ObjectAlign: Neuro-Symbolic Object Consistency Verification and Correction

Anonymous CVPRW submission

Paper ID 17

Abstract

Video editing and synthesis often introduce object inconsistencies, such as frame flicker and identity drift that degrade perceptual quality. To address these issues, we introduce ObjectAlign, a novel framework that seamlessly blends perceptual metrics with symbolic reasoning to detect, verify, and correct object-level and temporal inconsistencies in edited video sequences. The novel contributions of ObjectAlign are as follows: First, we propose learnable thresholds for metrics characterizing ***object consistency*** *(i.e. CLIP-based semantic similarity, LPIPS perceptual distance, histogram correlation, and SAM-derived object–mask IoU). Second, we introduce a neuro-symbolic verifier that combines two components: (a) a formal, SMT-based check that operates on masked object embeddings to provably guarantee that object identity does not drift, and (b) a temporal fidelity check that uses a probabilistic model checker to verify the video's formal representation against a temporal logic specification (Φ). A frame transition is subsequently deemed "consistent" based on a single logical assertion that requires satisfying both the learned metric thresholds and this unified neuro-symbolic constraint, ensuring both low-level stability and high-level temporal correctness. Finally, for each contiguous block of flagged frames, we propose a neural network based interpolation for adaptive frame repair, dynamically choosing the interpolation depth based on the number of frames to be corrected. By decoupling consistency verification from the heavy generative diffusion process, ObjectAlign operates with* ***minimal computational overhead****, making it highly suitable for resource-constrained edge deployments. Our results show up to 1.4 point improvement in CLIP Score and up to 6.1 point improvement in warp error compared to SOTA baselines on the DAVIS and Pexels video datasets.*

1. Introduction

Recent advances in artificial intelligence have significantly enhanced the quality, realism, and efficiency of synthetic image and video generation models [14, 18, 35, 44]. These improvements have broadened applications in content creation, real-time video editing, and interactive media [10, 27, 33]. Despite these strides, a critical yet often overlooked challenge persists, namely *maintaining consistent object representation* across different video frames. This is important since subtle inconsistencies, including semantic drift, visual flickering, or transient artifacts, frequently arise during video synthesis and editing, diminishing the visual coherence and perceptual realism [14].

Current diffusion-based editing methods [14, 32, 37, 42, 53], predominantly use extended attention mechanisms to propagate information across frames to maintain temporal coherence. However, extending attention across multiple frames significantly increases the computational cost and memory requirements, often becoming prohibitively expensive[14, 37] and unsuitable for deployment on memory-constrained edge devices. Moreover, these approaches do not provide formal guarantees for consistency, leaving room for errors that degrade the video quality.

To overcome these limitations, there is an emerging need for robust verification methods capable of *provably* ensuring consistency between frames. Unlike methods relying solely on perceptual metrics which may still miss subtle inconsistencies or offer no formal assurances, a provable guarantee, such as that provided by a Satisfiability Modulo Theories (SMT) solver [11], can offer a mathematically-grounded assertion that specified consistency constraints (e.g., bounds on semantic feature drift) are met. This is crucial for detecting errors that evade heuristic checks.

In this paper, we propose **ObjectAlign**, a neuro-symbolic framework that can rigorously verify and adaptively repair object-level and temporal inconsistencies in edited video sequences. Our approach bridges perceptual metrics with symbolic verification techniques, ensuring both practical performance and formal consistency guarantees. To this end, we introduce three key contributions:

- First, we propose a new methodology that integrates multiple perceptual and semantic metrics, including CLIP-based semantic similarity, LPIPS perceptual distance, color histogram correlation, and segmentation-mask IoU, into a unified, *learnable threshold-based classifier* for identifying object-level inconsistencies. By learning thresholds

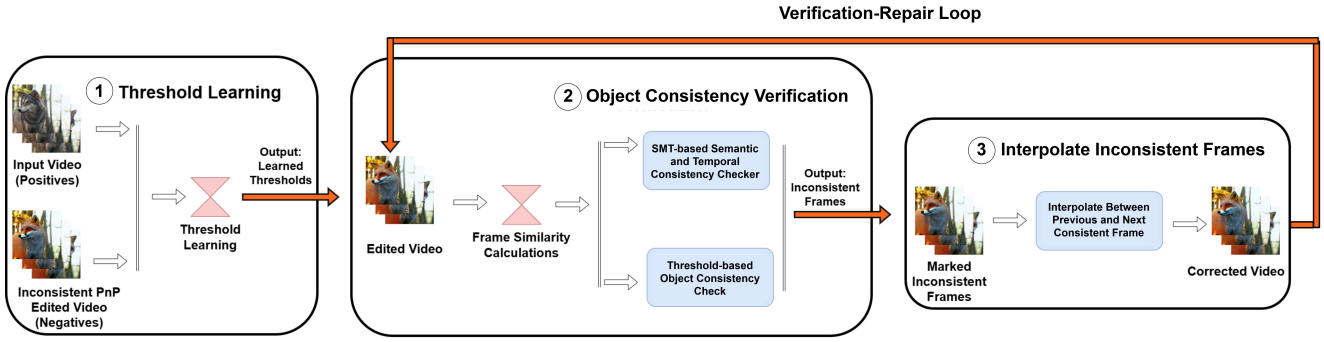

Figure 1. **Overview of ObjectAlign.** ① We first learn per-metric consistency thresholds from "positive" original video clips and "negative" inconsistently edited clips. ② Next, for each consecutive frame pair in a newly edited video, we compute semantic and perceptual similarities and apply both the learned threshold checks and an SMT-based object consistency check on the embeddings to flag inconsistent transitions. ③ Finally, each contiguous block of flagged frames is repaired by adaptively interpolating between the nearest preceding and succeeding consistent keyframes, with the interpolation depth chosen according to the segment length. The corrected frames can then be re-verified in a closed loop until no inconsistencies remain.

directly from data, our approach offers both flexibility and interpretability in inconsistency detection.

- Second, we introduce a formal verification method to *provably verify semantic and temporal consistency*. Specifically, we embed object features as constraints within a symbolic reasoning framework, enforcing per-dimension semantic bounds on masked CLIP embeddings. We also ensure temporal fidelity verification through a probabilistic model checker to verify the video satisfies a given temporal logic specification (Φ). This ensures a mathematically grounded guarantee of semantic and temporal consistency within defined thresholds.
- Finally, we develop an *adaptive interpolation strategy* for correcting flagged inconsistencies. Our repair mechanism dynamically adjusts the interpolation depth based on the number of contiguous inconsistent frames identified, reconstructing corrupted frames from adjacent consistent keyframes, thus preserving a smooth temporal coherence.

