significant computational cost. More recently, 3D Gaussian Splatting (3DGS) [26] offers NeRF-level realism with far more efficient training and rendering, establishing a critical foundation for creative tasks such as cinematic camera control and 3D style transfer. Motivated by these advantages, we adopt 3DGS as the input 3D scenes in our Cinematic-Guided Camera Language Transfer framework.

2.2. Cinematic Feature

Cinematic features have been studied for decades. Early methods [6, 51, 52, 59] focus on shot size classification and analysis, which is determined by how much of the screen a subject fills. Camera motion is another key feature. For camera motion classification, CAMHID [17] takes motion vector as camera motion descriptors, while Derue et al. [13] leverage optical flow, and MUL-MOVE-Net [10] extends this to optical-flow histograms. For camera movement analysis, CameraBench [35] proposes annotations and a taxonomy of motion primitives. SGNet [50] and LWSRNet [33] use view scale and camera movement for shot classification. MovieNet [22] annotates view scale and camera movement to support broader film-understanding tasks. Lu et al. [41] incorporate composition with shot size and movement for analyzing film shot attributes. CineScale2 [53] extends cinematic analysis to camera angles, proposing a CNN-based framework for automatic angle recognition. Recent benchmarks incorporate filmic attributes with language models. CineTechBench [63], FilMaster [21], and ShotBench [37] annotate multiple cinematic dimensions but focus on evaluating the video generation performance of vision-language models, which is not suitable for cinematic-guided camera language transfer task. We define cinematic features rooted in film grammar for cinematic-guided camera language transfer task: shot size, framing, camera angle, camera movement, lens choice, camera position, zooming and focus, which encompassing the cinematic visual feature from classical film theory.

2.3. Camera Control in Virtual Cinematography

Camera Control has been long studied in computer graphics and virtual cinematography [11]. A naive example-based solution is to reconstruct the camera path from a film clip (typically via SfM) and replay it in a new scene [14, 29]. However, differences in subject distance and scene scale often cause composition drift and scale mismatch, degrading shot size and parallax. A second strategy treat camera control as a sequence prediction problem: Huang et al. [20] incorporate the video contents and previous camera movements to predict the future camera movements, while Jiang et al. [23, 24] train example-driven LSTM controllers using a cinematic feature space (camera angle, distance, composition, character configuration, motion). Although these methods produce smooth trajectories, they did not model full cinematic visual language. A third strategy formulates camera control as a constraint-satisfaction or optimization problem by incorporating predefined metrics. Text-conditioned generation [25, 38] allow users to specify shots via natural language. Others adopt visual metrics to guide the optimization. For example, Galvane et al. [15] formulate camera parameters as a search or optimization problem to maximize view quality metrics. GAIT [65] adopted reinforcement learning to auto-generate camera trajectories in 3D indoor environments by maximizing a learned aesthetic scoring function. Jaws [62] optimize camera parameters using optical flow and pose. However, these optimization methods do not capture the full cinematic visual feature.

3. Preliminaries

Problem Formulation. Cinematic-Guided Camera Language Transfer enables intuitive replication of a reference video clip's camera language onto novel 3D scenes. Following [62], we formulate it as an inverse rendering-style optimization problem. Specifically, given a 3D scene S and a reference video clip $\mathcal{V}_{\text{ref}} = \{\mathcal{F}^i_{\text{ref}}\}_{i=1}^N$ comprising N frames, we aim to synthesize a novel video clip $\mathcal{V}_{\text{S}} = \{\mathcal{F}^i_S\}_{i=1}^N$ by transferring the camera language of $\mathcal{V}_{\text{ref}}$ onto S and rendering it accordingly:

$$\begin{aligned}\mathcal{V}_{\text{S}} &= \text{Render}(C, S) \\ &= \text{Render}(\text{CLTrans}(\mathcal{V}_{\text{ref}}, S), S)\end{aligned} \tag{1}$$

where $\text{Render}(\cdot,\cdot)$ represents the native rendering methods associated with the input 3D scene (*e.g.*, NeRF, 3DGS); and $C = \text{CLTrans}(\mathcal{V}_{\text{ref}}, S) = \{\mathcal{I}^i_S, \mathcal{E}^i_S\}_{i=1}^N$ denotes the frame-wise intrinsic and extrinsic camera parameters, whose optimum $\hat{C}$ is obtained via solving an optimization problem:

$$\hat{C} = \arg\min_{C} \ \mathcal{L}_{\text{CL}}\left(\mathcal{V}_S, \mathcal{V}_{\text{ref}}\right), \tag{2}$$

where $\mathcal{L}_{\text{CL}}$ is a loss function capturing the camera language of the given video clips.

Camera Parameters. Following [62], we define the camera parameters C comprising intrinsics and extrinsics as:

- Intrinsic parameters $\mathcal{I} = \gamma$, where γ is a focal length scaling factor. This simplified version has been widely adopted in prior works [47, 54] due to its favorable optimization properties.
- Extrinsic parameters $\mathcal{E} = (\mathbf{t}, \boldsymbol{\theta})$ defining the camera pose, where $\mathbf{t} = (t_x, t_y, t_z)$, $\boldsymbol{\theta} = (\theta_{\text{roll}}, \theta_{\text{pitch}}, \theta_{\text{yaw}})$ denotes the camera translation and rotation in $SE(3)$, respectively.

4. Method

As with most optimization problems, our solution (Eq. 2) is characterized by three key components: (i) loss function, (ii) initialization strategy, and (iii) optimization procedure.

In this work, we first draw inspiration from professional cinematography literature [9, 44] to identify key cinematic features for camera language transfer, and show that the formulation in [62] is suboptimal in this regard (Sec. 4.1). We then introduce a novel loss design, grounded in these cinematic features, that comprehensively captures the camera language of the given video clips (Sec. 4.2). Finally, building on this loss, we present the corresponding optimization procedure and initialization strategy (Sec. 4.3).

