Original Research

Perceptual Metrics for Evaluating Spatial and Temporal Consistency in Edited Visual Media

Faisal Khan[1] and Imran Shah[2]

[1]Department of Computer Science and Engineering, University of Malakand, Chakdara Road, Lower Dir 18800, Khyber Pakhtunkhwa, Pakistan.
[2]Department of Computer Science and Engineering, Khwaja Fareed University of Engineering and Information Technology, Abu Dhabi Road, Rahim Yar Khan 64200, Punjab, Pakistan.

Abstract
Contemporary image and video editing systems enable diverse spatially localized and temporally extended modifications, including object insertion, background replacement, relighting, retiming, and generative synthesis. As these operations become more accessible and numerous, the assessment of visual quality requires metrics that explicitly account for spatial and temporal consistency rather than only global distortion with respect to a nominal reference. Spatial consistency concerns how edited regions integrate with surrounding content in terms of geometry, appearance, and semantics, whereas temporal consistency concerns how modifications evolve over time without causing flicker, motion discontinuities, or structural drift. Human observers judge these properties jointly, integrating local evidence across space and time under constraints of visual attention and memory, so computational metrics that ignore these interactions may correlate weakly with perceived plausibility. This text discusses the formulation of perceptual metrics designed to evaluate spatial and temporal consistency in edited visual media, emphasizing representations that combine low-level gradients, mid-level structures, and high-level learned features. It examines how spatiotemporal derivatives, graph-based regularity measures, and probabilistic models of human preference can be used to define differentiable objective functions suitable both for evaluation and for guiding optimization-based editing algorithms. It also considers numerical issues associated with large-scale video data, including sampling strategies, stability of gradient-based optimization, and computational trade-offs. Finally, it outlines experimental protocols for benchmarking such metrics against human judgments, with attention to the diversity of editing operations and viewing conditions, and identifies open questions in aligning metric predictions with human perception of edited visual media.

1. Introduction

Editing of visual media has shifted from manual frame-by-frame manipulation to workflows that heavily rely on algorithmic and learning-based components [1]. Modern pipelines support operations such as object removal, inpainting, compositing, style transfer, relighting, and temporally coherent generative synthesis across long sequences. These operations are increasingly applied not only in film production but also in user-generated content, where automatic or semi-automatic tools perform complex transformations that would otherwise be impractical. As a result, quantitative assessment of edited content needs to move beyond simple signal fidelity toward measures that reflect whether the edits produce spatially and temporally coherent imagery that remains subjectively plausible to human observers [2].

Traditional full-reference image and video quality metrics are typically defined as distances between an original signal and a processed version, often focusing on pixel-wise errors, frequency-domain differences, or local structural similarity. In edited media, however, the objective is rarely to reproduce an exact reference [3]. Instead, editing operations intentionally introduce new structures, remove content, or alter appearance while remaining consistent with the surrounding scene. Portions of the sequence may have no ground-truth counterpart, and even when a reference exists, the perceptual acceptability of deviations depends strongly on how they integrate spatially and temporally. A small local discrepancy can be unacceptable if it breaks global illumination or motion coherence, whereas a larger deviation may be tolerated if it remains consistent with scene semantics and temporal dynamics.

Spatial consistency in edited visual media can be informally described as the property that edited regions do not appear pasted, floating, or out of place relative to the underlying scene. This involves geometric compatibility, including perspective and scale; photometric compatibility, including color, shading, and contrast; and semantic compatibility, including object identity and interaction with context [4]. Temporal consistency refers to the continuity of these properties across time, such that motion, deformation, and appearance changes in edited regions align with the physical dynamics of the scene and with camera motion. Typical temporal artifacts include flickering textures, inconsistent shadows, temporal aliasing, or abrupt changes in object shape that violate expected motion trajectories or temporal integration in the visual system.

To evaluate these properties computationally, metrics need to encode spatiotemporal dependencies across multiple scales, often using representations that extend beyond raw pixel intensities. At the same time, metrics must be tractable for high-resolution and long-duration sequences and, in many applications, differentiable with respect to the edited content so that they can be used as objective functions during optimization or learning. These requirements motivate formulations based on linear operators, multiscale transforms, and learned feature spaces, combined with statistical models that connect metric values to subjective judgments of spatial and temporal coherence.

Perceptual metrics for edited media face several structural challenges [5]. First, edits may be localized to specific regions, making it necessary to weight errors differently in edited and unedited areas while still accounting for global context. Second, edited frames can exhibit long-range dependencies; for example, a relit object must remain consistent with distant cast shadows, or a synthesized background must respect the parallax implied by camera motion. Third, human observers may exhibit different sensitivities to inconsistencies depending on viewing conditions, task, and the presence of distractors, which complicates the mapping from metric scores to perceived quality. Addressing these challenges requires a combination of perceptual modeling, mathematical formulation, and empirical calibration.