Indeed, as shown in Figure 1, ObjectAlign effectively integrates learnable perceptual metrics, formal semantic verification, and adaptive interpolation-based correction into a unified end-to-end pipeline. Our evaluation demonstrates that ObjectAlign reduces perceptual flickering and semantic drift, decreasing the warp error [30] from 107.4 to 101.3 compared to Plug and Play Diffusion (PnP) [53] on clips from the DAVIS [45] and Pexels [1] video datasets.

The remainder of this paper is structured as follows: Section 2 discusses related work in video synthesis and formal verification techniques. Section 3 provides necessary background on diffusion models, perceptual metrics, and SMT solvers. Section 4 describes the ObjectAlign methodology and technical innovations in detail. Experimental results and ablations are presented in Section 5. Finally, Section 6 summarizes our main contributions.

2. Related work

2.1. Video Editing and Object Consistency

Recent works have explored training-free frameworks for improving or stylizing text-to-video generation by leveraging pre-trained text-to-image (T2I) models to edit video frames [21, 23, 54, 58]. Approaches such as SDEdit [42], InstructPix2Pix [4], and ControlNet [62] provide general-purpose image editing capabilities that have been adapted for video by applying them frame-by-frame or with additional guidance. Several methods enhance video generation through refined text prompts [24, 36], or by combining text and image modalities for editing [61]. Plug-and-Play Diffusion [53] and Free2Guide [24] further enable flexible, training-free editing. Dreamix [43] and Tune-A-Video [55] demonstrate the use of video diffusion models and spatio-temporal tuning for improved consistency and style transfer. Real-time editing approaches such as StreamDiffusion and StreamV2V [27, 33] enable efficient video editing.

A key limitation of these approaches lies in their difficulty to maintain temporal coherence and object consistency across frames. Methods like TokenFlow [14], Rerender [58], and VideoP2P [34] address this by identifying key frames [5] and propagating features across frames. Other approaches, such as Ground-A-Video [21], FateZero [46], and Ada-VE [37], rectify cross-frame attention or integrate motion cues to improve consistency. Despite these advances, cross-frame attention remains computationally expensive and does not provide formal guarantees of consistency [14].

2.2. Neuro-Symbolic Verification

Neuro-symbolic methods aim at integrating the advancement of neural networks with the rigor of symbolic reasoning [9, 16]. Neuro-symbolic methods use symbolic reasoning to provide formal guarantees in various domains. Specifically, in image and video synthesis, formal verification approaches

such as SMT [11] and temporal logic [7, 50] can rigorously validate the consistency and semantic correctness of the generated content.

Recently, neuro-symbolic verification has been be explored for video searching, editing, and evaluation tasks [7, 50]. Video classification employs graph-based relational modeling [13, 52], while event detection leverages spatiotemporal pattern recognition in video streams [31, 40, 57]. Neuro-symbolic frameworks enhance video question-answering [6, 59], with applications extending to robotic action planning [15, 28, 51] and safety verification in autonomous driving systems [22, 41]. These methods either construct graph structures [39, 56, 60], use latent-space representations as symbolic representations [3, 29, 49], or use formal language methods [2] to design specifications.

In contrast with this prior work, our ObjectAlign uniquely combines learnable perceptual metrics with symbolic constraints using the SMT solving, thus providing formal guarantees for object consistency in video editing. Additionally, we complement our verification with adaptive interpolation to repair inconsistencies dynamically. ObjectAlign is the first work to explore object consistency correction by post-processing inconsistent frames identified by learnable perceptual metrics or formal verification.

3. Preliminaries

3.1. Latent Diffusion Models

Diffusion models [12, 17] are generative models comprised of two main stochastic phases: (a) a *forward process* that progressively adds noise to data, and (b) a *reverse process* that learns to remove this noise to generate data.

The **forward process** is typically formulated as a fixed Markov chain that gradually introduces Gaussian noise to an initial data sample $\mathbf{z}_0$ over T discrete time steps. If $\mathbf{z}_0$ is a sample from the true data distribution p_{data} (e.g., a clean image), this process yields a sequence of increasingly noisy samples $\mathbf{z}_1, \ldots, \mathbf{z}_T$. The final sample $\mathbf{z}_T$ is ideally distributed according to a normal distribution, $\mathcal{N}(0, \mathbf{I})$, where $\mathbf{I}$ is the identity covariance matrix. The transition at each step t is defined as:

$$q(\mathbf{z}_t \mid \mathbf{z}_{t-1}) = \mathcal{N}\big(\mathbf{z}_t;\, \sqrt{\alpha_t}\,\mathbf{z}_{t-1},\, (1-\alpha_t)\,\mathbf{I}\big), \qquad (1)$$

where $\mathbf{z}_{t-1}$ is the data sample at the previous time step, $\mathbf{z}_t$ is the sample at the current time step, and α_t is a parameter derived from a predefined noise schedule (e.g., $\alpha_t = 1 - \beta_t \in (0,1)$, where $\beta_t \in (0,1)$ are small positive constants representing variance schedules).

The **reverse process** aims to reverse this noising procedure. It starts with a sample $\mathbf{z}_T \sim \mathcal{N}(0, \mathbf{I})$ and iteratively denoises it to produce a sample $\mathbf{z}_0$ that resembles data from the true distribution p_{data}. This process is also a Markov chain, parameterized by a neural network with parameters θ. The model is trained to predict the conditional probability distribution $p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$ for each step t:

$$p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t) = \mathcal{N}\big(\mathbf{z}_{t-1};\, \boldsymbol{\mu}_\theta(\mathbf{z}_t, t),\, \sigma_t^2\,\mathbf{I}\big), \qquad (2)$$

where $\boldsymbol{\mu}_\theta(\mathbf{z}_t, t)$ is the mean of the Gaussian distribution for $\mathbf{z}_{t-1}$, predicted by the neural network conditioned on the noisy sample $\mathbf{z}_t$ and the time step t. The term σ_t^2 represents the variance at time step t, which is often predefined or learned as part of the noise schedule. The neural network is trained to make $p_\theta(\mathbf{z}_{t-1} \mid \mathbf{z}_t)$ accurately approximate the true posterior $q(\mathbf{z}_{t-1} \mid \mathbf{z}_t, \mathbf{z}_0)$.

3.2. Metrics for Object Consistency Verification

To assess object consistency between video frames, ObjectAlign employs a combination of perceptual metrics and formal verification techniques. Let $f_i, f_j \in \mathcal{I}$ be two video frames, where $\mathcal{I}$ denotes the entire space of all video frames.