4.1. Key Cinematic Features for Camera Language

As shown in Eqs. 1,2, our goal is to estimate camera parameters that reproduce the look and feel of the reference clip's cinematography, even when the underlying scene content differs. This task is challenging because it demands matching high-level cinematic visual features conveyed through camera language, rather than merely replicating the raw camera trajectories and settings of the reference video [7]. Therefore, the key lies in identifying the key cinematic features that fundamentally shape how visual storytelling is expressed through camera work. Drawing from professional cinematography literature [9, 44], we identify the following key cinematic features for camera language:

- **Shot Size**: Determines the proportion of the subject (typically a character) within the frame, influencing *narrative intimacy* and *visual emphasis*.
- **Framing**: Defines spatial composition of subjects within the image plane, shaping *visual balance* and directing *audience attention*.
- **Camera Angle**: Encodes camera-to-subject orientation, modulating *power dynamics* and *viewer alignment*.
- **Camera Movement**: Reflects temporal camera displacement, producing *perceived motion* and *rhythm*.
- **Lens Choice**: Models *perceived depth* and *spatial compression*, ultimately, the construction of *filmic space* [7].

- ***Camera Position**: Camera-to-subject location, affecting *shot size*, *framing*, *camera angle,* and *movement*.
- ***Zooming**: Dynamically modifies *shot size* without camera translation.
- **Focus**: Closely tied to depth of field, determines which parts of the scene appear sharp or blurred, *guiding attention* and *suggesting emotional or narrative focus*.

where * shows that the feature is closely related to other features. **Please see Supplementary Sec. 8 for details.**

Remark on [62]. Although the pioneering work [62] yields promising results, it approaches the problem from a purely computer vision perspective. Specifically, it implements $\mathcal{L}_{\text{CL}}$ (Eq. 2) as a matching of pose and optical flow between the rendered video and the reference input. However, this approach overlooks alignment with the key cinematic features identified above. For instance, direct pose matching often introduces errors in shot size, as the poses of the main character in the 3D scene and the reference video typically differ. We refer the audience to Sec. 6.3 and the supplementary material for results and analysis.

4.2. Camera Language Losses

In this section, we formalize the eight key cinematic features (Sec. 4.1) into *five* loss functions as follows. Note that as mentioned in Sec. 4.1, (i) Camera position influences shot size, framing, camera angle, and movement. Its effects are thus implicitly captured by these components and not modeled separately. (ii) Zooming is functionally encompassed within our shot size formulation and is not treated as an independent factor as well. (iii) Focus is not modeled due to representation limitations (*i.e.*, depth-of-field) in the 3DGS framework and is left for future work. Please see the supplementary materials for more details.

4.2.1 Shot Size Loss

As defined in [28], shot sizes are typically categorized based on the relative positions of *five* key human joints, including (i) head top, (ii) chest, (iii) waist, (iv) knees, and (v) feet. Please see Supplementary Sec. 11.2 for more details. Accordingly, we propose a novel shot size loss as:

$$L_{\text{shotsize}} = \|d^{\text{ref}} - d^{S}\| \tag{3}$$

where d^{ref} and d^{S} are the normalized maximum horizontal/vertical distances among the visible key joints in the corresponding reference and rendered frames $\mathcal{F}^{i}_{\text{ref}} \in \mathbb{R}^{H_{\text{ref}} \times W_{\text{ref}} \times 3}$ and $\mathcal{F}^{i}_{S} \in \mathbb{R}^{H_S \times W_S \times 3}$, respectively; and the choice between horizontal and vertical distances is determined by whichever is greater in $\mathcal{F}^{i}_{\text{ref}}$. Formally, let $\mathcal{J}^{\text{vis}}_{\text{ref}} \subseteq \mathcal{J} = \{j_{\text{headtop}}, j_{\text{chest}}, \dots, j_{\text{feet}}\}$ denote the visible set of the five key joints in $\mathcal{F}^{i}_{\text{ref}}$, we have:

$$d^{\text{ref}} = \max_{j_i, j_k \in \mathcal{J}^{\text{vis}}_{\text{ref}}} \left(\frac{\|x_{j_i} - x_{j_k}\|}{W_{\text{ref}}}, \frac{\|y_{j_i} - y_{j_k}\|}{H_{\text{ref}}} \right) \tag{4}$$

and

$$d^{S} = \frac{\|x_{j_a} - x_{j_b}\|}{W_S} \text{ or } \frac{\|y_{j_c} - y_{j_d}\|}{H_S} \tag{5}$$

where the choices of (x, j_a, j_b) or (y, j_c, j_d) depend on the results of Eq. 4 for consistency.

Comparison with Previous Works. Previous works estimate shot size either from the normalized area of the subject [22, 50, 51, 61, 66] or from the full-body pose of the main character [62]. However, both approaches are suboptimal: the former is highly sensitive to pose variations and subject shapes that are irrelevant to shot size, while the latter enforces overly strict alignment of the entire pose, including joints (e.g., arms) that have little bearing on shot size. In contrast, our shot size loss adheres closely to the definition in cinematography literature and is robust to pose, viewpoint, and body-shape variations unrelated to shot size, thereby ensuring faithful transfer of camera language.