The following sections discuss the perceptual foundations relevant to spatial and temporal consistency in edited media, propose mathematical formulations that represent edited sequences as spatiotemporal tensors equipped with differential and probabilistic structures, and describe learning-based approaches that fit parametric metrics to human judgments [6]. Numerical aspects, such as stability of gradient-based optimization and efficient sampling in space-time, are considered alongside experimental protocols for validating metrics. Throughout, emphasis is placed on the interplay between spatial and temporal aspects of perception and on how this interplay can be reflected in reproducible, differentiable metrics applicable to a broad spectrum of editing tasks.

2. Perceptual Foundations of Spatial and Temporal Consistency

Human perception of edited visual media is governed by mechanisms that jointly integrate spatial and temporal context. Spatial visual processing is organized hierarchically, with early stages responding to local edges, orientations, and textures, and later stages representing object-level structure and scene layout. Temporal processing integrates information over windows that depend on motion speed, contrast, and spatial frequency content [7]. Within these hierarchies, the perception of consistency is not limited to local agreement of pixel values but depends on relational properties such as continuity of contours, alignment of motion trajectories, and coherence of illumination across surfaces.

Spatial consistency in edited frames involves the alignment of several cues that observers use to infer three-dimensional structure and material properties. Geometric cues include perspective convergence, relative size, and occlusion relations, all of which constrain the plausible placement of synthesized or moved objects. Photometric cues include color balance, shading gradients, specular highlights, and cast shadows, which jointly inform the interpretation of surface reflectance and lighting. When edited

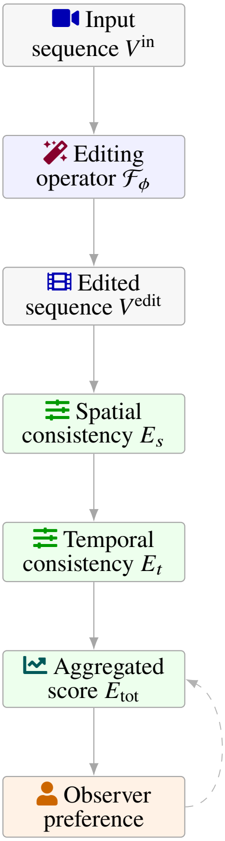
Input sequence V^{in}
Editing operator $\mathcal{F}_\phi$
Edited sequence V^{edit}
Spatial consistency E_s
Temporal consistency E_t
Aggregated score E_{tot}
Observer preference

Figure 1: Overview of a perceptual pipeline in which an editing operator transforms an input sequence into an edited sequence that is evaluated by spatial and temporal consistency modules, aggregated into a scalar score, and compared against human observer preferences.

content violates these constraints, such as by introducing an object with inconsistent shading direction or inconsistent blur relative to its depth, observers may interpret the region as pasted or artificial even if local pixel-level distortions are small.

Texture and noise statistics also contribute to spatial consistency [8]. Surfaces within a scene often exhibit characteristic spatial frequency spectra and correlation structures. Editing operations that replace or generate textures must preserve local anisotropy, orientation, and granularity patterns that are consistent with the surrounding region. Mismatches in noise level, such as a denoised foreground inserted into a noisy background, can be particularly noticeable because the visual system is sensitive to unnatural homogeneity or grain differences. Spatial consistency metrics therefore benefit from descriptors that capture such statistics at multiple scales, allowing a distinction between acceptable texture variation and disruptive inconsistency.

Temporal consistency is shaped by mechanisms of motion perception and temporal integration [9]. Observers tend to integrate information across short time intervals, forming expectations about the trajectories of objects and the evolution of illumination. When an object is edited across frames, any discontinuity in position, shape, shading, or occlusion that violates these expectations can produce perceptual artifacts such as jitter, flicker, or implausible motion. For example, if a synthesized object moves at a speed inconsistent with the background parallax implied by camera motion, or if shadows and reflections fail to update coherently over time, the resulting inconsistency can be detected even when single frames appear visually plausible.

Editing operation	Spatial objective	Temporal objective	Typical artifacts
Object insertion / compositing	Geometric and photometric alignment with scene layout	Consistent motion and occlusion across frames	Floating objects, misaligned scale, inconsistent shadows or blur, popping edges
Background replacement / inpainting	Fill holes plausibly with coherent textures and structures	Stable background evolution under camera motion	Texture seams, perspective errors, parallax violations, temporal flicker in fill regions
Relighting / recoloring	Consistent shading, color balance, and contrast across surfaces	Smooth evolution of lighting with motion and scene dynamics	Shadow direction changes, inconsistent specular highlights, temporally unstable white balance
Retiming / motion editing	Maintain plausible geometry in interpolated or removed frames	Smooth trajectories and physically plausible timing	Motion jitter, temporal aliasing, ghosting, duplicated or missing limbs
Generative synthesis (video diffusion, GANs)	Globally coherent layout and semantics in each frame	Long-range temporal coherence without structural drift	Structurally drifting objects, identity switches, texture crawling, unstable global illumination

Table 1: Representative editing operations and associated spatial and temporal challenges in edited visual media.

