Automated Adaptive Cinematography for User Interaction in Open World

Zixiao Yu, Xinyi Wu, Haohong Wang, Aggelos K. Katsaggelos, *Life Fellow, IEEE*, and Jian Ren, *Senior Member, IEEE*

***Abstract*—Advancements in wearable technology and their capacity to interpret user movements, transforming them into interactive actions in virtual environments, have sparked an increased demand for user flexibility within these spaces. A direct outcome of this growing trend is the imperative need for automated cinematography in expansive, open-world scenarios. Nevertheless, the task of interpreting these interactive sequences through automated cinematography in unconstrained environments involves significant computational challenges. In response to this, we introduce the Automated Adaptive Cinematography for Open-world Generative Adversarial Network (AACOGAN) -an innovative solution that addresses these issues. Contrary to traditional models, which require comprehensive prior knowledge about scenes, characters, and objects, AACOGAN identifies and models the relationships among user interactions, object positions, and camera movements during the process of user engagement. This novel approach allows the model to function effectively even in open-world scenarios riddled with numerous uncertain factors. In the experimental phase, we developed and employed the *MineStory Dataset*, designed specifically for automatic cinematography in open-world scenarios. We devised and implemented novel metrics that are more congruent with the distinctive features of open-world scenarios. These innovative metrics provide a more nuanced understanding of the performance and effectiveness of our proposed method. Experimental findings substantiate that AACOGAN significantly enhances automatic cinematography performance within open-world contexts, including an average augmentation of 73% in the correlation between user interactions and camera trajectories, and an increase of up to 32.9% in the quality of multi-focus scenes. Therefore, AACOGAN emerges as an efficient, and innovative solution for creating appropriate camera shots in a myriad of interactive motions in open-world scenarios.**

***Index Terms*—Automatic cinematography, deep-learning, efficient, GAN, multi-media application.**

Manuscript received 30 May 2023; revised 8 August 2023 and 25 October 2023; accepted 15 December 2023. Date of publication 26 December 2023; date of current version 24 July 2024. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Yong Luo. *(Corresponding author: Jian Ren.)*

Zixiao Yu and Jian Ren are with the Department of Electronics and Communication Engineering, Michigan State University, East Lansing, MI 48824 USA (e-mail: yuzixiao@msu.edu; renjian@msu.edu).

Xinyi Wu and Aggelos K. Katsaggelos are with the Department of Electronics and Communication Engineering, Northwestern University, Evanston, IL 60208-3109 USA (e-mail: xinyiwu2019.1@u.northwestern.edu; a-katsaggelos@u.northwestern.edu).

Haohong Wang is with TCL Research American, San Jose, CA 95110 USA (e-mail: haohong.wang@tcl.com).

An exemplary video footage can be found at https://youtu.be/pbSHF-uxomw.

Digital Object Identifier 10.1109/TMM.2023.3347092

I. Introduction

THE gaming industry's progression has been persistently propelled by the aspiration to provide users with increased autonomy in gameplay, which serves to elevate their pleasure and absorption within the game experience [1]. This has resulted in a trend towards the creation of open-world games, which provide users with a more expansive and unrestricted environment and, as a result, a more enriching experience. Cinematography is a crucial element in conveying character emotions and advancing the plot in media such as games and films, and it can significantly enhance the immersion of the audience if utilized effectively. However, camera placement always requires expert knowledge and also consumes a significant amount of time and budget in the production process. Automatic cinematography technology has been developed to solve this problem.

Historically, film directors could lean on scripts to guide camera strategy and lens selection. Such reliance streamlined the cinematographic process, ensuring the audience received an optimized visual experience, while simultaneously cutting down on production costs [2], [3]. Over time, these tactics crystallized into classical lens language techniques, distilled into a codified set of camera rules [4], [5].

However, these conventional, rule-based methodologies falter in open-world scenarios, marked by their inherent unpredictability (as shown in Table I). Players in these scenarios cherish their ability to sculpt and narrate their distinct stories through interactive gameplay, not just follow a preset narrative. Such unparalleled freedom fosters unique virtual experiences. Player-controlled avatars are emancipated from rigid scripts, enabling them to interact dynamically with the virtual world, breaking the shackles of fixed choices. This fluidity introduces unpredictability in elements like narrative flow and character interplay. Although some applications attempt to employ rule-based automatic cinematography, they often fall short, compromising the immersive quality of the game [6], [7]

Emerging deep learning technologies bring promise to the domain of automatic cinematography [4], [5]. These methodologies empower directors to derive lens language from expansive datasets, widening the scope of its applications. Consequently, deep learning-centric automatic cinematography becomes a formidable tool for open-world settings, significantly amplifying immersion.

In our research, a predominant challenge was devising an algorithmic approach for optimal camera placement to maximize

TABLE I
COMPARISON OF THREE DIFFERENT TYPES OF USER EXPERIENCES (TRADITIONAL MOVIE, PURPOSE-DRIVEN INTERACTIVE MOVIE OR GAME, AND OPEN WORLD) IN VARIOUS ASPECTS AS PERCEIVED BY THE AUDIENCE

	Traditional Movie	Purpose-driven Interactive Movie/Game	Open World
Whether the protagonist is controlled	NO	Yes	Yes
Script form	Single-line script	Multiple branching script	Script is present but users do not have to follow it
Interactive distance	Fixed	Fixed	Uncertain
Interactive choice **Interactive objects**	None Determine based on plot	Limited multiple choice Objects within selectable range	Any way captured by devices Any objects in the scene
Lens usage	Preset	From preset idioms	Cannot be preset
Script defines character emotions	Yes	Yes	No
Camera movement usage	Many/director determines	Many/director determines	In most cases, users only have first or third person fixed perspective
User experience mode	Watch	Interact driven by script/watch	Users can freely decide to watch/interact

shot quality. Our solution extends the toric surface model [8], infusing it with considerations of character emotions, actions, and environmental contexts, leading to nuanced camera choices in diverse open-world contexts.

Consistency, spanning character movements, emotions [9], and camera dynamics remain paramount for immersive gameplay [10], [11], [12]. Recent investigations underscore the efficacy of automatic camera movements tailored around individual characters, encapsulating both their physical and emotional states [13]. Such techniques surpass prior methods that solely track head movements, disregarding emotional undertones [14]. Our methodology builds upon these foundations, tailoring them for open-world applications.

Furthermore, to elevate the aesthetic resonance of our frames, we incorporated insights from professional film director shot selections into our model's loss function, optimizing our automated camera generation algorithm [15].

To the best of our knowledge, this paper is the first effort in the automated cinematography field that generates camera movement for multi-character interactions in an open-world environment.

The major contributions of this paper can be summarized as follows:

- We develop comprehensive quality assessment metrics tailored for automatic cinematography for open-world scenarios.
- We propose a novel auto cinematography framework called AACOGAN based on Generative Adversarial Networks (GANs). AACOGAN is designed to generate camera movements that are consistent with the interactive actions of the characters in the open-world environment.
- We design a unique input criterion that includes all the factors that need to be considered for camera selection in open-world scenarios.

The rest of the paper is organized as follows: Section II provides a literature review on computational cinematography, video editing, and video understanding. Section III presents our proposed framework, problem formulation, and dynamic programming solution. Section IV discusses experimental results and the potential impact of video understanding errors on the framework. Section V concludes the work.

II. RELATED WORK

Cinematography plays an essential role in creating immersion and realism for audiences. Through dynamic camera movements and meticulously designed shots, it directs attention, establishes the game world's ambiance, and communicates information and emotions [12], [16].

Historically, the film industry relied on director-led strategies, where camera scripts were derived from the story's plot [14], [17]. This method, while effective, requires extensive cinematography expertise and considerable manual intervention. Contemporary research has shifted towards identifying cinematic idioms—recurring camera movements tailored for specific scene types [18], [19]. These idioms, and their sophisticated counterparts, have been designed for films or games with explicit narratives, streamlining the filmmaking process [5], [20], [21]. The relationship between lens language and cinematography, especially its potential to guide user emotions, has been explored extensively [6], [7]. However, for open-world scenarios, where cinematography depends on character dynamics and environment interactions, this idiom-based approach becomes inappropriate.

An alternative automated cinematography approach, as described in [4], [22], may be more fitting for open-world scenarios. By integrating various lens language elements into training data, this method adapts to character actions and environmental interactions. Notably, GAN-based cinematography focuses on individual character shots, considering actions and emotions, but often overlooks inter-character and object dynamics [13], which are vital in open-world contexts.