Perceptual Consistency Metrics. We utilize four established metrics to capture different aspects of visual and semantic similarity:

- **Learned Perceptual Image Patch Similarity (LPIPS):** This metric quantifies low-level perceptual similarity. The LPIPS distance between any two frames f_i, f_j is given by:

$$\text{LPIPS}(f_i, f_j) = \big\|\phi(f_i) - \phi(f_j)\big\|_2, \qquad (3)$$

  where $\phi\colon \mathcal{I} \to \mathbb{R}^d$ is a deep feature extractor [63] and $\|x\|_2$ denotes the Euclidean norm. Smaller LPIPS values indicate that the frames are more similar at a patch-perceptual level.
- **CLIP-based Semantic Similarity:** To measure high-level semantic alignment, we use the cosine similarity between image embeddings from a Contrastive Language-Image Pre-training (CLIP) model [47]:

$$\text{Sim}_{\text{CLIP}}(f_i, f_j) = \frac{\langle e(f_i),\, e(f_j)\rangle}{\|e(f_i)\| \cdot \|e(f_j)\|_2}, \qquad (4)$$

  where $e\colon \mathcal{I} \to \mathbb{R}^k$ is the CLIP image encoder, $\langle x, y\rangle$ denote the standard dot-product and $\|x\|_2$ denotes the Euclidean norm. Values closer to 1 signify stronger semantic correspondence between the frames.
- **Histogram Correlation:** To check for significant color shifts between frames, we compute the correlation between their color histograms. Let $h(f) \in \mathbb{R}^c$ be the flattened and normalized color histogram vector for frame f, where c represents the dimensionality. The histogram correlation is:

$$\text{Sim}_{\text{Hist}}(f_i, f_j) = \frac{h(f_i)^\top h(f_j)}{\|h(f_i)\|_2 \cdot \|h(f_j)\|_2}, \qquad (5)$$

  where $(\cdot)^\top$ denotes transpose. Values closer to 1 indicate higher similarity in the overall color distributions.

- **Mask IoU:** For object-level geometric consistency, we compute the IoU of foreground object masks. Let $M : \mathcal{I} \to \{0,1\}^{H\times W}$ be the binary foreground mask obtained for a frame (e.g., via the Segment Anything Model (SAM) [26]), where H and W are the frame height and width. The IoU is:

$$\mathrm{IoU}\big(M(f_i), M(f_j)\big) = \frac{|M(f_i) \cap M(f_j)|}{|M(f_i) \cup M(f_j)|}. \quad (6)$$

  This value, ranging from 0 to 1, quantifies the spatial overlap of the primary objects.

These perceptual metrics provide complementary empirical checks on object consistency, covering low-level appearance, high-level semantic content, color distribution, and object geometry, respectively. However, taken alone, they cannot inherently provide formal guarantees of coherence.

Formal Verification with SMT Solvers. To address the limitations of purely metric-based approaches and introduce rigorous consistency checks, ObjectAlign provides formal verification using SMT solvers. SMT solvers can determine the satisfiability of logical formulas with respect to background theories, enabling us to enforce provable bounds on specific features. In our context, we use an SMT solver to enforce *semantic stability* by asserting bounds on the object drift. We use object masks M_i to compute separate embeddings for the foreground object $e(f_i, M_i)$ and the background $e(f_i, \neg M_i)$. We then use an SMT solver to formally verify a *conjunctive* formula that ensures both **object identity stability** and **background stability**:

$$\begin{aligned}&\big(\forall j \,\big|\, e_j(f_i, M_i) - e_j(f_{i+1}, M_{i+1}) \,\big| \le \epsilon_s\big) \wedge \\ &\big(\forall j \,\big|\, e_j(f_i, \neg M_i) - e_j(f_{i+1}, \neg M_{i+1}) \,\big| \le \epsilon_{bg}\big)\end{aligned} \quad (7)$$

where ϵ_s and ϵ_{bg} are semantic drift tolerances. An SMT solver checks if this set of constraints is satisfiable; if it is, then we have a formal guarantee that no individual semantic feature dimension has drifted beyond the specified tolerances ϵ_s and ϵ_{bg}. ObjectAlign leverages this neuro-symbolic verification to complement the aforementioned learned perceptual metrics, providing a more robust and reliable consistency assessment than using the perceptual metrics alone.

4. Proposed Methodology

ObjectAlign consists of three stages executed in a closed verification–repair loop (Fig. 1): ① *metric–based scoring with learned thresholds*, ② *neuro-symbolic consistency verification*, and ③ *adaptive frame repair via neural interpolation*. The loop repeats until every neighbouring frame pair satisfies *all* consistency constraints.

4.1. Inconsistency identification (Step ① in Fig. 1)

4.1.1. Metric Based Consistency Scoring

Feature vector. For two consecutive frames f_i, f_{i+1}, we extract (a) cosine similarity of CLIP embeddings $S_{\cos}$, (b) color–histogram correlation S_{hist}, (c) mask–IoU S_{iou}, and (d) perceptual distance D_{lpips}. We invert LPIPS so that *larger* values denote higher consistency, i.e. $\tilde{S}_{\text{lpips}} = -D_{\text{lpips}}$. Hence the feature vector is $\mathbf{s}_i = \big[\, S_{\cos},\, S_{\text{hist}},\, S_{\text{iou}},\, \tilde{S}_{\text{lpips}}\big]^\top$. These specific metrics are chosen for their complementary strengths in assessing frame-to-frame object consistency: CLIP [47] similarity captures high-level semantic content alignment, LPIPS [63] evaluates low-level perceptual appearance, color histogram correlation checks for drastic color shifts, and mask IoU quantifies object-level geometric overlap and spatial stability, thereby providing a comprehensive empirical check as noted in Section 3.

Learnable thresholds. We treat each dimension of the feature vector $\mathbf{s}_i$ independently and learn a threshold vector $\boldsymbol{\tau} = \big[\tau_{\cos}, \tau_{\text{hist}}, \tau_{\text{iou}}, \tau_{\text{lpips}}\big]^\top$ from a small *positive* set $\mathcal{P}$ (adjacent frames from the unedited video) and a *negative* set $\mathcal{N}$ (pairs of original vs. edited frames, considered inconsistent). For each frame pair i, we compute the element-wise difference vector Δ_i between its feature vector $\mathbf{s}_i$ and the learned threshold vector $\boldsymbol{\tau}$: $\Delta_i = \mathbf{s}_i - \boldsymbol{\tau}$. The probability that a pair is consistent for a single threshold k ($P_k(i)$) is modeled by the sigmoid function:

$$P_k(i) \;=\; \sigma\big(\lambda\, \Delta_k\big), \quad \sigma(z) = \frac{1}{1+e^{-z}}, \quad (8)$$

where λ is a sharpness constant. The four thresholds in $\boldsymbol{\tau}$ are simultaneously optimized by minimizing the binary cross-entropy loss:

$$\begin{aligned}\mathcal{L}_{\text{BCE}} = -\frac{1}{|\mathcal{P}|+|\mathcal{N}|} \sum_{i\in\mathcal{P}\cup\mathcal{N}} \Big[& y_i \log(P_k(i)) \\ & + (1-y_i)\log(1-P_k(i))\Big]\end{aligned} \quad (9)$$

where $y_i = 1$ for $i \in \mathcal{P}$ and $y_i = 0$ for $i \in \mathcal{N}$. Optimization is performed using Adam [25].