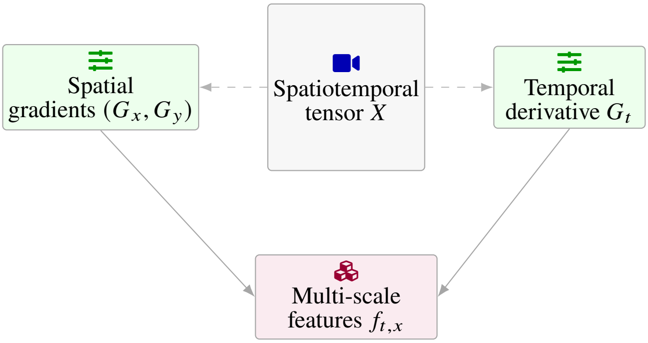
Spatial gradients (G_x, G_y)
Spatiotemporal tensor X
Temporal derivative G_t
Multi-scale features $f_{t,x}$

Figure 2: Representation of an edited video as a spatiotemporal tensor X with linear spatial and temporal derivative operators generating gradients that are pooled into multi-scale feature vectors used to quantify local spatial and temporal consistency.

Temporal artifacts in edited media interact with spatial structure. Flicker in high-frequency texture may be more tolerable in peripheral regions or in areas of high motion, whereas flicker in a static central object can be highly salient [10]. The visual system exhibits temporal contrast sensitivity that depends on spatial frequency, leading to differential detectability of temporal modulation across scales. This implies that temporal consistency metrics should weight deviations differently across spatial frequencies and locations and that purely frame-wise spatial metrics are insufficient to capture perceived quality in dynamic content.

Another perceptual consideration is the role of attention and task. Observers may be more sensitive to inconsistencies in semantically important or attended regions, such as faces, text, or user-specified

Spatial aspect	Perceptual cues	Example inconsistency
Geometry	Perspective convergence, relative size, occlusion order, depth-of-field blur	Inserted object has incorrect scale or vanishing point, or violates foreground/background ordering
Photometry / illumination	Color balance, shading gradients, cast shadows, specular highlights, exposure	Shadow direction mismatch, too sharp or too soft shadows, inconsistent noise or exposure
Semantics and context	Object identity, affordances, interactions, scene category	Implausible object placement (e.g., car on wall), mismatched style relative to environment
Texture and noise statistics	Spatial frequency spectra, anisotropy, grain level, local correlations	Over-smoothed foreground in noisy background, repetitive or aliased synthesized textures
Salience and region importance	Faces, text, central objects, user-selected regions of interest	Minor spatial errors in highly salient areas dominating perceived quality

Table 2: Key components of spatial consistency and typical ways in which edited regions can violate them.

Temporal factor	Description	Typical artifact	Perceptual considerations
Motion alignment	Agreement of object motion with camera motion and scene geometry	Jitter, sliding objects, inconsistent parallax	High sensitivity for salient objects and rigid structures
Illumination evolution	Time-varying shadows, reflections, and shading consistent with dynamics	Shadows that lag or lead motion, popping highlights	Violations visible even when single frames look plausible
Shape and structure coherence	Smooth evolution of object contours and topology	Abrupt shape changes, limb popping, structural drift	More noticeable for familiar objects (faces, bodies)
Texture and noise stability	Temporal behavior of high-frequency content and grain	Texture flicker, temporal aliasing, unstable denoising	Sensitivity depends on spatial frequency and retinal eccentricity
Temporal masking and integration	Pooling of information over temporal windows	Overly local temporal metrics missing long-range drifts	Perceptual impact depends on motion speed and task

Table 3: Temporal consistency factors that influence perceived plausibility of edited video sequences.

regions of interest. Editing tools often target precisely these regions, making perceptual sensitivity particularly high. Moreover, when observers expect manipulation, as in certain synthetic or augmented reality scenarios, they may adopt different criteria than when viewing content presumed to be unedited [11]. Metrics that aim to predict subjective judgments must therefore consider how salience maps, semantic segmentation, and task-specific viewing strategies influence the effective weighting of spatial and temporal inconsistencies.

Finally, perceptual thresholds for detecting inconsistencies can be modeled as just-noticeable-difference regions in a high-dimensional space of spatiotemporal distortions. In this view, an edited

Component	Notation / operator	Role in metric formulation
Video tensor representation	V^{edit}, V^{ref}, matrix $X \in \mathbb{R}^{3\|\Omega\| \times T}$	Encodes frames as a spatiotemporal tensor or matrix for linear-algebraic manipulation
Spatial and temporal gradients	G_x, G_y, G_t, discrete ∇ operators	Measure local changes in space and time, form the basis of derivative-based energies
Feature descriptors	$f_{t,x} \in \mathbb{R}^d$, feature matrices F_t	Capture low-, mid-, and high-level structure beyond raw pixels
Motion fields	$u_t(x)$, motion-compensated sampling locations	Relate corresponding points across frames for temporal consistency measures
Gram matrices and covariances	$G_t = F_t F_t^\top$	Encode global texture and correlation statistics for style and texture consistency
Graph structures and Laplacian	Graph $G = (\mathcal{V}, \mathcal{E})$, Laplacian L	Model smoothness and propagation of inconsistency over space-time
Continuous space-time view	$V(x, y, t)$ with $\partial_x V, \partial_y V, \partial_t V$	Links discrete metrics to continuous differential formulations and physical models

Table 4: Mathematical building blocks used to represent edited sequences and define consistency metrics.