4.2.2 Framing Loss

Framing refers to the spatial arrangement and composition of significant visual elements in a film frame [28] (please see Supplementary Sec. 11.4 for details). Notably, framing is often co-determined with shot size as determining a subject's spatial placement also involves determining how much space they occupy in a frame. Thus, our framing loss focuses on capturing the spatial placement of a subject, as its size is already captured in Eq. 3. However, given the inevitable differences in content between the reference video and the input 3D scene, perfectly matching all spatial elements through camera adjustment is infeasible. Fortunately, among these elements, human characters are most often the primary narrative focus and serve as the dominant compositional anchors in the frame. Guided by this cinematic principle, we follow previous works [20, 23, 62] and focus on character placement as the key visual anchor for framing alignment. Accordingly, we represent character placement using centroids of visible key joints, which serve as a compact descriptor of the character's overall spatial location in the frame, and have:

$$L_{\text{framing}} = \sqrt{\left(\bar{x}^{\text{ref}} - \bar{x}^{S}\right)^2 + \left(\bar{y}^{\text{ref}} - \bar{y}^{S}\right)^2} \tag{6}$$

where $\bar{x}^{\text{ref}}$ and $\bar{y}^{\text{ref}}$ are the centroid coordinates of the set of visible joints $\mathcal{J}^{\text{vis}}_{\text{ref}}$ in frame $\mathcal{F}^{i}_{\text{ref}}$ (Sec. 4.2.1) that:

$$(\bar{x}^{\text{ref}}, \bar{y}^{\text{ref}}) = \frac{1}{|\mathcal{J}^{\text{vis}}_{\text{ref}}|} \sum_{j_i \in \mathcal{J}^{\text{vis}}_{\text{ref}}} (x_{j_i}, y_{j_i}) \tag{7}$$

And $\bar{x}^{S}$ and $\bar{y}^{S}$ are calculated in a similar way with the same set of joints in the rendered frame $\mathcal{F}^{i}_{S}$.

Comparison with Previous Works. Interestingly, [62] achieves framing implicitly through a full-pose matching loss, which inherits similar shortcomings to those in shot size estimation (*e.g.*, sensitivity to framing-irrelevant joints such as the arms). In contrast, our method explicitly models the cinematographic intent of subject placement while remaining robust to variations in pose, orientation, and articulation, thereby providing a stable framing transfer across heterogeneous scenes.

4.2.3 Filmic Space Loss

Filmic space is the spatial structure perceived within a film frame [28], which can be characterized by the depth, proximity, size, and proportions of objects and places within the image (please see Supplementary Sec. 11.5 for more details). To encode these properties in a manner consistent with human perception and robust to the monocular scale ambiguity inherent in films, we adopt perceptual depth, rather than absolute depth, as the basis to capture the filmic space feature of an input frame.

Following classical mise-en-scène conventions [7], we segment each frame into three coarse depth layers by thresholding the perceptual depth value of each pixel: (i) foreground ($\mathcal{F}$), (ii) character ($\mathcal{C}$), and (iii) background ($\mathcal{B}$). This tripartite scheme reflects both classical film language and cognitively natural: observers coarsely "chunk" depth into near/mid/far zones, supporting stable perception of spatial layout and narrative salience.We then propose our filmic space loss using the *log*-form of relative depth ratios between the three depths layers as:

$$L_{\text{space}} = \| \log d_{\text{bc}}^{\text{ref}} - \log d_{\text{bc}}^{S} \| + \| \log d_{\text{fc}}^{\text{ref}} - \log d_{\text{fc}}^{S} \| \quad (8)$$

where d_{bc} and d_{fc} are the relative depth ratios between ($\mathcal{B}$, $\mathcal{C}$) and ($\mathcal{F}$, $\mathcal{C}$), respectively, that:

$$d_{\text{bc}} = \frac{d_{\mathcal{B}}}{d_{\mathcal{C}}}, \qquad d_{\text{fc}} = \frac{d_{\mathcal{F}}}{d_{\mathcal{C}}} \quad (9)$$

where $d_{\mathcal{B}}$, $d_{\mathcal{C}}$, and $d_{\mathcal{F}}$ are the representative depths of the three depth layers, respectively, that:

$$d_{\mathcal{K}} = \arg\max_{d(p)} \Pr[d(p) \mid p \in \mathcal{K}], \quad \mathcal{K} \in \{\mathcal{F}, \mathcal{C}, \mathcal{B}\}, \quad (10)$$

where $\Pr[\cdot]$ is the probability estimated by applying kernel density estimation (KDE) on depth values $d(p)$ of pixel p at depth layer $\mathcal{K} \in \{\mathcal{F}, \mathcal{C}, \mathcal{B}\}$.

Discussion. Our loss features two novel designs:

- *Representative Depth Value.* Because direct per-pixel depth matching between reference and rendered frames becomes invalid under scene content differences, we instead represent each layer by the mode of its depth distribution as a stable depth estimate. This choice (i) suppresses noise and small occlusions more effectively than means or medians in multi-modal cases, (ii) captures the "prevailing distance" of the layer, and (iii) produces a compact, semantically grounded descriptor aligned with the (foreground, character, background) schema.
- *Relative Depth Ratio.* To obtain a scale-robust perceptual descriptor, we compute two relative depth ratios d_{bc} and d_{fc}. These ratios encode perceived depth separation and offer three advantages: (i) invariance to global monocular depth scaling, (ii) direct correspondence to perceptual separation ("how far the character sits from foreground/background"), and (iii) a clear mapping to cinematic intent ("deep" vs. "flat" staging).

To the best of our knowledge, we are the first to introduce a loss function for modeling filmic space. Consequently, no direct comparison with prior approaches is available.

4.2.4 Camera Movement Loss

Camera movement refers to the changing position or orientation of the camera over time, resulting in perceived relative motion of scene elements within the frame [28] (please see Supplementary Sec. 11.7 for details). Recognizing that camera movement is largely conveyed through scene motion parallax [18, 60], where nearer objects exhibit greater displacement than distant ones, we propose a novel camera movement loss based on a novel depth-layered optical flow decomposition strategy:

$$L_{\text{cam-move}} = \frac{1}{3} \sum_{\mathcal{K} \in \{\mathcal{F}, \mathcal{C}, \mathcal{B}\}} L_{\text{opti-flow}}^{\mathcal{K}}. \quad (11)$$

where $\mathcal{F}$, $\mathcal{C}$, and $\mathcal{B}$ are the three depth layers obtained in Sec. 4.2.3; the optical flow loss $L_{\text{opti-flow}}^{\mathcal{K}}$ of depth layer $\mathcal{K}$ is:

$$L_{\text{opti-flow}}^{\mathcal{K}} = \|O_{\text{ref}}^{\mathcal{K}} - O_{S}^{\mathcal{K}}\|_2, \quad (12)$$

where $O_{\text{ref}}^{\mathcal{K}}$ and $O_{S}^{\mathcal{K}}$ represent the optical flows of the reference and rendered videos at depth layer $\mathcal{K}$, respectively, measured using the endpoint error (EPE) distance [4].