Optimizing camera position selection is critical in expansive environments, given the computational challenges posed by numerous options. Limiting camera movements to widely accepted spaces ensures efficient computations without compromising quality [8], [23]. However, a persistent challenge in automated cinematography is the absence of objective evaluation criteria. Incorporating aesthetic assessment models, rooted in widely recognized aesthetic benchmarks [24], [25], [26], into AI-driven cinematography could enhance the overall frame quality.

III. PROBLEM FORMULATION

Evaluating the quality of lenses generated by automated photography technology presents a complex challenge, given the absence of universally acknowledged objective evaluation standards [27]. Traditional user research, employed for outcome assessment, may be hampered by biases and limitations in addressing dynamic user demands, potentially impeding system enhancements. Therefore, the development of objective evaluation criteria is vital for improving research results.

A. Quality Measurement

In open-world environments, user-directed character movements are often primarily focused on the character itself rather than the entire scene. To effectively address these unique requirements for automatic cinematography in open-world contexts, we propose an equation with rational metrics to evaluate the quality of the automatically generated camera q, as detailed below:

$$q = Q(D_{\mathsf{corr}}(C, A),\ R(C),\ S_{\mathsf{aes}}(C)), \tag{1}$$

where $Q(\cdot)$ is the quality function, C is the generated camera trajectory, and A is the position and rotation of the skeletal bone during character interactive movement. $D_{\mathsf{corr}}(\cdot)$ is the function that calculates the similarity between the camera and character movement trajectories. $R(\cdot)$ is the function that calculates the ratio of the all-character captured frame during the interaction, and $S_{\mathsf{aes}}(\cdot)$ calculates the aesthetic score of the frames captured by the given C.

Undoubtedly, the employment of cinematic language as an essential component of the artistic domain is contingent upon subjective assessments. Consequently, a preferable approach might encompass enabling automatic cinematography systems to learn from human film directors in the gaming industry [4], rather than relying exclusively on fixed algorithms for camera movement generation. By leveraging pertinent data to mimic the cinematic language habits of directors and iteratively refining the generated algorithm based on user feedback, neural network technology could effectively replicate directorial expertise. During the system training phase of our experiment, the preceding quality measurement defined in (1) can be expanded to Q_{ref} as follows:

$$q_{\mathsf{ref}} = Q_{\mathsf{ref}}(D_{\mathsf{corr}}(C, A), R(C), S_{\mathsf{aes}}(C), \mathsf{Dis}(C, \hat{C})), \tag{2}$$

where $\mathsf{Dis}(\cdot)$ represents the distance metric that measures the dissimilarity between the C and the ground truth camera motion $\hat{C}$ obtained from human film directors, who authored the actual in-game camera movements.

It should be noted that the approach we use to define Q_{ref} is not necessarily a standard one. Moving forward, we will provide a detailed explanation of each component that makes up this Q_{ref}.

In gaming environments, players exert control over both in-game settings and character functionalities, which can influence camera trajectories based on predefined rules [11]. This dynamic is similarly observed in open-world contexts, where player actions predominantly alter the environment and game status. Maintaining consistency in character control can reduce user disorientation, as highlighted by [28]. An important metric to evaluate coherence is the congruence between character skeletal nodes' trajectories and camera motion during interactions. The disparity between these trajectories can be quantified using the correlation distance d_{corr} for each position and rotation axis.

$$\begin{aligned} d_{\mathsf{corr}} &= D_{\mathsf{corr}}(C, A) \\ &= \sqrt{1 - \frac{\left(\sum_{t=1}^{n-1}(f_t - \bar{f})(p_t - \bar{p})\right)^2}{\sum_{t=1}^{n-1}(f_t - \bar{f})^2 \sum_{t=1}^{n-1}(p_t - \bar{p})^2}}, \end{aligned} \tag{3}$$

f_t and p_t are the frame and position coordinates of the t-th point on the trajectory C, n is the number of frames for the action and camera movement, and $\bar{f}$ and $\bar{p}$ are the mean values of the f and p coordinates, respectively.

The absence of a predefined script in open-world environments poses a significant challenge to automated filming techniques, especially when it comes to focusing on multiple points [8]. In scenarios involving multiple parties, it is crucial for the camera to capture the entire interaction process and all characters comprehensively, not solely the character currently under user control. To address this challenge, a potential strategy is to facilitate the camera's extended capture of all involved characters throughout the camera movement process. An effective metric for evaluating the efficacy of capturing all involved characters r throughout the camera movement process is the ratio of the number of frames in which all interactive characters appear in the frame to the total number of frames n used for the interaction, which can be calculated by:

$$\begin{aligned} r &= R(C) \\ &= \frac{\sum_{t=0}^{t=n} R_{\mathsf{frame}}(f_t)}{n} \times 100\%, \end{aligned} \tag{4}$$

where $R_{\mathsf{frame}}(\cdot)$ result equals 1 when all the interactive characters are present within ft, otherwise it equals 0.

Although objective criteria for evaluating the quality of imagery generated by automated cinematography techniques remain elusive, aesthetic evaluation models have been widely acknowledged for images as presented in [26], [29]. As a sequence of images, the video captured during camera movement can be evaluated objectively in terms of aesthetics by calculating the aesthetic score of each frame captured during the camera movement process. Integrating aesthetic models into the automated cinematography system can improve the conformity of captured imagery with objective standards. This aesthetic score can be calculated by:

$$\begin{aligned} s_{\mathsf{aes}} &= S_{\mathsf{aes}}(C) \\ &= \frac{\sum_{t=1}^{t=n} \mathsf{AES}(f_t)}{n}, \end{aligned} \tag{5}$$

where f_t is the visual content captured by the C at t-th frame, AES is the model for image aesthetic evaluation, and n is the number of frames for the camera movement.

Emotional states of characters play a pivotal role in automated camera movement decisions, influencing shot selection based on their emotional intensities [12], [13]. Notably, screen motion intensity has been found to correlate with viewer arousal [9]. Thus, even with consistent interactive actions, diverse emotional contexts necessitate variations in camera movement dynamics to aptly represent the character's emotional nuances. However, assessing the exact relationship between emotions and camera movement can be challenging due to individual differences in perceiving camera movement responses to emotions. This difference will be directly reflected in the actual camera trajectory and can be considered a component of $\mathsf{Dis}(C, \hat{C})$. The $\mathsf{Dis}(\cdot)$ is employed to compute the distance between the generated and

real camera drive data, consisting of two parts: MSE and Euclidean distance, represented as follows:

$$\begin{aligned} \text{dist} &= \text{Dis}(C, \hat{C}) \\ &= \text{MSE}(C, \hat{C}) + \text{Euclidean}(C, \hat{C}). \end{aligned} \tag{6}$$

In response to the aforementioned problems and challenges, a deep-learning-based generative model, AACOGAN, which is based on GANs and is developed to enable automatic cinematography in the open-world environment. GANs have demonstrated impressive results in generating synthetic data samples that resemble data from a training dataset, commonly used for generating images [30] and audio [31] signals. The essence of camera movement in cinematography is the variation of the camera's position and rotation along different axis over time, which is comparable to the time-varying intensity of audio signals.

B. AACOGAN Architecture

Based on the aforementioned factors that influence the generated camera trajectory in AACOGAN, we have designed the input (green block of Fig. 2) for the generation model.

Character skeletal animation, also known as skeletal animation (Fig. 3), is a widely adopted technique in the animation field that enables the generation of realistic and intricate movement. This technique involves continuously recording A of the skeletal bones during character interactive movement, as well as computing and documenting the speed (*AV*) and acceleration (*AD*) of these bones. By including the speed and acceleration of the skeletal bones as input features, the generation model is able to make more accurate predictions about the position of the camera, particularly in cases of continuous camera movement. The initial camera position is also significant, as it cannot be forecasted by the generation model. In addition to the 3D coordinates and rotation (*IniCAM*) of the camera's initial position within the virtual environment, the relative coordinates on a toric surface (*IniTheta*) have been considered as input parameters. This is because the movement of the camera is confined to this surface in the experiment.

Character and object position, as well as the position relationship between the user-controlled character (*MPOS*) and the target interactive objects (*TPOS*), are crucial factors in determining the camera's path. In the experiment, the 3D coordinates of both the character and the interactive objects are recorded. It should be noted that there may be multiple interactive objects, and thus a set of vectors representing their positions have been included in the experiment, with a maximum of five vectors. The character's emotional intensity Emo is a crucial factor in the selection of lens language. Compared to simple emotion categorization (dividing emotions into categories like happy, sad, angry, etc.), emotional intensity is a better representation of the impact of emotions on cinematographic language [32]. This is because even the same emotion might require completely different camera presentations, such as the difference between mild anger and uncontrollable rage. In our experiment, Emo is represented by numerical values ranging from 0 to 1, indicating the level of impact that the emotion has on the character's movements, where the intensity of the emotion increases with higher numbers. Therefore, the input, O, for the generator can be represented as the collection of the above factors.