4.1.2. Neuro-Symbolic Verification (Step ② in Fig. 1)

While the metric classifier is effective in practice for capturing perceptual inconsistencies, it offers no *formal* guarantee against all forms of object drift, particularly subtle semantic shifts that may fall within learned perceptual thresholds but still represent a logical inconsistency. The SMT-based verification step (see ② in Fig 1) addresses this by combining low-level feature stability with high-level temporal fidelity.

Given that the scalar perceptual metrics (e.g., S_{hist}) are directly evaluated against their learned thresholds (Eq. (12)), SMT verification is reserved for the high-dimensional CLIP

embeddings to enforce semantic stability. We therefore impose an SMT constraint on the **masked CLIP embeddings** (introduced in Sec. 3.2). Specifically, we verify the stability of both the foreground object $e(f, M)$ and the background $e(f, \neg M)$ independently, defining this semantic stability constraint as $\mathcal{C}_{\text{neuro}}$:

$$\begin{aligned}\mathcal{C}_{\text{neuro}} \equiv \big(\forall j \,\big|\, e_j(f_i, M_i) - e_j(f_{i+1}, M_{i+1}) \,\big| \leq \epsilon_s\big) \wedge \\ \big(\forall j \,\big|\, e_j(f_i, \neg M_i) - e_j(f_{i+1}, \neg M_{i+1}) \,\big| \leq \epsilon_{bg}\big)\end{aligned} \tag{10}$$

We complement this stability check with a high-level temporal fidelity metric [8, 50]. This component calculates a *satisfaction probability* by verifying the video's formal representation (automaton $\mathcal{A}_\nu$) against the text prompt's temporal logic specification (Φ) using a probabilistic model checker function, Ψ. A video is considered formally verified *only if* it satisfies both the low-level stability constraints ($\mathcal{C}_{\text{neuro}}$) and the high-level temporal requirements. We define this unified neuro-symbolic constraint, $\mathcal{P}_{\text{formal}}$, as the logical conjunction of these two conditions:

$$\mathcal{P}_{\text{formal}} \equiv \mathcal{C}_{\text{neuro}} \wedge (\Psi(\mathcal{A}_\nu, \Phi) \geq \tau) \tag{11}$$

Here, $\mathcal{P}_{\text{formal}}$ is satisfied if and only if the SMT solver finds the frame-to-frame drift constraints $\mathcal{C}_{\text{neuro}}$ (the first conjunct) satisfiable for all frames, *and* the probabilistic model checker finds that the temporal fidelity $\Psi(\mathcal{A}_\nu, \Phi)$ meets or exceeds a specified probability threshold τ (the second conjunct).

4.1.3. Joint Consistency Criterion

A transition is declared *consistent* ($C(i) = 1$) if and only if (iff) *all* thresholds are satisfied simultaneously *and* the formal constraints are met:

$$\begin{aligned}C(i) = &\Big(S_{\cos} \geq \tau_{\cos}\Big) \wedge \Big(S_{\text{hist}} \geq \tau_{\text{hist}}\Big) \wedge \\ &\Big(S_{\text{iou}} \geq \tau_{\text{iou}}\Big) \wedge \Big(D_{\text{lpips}} \leq \tau_{\text{lpips}}\Big) \wedge \\ &\Big(P_{\text{formal}}(i) = 1\Big)\end{aligned} \tag{12}$$

All indices with $C(i) = 0$ form the inconsistent set $\mathcal{I}$.

Formal Consistency Guarantees. The joint consistency criterion $C(i)$ defined in Eq. (12) combines the learned threshold checks with a formal SMT constraint. Our joint consistency criterion assures that a frame-pair declared consistent by our pipeline ($C(i) = 1$) exhibits a bounded drift according to every metric and the formal semantic check included in our criteria. These formal bounds underpin ObjectAlign's robustness in improving video consistency.

4.2. Adaptive Frame Repair (Step ③ in Fig. 1)

The adaptive frame repair stage (see ③ in Fig 1) fundamentally relies on the presence or eventual emergence of consistent anchor frames surrounding any block of identified inconsistencies. This principle can be conceptualized using Linear Temporal Logic (LTL) [38].

Algorithm 1 ObjectAlign verification–repair loop

Input: edited video $V = \{f_0, \ldots, f_{T-1}\}$ ▷ T is the total number of frames in video V.
Learn thresholds $\boldsymbol{\tau}$ on positive and negative set ▷ Sec. 4.1
repeat
  $\mathcal{I} \leftarrow \emptyset$
  for $i = 0$ **to** $T - 2$ **do** ▷ Iterate over all $T - 1$ frame transitions (f_i, f_{i+1})
    compute $P_{\text{metric}}(i)$, $P_{\text{formal}}(i)$
    if $C(i)$ (Eq. 12) is **false then**
      $\mathcal{I} \leftarrow \mathcal{I} \cup \{i\}$
  for each contiguous run $[i_s, i_e] \subseteq \mathcal{I}$ **do** ▷ i_s, i_e: start/end indices of a run of inconsistent transitions
    $k \leftarrow i_e - i_s + 1$ ▷ Number of frames $f_{i_s}, \ldots, f_{i_e}$ to repair
    replace $f_{i_s}, \ldots, f_{i_e}$ with corrected frames ▷ Sec. 4.2
until $\mathcal{I} = \emptyset$
return verified & corrected video V

Let AP_{IB} be an atomic proposition that is true when a contiguous block of frames is currently identified as an '*InconsistentBlock*' requiring repair. Let AP_{CAB} be true if a '*ConsistentAnchorBefore*' (i.e., a suitable frame f_{i_s-1}) exists or is established, and AP_{CAA} be true if a '*ConsistentAnchorAfter*' (i.e., a suitable frame f_{i_e+1}) exists or is established. The iterative verification-repair loop of ObjectAlign (Algorithm 1) operates under the premise that the video sequence will eventually satisfy the property:

$$\Box(AP_{IB} \implies (\Diamond AP_{CAB} \wedge \Diamond AP_{CAA})) \tag{13}$$

This LTL formula asserts that it is always ($\Box$) the case that if an inconsistent block requiring repair (AP_{IB}) exists, then eventually ($\Diamond$) a consistent anchor frame will be found or established before it (AP_{CAB}), and eventually ($\Diamond$) a consistent anchor frame will be found or established after it (AP_{CAA}), thus enabling interpolation. Our framework aims to progressively achieve this state, allowing for repair even when initial edits contain extended inconsistent segments.