Comparison with Previous Works. Our camera movement loss offers two distinct advantages over the global optical flow matching loss in [62]:

- First, it accounts for motion parallax by decomposing optical flow matching across depth layers. This was neglected in [62], which matches only global optical flow between the reference and rendered videos. As a result, when the depth structures of the reference and rendered frames are not perfectly aligned—as is typical in practice—foreground motion is averaged with background parallax, especially near depth discontinuities.
- Second, it balances the contribution of foreground, character, and background, thereby avoiding biased optical flow matching. Specifically, since endpoint error (EPE) is implicitly weighted by pixel count, the global optical flow matching in [62] is dominated by large background

regions and thus obscures character dynamics, especially when the foreground and the background differ greatly in depth and motion patterns (*e.g.*, dolly zoom or bullet-time effects).

4.2.5 Camera Angle Loss

Camera angle refers to the placement of the camera relative to the subject [28] ((please see Supplementary Sec. 11.10). Thus, we tie it to the relative orientation between the camera and the subject. For each frame, we infer three angles $\mathbf{a} = (\psi, \theta, \phi)$ (yaw, pitch, roll). We minimize the difference of this angle between reference and rendered frames to preserve consistent camera angle. We convert degrees to radians and take the component-wise absolute error:

$$\Delta\mathbf{a} = \left|\mathrm{rad}(\mathbf{a}^{S}) - \mathrm{rad}(\mathbf{a}^{\mathrm{ref}})\right|. \tag{13}$$

Our per-frame camera-angle loss sums the per-axis errors:

$$L_{\text{angle}} = |\Delta\psi| + |\Delta\theta| + |\Delta\phi|. \tag{14}$$

4.2.6 Overall Loss Function

In summary, we have the overall loss function $\mathcal{L}_{\text{CL}}$ as:

$$\begin{gathered}\mathcal{L}_{\text{CL}} = \lambda_1 L_{\text{shotsize}} + \lambda_2 L_{\text{framing}} + \\ \lambda_3 L_{\text{space}} + \lambda_4 L_{\text{cam-move}} + \lambda_5 L_{\text{angle}}\end{gathered} \tag{15}$$

where we set $\lambda_1 \ldots, \lambda_5$ are weighting coefficients, empirically determined to balance the scale of different loss terms.

4.3. Optimization and Initialization

Optimization Procedure. Notice, shot size is jointly determined by both the intrinsic and extrinsic camera parameters. For a given shot size, the desired framing can be achieved either by adjusting the intrinsic parameters (e.g., focal length) or by modifying the extrinsic parameters (e.g., moving the camera closer to or farther from the subject). In our experiments, we observe that the corresponding feature space exhibits lots of local minima. When employing gradient-based optimizers such as the Adam optimizer [27], the optimization process easily falls into local minima. To address this, we adopt the gradient-free Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [16], which is run for up to 100 iterations, with early stopping if the total loss does not decrease over 20 consecutive steps. The search ranges of parameters were set as follows: the translation vector $\mathbf{v}_i \in [-5.0, 5.0]^3$, the rotation axis $\mathbf{w}_i \in [-1.0, 1.0]^3$, and the rotation angle $\theta_i \in [-\frac{\pi}{2}, \frac{\pi}{2}]$. The focal length scaling factor γ_i is optimized within $[0.0, 5.0]$. Notably, this gradient-free approach allows us to incorporate non-differentiable operations—such as the selection of valid key joints based on confidence thresholds—without affecting the overall optimization process. Moreover, the framework transfers easily to other 3D scene representations(NeRFs[5, 32, 45, 46], 3D point-cloud[30], and Unity[58]). By contrast, Jaws[62] relies on a differentiable NeRF and assumes dense multi-view as input. As a result, under the sparse-view reconstructions common in practice, it often fails (see Fig.3).

Initialization Strategy. Following JAWS [62], we adopt the same initialization strategy to make a fair comparison. The initial view is selected by the user from the input images used to train the 3D scene representation.

5. Implementation Details

We represent the input 3D scene S using 3DGS [26] due to its high quality and efficiency. To extract their cinematic features, we leverage several state-of-the-art models. See Supplementary Sec. 9 for more details.

6. Experiments

6.1. Experimental Setup

Datasets. Our dataset comprises 3D scenes S and reference videos $\mathcal{V}_{\text{ref}}$:

- Our dataset includes both outdoor (selected from DL3DV [36], ENeRF-Outdoor [34]) and indoor scenes (DyNeRF [32], Mobile-Stage [48, 67]). All the selected scenes have at least one human or character-like subject.
- Our reference videos are selected from the CameraBench [35] and CondensedMovies [3] datasets, each a single-shot clip with one character. To cover diverse and representative cinematic motion styles, we include canonical shot types defined in classical film theory [7, 43], including both basic and classic complex shots.
  - The basic shots include: (i) Push in (camera moves forward), (ii) Pull out (camera moves backward), (iii) Pan (camera moves horizontally), (iv) Tilt (camera moves vertically), (v) Orbit (camera circles around a subject), (vi) Zoom (lens-based magnification), and (vii) Crash Zoom (a rapid zoom in or out).
  - The classic complex shots include: (i) Dolly Zoom (simultaneous zoom and dolly movement that alters background perspective while maintaining subject scale), and (ii) Dutch Angle (camera is tilted to create a sense of unease or disorientation).