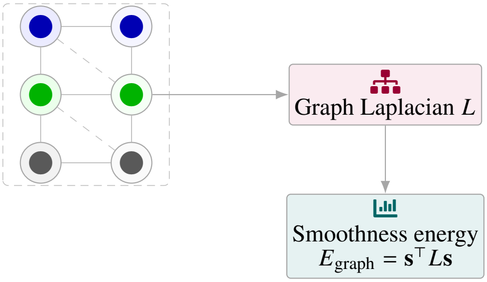
Graph Laplacian L
Smoothness energy $E_{\text{graph}} = \mathbf{s}^\top L\mathbf{s}$

Figure 3: Graph-based formulation in which spatiotemporal samples are nodes of a locally connected grid, and a weighted graph Laplacian transforms a scalar inconsistency field into a smoothness energy that penalizes spatial and temporal irregularities in edited regions.

sequence is acceptable if the perturbation it introduces lies within a perceptual tolerance region relative to the unedited sequence, taking into account masking effects from motion, texture, and luminance variation. This perspective naturally connects to probabilistic models in which metric outputs are interpreted as likelihoods or confidence scores for perceptual equivalence. It also highlights the importance of nonlinear pooling across space and time, because the detection of inconsistencies often depends on the maximum or local concentration of artifacts rather than on simple global averages [12].

Energy term	Definition sketch	Targeted aspect	Remarks
Spatial neighborhood energy $E_s(t)$	Pairwise penalties $\psi(\|f_{t,x} - f_{t,y}\|_2)$ over $(x,y) \in \mathcal{E}_s$	Local spatial smoothness and boundary transitions	Weights $w_{x,y}$ can emphasize edited vs. unedited boundaries
Temporal consistency energy $E_t(t)$	Motion-compensated differences $\phi(\|f_{t+1,x+u_t(x)} - f_{t,x}\|_2)$	Inter-frame coherence under motion	Requires reliable motion or robust penalties for flow errors
Sequence-averaged terms $\bar{E}_s, \bar{E}_t$	Temporal averages over t	Global spatial and temporal consistency	Allow adaptive weighting by content or motion statistics
Combined metric E_{tot}	$E_{\text{tot}} = \alpha\bar{E}_s + \beta\bar{E}_t$	Joint space-time measure	Coefficients α, β can be tuned or learned from data
Gram-based energy E_{gram}	Frobenius norm $\|G_t^{\text{edit}} - G_t^{\text{ref}}\|_F^2$	Global texture/style consistency	Can also be extended across time for temporal style stability
Graph smoothness energy E_{graph}	Quadratic form $\mathbf{s}^\top L\mathbf{s}$ over inconsistency field **s**	Propagation of local inconsistencies in space-time	Connects to spectral analysis and regularization in optimization

Table 5: Examples of energy terms used to quantify spatial and temporal consistency in edited video.

Design choice	Typical options	Influence on metric behavior
Input representation	Raw RGB frames, gradients, motion-compensated stacks, pretrained features	Controls invariance to benign variations and sensitivity to structural inconsistencies
Network architecture	2D CNNs with frame pooling, 3D CNNs, transformers over space-time tokens	Determines receptive field and ability to capture long-range temporal dependencies
Supervision signal	Pairwise preferences, ratings, mixed objectives with auxiliary tasks	Aligns metric scale with human judgments and task-specific priorities
Pooling strategy	Average pooling, max or power pooling, attention-weighted aggregation	Shapes how local inconsistency maps are summarized into a scalar score
Multiscale processing	Spatial and temporal pyramids, multi-resolution branches	Enables detection of both fine-grained artifacts and global coherence issues
Differentiability and integration	Fully differentiable pipeline, differentiable motion modules, surrogate terms	Allows gradients to flow into editing models for optimization-based workflows

Table 6: Design choices for learning-based perceptual metrics and their qualitative impact.