The architecture of the AACOGAN generator model is represented in Fig. 1. The generator is designed to learn the pattern of the ground truth camera drive data sequence, $\hat{C} = [\hat{c_0}, \hat{c_1}, \ldots, \hat{c_T}]$, for each interactive action and generate a corresponding sequence of camera drive data, $C = [c_0, c_1, \ldots, c_T]$, based on the sequence of observed input features, $O = [o_0, o_1, \ldots, o_T]$, where T represents the number of samples of the interactive action over the duration.

As depicted in Fig. 2, the AACOGAN generator is a neural network designed to generate a camera drive data sequence based on a given input sequence of observed features, utilizing an encoder-decoder GAN structure [33], [34]. The encoder, a function that processes a sequence of partial input features (A), produces a latent representation through feature extraction [35].

Some interactive actions, such as running or walking, involve the entire body and can impact the camera's overall path in the virtual environment. To address potential connections between various body parts and movements, in the proposed AACOGAN architecture (Fig. 3), the encoder separates input data into distinct body parts using specialized neural network layers known as Body Part Blocks (BPBs). These BPBs, trained to isolate and encode specific body regions, facilitate the learning of fine-grained representations for each region, establishing a deeper correlation between character and camera movements. This separation offers multiple benefits, including the effective capture of various body parts' engagement during different actions and enabling the encoder to focus on specific input data segments. This is particularly useful for capturing relevant body parts during intense or vigorous movements.

Subsequently, this collective feature representation undergoes a Linear Block (LB) operation, primarily designed to optimize the input data's structural compatibility with subsequent computations. This process derives a latent representation of the skeletal bones A as per the following equation:

$$A_{\text{latent}} = \text{LB}(\text{BPB}(A_{\text{head}}), \ldots, \text{BPB}(A_{\text{fully}})). \tag{7}$$

In particular, the latent representation of A retains the positions and rotations of skeletal bones, encompassing crucial information for generating camera drive data. The remaining data decodes this latent representation at various stages and from distinct perspectives based on its type. Firstly, *AV* and *AD* which are derived from *A* are processed through two independent LBs and concatenated, serving as supplementary information about the skeletal bones' position and rotation during movement. The positional data of the camera, including *IniCAM* and *IniTheta*, are processed separately through two independent LBs and concatenated. These features enhance the model's ability to establish a robust correlation between the initial camera position and camera movements. Features regarding the positions of the character and interactive object(s), *MPOS* and *TPOS*, are involved in the decoding of the latent representation after processing through two other independent LBs.

Emo is a critical component of the proposed model, integrated with other data by the generator through a single LB.

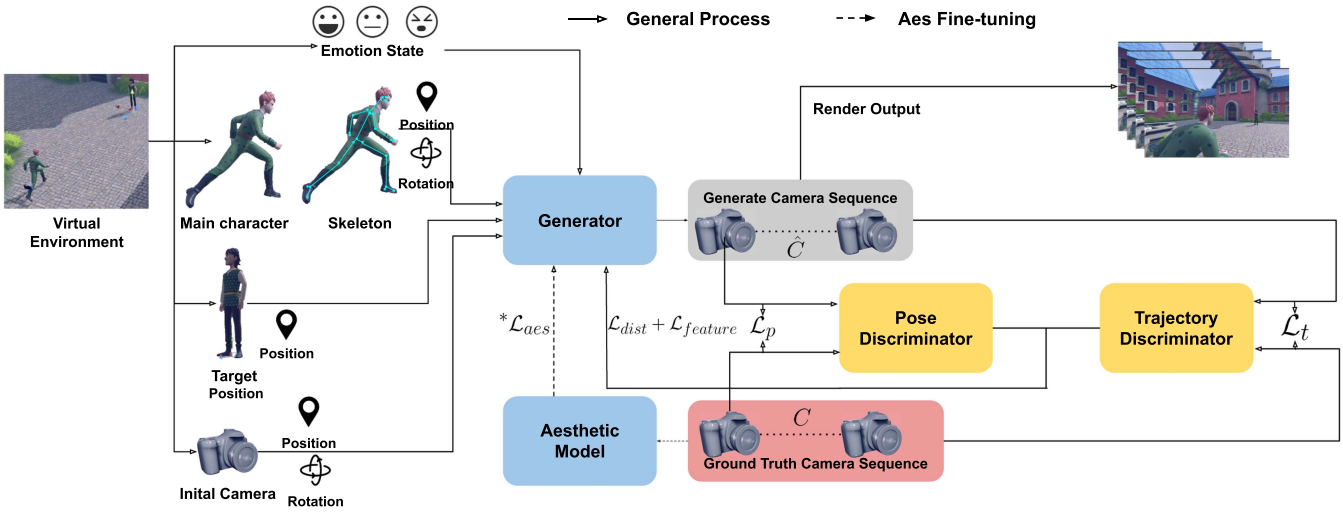

Fig. 1. AACOGAN architecture overview: C is the ground truth camera trajectory, $\hat{C}$ is the generated camera trajectory, and $\mathcal{L}$ is the loss.

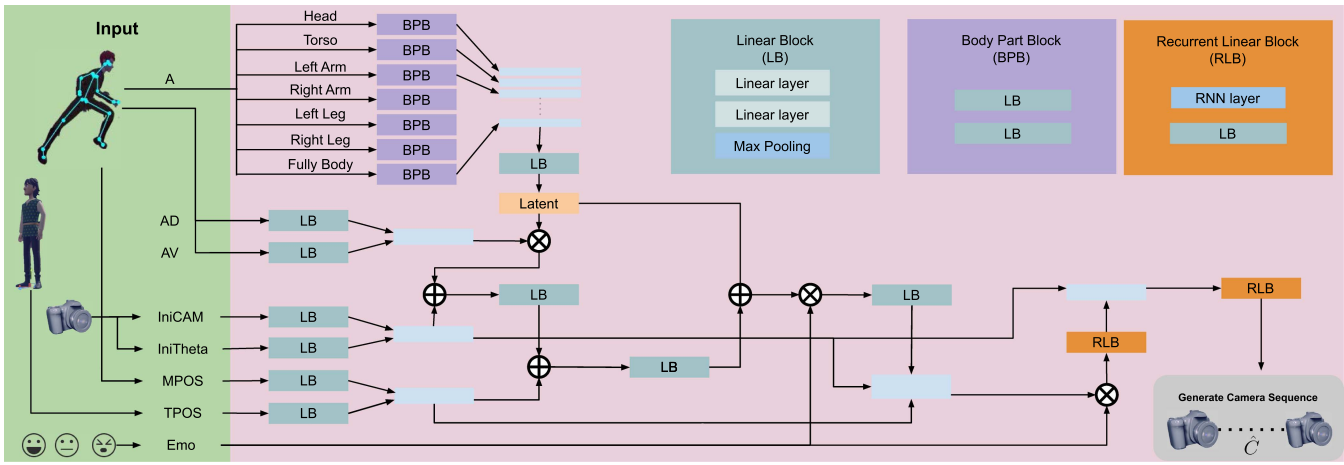

Fig. 2. AACOGAN generator architecture overview. The green blocks are the input for the generator model including skeletal key point position (A), point speed (AV), point acceleration (AD), initial camera parameters (CAM), initial theta value on camera moving surface (IniTheta), user-controlled character position (MPOS) and target objects position (TPOS). The block with "Latent" is the model to generate the latent representation through feature extraction. The detailed explanation of this overview is in Section III-B.

This integration enhances the model's capacity to establish a strong relationship between camera movements and character emotions. Since Emo is one-dimensional and remains relatively stable within a single interactive action, it is incorporated as a coefficient on all intermediate outputs during the second fusion for decoding.

The camera's movement is represented by data that continuously varies over time, requiring two Recurrent Linear Blocks (RLBs) in the generator's final portion to decode the time-varying characteristics of the latent representation.

AACOGAN employs two discriminators, inspired by [36], to evaluate the performance of the generator from two distinct perspectives, thus evaluating and enhancing the generator's performance for multiple requirements.