Metrics. Based on our Camera-Language loss, we identify cinematic visual metrics for quantitative experiment from a professional cinematography perspective. Specifically, our Shot Size Loss can be used to evaluate the main character's screen occupancy relative to the reference. Framing loss evaluates the on-screen position of the main character. Filmic space loss evaluates the perceived depth structure and further enforces focal-length consistency. Cam-

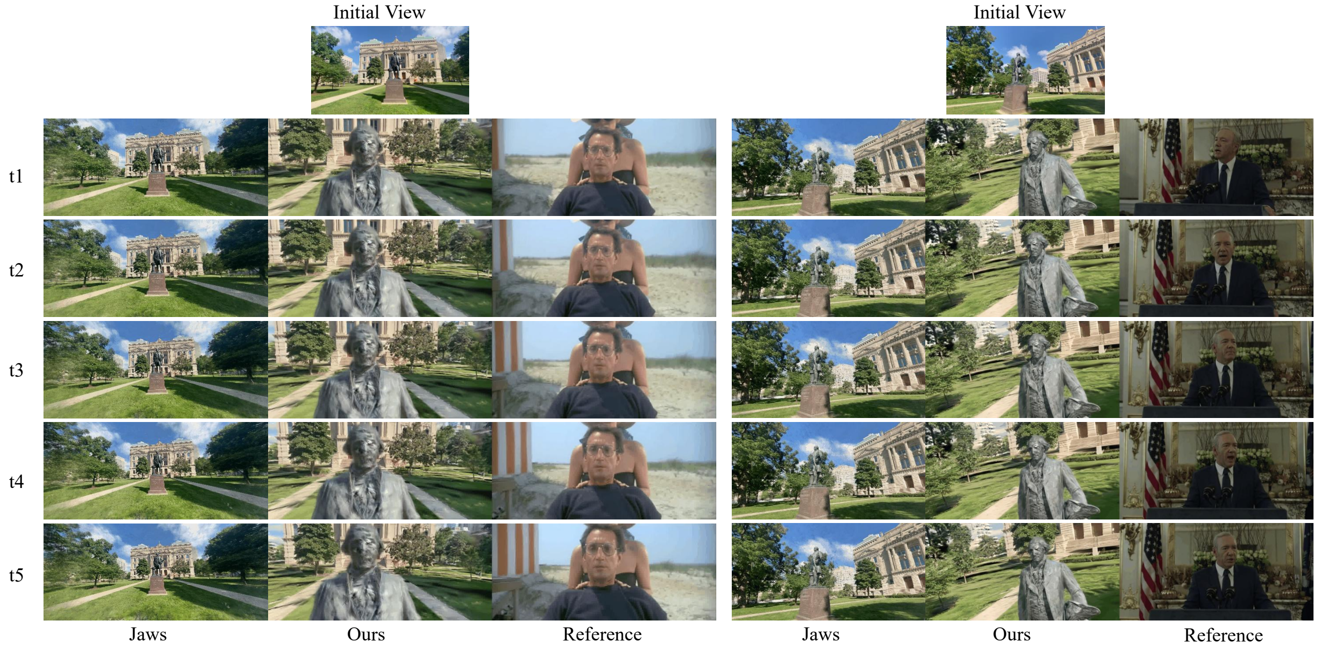

Figure 2. Qualitative results of the dolly zoom (left) and rotating (right) example.Our results show that our cinematic visual feature consistent with the reference frame.

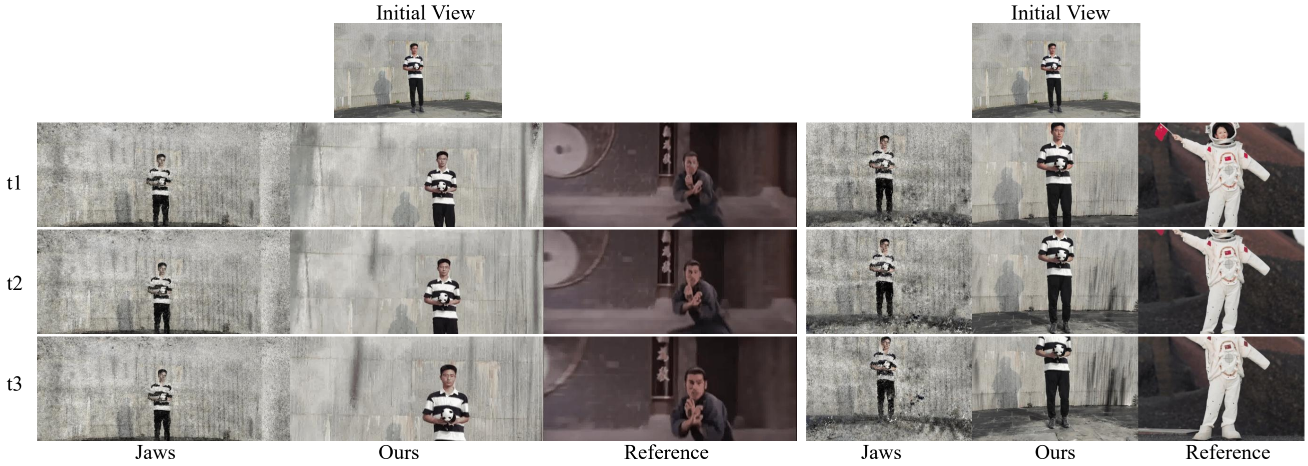

Figure 3. Qualitative results of the crash zoom (left) and Boom (right) example. Our results show that our cinematic visual feature (specifically, shot size, framing, camera movement) consistent with the reference frame.

era movement loss evaluates temporal differences in camera motion. Camera angle loss evaluates the relative camera–subject orientation.

6.2. Quantitative results

We use our cinematic feature metric to evaluate the results of Jaws[62] and our model by computing the mean difference of each frame at each rendered video clip. Our method achieves consistently lower errors as shown in Table.1.(Note: Frames with severe rendering failures in JAWS are excluded from the evaluation.)