3. Mathematical Formulation of Consistency Metrics

A convenient starting point for formalizing perceptual metrics is to represent a video sequence as a discrete spatiotemporal tensor. Let the spatial domain be a finite grid

$$\Omega \subset \mathbb{Z}^2$$

Numerical issue	Underlying cause	Mitigation strategy
Unstable gradients	Highly nonlinear penalties, sharp activations, unnormalized features	Use smooth robust losses, normalization layers, and gradient clipping
Poor conditioning of loss landscape	Strong anisotropy of operators (e.g., $G^\top G$), unbalanced scales	Preconditioning, feature normalization, residual formulations, multi-term balancing
Interpolation artifacts in temporal terms	Non-differentiable or coarse sampling at $x + u_t(x)$	Use bilinear/bicubic interpolation, antialiasing filters, and flow regularization
High computational cost on long videos	Dense evaluation over all pixels and frames	Importance sampling in space-time, salience-guided subsampling, patch-based evaluation
Sensitivity to minor perturbations	Large Lipschitz constant or poorly regularized network weights	Architectural constraints (e.g., spectral normalization) and explicit robustness regularizers
Differentiation through iterative editors	Backpropagation through long unrolled optimization or recurrent loops	Truncated backpropagation, implicit differentiation, or staged surrogate losses

Table 7: Numerical and optimization challenges when using perceptual metrics as objective functions for editing.

with cardinality $|\Omega| = HW$, and let the temporal index set be

$$\mathcal{T} = \{1, \dots, T\}.$$

An edited sequence is denoted

$$V^{\text{edit}} = \left\{[13] I_t^{\text{edit}}\right\}_{t\in\mathcal{T}},$$

where each frame

$$I_t^{\text{edit}} : \Omega \to \mathbb{R}^3$$

assigns a color vector to each pixel. An optional reference sequence

$$V^{\text{ref}} = \left\{I_t^{\text{ref}}\right\}_{t\in\mathcal{T}}$$

may be available in cases where the edit is derived from original footage. A binary mask [14]

$$M_t : \Omega \to \{0, 1\}$$

can indicate edited regions, with $M_t(x) = 1$ for pixels directly affected by editing operations.

Pixels and frames can be vectorized for linear-algebraic manipulation. For each frame t, the vectorized image

$$\mathbf{i}_t \in \mathbb{R}^{3|\Omega|}$$

is formed by stacking color channels and pixels in a fixed order. The sequence can then be regarded as a matrix [15]

$$X \in \mathbb{R}^{3|\Omega|\times T},$$

whose columns correspond to frames. This representation facilitates the use of linear operators that act along spatial or temporal dimensions, such as finite-difference gradients, multiscale transforms, or low-rank projections. For example, a spatial gradient operator in the horizontal direction can be represented

Stage	Key decisions	Example choices	Notes
Source dataset design	Scene diversity, motion types, resolution, duration	Indoor/outdoor, static vs. dynamic, natural vs. man-made scenes	Should reflect target application domains and viewing scenarios
Editing operation coverage	Types of edits and artifact severity levels	Compositing, inpainting, style transfer, relighting, retiming, generative edits	Include both controlled synthetic artifacts and realistic pipeline failures
Subjective testing protocol	Comparison vs. rating, presentation setup, quality control	Pairwise A/B tests with replay, calibrated displays, crowd-sourcing filters	Design influences reliability and variance of collected judgments
Statistical analysis	Performance metrics and uncertainty estimates	Rank and linear correlations, preference prediction accuracy, bootstrap CIs	Separate training, validation, and test content to avoid overfitting
Scale calibration	Mapping raw scores to perceptual scales or acceptance thresholds	Nonlinear regression to mean opinion scores, JND-based margins	Enables interpretable thresholds for deployment decisions
Generalization and stress tests	Cross-dataset and out-of-distribution evaluation	Testing on new content, extreme edits, compression variants	Reveals failure modes and robustness of the metric

Table 8: Elements of evaluation protocols for assessing perceptual metrics against human judgments.

as a matrix

$$G_x \in \mathbb{R}^{3|\Omega|\times 3|\Omega|},$$

so that

$$G_x \mathbf{i}_t$$

approximates the discrete derivative of frame t along the x-axis [16]. Analogous operators

$$G_y \quad \text{and} \quad G_t$$

can be defined for vertical and temporal derivatives.

Spatial consistency metrics can be expressed as functionals of spatial derivatives and local feature statistics. For each frame t and pixel $x \in \Omega$, let $f_{t,x} \in \mathbb{R}^d$ denote a feature vector derived from local neighborhoods of I_t^{edit}. The feature extractor can encompass linear filters, multiscale transforms, or learned convolutional embeddings. A generic spatial consistency energy for frame t can be written as

$$E_s(t) = [17] \frac{1}{|\mathcal{E}_s|} \sum_{(x,y)\in\mathcal{E}_s} w_{x,y}\, \psi\big(\|f_{t,x} - f_{t,y}\|_2\big),$$

where $\mathcal{E}_s$ is a set of spatial neighbor pairs that encode adjacency relations, $w_{x,y}$ are spatial weights that may depend on distance or semantic boundaries, and ψ is a nonnegative function that quantifies inconsistency between neighboring features. When features and weights are chosen to emphasize boundaries between edited and unedited regions, $E_s(t)$ captures how smoothly edited content transitions into its context.