Given a contiguous sequence of frames marked as inconsistent, we apply adaptive neural network based interpolation using RIFE [19]. Let $[i_s, i_e] \subseteq \mathcal{I}$ represent a maximal run of $k = i_e - i_s + 1$ inconsistent frames. To reconstruct these frames, we first identify the closest consistent frames immediately preceding and following this sequence: f_{i_s-1} and f_{i_e+1} (whose existence is anticipated by the property in Eq. (13)). We then dynamically select the interpolation depth (γ) as a function of the number of frames needing repair (k), defined by: $\gamma = \lceil \log_2(k + 1) \rceil$. This adaptive interpolation depth ensures that longer runs of inconsistencies are addressed with deeper interpolation, generating sufficient intermediate frames to preserve smooth and coherent motion.

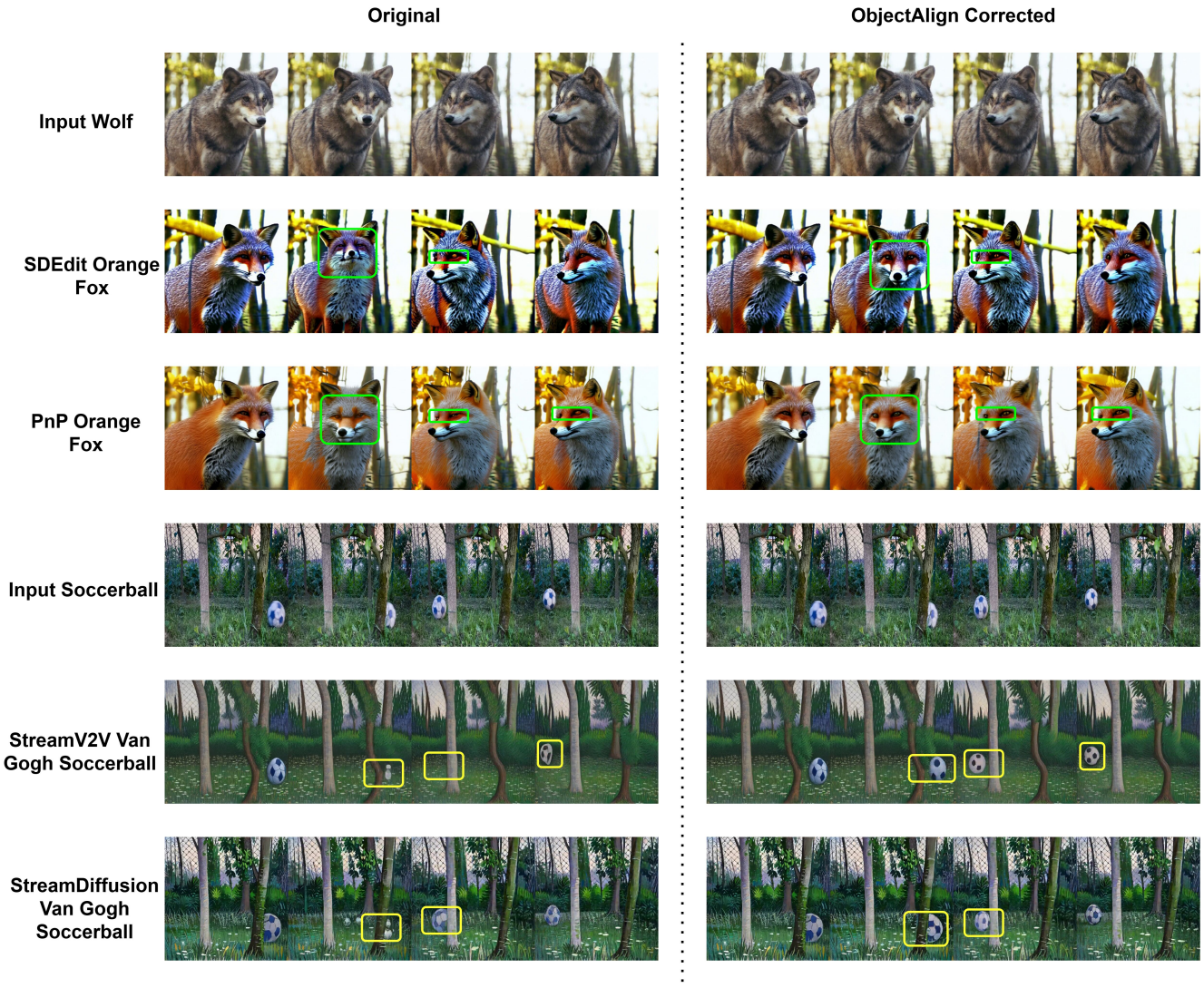

Figure 2. **Qualitative comparison of ObjectAlign corrections across different editing pipelines.** Before ObjectAlign correction (left), both SDEdit and PnP in the "Orange Fox" edits incorrectly alter the wolf's shape and color across consecutive frames (*highlighted in green boxes*). Similarly, StreamV2V and StreamDiffusion in the "Van Gogh Soccerball" edits cause the soccerball to intermittently disappear and reappear (*highlighted in yellow boxes*). These inconsistencies are accompanied by noticeable color and style drift, perceptual flickering, and identity misalignment. After applying ObjectAlign (right), these issues are effectively mitigated, resulting in greater semantic and temporal consistency.

The repaired frames replace the original inconsistent frames, producing an updated, more consistent video sequence.

The ObjectAlign pipeline re-runs Sections 4.1 to 4.2 on V until $\mathcal{I} = \emptyset$ as shown in Algorithm 1. ObjectAlign is able to provide both *empirical* quality and *formal* consistency guarantees through its neuro-symbolic verification pipeline.

5. Experimental Results

5.1. Experimental Setup

Our experiments utilize a dataset of 95 text-to-video prompt pairs obtained from the DAVIS [45] and Pexels [1] datasets. The prompts include both manually designed descriptions and those inspired from [33, 53]. The videos cover a diverse array of subjects including animals and humans. The prompts encompass various scenarios involving object edits, style changes, and structural modifications. Our primary baselines for comparison are PnP [53], SDEdit [42], StreamV2V [33], and StreamDiffusion [27] without our object consistency corrections. Evaluations focus on object-level consistency and semantic fidelity using established metrics such as CLIP [47] scores and Warp Error [30]. All experiments were performed on one Nvidia 3090 GPU.