6.3. Qualitative results

We test our method on both basic (Fig.3) and classic complex shots dolly zoom(Fig.2 left). Baseline method generally fail when background and character move differently. Specifically, baseline method failed to handle the shot size. In contrast, our method accurately clones visual feature from the reference video: the tree in the background become smaller, while the character become bigger.

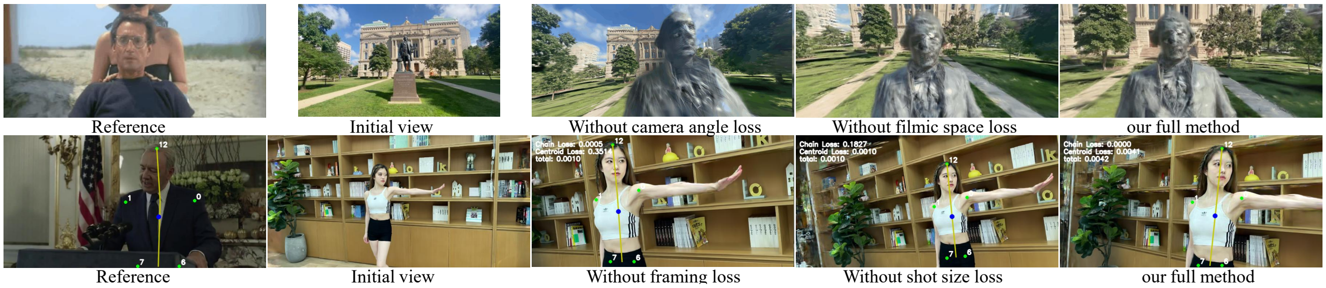

Figure 4. Ablation study for camera angle, filmic space, framing, and shot size losses.

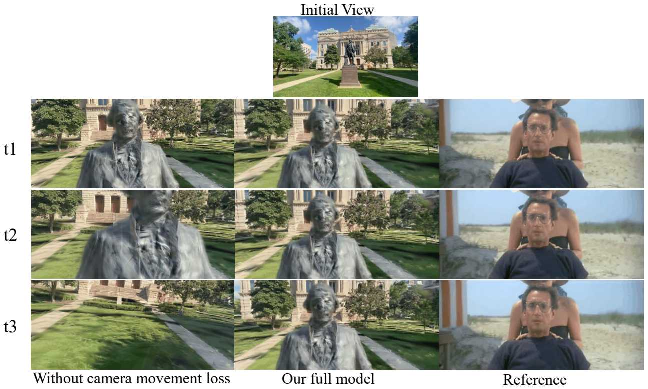

Figure 5. Ablation study for camera movement loss.

Table 1. Quantitative comparison of cinematic feature metrics. Lower values indicate better alignment with the reference. Our method outperforms JAWS on all reported metrics.

Metric	JAWS [62]	Ours
Shot Size	0.093017	0.080433
Framing	0.447699	0.380040
Filmic Space	2.905601	0.475843
Camera Movement	30.288347	1.952139
Camera Angle	2.905543	2.603337

6.4. User Study

We conducted a user study with 11 participants, recruited from the computer science department as unpaid volunteers with no formal training in cinematography. The study evaluated (1) the fidelity of visual style to the reference video, (2) smoothness, and (3) naturalness of the generated video covering both basic shots and classic complex shots. For each sample, participants viewed two anonymized videos, one from the baseline Jaws[62] and one from our method, presented in random order alongside the reference. They were asked to select the preferred video for each criterion.

As shown in Table 2, across all three criteria, our method is consistently preferred over the baseline Jaws. In particular, for visual style fidelity, our method is preferred in all trials, indicating a strong alignment with the reference videos. Substantial improvements are also observed in smoothness (84.85%) and naturalness(81.82%).

Table 2. User preference evaluation between our method and Jaws[62]. Each percentage represents the ratio of pairwise comparisons in which ours was preferred by participants over Jaws.

	Visual Style	Smoothness	Naturalness
Ours (%)	100.00	84.85	81.82
Jaws[62] (%)	0.00	15.15	18.18

6.5. Ablation study

Fig. 4 and Fig. 5 present our ablation results. Removing the framing or shot size loss leads to inaccurate subject placement or scale, while removing the filmic space loss produces overly deep perspective inconsistent with the reference. Excluding the camera movement loss causes temporal instability, and removing the camera angle loss yields misaligned subject orientation. Our full model contributes to best preserving the intended cinematic expression. Please see Sec. 10 in the supplementary materials for more details.

7. Conclusion

We address the task of Cinematic-Guided Camera Language Transfer, aiming to reproduce the cinematic camera language of a reference video in a new 3D scene. While prior work approached this challenge with generic computer vision techniques, it overlooked core cinematographic principles, resulting in inaccurate framing, motion mismatches, and loss of expressive intent. To address this gap, we systematically grounded the task in professional cinematography literature, identifying eight fundamental cinematic features and encoding them into five novel camera language losses, which not only enable more effective and consistent transfer of camera language, but also provide quantitative metrics for evaluating cinematographic fidelity. Extensive experiments show that our method substantially outperforms existing approaches, better preserving both narrative intent and cinematic visual style of reference clips.