Temporal consistency can be modeled by relating features across successive frames while compensating for motion. Let $u_t(x) \in \mathbb{R}^2$ denote a motion field mapping pixel x in frame t to a corresponding

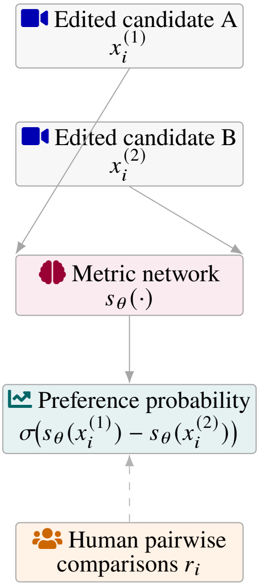

Figure 4: Learning-based perceptual metric driven by pairwise comparisons: two edited sequences are scored by a shared spatiotemporal network whose outputs are mapped to a preference probability and fitted to human comparison labels.

location in frame $t+1$. For each pixel x, define the motion-compensated feature discrepancy [18]

$$d_t(x) = \left\| f_{t+1,\, x+u_t(x)} - f_{t,x} \right\|_2.$$

A temporal consistency energy for the pair of frames t and $t+1$ is then

$$E_t(t) = [19] \frac{1}{|\Omega|} \sum_{x\in\Omega} \phi\big(d_t(x)\big),$$

where ϕ is a penalty that may be robust to small deviations but grows for larger inconsistencies. Aggregating over time yields a sequence-level temporal consistency measure

$$\bar{E}_t = \frac{1}{T-1} \sum_{t=1}^{T-1} E_t(t) [20].$$

Spatial and temporal consistency are not independent, and a metric that aims to reflect perception often combines them. A simple linear combination of averaged energies can be defined as

$$E_{\text{tot}} = \alpha\, \bar{E}_s + \beta\, \bar{E}_t,$$

where

$$\bar{E}_s = \frac{1}{T} \sum_{t=1}^{T} E_s(t),$$

and $\alpha, \beta \ge 0$ control the relative weighting of spatial and temporal contributions. In practice, these weights may depend on motion magnitude, scene content, or viewing conditions [21]. For example, temporal consistency might be given more weight in highly dynamic scenes where motion dominates perception, while spatial consistency may be more important in nearly static footage.

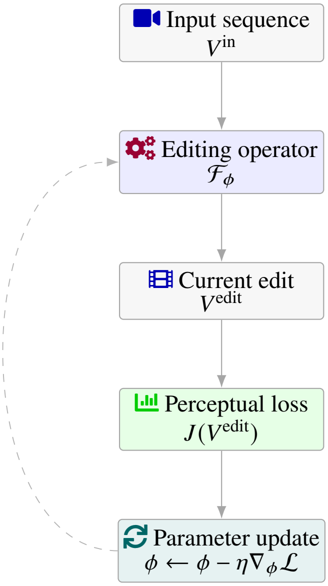

Figure 5: Integration of a differentiable perceptual consistency metric into an optimization loop, where the editing operator parameters are updated using gradients of a loss that includes spatial and temporal consistency terms.

Beyond local pairwise interactions, global consistency can be analyzed using linear algebraic constructs such as covariance matrices and Gram matrices of features. Let $F_t \in \mathbb{R}^{d \times N_t}$ be a matrix whose columns are feature vectors $f_{t,x}$ for pixels or patches indexed by x, where N_t is the number of sampled locations in frame t. The spatial Gram matrix

$$G_t = F_t F_t^\top$$

encodes inner products between feature channels and captures global texture and correlation statistics [22]. Differences between Gram matrices of edited and reference frames, or between edited frames over time, can serve as measures of style and texture consistency. For instance, one can define

$$E_{\text{gram}} = \frac{1}{T} \sum_{t=1}^{T} \left\| G_t^{\text{edit}} - G_t^{\text{ref}} \right\|_F^2,$$

where $\|\cdot\|_F$ denotes the Frobenius norm. Variants that compare Gram matrices across time capture whether global texture statistics evolve smoothly or exhibit unnatural temporal fluctuations [23].

Graph-based formulations provide a bridge to discrete mathematics and spectral analysis. Consider a graph $G = (\mathcal{V}, \mathcal{E})$ whose vertices represent spatiotemporal locations labeled by (x, t) and whose edges connect spatial or temporal neighbors. Let $s_{x,t} \in \mathbb{R}$ denote a scalar field measuring local inconsistency, such as the magnitude of a feature difference. A smoothness energy on the graph can be written as

$$E_{\text{graph}} = \frac{1}{2} \sum_{((x,t),(y,s)) \in \mathcal{E}} w_{(x,t),(y,s)} \left(s_{x,t} - s_{y,s} \right)^2,$$

where $w_{(x,t),(y,s)}$ are edge weights that may reflect spatial distance, temporal separation, or feature similarity. This energy can be expressed as a quadratic form [24]

$$E_{\text{graph}} = \mathbf{s}^\top L \mathbf{s},$$

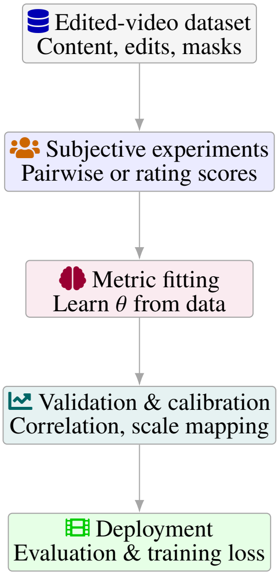

Figure 6: Experimental protocol for evaluating and deploying perceptual consistency metrics, from curated datasets and subjective experiments through metric training, validation and calibration, to use as both an evaluation tool and an optimization objective in editing systems.