Accurately capturing subtle variations in the camera's position is vital to ensure the generator produces camera driving parameters closely resembling the ground truth. Given that minor differences in the shooting angle can significantly affect the overall outcome (example details shown in the experiment about the aesthetic score in Section IV-B), properly evaluating the generator's output is essential. Consequently, the pose discriminator evaluates individual data points, c_t, generated from the input features o_t.

Conversely, the trajectory discriminator assesses the complete camera drive data, C, for an entire interactive action generated by O. This discriminator is pivotal in ensuring the overall coherence and realism of the generated camera movement. By evaluating the entire sequence of camera driving data, it can determine whether the generated camera movement adheres to a plausible and natural trajectory, rather than consisting of unrelated or jarring movements. This aspect is particularly important for maintaining immersion and ensuring a seamless user experience. As such, it is critical that the trajectory discriminator accurately assesses the quality of the generated camera movement, as inaccuracies could lead to unrealistic or incoherent camera movements.

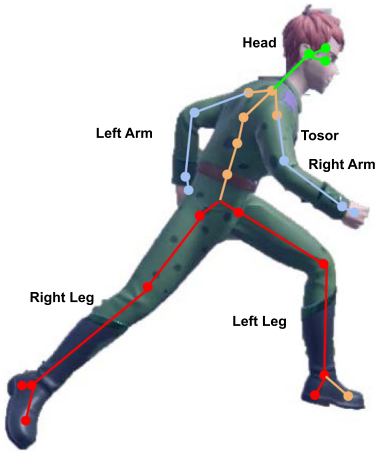

Fig. 3. 3D Character skeleton illustration. Each tetrahedron corresponds to a bone in the skeletal animation.

C. Loss Functions and Algorithm

Our goal is to maximize the $\mathsf{q}_{\mathsf{ref}}$ while minimizing the network loss during the training. Let θ represent the parameters of this camera drive data generator. Additionally, let ψ_p and ψ_t denote the parameters of the camera pose and trajectory discriminators, respectively. Then the objective function derivatives from (2) for the AACOGAN can be expressed as follows:

$$\max_{\theta} Q_{\mathsf{ref}}(\cdot) = \min_{\theta} \max_{\psi_p,\psi_t} \omega_0 \mathcal{L}_{\mathsf{dist}} + \omega_1 \mathcal{L}_{\mathsf{corr}} + \omega_2 \tilde{\mathcal{L}}_{\mathsf{aes}} + \omega_3 \mathcal{L}_p + \omega_4 \mathcal{L}_t, \tag{8}$$

where $\omega_0, \ldots, \omega_4$ are the weight factors used to balance the loss terms, the loss functions $\mathcal{L}_{\mathsf{dist}}$ and $\mathcal{L}_{\mathsf{corr}}$ represent the distance function $\mathsf{Dis}(\cdot)$ and the correlation distance function $\mathsf{D}_{\mathsf{corr}}(\cdot)$ for the AACOGAN generator model. They are employed to compute the distance and similarity between C and $\hat{C}$. $\tilde{\mathcal{L}_{\mathsf{aes}}}$ is a loss function representation of $S_{\mathsf{aes}}(\cdot)$ which utilizes an anesthetic assessment model (AES) to evaluate the conformance of the results to general aesthetic standards. This loss function differs from others in that it requires the use of the resulting captured frames for evaluation, whereas the generator only generates camera drive data. As a result, it is utilized as a separate fine-tuning mechanism for the generator model at the conclusion of training. Following the completion of a training phase, the generator is used to generate a set of parameters C for each O in the training set. Each set of these C values is used to capture interactive actions in a virtual environment as a camera shot, resulting in a corresponding video clip. Each video clip can be represented as a sequence of images, and an AES is employed to evaluate these images. Due to the camera trajectories being pre-designed within the optimization space of a toric surface, as applied in our experiments, we discern important insights from the data analysis of professional directors that we collect. The majority of cases highlighted that the two most critical elements determining the quality of the shot and the camera trajectory are the starting and ending points. Therefore, the beginning and ending frames in this sequence are given more weight in the calculation of the loss value as follows:

$$\tilde{\mathcal{L}_{\mathsf{aes}}} = \frac{1}{T} \sum_{t=1}^{T} \alpha_t (\mathsf{AES}_{\max} - \mathsf{AES}(c_t)), \tag{9}$$

where $\mathsf{AES}_{\max}$ is the max score of the employed AES and α_t is the weight factor for different frames over t.

As the number of characters captured by C can only be calculated after the actual video generation, there is no function based on $R(\cdot)$ in the loss function for the AACOGAN generator.

The discriminator loss function has two parts. The pose discriminator loss function, $\mathcal{L}_p$, which is similar to the $\mathcal{L}_{\mathsf{dist}}$ that calculates the pose difference C and $\hat{C}$ at each t. The trajectory discriminator loss function, $\mathcal{L}_t$, which is similar to the $\mathcal{L}_{\mathsf{corr}}$ that calculates the trajectory difference between C and $\hat{C}$. The discriminator loss can be defined as follows:

$$L_p = \sum_{t=1}^{T} (-\mathbb{E}[\log D(c_t, \hat{c}_t)] - \mathbb{E}[\log(1 - D(G(o_t)))]), \tag{10}$$

$$L_t = -\mathbb{E}\left[\log D(C, \hat{C})\right] - \mathbb{E}\left[\log(1 - D(G(O)))\right], \tag{11}$$

where the D is the discriminator and G is the generator.

The pseudo-code for training AACOGAN is given in Algorithm 1. In the first and second loops, ψ_p is updated separately, and θ is updated in both loops. The third loop is implemented to refine θ based on aesthetic aspects for fine-tuning.

Due to the precise camera parameters for each frame and the presence of noise during data generation, the resulting camera drive data may exhibit minor fluctuations between frames. These continuous, randomly oriented fluctuations can disrupt image continuity and reduce user immersion. To address this issue, our system's postprocessing applies smoothing to the final output data to create a more continuous curve by minimizing these fluctuations.

The smoothing process, which aims to refine the data points and generate a smoother curve in a 2D coordinate system, is achieved using a Savitzky-Golay filter [37]. To ensure the accuracy of the smoothed result in representing the underlying data, a window size of 5 is employed for the filter. The outcome is then evaluated using four different polynomials of varying degrees, ranging from 2 to 5. The smoothed result that most closely resembles the original data points is selected, thus preserving data curvature while reducing potential information loss due to the smoothing process. This step can be represented as follows:

$$C_{\mathsf{smooth}} = \min_{p=2}^{5} \mathsf{MSE}\{\mathsf{savitzky}_{\mathsf{golay}}(G(O), 5, p), G(O)\} \tag{12}$$

IV. Experiment

In this section, the evaluation of the proposed AACOGAN is carried out using both objective and subjective metrics. To the best of our knowledge, there is no prior study on automatic cinematography in open-world scenarios. Therefore, we draw comparisons between our results and conventional automatic cinematography techniques [4], [13], [20], [38] commonly employed in general games and films. These four reference works

Algorithm 1: Training Procedure of AACOGAN

Input The extracted features defined in the pre-processing $O_n = [o_{n1}, o_{n2}, \ldots, o_{nT}]$ and the corresponding ground truth camera drive data $C_n = [c_{n1}, c_{n2}, \ldots, c_{nT}]$ for $n = 1, 2, \ldots, N$.
Output Generator parameters θ and two discriminator parameters ψ_p and ψ_t.

for $epoch = 1$ to max_epoch **do**
  for $iter_p = 1$ to k_p **do**
    Sample a mini-batch of input features and camera drive data pairs in frame $\{(o_t, c_t)\}$ from the training set.
    Generate a single camera drive data point $\hat{c}_t$.
    Calculate the $\mathcal{L}_{k_p} = \mathcal{L}_{\text{dist}} + \mathcal{L}_{\text{corr}} + \mathcal{L}_p$
    Update ψ_p and θ.
  end for
  for $iter_t = 1$ to k_t **do**
    Sample a mini-batch of input features and camera drive data for the entire interactive action pairs in $\{(O, C)\}$ from the training set.
    Generate camera drive data for the entire integrative action $\hat{C}$.
    Calculate the $\mathcal{L}_{k_t} = \mathcal{L}_{\text{dist}} + \mathcal{L}_{\text{corr}} + \mathcal{L}_t$
    Update ψ_t and θ.
  end for
end for
for $iter_t = 1$ to k_t **do**
  Sample a mini-batch of input features and aesthetic score pair for each interactive action.
  Calculate the $\tilde{\mathcal{L}_{aes}}$
  Update θ.
end for

have tackled the problem of automatic cinematography using two different approaches: a rule-based auto-cinematography approach [4], [20] and a behavior learning approach [13], [38]. For [20], we have adjusted the original loss function from dialogue-focus to character-action emphasis since the open-world scenarios are more focus on the actions. Other significant parameters like screen position, shot types, visibility, etc, have been retained.