References

[1] *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, New York, NY, USA, 1996. Association for Computing Machinery. 2

[2] Dui Ardal, Simon Alexandersson, Mirko Lempert, and André Tiago Abelho Pereira. A collaborative previsualization tool for filmmaking in virtual reality. In *Proceedings of the 16th ACM SIGGRAPH European Conference on Visual Media Production*, pages 1–10, 2019. 1

[3] Max Bain, Arsha Nagrani, Andrew Brown, and Andrew Zisserman. Condensed movies: Story based retrieval with contextual embeddings. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 6

[4] Simon Baker, Daniel Scharstein, James P Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *International journal of computer vision*, 92(1):1–31, 2011. 5

[5] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022. 2, 6

[6] Sergio Benini, Michele Svanera, Nicola Adami, Riccardo Leonardi, and András Bálint Kovács. Shot scale distribution in art films. *Multimedia Tools and Applications*, 75(23): 16499–16527, 2016. 2

[7] David Bordwell and Kristin Thompson. *Film art: An introduction*. 3, 5, 6, 1, 2, 8

[8] G Briand, F Bidgolirad, JF Szlapka, JM Lavalou, M Lanouiller, M Christie, J Lvoff, P Bertolino, and E Guillou. On-set previsualization for vfx film production. In *IBC2014 Conference*, pages 12–1. IET, 2014. 1

[9] Daniel Chandler and Rod Munday. *A dictionary of media and communication*. Oxford University Press, USA, 2011. 1, 2, 3

[10] Zeyu Chen, Yana Zhang, Lianyi Zhang, and Cheng Yang. Ro-textcnn based mul-move-net for camera motion classification. In *2021 IEEE/ACIS 20th International Fall Conference on Computer and Information Science (ICIS Fall)*, pages 182–186. IEEE, 2021. 2

[11] Marc Christie, Patrick Olivier, and Jean-Marie Normand. Camera control in computer graphics. In *Computer graphics forum*, pages 2197–2218. Wiley Online Library, 2008. 3

[12] Yuanjie Dang, Chong Huang, Peng Chen, Ronghua Liang, Xin Yang, and Kwang-Ting Cheng. Imitation learning-based algorithm for drone cinematography system. *IEEE Transactions on Cognitive and Developmental Systems*, 14(2):403–413, 2020. 1

[13] François-Xavier Derue, Mohamed Dahmane, Marc Lalonde, and Samuel Foucher. Exploiting semantic segmentation for robust camera motion classification. In *International Conference on Image Analysis and Recognition*, pages 173–181. Springer, 2017. 2

[14] Steven M Drucker, Tinsley A Galyean, and David Zeltzer. Cinema: A system for procedural camera movements. In *Proceedings of the 1992 symposium on Interactive 3D graphics*, pages 67–70, 1992. 3

[15] Quentin Galvane, Marc Christie, Chrsitophe Lino, and Rémi Ronfard. Camera-on-rails: automated computation of constrained camera paths. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, pages 151–157, 2015. 3

[16] Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary computation*, 9(2):159–195, 2001. 6

[17] Muhammad Abul Hasan, Min Xu, Xiangjian He, and Changsheng Xu. Camhid: Camera motion histogram descriptor and its application to cinematographic shot classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(10):1682–1695, 2014. 2

[18] Hermann Ludwig Ferdinand von Helmholtz, James Powell Cocke Southall, et al. Treatise on physiological optics. *(No Title)*, 1925. 5, 3

[19] Alfred Hitchcock. Vertigo [film]. Motion picture, 1958. 1

[20] Chong Huang, Chuan-En Lin, Zhenyu Yang, Yan Kong, Peng Chen, Xin Yang, and Kwang-Ting Cheng. Learning to film from professional human motion videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4244–4253, 2019. 3, 4, 2

[21] Kaiyi Huang, Yukun Huang, Xintao Wang, Zinan Lin, Xuefei Ning, Pengfei Wan, Di Zhang, Yu Wang, and Xihui Liu. Filmaster: Bridging cinematic principles and generative ai for automated film generation. *arXiv preprint arXiv:2506.18899*, 2025. 2

[22] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *European conference on computer vision*, pages 709–727. Springer, 2020. 2, 4, 1

[23] Hongda Jiang, Bin Wang, Xi Wang, Marc Christie, and Baoquan Chen. Example-driven virtual cinematography by learning camera behaviors. *ACM Trans. Graph.*, 39(4):45, 2020. 3, 4, 2

[24] Hongda Jiang, Marc Christie, Xi Wang, Libin Liu, Bin Wang, and Baoquan Chen. Camera keyframing with style and control. *ACM Transactions on Graphics (TOG)*, 40(6): 1–13, 2021. 3

[25] Hongda Jiang, Xi Wang, Marc Christie, Libin Liu, and Baoquan Chen. Cinematographic camera diffusion model. In *Computer Graphics Forum*, page e15055. Wiley Online Library, 2024. 3

[26] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 2, 6, 3

[27] DP Kingma. Adam: a method for stochastic optimization. In *Int Conf Learn Represent*, 2014. 6

[28] Annette Kuhn and Guy Westwell. *A dictionary of film studies*. Oxford University Press, USA, 2012. 4, 5, 6, 1, 2

[29] Christian Kurz, Tobias Ritschel, Elmar Eisemann, Thorsten Thormählen, and Hans-Peter Seidel. Camera motion style transfer. In *2010 Conference on Visual Media Production*, pages 9–16. IEEE, 2010. 3

[30] Marc Levoy and Turner Whitted. The use of points as a display primitive. 1985. 2, 6

[31] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10863–10872, 2019. 3

[32] Tianye Li, Mira Slavcheva, Michael Zollhoefer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard Newcombe, et al. Neural 3d video synthesis from multi-view video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5521–5531, 2022. 2, 6

[33] Yuzhi Li, Tianfeng Lu, and Feng Tian. A lightweight weak semantic framework for cinematographic shot classification. *Scientific Reports*, 13(1):16089, 2023. 2

[34] Haotong Lin, Sida Peng, Zhen Xu, Yunzhi Yan, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Efficient neural radiance fields for interactive free-viewpoint video. In *SIGGRAPH Asia Conference Proceedings*, 2022. 6

[35] Zhiqiu Lin, Siyuan Cen, Daniel Jiang, Jay Karhade, Hewei Wang, Chancharik Mitra, Tiffany Ling, Yuhan Huang, Sifan Liu, Mingyu Chen, et al. Towards understanding camera motions in any video. *arXiv preprint arXiv:2504.15376*, 2025. 2, 6

[36] Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22160–22169, 2024. 6

[37] Hongbo Liu, Jingwen He, Yi Jin, Dian Zheng, Yuhao Dong, Fan Zhang, Ziqi Huang, Yinan He, Yangguang Li, Weichao Chen, et al. Shotbench: Expert-level cinematic understanding in vision-language models. *arXiv preprint arXiv:2506.21356*, 2025. 2