where $\mathbf{s}$ stacks all $s_{x,t}$ into a vector and L is the graph Laplacian. Spectral properties of L characterize how inconsistency signals propagate over space and time, which is relevant when metrics are used as regularizers in optimization.

Differential operators defined on continuous space-time provide an alternative perspective that connects to tensor calculus. Viewing the video as a function

$$V : \Omega_c \times [0, T_c] \to \mathbb{R}^3,$$

with continuous spatial domain Ω_c and continuous time interval $[0, T_c]$, one can consider the spatiotemporal gradient tensor [25]

$$\nabla V = \big(\partial_x V,\ \partial_y V,\ \partial_t V\big), [26]$$

and define consistency energies that penalize deviations from smoothness or from an underlying physical model. For instance, in regions expected to follow brightness constancy along motion trajectories, temporal derivatives along the flow field can be used to measure inconsistency. Discrete metrics arise by sampling and approximating these derivatives with finite differences, linking continuous and discrete formulations.

4. Learning-Based Perceptual Metrics

While hand-designed functionals of derivatives, features, and graph structures provide interpretable metrics, their parameters must be chosen to align with human judgments. Learning-based perceptual metrics adopt a parametric mapping from edited sequences to scalar scores and fit this mapping using subjective data [27]. Let x denote an edited video, represented by its tensor of pixel values or by derived features, and let $s_\theta(x) \in \mathbb{R}$ be a metric parameterized by θ. The goal is to learn θ such that the ordering induced by s_θ correlates with perceived spatial and temporal consistency for a variety of edits.

Human annotations are often collected in the form of pairwise comparisons. Each sample consists of a pair $(x_i^{(1)}, x_i^{(2)})$ of edited sequences and a binary label $r_i \in \{0, 1\}$ indicating which sequence is judged more consistent by observers. A probabilistic model can be used to connect metric scores to comparison outcomes. A common choice is a logistic model,

$$\mathbb{P}(r_i = 1) = \sigma\big([28] s_\theta(x_i^{(1)}) - s_\theta(x_i^{(2)})\big),$$

where

$$\sigma(z) = \frac{1}{1 + \mathrm{e}^{-z}}$$

is the sigmoid function. The parameters θ are estimated by minimizing the negative log-likelihood

$$L(\theta) = -\sum_{i=1}^{N} \log \mathbb{P}_\theta(r_i),$$

possibly with regularization terms to control complexity [29]. This formulation encourages the learned metric to assign higher scores to sequences that are more often judged consistent.

The mapping s_θ is typically implemented by a neural network that processes spatiotemporal data. For spatial consistency, networks operating on individual frames or on spatial patches can be used, with temporal context provided implicitly via aggregation of scores across frames. For temporal consistency, architectures that explicitly process sequences, such as three-dimensional convolutions or attention mechanisms over space-time tokens, can model long-range temporal dependencies. The input to the network may consist of raw frames, derivatives, motion-compensated differences, or higher-level features extracted from networks trained for recognition tasks [30]. The choice of representation influences the ability of the metric to generalize across content and editing operations.

Multiscale processing is particularly important when learning perceptual metrics for edited media. Spatial pyramids and temporal pyramids allow the network to compute descriptors at different resolutions and frame rates, capturing both fine-grained artifacts such as edge halos and coarse artifacts such as inconsistent global illumination or motion. A metric can internally produce intermediate representations that approximate human sensitivity curves across spatial and temporal frequencies, with learnable pooling strategies that emphasize salient inconsistencies. For example, the network may be trained to compute local inconsistency maps and then pool them using nonlinear operations such as maxima, powered means, or attention-weighted averages to produce a single scalar score [31].

From a probabilistic standpoint, a learned metric defines a latent perceptual scale on which edited sequences are embedded. Under the logistic comparison model, differences in scores approximate log-odds of preference for one sequence over another in terms of spatial and temporal coherence. This interpretation suggests connections to psychometric functions and just-noticeable-difference thresholds. By analyzing the distribution of score differences for pairs of sequences that observers judge as indistinguishable, one can estimate a margin around zero within which changes in the metric are not perceptually significant. This margin can be used to assess the reliability of metric differences when comparing editing methods.