Table 1. **ObjectAlign Improvements.** Comparison of video quality and consistency metrics *before* (Original) and *after* applying ObjectAlign (Edited) to videos processed by different base editing methods (PnP, SDEdit, StreamV2V, and StreamDiffusion). Scores are shown for different edit categories, with (+Improvement) indicating the improvement attributed to ObjectAlign. Higher scores are better for CLIP Score and VBench [20] motion smoothness, subject consistency, and background consistency metrics; lower is better for Warp Error.

Edit Category		PnP [53]		SDEdit [42]		StreamV2V [33]		StreamDiffusion [27]	
		Orig	Edited	Orig	Edited	Orig	Edited	Orig	Edited
Object Edits	CLIP Score ↑	97.0	98.1 (+1.1)	96.7	98.1 (+1.4)	97.1	97.9 (+0.8)	95.2	96.0 (+0.8)
	Warp Error ↓	107.4	101.3 (+6.1)	105.5	100.8 (+4.7)	100.5	98.8 (+1.7)	108.7	103.5 (+5.2)
	Motion Smoothness ↑	0.917	0.930 (+0.013)	0.903	0.925 (+0.022)	0.916	0.935 (+0.019)	0.887	0.901 (+0.014)
	Subject Consistency ↑	0.913	0.925 (+0.012)	0.900	0.915 (+0.015)	0.920	0.931 (+0.011)	0.884	0.899 (+0.015)
	Background Consistency ↑	0.921	0.925 (+0.004)	0.904	0.917 (+0.013)	0.917	0.924 (+0.007)	0.892	0.908 (+0.016)
Style Edits	CLIP Score ↑	97.6	98.2 (+0.6)	97.3	98.0 (+0.7)	97.5	97.9 (+0.4)	95.8	96.4 (+0.6)
	Warp Error ↓	106.3	101.6 (+4.7)	105.3	100.5 (+4.8)	99.5	98.6 (+0.9)	107.2	103.3 (+3.9)
	Motion Smoothness ↑	0.938	0.973 (+0.035)	0.930	0.958 (+0.028)	0.933	0.940 (+0.007)	0.920	0.936 (+0.016)
	Subject Consistency ↑	0.905	0.937 (+0.032)	0.904	0.925 (+0.021)	0.912	0.928 (+0.016)	0.896	0.908 (+0.012)
	Background Consistency ↑	0.913	0.932 (+0.019)	0.903	0.920 (+0.017)	0.915	0.928 (+0.013)	0.900	0.913 (+0.013)
Overall Average	CLIP Score ↑	97.3	98.2 (+0.9)	97.0	98.1 (+1.1)	97.3	97.9 (+0.6)	95.5	96.2 (+0.7)
	Warp Error ↓	106.9	101.5 (+5.4)	105.4	100.7 (+4.7)	100.0	98.7 (+1.3)	108.0	103.4 (+4.6)
	Motion Smoothness ↑	0.928	0.952 (+0.024)	0.917	0.942 (+0.025)	0.925	0.938 (+0.013)	0.904	0.919 (+0.015)
	Subject Consistency ↑	0.909	0.931 (+0.022)	0.902	0.920 (+0.018)	0.916	0.930 (+0.014)	0.890	0.904 (+0.014)
	Background Consistency ↑	0.917	0.929 (+0.012)	0.905	0.919 (+0.014)	0.916	0.926 (+0.010)	0.896	0.911 (+0.015)

5.2. Qualitative Results

Figure 2 presents visual comparisons between ObjectAlign and the baseline PnP [53] and SDEdit [42] methods on multiple challenging scenarios. We observe that ObjectAlign significantly reduces perceptual flicker, artifact generation, and object drift. In particular, ObjectAlign effectively maintains stable object identities across frames, producing results noticeably smoother and more temporally coherent than baseline methods.

5.3. Quantitative Results

CLIP Score. We measure semantic consistency using the CLIP similarity score [47], defined as the cosine similarity of CLIP embeddings between consecutive frames. Higher scores reflect greater semantic stability (↑). When ObjectAlign is applied to correct the outputs of various base editing methods, it consistently enhances semantic stability. For instance, drawing from the "Overall Average" results in Table 1, applying ObjectAlign to videos edited by PnP improves the CLIP Score from an original 97.3 to 98.2. For SDEdit, the score increases from 97.0 to 98.1; for StreamV2V [33], it improves from 97.3 to 97.9; and for StreamDiffusion [27], the score is enhanced from an original 95.5 to 96.2. This demonstrates that ObjectAlign effectively preserves or improves semantic content preservation across frames when applied to a range of editing techniques.

Warp Error. Temporal coherence is evaluated via Warp Error [30], which computes pixel-wise discrepancies after warping edited frames by the original video's optical flow. Lower Warp Error indicates greater temporal consistency (↓). When applied to various base editing methods, ObjectAlign consistently reduces their Warp Error, thereby enhancing temporal coherence. For instance, as detailed in Table 1 (Overall Average section), ObjectAlign improves the Warp Error for PnP from an original score of 106.9 down to 101.5. Similarly, for SDEdit, the error is reduced from 105.4 to 100.7; for StreamV2V, from 100.0 to 98.7; and for StreamDiffusion, from 108.0 to 103.4. These results confirm that our method produces more temporally consistent videos when used to correct the outputs of established editing techniques.

VBench Perceptual Metrics. To further assess video quality across diverse perceptual dimensions, we employ metrics from the VBench benchmark [20], specifically Motion Smoothness, Subject Consistency, and Background Consistency. For these metrics, higher scores are preferable (↑). As shown in Table 1, ObjectAlign consistently improves these scores when applied to the outputs of different editing methods across both object and style edit categories. For instance, when ObjectAlign is applied to videos edited using PnP, the Motion Smoothness score increases from 0.928 to **0.952**, and Subject Consistency improves from 0.909

to **0.931**. Similarly, for a baseline like StreamDiffusion, ObjectAlign enhances Motion Smoothness from 0.904 to **0.919** and Subject Consistency from 0.890 to **0.904**. These examples, representative of the broader findings in Table 1, indicate enhanced visual quality in terms of smoother motion, more stable subject appearance, and more coherent backgrounds. Further details on the VBench benchmark can be found in the original VBench documentation [20].

Runtime Efficiency. The runtime overhead introduced by ObjectAlign is minimal, requiring approximately 3% additional computation time compared to baseline PnP [53] editing, and 4% additional runtime compared to SDEdit [42]. The runtime overhead is primarily due to adaptive interpolation and SMT-based verification. The average processing time per frame remains acceptable for practical use, enabling ObjectAlign integration into existing image and video-editing workflows even in compute-limited edge environments. Overall, ObjectAlign achieves a superior balance between semantic consistency, temporal coherence, and computational efficiency compared to existing methods.