[38] Xinhang Liu, Yu-Wing Tai, and Chi-Keung Tang. Chatcam: Empowering camera control through conversational ai. *arXiv preprint arXiv:2409.17331*, 2024. 3

[39] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: learning dynamic renderable volumes from images. *ACM Transactions on Graphics (TOG)*, 38(4):1–14, 2019. 2

[40] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169, 1987. 2

[41] Fengtian Lu, Yuzhi Li, and Feng Tian. Exploring challenge and explainable shot type classification using sam-guided approaches. *Signal, Image and Video Processing*, 18(3):2533–2542, 2024. 2

[42] Mark Roberts Motion Control. Bolt high-speed cinebot: Motion control solutions, 2025. Describes the Bolt high-speed camera robot; it can accelerate from standstill to high speed and stop within fractions of a second, reach speeds up to 12 m/s on track, carry a 20 kg payload, and features a 6-axis arm with 2 m reach and 3.5 m height:contentReference[oaicite:8]index=8. 1

[43] Joseph V Mascelli. *The five C's of cinematography*. Grafic Publications Hollywood, 1965. 6

[44] David Mia. Cinematographic language: The role of the camera in constructing mood and atmosphere. 2023. 1, 2, 3

[45] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 6

[46] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022. 2, 6

[47] Jong-Ik Park, Carlee Joe-Wong, and Gary K Fedder. Mzen: Multi-zoom enhanced nerf for 3-d reconstruction with unknown camera poses. *arXiv preprint arXiv:2508.05819*, 2025. 3

[48] Sida Peng, Zhen Xu, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Hujun Bao, and Xiaowei Zhou. Animatable implicit neural representations for creating realistic avatars from videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024. 6

[49] Pragathi Praveena, Bengisu Cagiltay, Michael Gleicher, and Bilge Mutlu. Exploring the use of collaborative robots in cinematography. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–6, 2023. 1

[50] Anyi Rao, Jiaze Wang, Linning Xu, Xuekun Jiang, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A unified framework for shot type classification based on subject centric lens. In *European Conference on Computer Vision*, pages 17–34. Springer, 2020. 2, 4, 1

[51] Mattia Savardi, Alberto Signoroni, Pierangelo Migliorati, and Sergio Benini. Shot scale analysis in movies by convolutional neural networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2620–2624. IEEE, 2018. 2, 4, 1

[52] Mattia Savardi, András Bálint Kovács, Alberto Signoroni, and Sergio Benini. Cinescale: A dataset of cinematic shot scale in movies. *Data in Brief*, 36:107002, 2021. 2

[53] Mattia Savardi, András Bálint Kovács, Alberto Signoroni, and Sergio Benini. Recognition of camera angle and camera level in movies from single frames. In *Proceedings of the 2023 ACM International Conference on Interactive Media Experiences Workshops*, pages 79–85, 2023. 2

[54] Hannah Schieber, Fabian Deuser, Bernhard Egger, Norbert Oswald, and Daniel Roth. Nerftrinsic four: An end-to-end trainable nerf jointly optimizing diverse intrinsic and extrinsic camera parameters. *Computer Vision and Image Understanding*, 249:104206, 2024. 3

[55] Steven Spielberg. Jaws [film]. Motion picture, 1975. 1

[56] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 3

[57] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020. 3

[58] Unity Technologies. Unity game engine, version 6.2. https://unity.com/releases/unity-6, 2025. Accessed: 2025-08-17. 1, 6

[59] Bartolomeo Vacchetti and Tania Cerquitelli. Cinematographic shot classification with deep ensemble learning. *Electronics*, 11(10):1570, 2022. 2

[60] Nicholas J Wade. Helmholtz at 200. *i-Perception*, 12(4): 20416695211022374, 2021. 5, 3

[61] Hee Lin Wang and Loong-Fah Cheong. Taxonomy of directing semantics for film shot classification. *IEEE transactions on circuits and systems for video technology*, 19(10):1529–1542, 2009. 4, 1

[62] Xi Wang, Robin Courant, Jinglei Shi, Eric Marchand, and Marc Christie. Jaws: Just a wild shot for cinematic transfer in neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16933–16942, 2023. 1, 3, 4, 5, 6, 7, 8, 2

[63] Xinran Wang, Songyu Xu, Xiangxuan Shan, Yuxuan Zhang, Muxi Diao, Xueyan Duan, Yanhua Huang, Kongming Liang, and Zhanyu Ma. Cinetechbench: A benchmark for cinematographic technique understanding and generation. *arXiv preprint arXiv:2505.15145*, 2025. 2

[64] Zehan Wang, Ziang Zhang, Tianyu Pang, Chao Du, Hengshuang Zhao, and Zhou Zhao. Orient anything: Learning robust object orientation estimation from rendering 3d models. *arXiv preprint arXiv:2412.18605*, 2024. 3

[65] Desai Xie, Ping Hu, Xin Sun, Soren Pirk, Jianming Zhang, Radomír Mech, and Arie E Kaufman. Gait: Generating aesthetic indoor tours with deep reinforcement learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7409–7419, 2023. 3

[66] Min Xu, Jinqiao Wang, Muhammad A Hasan, Xiangjian He, Changsheng Xu, Hanqing Lu, and Jesse S Jin. Using context saliency for movie shot classification. In *2011 18th IEEE International Conference on Image Processing*, pages 3653–3656. IEEE, 2011. 4, 1

[67] Zhen Xu, Sida Peng, Haotong Lin, Guangzhao He, Jiaming Sun, Yujun Shen, Hujun Bao, and Xiaowei Zhou. 4k4d: Real-time 4d view synthesis at 4k resolution. 2023. 6

[68] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 3

[69] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in Neural Information Processing Systems*, 37:21875–21911, 2024. 3