In [38], a Recurrent Neural Network (RNN) is used to capture the essence of cinematography, while [13] focuses on individual characters and employs a GAN-based model to generate camera movement according to the characters' emotions and actions. More experiment example video footage can be found at https://youtu.be/vhvgvE-DU2Y.

A. MineStory Dataset

Given the complexity and dynamism of user interactions within open-world environments, we developed a novel interactive action dataset named the MineStory dataset. This dataset, created using motion capture techniques, trains the AACOGAN to adapt to the diverse and ever-evolving nature of user actions in such virtual environments.

The MineStory dataset encompasses a total of 546 distinct actions, including most actions typically used in animation production and some actions specially tailored for our product. Each action is captured by a standard protocol that involves 25 joint nodes distributed throughout the character's entire body. The positional and rotational data of each node is captured at a rate of 30 frames per second, yielding a comprehensive skeletal animation.

To train the AACOGAN, we need a set of camera trajectory data for each action, as directed by a human operator in an open-world environment. We employ the toric surface method, as introduced in [8], to simplify the process of generating this data. This method conceptualizes the toric surface as a two-dimensional plane for creating camera movement trajectories, which can then be projected into a three-dimensional virtual space. This approach allows for the efficient generation of extensive camera trajectory data with a limited number of human operators. In our experiments, we generated 2 to 8 distinct sets of camera trajectory data for each interactive action. This data generation employed a randomized combination of parameters tailored to the unique aspects of the interactive activities, such as the characters' emotional states and the distance of interaction.

B. Training Detail

For the general training process, we trained the AACOGAN by using the Minestory dataset with a batch size of 128 and a learning rate of 0.0002. Adam optimizer [39] was employed with $\beta_1 = 0.5$ and $\beta_2 = 0.999$. Due to the varying durations required for each action performance, we standardized all actions based on the one with the longest completion time. The missing camera data and character skeleton data were filled using the information from the last frame. The model was trained for 3,000 epochs, with the loss function consisting of adversarial loss, pose, and trajectory discriminator loss. The weight for adversarial loss and discriminator loss in the overall loss function was set to 0.5 for both scenarios. All layer dropout rate was set to 0.2.

For the aesthetic model fine-tuning phase, we began by fine-tuning the NIMA aesthetic model [26], using frames captured from the Minestory dataset. Subsequently, we employed this fine-tuned NIMA model to evaluate frames captured by the output camera sequence from AACOGAN. This evaluation served as the aesthetic loss ($\mathcal{L}_{aes}$), which was then used to further fine-tune the AACOGAN.

C. Objective Numerical Comparison

The comparisons between the AACOGAN and the baseline model have been performed with regard to several important aspects of automatic photography technology in open-world scenarios.

1) Precise Difference in Camera Position and Rotation: The camera's location and orientation in the environment are specified by a set of six parameters, three of which pertain to position and three to rotation. This metric directly compares the

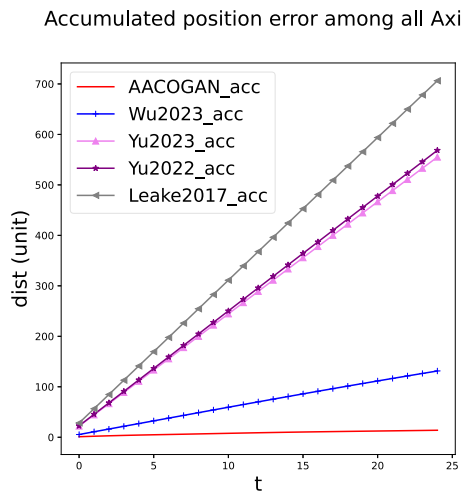

Fig. 4. Accumulated difference between parameters of the generated camera positions and the ground truth camera position calculated over time, with the error distance expressed in unit distance.

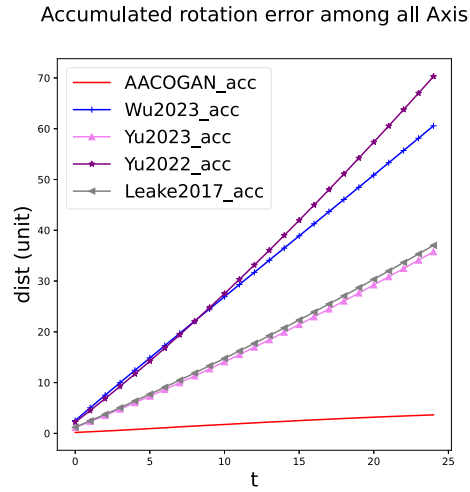

Fig. 5. Accumulated differences between the generated camera parameters and the ground ones truth over time are shown for all the rotation parameters. The error distance is expressed in radians.

difference between the generated camera position and rotation parameters with the ground truth for each approach. Figs. 4 and 5 present the sum of differences of all the x, y, and z axes between the generated camera parameters and the ground truth parameters for position parameters and rotation parameters are presented, respectively, over time. Compared to other methods, AACOGAN exhibits the least deviation in position from the ground truth in terms of results. The results show that there has been a reduction in the average positional error of 1.56 units (93.6%) and an average reduction in the rotational error of 1.11 radians (55%), where the 'unit' for distance measurement is the unit distance for object positioning in the virtual environment.

Fig. 6 offers a comprehensive depiction of the accumulated distance error for position parameters along the x, y, and z axes. The upper section of Fig. 6 displays the accumulated difference between generated camera parameters and ground truth over time, for each axis individually. AACOGAN exhibits the smallest accumulated discrepancies across all scenarios in comparison to other methods. It is crucial to recognize that mere similarity in camera parameter numerical values does not guarantee similarity in camera movement. The camera's trajectory in three-dimensional space throughout the shot is of utmost significance. Consequently, we extended our evaluation by comparing the variation curves of generated camera parameters for each method along each axis over time. The bottom section of Fig. 6 demonstrates the variation of camera rotation parameters over time, with the visual similarity between the AACOGAN-generated camera parameters and ground truth parameters being more distinct than the two baseline models.

Fig. 7 displays a visual comparison of generated camera shots from different methods in terms of shooting position and angle. When contrasted with ground truth camera shots, the AACOGAN-generated frames exhibit a higher degree of similarity, corroborating the numerical results.

In summary, the camera movement generated by the AACOGAN model more closely approximates the ground truth, indicating that our method is more adept at learning the film director's lens language in open-world scenarios.

2) Correlation of Trajectories: The correlation distance metric is employed to calculate the similarity between generated camera motion and the actual motion of subjects during interactive actions. As previously mentioned in [10], [11], when camera movement closely mirrors the subjects' movements, it can enhance the sense of immersion for the audience. The significance of body parts in real action is often gauged by their range of motion and velocity.

In this experiment, the joint exhibiting the largest range of motion and fastest movement was selected as a reference to assess the correlation between subject and camera movements. Table II presents the similarity between various camera parameter curves generated by different automatic cinematography methods and the ground truth, quantified using correlation distance. The camera movement generated by AACOGAN is evidently more similar to the actual subject movement than other methods, especially near peaks of subject movement.

The similarity between the camera trajectory generated by AACOGAN and the manually created camera exhibits the lowest discrepancy at 27%, while other methods display a minimum discrepancy of 95.3%. Compared to the other two references, the camera trajectory produced by AACOGAN demonstrates a decrease of 0.78 (73%) in the average correlation distance. This synchronized movement between the camera and the action can provide a superior experience for the user.

Fig. 8 illustrates the comparison between the camera transport mirror trajectory generated by different methods and the skeletal animation motion trajectory for the x-axis in the experiment environment. The y-axis in Fig. 8 represents the position of the character's skeletal joint (right y-axis) or the camera position (left y-axis) relative to the previous time step for each time step. It can be observed that the trajectory generated by AACOGAN has a higher similarity to the motion trajectory. The corresponding example video footage can be found at https://youtu.be/M24bHDvDnqk.