5.4. Ablation Studies

Ablation Study on Diffusion-Based Inpainting for Frame Repair. To evaluate the efficacy of our adaptive interpolation for frame repair (Section 4.2), we conducted an ablation study comparing it against an alternative approach using a pre-trained Stable Diffusion (SD) inpainting pipeline [48]. For this experiment, inconsistent frame outputs of PnP [53] were targeted for repair. Segmentation masks obtained via SAM [26] from a consistent reference frame guided the inpainting region, and textual prompts were provided. As shown in Table 2, the SD inpainting method yielded minimal beneficial impact on key metrics such as CLIP Score and Warp Error when applied to repair inconsistent PnP outputs, improving them by only 0.1 points. In contrast, ObjectAlign's interpolation demonstrates substantial improvements on CLIP Score [47] and warp error [30] for the same PnP outputs. These findings support our choice of targeted interpolation from consistent anchor frames for frame repair.

Table 2. **Ablation Study: Frame Repair Methods for PnP Outputs.** Comparison of ObjectAlign's RIFE-based interpolation against Stable Diffusion (SD) inpainting for repairing frames from the PnP baseline. "Original" refers to PnP output before repair. SD Inpainting refers to scores after the inpainting based repair method. Improvements (+Value) are relative to PnP (Original). The ablation study is performed over 18 edited video sequences.

Metric	Original	SD Inpainting	Interpolation (ObjectAlign)
CLIP Score ↑	97.0	97.1 (+0.1)	97.7 (+0.7)
Warp Error ↓	106.4	106.6 (-0.2)	100.3 (+6.1)

Ablation Study on Verification Checks. We ablate the impact of individual consistency verification checks—semantic similarity (CLIP cosine), perceptual similarity (LPIPS), histogram correlation, object-mask overlap (IoU), and the SMT-based semantic embedding constraint—on identifying inconsistent frames. This study is performed over 35 edited video sequences of lengths between 40 and 280 frames each. As detailed in Table 3, we observe that the IoU-based object-mask consistency check is most frequently triggered (22.3% of total frames), reflecting its sensitivity to spatial discrepancies in segmentation masks. The SMT-based embedding constraint triggers second most often (16.6%), underscoring the benefit of formal semantic bounds. The perceptual LPIPS check triggers third (15.5%), highlighting its effectiveness at detecting subtle visual artifacts. Histogram correlation and CLIP-based semantic similarity check flag inconsistencies less frequently (8.7% and 7.1%, respectively), indicating that color and global semantic shifts are comparatively rarer. Overall, the combined use of complementary verification checks ensures robust detection of diverse inconsistency types, each targeting different aspects of perceptual, spatial, and semantic coherence.

Table 3. **Ablation on individual consistency verification checks.** We report the percentage of frames flagged as inconsistent by each verification check over all sequences. Higher percentage indicates greater sensitivity of the check in detecting inconsistencies.

Verification Check	IoU	SMT	LPIPS	Histogram	CLIP Cosine
Percentage flagged (%)	22.3	16.6	15.5	8.7	7.1

6. Conclusion

In this paper, we have introduced ObjectAlign, a neuro-symbolic framework designed to detect, formally verify, and adaptively correct object-level inconsistencies in edited video sequences. Our approach integrates learnable perceptual metrics, neuro-symbolic verification, and adaptive neural network based interpolation to ensure semantic fidelity, temporal fidelity, and visual coherence.

Experimental evaluations demonstrate ObjectAlign's ability to substantially reduce semantic drift and visual artifacts, achieving superior performance in both perceptual consistency and temporal coherence, compared to existing baseline methods. Furthermore, ablation studies confirm the importance of each component in our design, highlighting the effectiveness of combining learnable consistency thresholds, symbolic reasoning, and adaptive interpolation. ObjectAlign thus represents an important step towards provably consistent and visually stable video editing.

References

[1] https://www.pexels.com. 2, 6

[2] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. The MIT Press, 2008. 3

[3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *International Conference on Machine Learning*, pages 813–824. PMLR, 2021. 3

[4] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2022. 2

[5] Duygu Ceylan, Chun-Hao Paul Huang, and Niloy Jyoti Mitra. Pix2video: Video editing using image diffusion. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23149–23160, 2023. 2

[6] Zhenfang Chen, Kexin Yi, Yunzhu Li, Mingyu Ding, Antonio Torralba, Joshua B Tenenbaum, and Chuang Gan. Comphy: Compositional physical reasoning of objects and events from videos. *arXiv preprint arXiv:2205.01089*, 2022. 3

[7] Minkyu Choi, Harsh Goel, Mohammad Omama, Yunhao Yang, Sahil Shah, and Sandeep Chinchali. Towards neuro-symbolic video understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Milan, Italy, 2024. Springer. 3

[8] Minkyu Choi, S P Sharan, Harsh Goel, Sahil Shah, and Sandeep Chinchali. We'll fix it in post: Improving text-to-video generation with neuro-symbolic feedback, 2025. 5

[9] Brandon C. Colelough and William Regli. Neuro-symbolic ai in 2024: A systematic review, 2025. 2

[10] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *arXiv preprint arXiv:2209.04747*, 2022. 1

[11] Leonardo De Moura and Nikolaj Bjørner. Satisfiability modulo theories: introduction and applications. *Commun. ACM*, 54(9):69–77, 2011. 1, 3

[12] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 3

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *IEEE/CVF International Conference on Computer Vision*, pages 6201–6210. IEEE, 2019. 3

[14] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023. 1, 2

[15] Mohammadhosein Hasanbeig, Yiannis Kantaros, Alessandro Abate, Daniel Kroening, George J Pappas, and Insup Lee. Reinforcement learning for temporal logic control synthesis with probabilistic satisfaction guarantees. In *2019 IEEE 58th conference on decision and control (CDC)*, pages 5338–5343. IEEE, 2019. 3

[16] Pascal Hitzler, Aaron Eberhart, Monireh Ebrahimi, Md Kamruzzaman Sarker, and Lu Zhou. Neuro-symbolic approaches in artificial intelligence. *National Science Review*, 9(6): nwac035, 2022. 2

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[18] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv:2204.03458*, 2022. 1

[19] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 5

[20] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 7, 8

[21] Hyeonho Jeong and Jong Chul Ye. Ground-a-video: Zero-shot grounded video editing using text-to-image diffusion models. *ArXiv*, abs/2310.01107, 2023. 2

[22] Susmit Jha, Vasumathi Raman, Dorsa Sadigh, and Sanjit A Seshia. Safe autonomy under perception uncertainty using chance-constrained temporal logic. *Journal of Automated Reasoning*, 60:43–62, 2018. 3

[23] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators, 2023. 2

[24] Jaemin Kim, Bryan S Kim, and Jong Chul Ye. Free^{2}guide: Gradient-free path integral control for enhancing text-to-video generation with large vision-language models, 2024. 2

[25] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[26] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023. 4, 8

[27] Akio Kodaira, Chenfeng Xu, Toshiki Hazama, Takanori Yoshimoto, Kohei Ohno, Shogo Mitsuhori, Soichi Sugano, Hanying Cho, Zhijian Liu, and Kurt Keutzer. Streamdiffusion: A pipeline-level solution for real-time interactive generation. *arXiv preprint arXiv:2312.12491*, 2023. 1, 2, 6, 7

[28] Hadas Kress-Gazit, Georgios E Fainekos, and George J Pappas. Temporal-logic-based reactive mission and motion planning. *IEEE transactions on robotics*, 25(6):1370–1381, 2009. 3

[29] Aliaksandr Kroshchanka, Vladimir Golovko, Egor Mikhno, Mikhail Kovalev, Vadim Zahariev, and Aleksandr Zagorskij. A neural-symbolic approach to computer vision. In *International Conference on Open Semantic Technologies for Intelligent Systems*, pages 282–309. Springer, 2021. 3

[30] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 170–185, 2018. 2, 6, 7, 8

[31] Nanjun Li, Faliang Chang, and Chunsheng Liu. Human-related anomalous event detection via spatial-temporal graph convolutional autoencoder with embedded long short-term memory network. *Neurocomputing*, 490:482–494, 2022. 3

[32] Feng Liang, Bichen Wu, Jialiang Wang, Licheng Yu, Kunpeng Li, Yinan Zhao, Ishan Misra, Jia-Bin Huang, Peizhao Zhang, Peter Vajda, et al. Flowvid: Taming imperfect optical flows for consistent video-to-video synthesis. *arXiv preprint arXiv:2312.17681*, 2023. 1

[33] Feng Liang, Akio Kodaira, Chenfeng Xu, Masayoshi Tomizuka, Kurt Keutzer, and Diana Marculescu. Looking backward: Streaming video-to-video translation with feature banks. *arXiv preprint arXiv:2405.15757*, 2024. 1, 2, 6, 7

[34] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8599–8608, 2024. 2

[35] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024. 1

[36] Yang Luo, Xuanlei Zhao, Mengzhao Chen, Kaipeng Zhang, Wenqi Shao, Kai Wang, Zhangyang Wang, and Yang You. Enhance-a-video: Better generated video for free, 2025. 2

[37] Tanvir Mahmud, Mustafa Munir, Radu Marculescu, and Diana Marculescu. Ada-ve: Training-free consistent video editing using adaptive motion prior. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 940–949. IEEE, 2025. 1, 2

[38] Zohar Manna and Amir Pnueli. *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer-Verlag, 1992. 5

[39] Effrosyni Mavroudi, Benjamín Béjar Haro, and René Vidal. Representation learning on visual-symbolic graphs for video understanding. In *European Conference on Computer Vision*, pages 71–90. Springer, 2020. 3

[40] Gérard G. Medioni, Isaac Cohen, François Brémond, Somboon Hongeng, and Ramakant Nevatia. Event detection and analysis from video streams. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(8):873–889, 2001. 3

[41] Noushin Mehdipour, Matthias Althoff, Radboud Duintjer Tebbens, and Calin Belta. Formal methods to comply with rules of the road in autonomous driving: State of the art and grand challenges. *Automatica*, 152:110692, 2023. 3

[42] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2022. 1, 2, 6, 7, 8

[43] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors, 2023. 2

[44] Mustafa Munir, Saloni Modi, Geffen Cooper, Huntae Kim, and Radu Marculescu. Three decades of low power: From watts to wisdom. *IEEE Access*, 12:19447–19458, 2024. 1

[45] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 2, 6

[46] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15886–15896, 2023. 2

[47] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 3, 4, 6, 7, 8

[48] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 8

[49] Soumalya Sarkar, Kin Gwn Lore, and Soumik Sarkar. Early detection of combustion instability by neural-symbolic analysis on hi-speed video. In *Proceedings of the NIPS Workshop on Cognitive Computation: Integrating Neural and Symbolic Approaches co-located with the 29th Annual Conference on Neural Information Processing Systems*, Montreal, Canada, 2015. CEUR-WS.org. 3

[50] SP Sharan, Minkyu Choi, Sahil Shah, Harsh Goel, Mohammad Omama, and Sandeep Chinchali. Neuro-symbolic evaluation of text-to-video models using formal verification. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 8395–8405, 2025. 3, 5

[51] Yasser Shoukry, Pierluigi Nuzzo, Ayca Balkan, Indranil Saha, Alberto L Sangiovanni-Vincentelli, Sanjit A Seshia, George J Pappas, and Paulo Tabuada. Linear temporal logic motion planning for teams of underactuated robots using satisfiability modulo convex programming. In *2017 IEEE 56th annual conference on decision and control (CDC)*, pages 1132–1137. IEEE, 2017. 3

[52] Du Tran, Heng Wang, Matt Feiszli, and Lorenzo Torresani. Video classification with channel-separated convolutional networks. In *IEEE/CVF International Conference on Computer Vision*, pages 5551–5560. IEEE, 2019. 3

[53] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 1, 2, 6, 7, 8

[54] Wen Wang, Yan Jiang, Kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models, 2024. 2

[55] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. *arXiv preprint arXiv:2212.11565*, 2022. 2

[56] Yu Xiong, Qingqiu Huang, Lingfeng Guo, Hang Zhou, Bolei Zhou, and Dahua Lin. A graph-based framework to bridge

movies and synopses. In *IEEE/CVF International Conference on Computer Vision*, pages 4591–4600. IEEE, 2019. 3

[57] Zhongwen Xu, Yi Yang, and Alexander G. Hauptmann. A discriminative CNN video representation for event detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1798–1807, Boston, MA, USA, 2015. IEEE Computer Society. 3

[58] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. *SIGGRAPH Asia 2023 Conference Papers*, 2023. 2

[59] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic VQA: disentangling reasoning from vision and language understanding. In *Advances in Neural Information Processing Systems*, pages 1039–1050, 2018. 3

[60] Dongran Yu, Bo Yang, Qianhao Wei, Anchen Li, and Shirui Pan. A probabilistic graphical model based on neural-symbolic reasoning for visual relationship detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10599–10608, New Orleans, LA, USA, 2022. IEEE. 3

[61] David Junhao Zhang, Dongxu Li, Hung Le, Mike Zheng Shou, Caiming Xiong, and Doyen Sahoo. Moonshot: Towards controllable video generation and editing with multimodal conditions. *arXiv preprint arXiv:2401.01827*, 2024. 2

[62] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023. 2

[63] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 3, 4