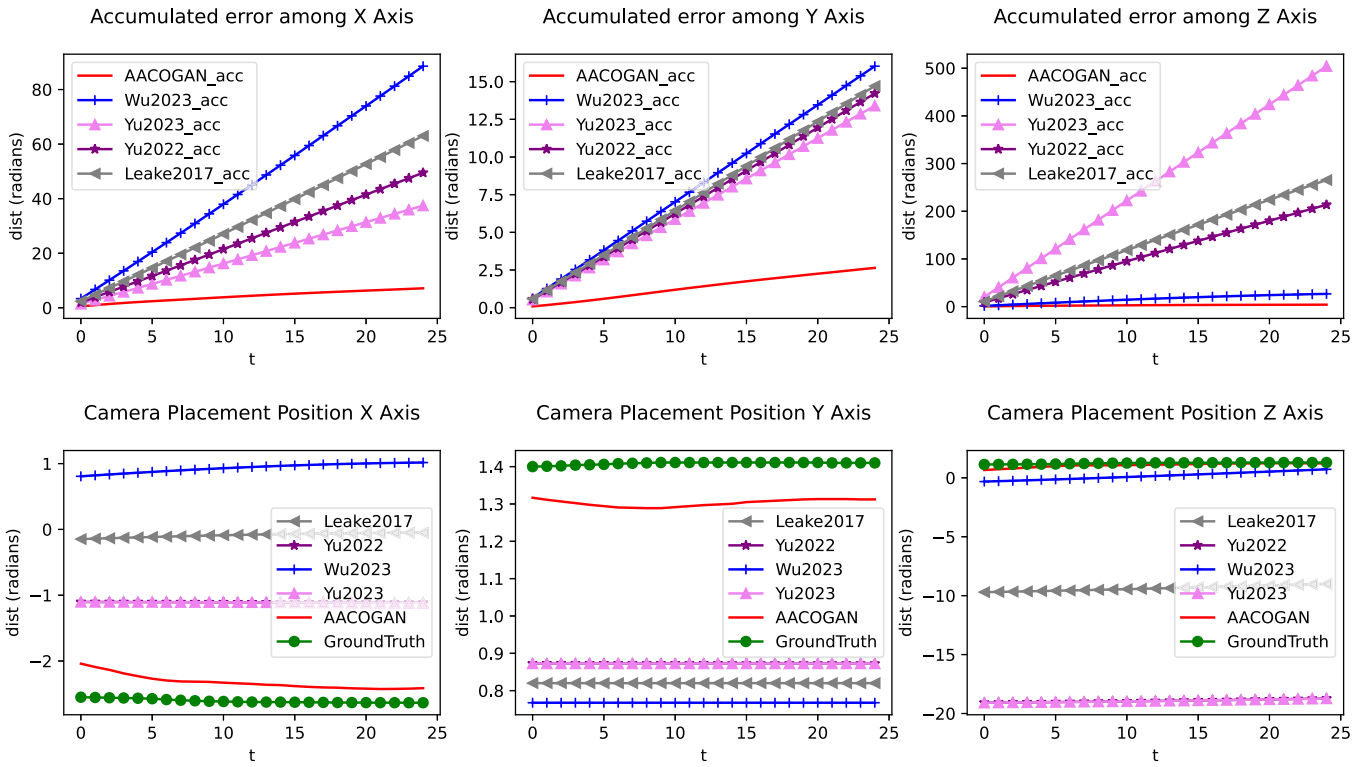

Fig. 6. Top: The accumulated differences between the generated camera parameters by different methods and the ground truth ones are displayed over time. The parameters for the x, y, and z-axis are shown individually. Bottom: the exact values of the parameters generated by different approaches for each axis at each time point are presented.

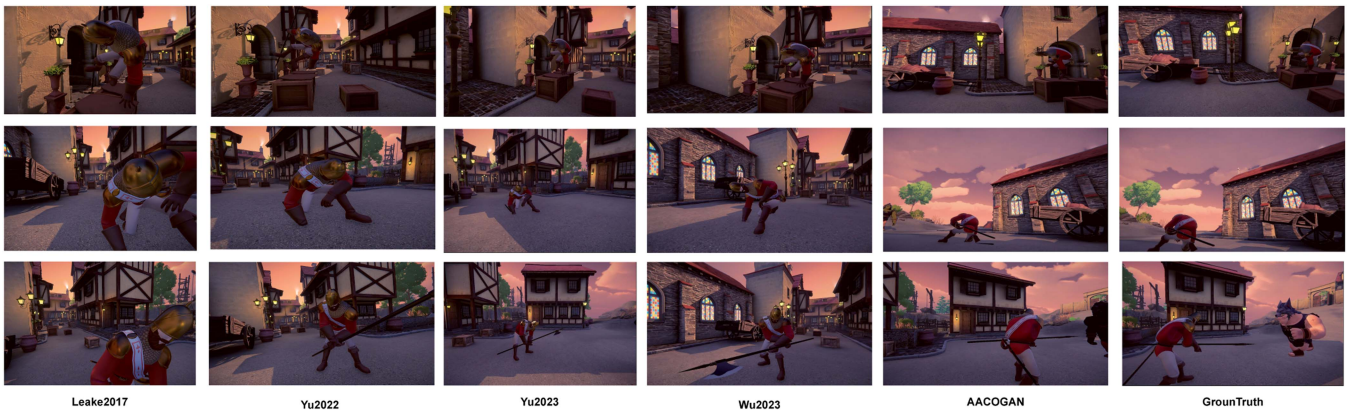

Fig. 7. Compare the frames captured by the virtual camera generated by different methods. From the visual inspection of these frames, we can intuitively observe the differences between the generated camera shots and the ground truth, in terms of position and orientation.

TABLE II
CORRELATION DISTANCE BETWEEN THE VARIOUS GENERATED CAMERA PARAMETERS AND ACTUAL SKELETON ANIMATION MOVEMENT AMONG DIFFERENT AXIS FOR A SINGLE ACTION

	Leake2017	Yu2022	Wu2023	Yu2023	Manual	AACOGAN
X axis D_{corr}	0.81	0.83	0.84	0.95	0.43	0.55
Y axis D_{corr}	1.31	1.29	1.25	1.18	0.005	0.066
Z axis D_{corr}	2.19	2.20	1.78	1.1	0.17	0.25
Average D_{corr}	1.43	1.44	1.29	1.07	0.2	0.288

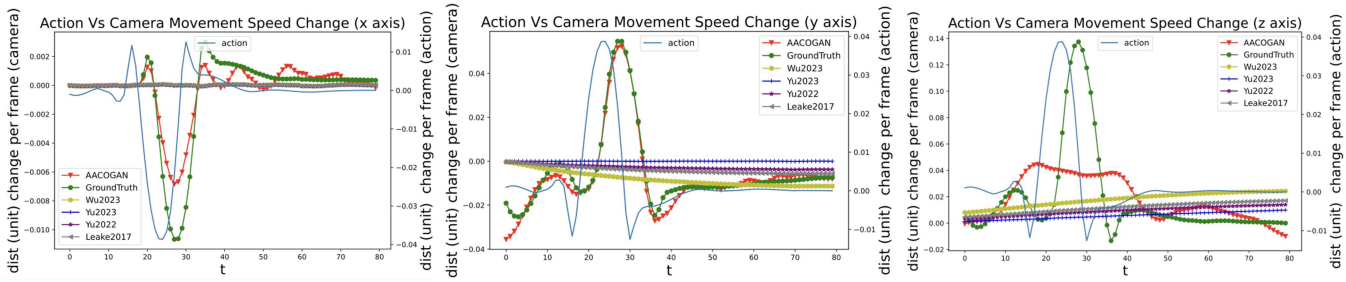

Fig. 8. Interactive action comparisons of the change curves of the different axis parameters of the camera position generated by different methods over a certain period of time with the corresponding skeletal animation motion curves of the interactive action.

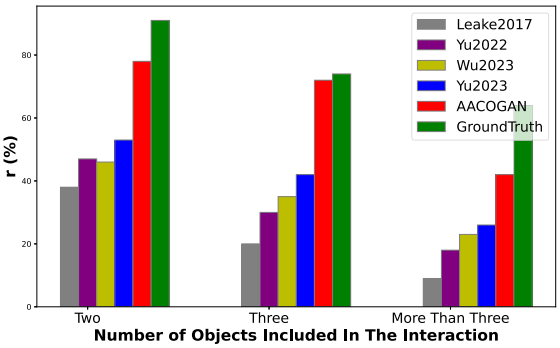

Fig. 9. Comparison of different methods for calculating the r value under varying numbers of individuals and scenes.

Fig. 10. Frames captured by different automatic cinematography techniques in a dance scene.

3) Multi-Focus: In open-world settings, interactive actions may not be solely centered on specific characters or objects. In such instances, the lens language employed by automatic cinematography technology should encompass as many related objects as feasible. This metric gauges the proportion of time during which camera movements, generated by various methods, successfully capture specified objects in multi-focus scenes.

Higher proportions generally imply superior performance and user experience. A comparison of the capture ratio for designated characters or objects during interactive actions between AACOGAN and alternative methods is depicted in Fig. 9. The results reveal that AACOGAN captures more relevant characters over extended durations, thereby augmenting users' overall viewing experience. In comparison to other reference methods, AACOGAN demonstrates an average improvement of 22% and up to 32.9% in multi-focal scene image capturing, contingent upon content.

Fig. 10 displays frames captured by distinct automatic cinematography techniques within a given multi-person interactive dance scene. Observing the image, the lens language generated by AACOGAN captures more comprehensive character images within the scene. The corresponding video footage example can be accessed at https://youtu.be/3KImvj9wabg.

4) Aesthetic Score: The AES [26], [29] offers an objective evaluation of images and assigns scores based on their aesthetic quality, with a higher score indicating a higher level of aesthetic appeal. This model is widely utilized in the realm of image and video content generation and provides valuable insights through artistic analysis of the output generated by these technologies. To the best of our knowledge, AACOGAN is the first work to apply this type of model in the field of automatic cinematography for shot generation. By incorporating aesthetic scores as part of the input for fine-tuning the generator of the AACOGAN, the resulting shots have higher aesthetic ratings than the original model without aesthetic-related adjustments.

In our experiments, employing AES led to a 9% average increase in the aesthetic score of images captured by the AACOGAN. For actions like over-wall jumping, illustrated in Fig. 11, the right-side camera shot generator captures richer lighting, background, and environmental information compared to the left-side perspective. This result is achieved by slightly lowering the camera position and raising the shooting angle. This distinction can be ascribed to the use of aesthetic scores in the

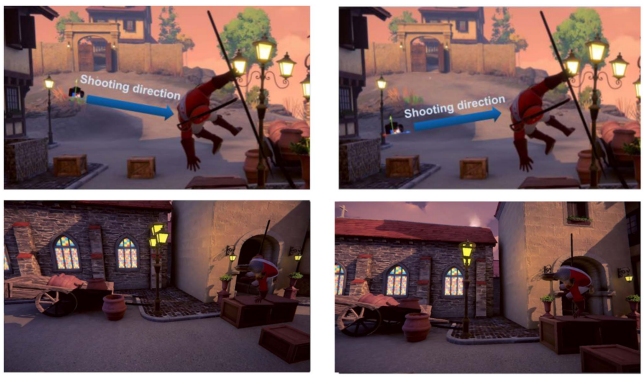

Fig. 11. Use of aesthetic scores as a part of the input for the generator of AACOGAN resulted in improved visuals, as seen in the comparison between the original camera shooting direction (left) and the fine-tuned version camera shooting direction (right) after incorporating aesthetic considerations.

TABLE III
REAL-TIME PERFORMANCE METRICS FOR AACOGAN MODEL WITH DIFFERENT INPUT LATENCY

Model	FLOPs(M)	FPS	Latency	Average D_{corr}
AACOGAN(5)	64.33	78.9	0.012	0.393
AACOGAN(10)	113.93	39.63	0.025	0.356
AACOGAN(20)	217.97	20.79	0.048	0.307
AACOGAN(30)	317.28	14.2	0.071	0.288

The number in parentheses after the model indicates the number of delayed frames. The FLOPs unit is in a million flops per second.

fine-tuning process of AACOGAN's generator, causing a preference for content-rich camera angles over monotonous ones, such as those oriented towards the sky or ground. Aesthetic assessment models typically favor images with more content. The corresponding video footage example can be accessed at https://youtu.be/_je_Gg_QQG0.

The benefit of this additional aesthetic fine-tuning for the generator is twofold. Firstly, it enhances the aesthetic quality of the automatically generated camera movement. Secondly, it preserves the quality of shots produced by the AACOGAN. This results in shots that better align with the preferences of a broader audience, without compromising the utilized lens language.

5) Real-Time Performance: In Table III, we present a detailed comparison of the performance of the AACOGAN model at various latency levels ranging from 5 to 30 frames. For each latency level, the table provides data on the frames per second (FPS) and floating point operations per second (FLOPS) metrics. We use average D_{corr} to represent the quality of the generated camera position. The data clearly demonstrate an increase in latency, which effectively allows the model to leverage more frame data, and improves the predictive capability of the model for the subsequent camera position. However, this is achieved at the expense of an increase in computational complexity, as shown by the higher FLOPS. Thus, it becomes a trade-off between real-time responsiveness and the quality of camera position prediction, necessitating careful tuning according to the specific demands of the gaming environment.

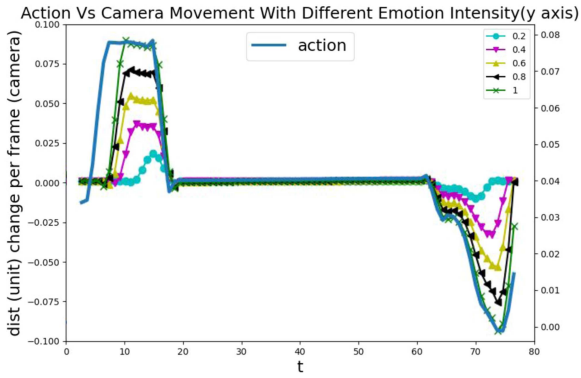

Fig. 12. Comparison of the change curves of the y-axis position parameter of the camera trajectory generated by different emotional states.

D. Subjective Evaluation

1) Emotional Reinforcement: The lens language also plays an essential role in expressing the atmosphere and emotions of characters. The emotion of a character can be analyzed through the user's facial expressions [40], speech [41], or body movements [42]. In different scenarios or characters' emotional states, the camera movement should reflect the emotions of the characters by varying to a greater or lesser degree.

This metric assesses the influence of a character's emotional state on camera trajectory, where stronger emotions should yield a more significant impact on camera movement. The comparison of the character's emotional state's effect on camera movements generated for identical interactive actions is depicted in Fig. 12. The y-axis value, representing emotion, denotes the stability of the character's emotional state; a value closer to 1 indicates an unstable or extremely unstable emotion (such as anger), while a value closer to 0 signifies a calmer state.

The y-axis in Fig. 12 represents the character's skeletal joint position (right y-axis) or the camera position (left y-axis) relative to the previous time step for each time step. As the character's emotional state intensifies, a more pronounced impact on camera movement is evident. Consequently, the camera movement generated by AACOGAN more accurately reflects the characters' actual emotions and creates a superior atmosphere compared to reference methods. The corresponding video footage example can be found at https://youtu.be/HfmotyfEWHw.

2) Human Evaluation: It is crucial to acknowledge that the previously mentioned evaluation methods do not entirely capture the superiority of AACOGAN's generated camera shots in open-world contexts compared to other automatic cinematography techniques concerning actual user experience. Consequently, enlisting real users to assess the generated shots is indispensable. In addition to a general ranking-based human evaluation, participants will be asked to appraise the shots in distinct aspects: 1) shot quality (Frames Quality), 2) shot consistency with the actual interaction (Consistence), 3) representation of characters and objects involved in the interaction (Completeness), and 4) the enhancement of emotions conveyed by the characters within the shot (Emotional Enhancement).

TABLE IV
RESULTS OF THE SUBJECTIVE EVALUATION

	leake2017	Yu2022	Wu2023	Yu2023	GroundTruth	AACOGAN
Frames Quality	4	4.16	4	4	**4.67**	4.5
Consistency	2.5	2.33	3.16	3	4.5	**4.67**
Completeness	2.5	2.83	2.3	2.16	**3.83**	**3.83**
Emotional	2	3	3.83	2	**4.6**	4.16
Overall Ranking	5	4.6	3.4	4.8	**1.4**	1.8

The scores for the four aspects of the shot quality are on a scale of 1-5, with 5 being the best. The final row shows the ranking of the four shots created by different methods, with a lower score indicating a better ranking.

As shown in Table IV, the results of the subjective evaluation indicate that compared to the baseline method, AACOGAN received the highest ranking in the sorting task. In comparison to other methods across various aspects, AACOGAN also received higher evaluations.

In summary, our experimental outcomes indicate that AACOGAN effectively generates camera parameters and produces shots more akin to those captured by human operators during interactive actions. This is demonstrated by comparing AACOGAN-generated parameters to baseline methods using metrics such as camera pose similarity, motion similarity, and character and object coverage. Moreover, aesthetic assessments and subjective evaluations conducted by human participants corroborate AACOGAN's superiority in creating visually appealing and interaction-aligned shots. Consequently, these findings suggest the potential of the AACOGAN method to enhance the quality of automatic cinematography in open-world interactive scenarios.

V. CONCLUSION

The advent of multimedia technology in the entertainment industry has bestowed upon consumers an unprecedented level of autonomy in their media consumption experiences. Within virtual open-world environments, automatic cinematography has emerged as an instrumental factor in delivering immersive experiences, catering to the users' growing demand for more engaging forms of interaction. In this study, we introduce AACOGAN, a technique for automatic cinematography designed for user-initiated interactions within open-world scenarios, leveraging GANs. This model incorporates various elements such as skeletal animations of interactive actions, positional relationships between interactive objects and characters, as well as character emotions, facilitating the effective generation of suitable camera movements for a wide array of interactive actions. Moreover, the integration of aesthetic scores into the generator's training process significantly enhances the quality of the shots generated. The results of our experiments substantiate the efficiency of the AACOGAN approach in producing camera shots that rival the quality of human-generated shots, demanding minimal input, thus leading to a more engaging user experience and substantially reducing costs.

REFERENCES

[1] P. Sweetser and D. Johnson, "Player-centered game environments: Assessing player opinions, experiences, and issues," in *Proc. Int. Conf. Entertainment Comput.*, 2004, pp. 321–332.

[2] H. Subramonyam, W. Li, E. Adar, and M. Dontcheva, "Taketoons: Script-driven performance animation," in *Proc. 31st Annu. ACM Symp. User Interface Softw. Technol.*, 2018, pp. 663–674.

[3] C. Liang, C. Xu, J. Cheng, W. Min, and H. Lu, "Script-to-movie: A computational framework for story movie composition," *IEEE Trans. Multimedia*, vol. 15, pp. 401–414, 2012.

[4] Z. Yu, H. Wang, A. K. Katsaggelos, and J. Ren, "A novel automatic content generation and optimization framework," *IEEE Internet Things J.*, vol. 10, no. 14, pp. 12338–12351, Jul. 2023.

[5] Z. Yu, E. Guo, H. Wang, and J. Ren, "Bridging script and animation utilizing a new automatic cinematography model," in *Proc. IEEE 5th Int. Conf. Multimedia Inf. Process. Retrieval*, 2022, pp. 268–273.

[6] P. Burelli, "Game cinematography: From camera control to player emotions," in *Emotion in Games*. Berlin, Germany: Springer, 2016, pp. 181–195.

[7] P. Burelli, "Virtual cinematography in games: Investigating the impact on player experience," in *Proc. 8th Int. Conf. Found. Digit. Games*, 2013, pp. 134–141.

[8] C. Lino and M. Christie, "Efficient composition for virtual camera control," in *Proc. ACM SIGGRAPH/Eurographics Symp. Comput. Animation*, 2012, pp. 65–70.

[9] R. F. Simons, B. H. Detenber, T. M. Roedema, and J. E. Reiss, "Emotion processing in three systems: The medium and the message," *Psychophysiology*, vol. 36, no. 5, pp. 619–627, 1999.

[10] M. Haigh-Hutchinson, *Real Time Cameras: A Guide for Game Designers and Developers*. Boca Raton, FL, USA: CRC Press, 2009.

[11] M. Christie, P. Olivier, and J.-M. Normand, "Camera control in computer graphics," *Comput. Graph. Forum*, vol. 27, pp. 2197–2218, 2008.

[12] B. Tomlinson, B. Blumberg, and D. Nain, "Expressive autonomous cinematography for interactive virtual environments," in *Proc. 4th Int. Conf. Auton. Agents*, 2000, pp. 317–324.

[13] X. Wu, H. Wang, and A. K. Katsaggelos, "The secret of immersion: Actor driven camera movement generation for auto-cinematography," 2023, *arXiv:2303.17041*.

[14] Q. Galvane, R. Ronfard, M. Christie, and N. Szilas, "Narrative-driven camera control for cinematic replay of computer games," in *Proc. 7th Int. Conf. Motion Games*, 2014, pp. 109–117.

[15] Y. Deng, C. C. Loy, and X. Tang, "Image aesthetic assessment: An experimental survey," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 80–106, Jul. 2017.

[16] A. McMahan, "Immersion, engagement, and presence: A method for analyzing 3-D video games," in *The Video Game Theory Reader*. Evanston, IL, USA: Routledge, 2013, pp. 67–86.

[17] N. Tandon, G. Weikum, G. d. Melo, and A. De, "Lights, camera, action: Knowledge extraction from movie scripts," in *Proc. 24th Int. Conf. World Wide Web*, 2015, pp. 127–128.

[18] D. B. Christianson et al., "Declarative camera control for automatic cinematography," in *Proc. AAAI Conf. Artif. Intell.*, 1996, pp. 148–155.

[19] M. Wang et al., "Write-a-video: Computational video montage from themed text," *ACM Trans. Graph.*, vol. 38, no. 6, 2019, Art. no. 177.

[20] M. Leake, A. Davis, A. Truong, and M. Agrawala, "Computational video editing for dialogue-driven scenes," *ACM Trans. Graph.*, vol. 36, no. 4, 2017, Art. no. 130.

[21] A. Louarn, M. Christie, and F. Lamarche, "Automated staging for virtual cinematography," in *Proc. 11th Annu. Int. Conf. Motion, Interact., Games*, 2018, pp. 1–10.

[22] Y. Dang et al., "Path-analysis-based reinforcement learning algorithm for imitation filming," *IEEE Trans. Multimedia*, vol. 25, pp. 2812–2824, 2023.

[23] C. Lino and M. Christie, "Intuitive and efficient camera control with the toric space," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 1–12, 2015.

[24] X. Lu, Z. Lin, H. Jin, J. Yang, and J. Z. Wang, "Rating image aesthetics using deep learning," *IEEE Trans. Multimedia*, vol. 17, pp. 2021–2034, 2015.

[25] X. Tian, Z. Dong, K. Yang, and T. Mei, "Query-dependent aesthetic model with deep learning for photo quality assessment," *IEEE Trans. Multimedia*, vol. 17, pp. 2035–2048, 2015.
[26] H. Talebi and P. Milanfar, "NIMA: Neural image assessment," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3998–4011, Aug. 2018.
[27] M. Radut, M. Evans, K. To, T. Nooney, and G. Phillipson, "How good is good enough? The challenge of evaluating subjective quality of AI-edited video coverage of live events," in *Proc. Workshop Intell. Cinematography Editing*, 2020, pp. 17–24.
[28] M. Haigh-Hutchinson, "Fundamentals of real-time camera design," in *Proc. Game Developers Conf.*, 2005, sec. 7.2, p. 17.
[29] L. Zhao et al., "Representation learning of image composition for aesthetic prediction," *Comput. Vis. Image Understanding*, vol. 199, 2020, Art. no. 103024.
[30] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," 2018, *arXiv:1809.11096*.
[31] C. Donahue, J. McAuley, and M. Puckette, "Adversarial audio synthesis," 2018, *arXiv:1802.04208*.
[32] T. Heiderich, "Cinematography techniques: The different types of shots in film," *Videomakers*, vol. 22, 2012, Art. no. 2020.
[33] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1125–1134.
[34] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2242–2251.
[35] I. Goodfellow et al., "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.
[36] C. Hardy, E. Le Merrer, and B. Sericola, "MD-GAN: Multi-discriminator generative adversarial networks for distributed datasets," in *Proc. IEEE Int. Parallel Distrib. Process. Symp.*, 2019, pp. 866–877.
[37] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.
[38] Z. Yu, C. Yu, H. Wang, and J. Ren, "Enabling automatic cinematography with reinforcement learning," in *Proc. IEEE 5th Int. Conf. Multimedia Inf. Process. Retrieval*, 2022, pp. 103–108.
[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[40] P. Tarnowski, M. Kołodziej, A. Majkowski, and R. J. Rak, "Emotion recognition using facial expressions," *Procedia Comput. Sci.*, vol. 108, pp. 1175–1184, 2017.
[41] S. G. Koolagudi and K. S. Rao, "Emotion recognition from speech: A review," *Int. J. Speech Technol.*, vol. 15, pp. 99–117, 2012.
[42] F. Ahmed, A. S. M. H. Bari, and M. L. Gavrilova, "Emotion recognition from body movement," *IEEE Access*, vol. 8, pp. 11761–11781, 2020.