Training data for learning-based metrics need to cover a range of editing operations and conditions [32]. Sequences may vary in resolution, duration, camera motion, and scene content, while edits may include compositing, inpainting, stylization, retiming, and generative synthesis. The metric must learn to focus on inconsistencies that are perceptually important while ignoring benign variations such as small style changes that do not disrupt coherence. To reduce dataset bias, training protocols can balance examples across content types and editing categories, and can include both extreme and subtle artifacts. Data augmentation techniques, such as synthetic perturbations of motion fields or controlled variations in lighting, can enrich the set of inconsistencies observed during training.

For deployment in optimization-based editing pipelines, learning-based metrics must be differentiable with respect to the edited content [33]. This requirement constrains both network architecture and preprocessing steps. For instance, motion estimation modules used within the metric should be differentiable if they are part of the computational graph, or approximated by fixed operators whose gradients can be computed analytically. Gradient flow from the metric back to the editing model can be analyzed using matrix calculus, with the Jacobian of the metric with respect to input pixels composing with the Jacobian of the editing transformation. Stability and conditioning of this composite mapping influence the efficiency of gradient-based optimization and may motivate the design of normalization layers or residual connections within the metric network.

5. Numerical Analysis and Optimization Aspects

When perceptual metrics are used not only for evaluation but also as objective functions for optimizing editing algorithms, numerical considerations become central [34]. Let $\mathcal{F}_\phi$ denote an editing operator parameterized by ϕ, mapping an input sequence V^{in} and optional auxiliary inputs to an edited sequence

$$V^{\text{edit}} = \mathcal{F}_\phi\big(V^{\text{in}}\big).$$

A perceptual metric

$$J(V^{\text{edit}})$$

is then used as part of a loss function

$$\mathcal{L}(\phi) = J\big(\mathcal{F}_\phi(V^{\text{in}})\big)[35] + R(\phi),$$

where $R(\phi)$ represents regularization terms. Gradient-based optimization updates ϕ according to estimates of $\nabla_\phi \mathcal{L}$, which requires efficient and stable computation of gradients of the metric with respect to its input.

If the metric is expressed in terms of linear operators and pointwise nonlinearities, its gradient structure can be analyzed using standard tools from linear algebra and multivariate calculus. For instance, consider a simplified spatial consistency term

$$J_s(V^{\text{edit}}) = \frac{1}{T}\sum_{t=1}^{T}\frac{1}{|\Omega|}\sum_{x\in\Omega}\psi\big([36]\|\nabla I_t^{\text{edit}}(x)\|_2\big),$$

where ∇ is a discrete gradient operator and ψ is differentiable. In matrix form, with $\mathbf{i}_t$ representing frame t, the gradient operator becomes a matrix G such that

$$G\mathbf{i}_t$$

stacks spatial derivatives. The gradient of J_s with respect to $\mathbf{i}_t$ is then

$$\nabla_{\mathbf{i}_t} J_s = \frac{1}{T|\Omega|} G^\top g_t,$$

where g_t is a vector whose entries are [37]

$$g_{t,k} = \psi'\big(\|(G\mathbf{i}_t)_k\|_2\big)\frac{(G\mathbf{i}_t)_k}{\|(G\mathbf{i}_t)_k\|_2},$$

with k indexing spatial locations. This expression illustrates how the transpose of the gradient operator propagates local inconsistency sensitivities back to pixel intensities. Stability depends on properties of $G^\top G$, such as its eigenvalue distribution, which can be analyzed using spectral methods [38].

Temporal consistency terms involve similar structures but incorporate motion compensation. For a metric component

$$J_t(V^{\text{edit}}) = \frac{1}{T-1} \sum_{t=1}^{T-1} \frac{1}{|\Omega|} \sum_{x \in \Omega} \phi\big(\|I_{t+1}^{\text{edit}}(x + u_t(x)) - I_t^{\text{edit}}(x)\|_2\big), \text{[39]}$$

gradients must flow through interpolation at noninteger locations $x + u_t(x)$. Numerical differentiation in this setting typically relies on spatially differentiable interpolation schemes, such as bilinear or bicubic interpolation, so that derivatives with respect to pixel values and motion vectors are well defined. Discretization choices for interpolation affect both smoothness and computational cost, and may introduce anisotropies or directional biases in gradient propagation.

The conditioning of the metric influences optimization dynamics. If the metric is highly sensitive to small changes in certain directions of the input space, the Hessian of the loss may exhibit large eigenvalue ratios, leading to slow convergence or instability for gradient-based methods [40]. Preconditioning strategies, such as normalizing features, using perceptually motivated transforms, or incorporating residual connections, can ameliorate these issues by reshaping the optimization landscape. For learned metrics implemented by neural networks, techniques such as weight normalization, spectral normalization, and carefully chosen activation functions can contribute to more predictable gradient magnitudes.

Computational efficiency is crucial for metrics applied to high-resolution, long-duration sequences. Direct evaluation of spatiotemporal energies over all pixels and frames may be prohibitively expensive. Sampling strategies can reduce cost while preserving metric reliability. For example, one can subsample frames in time according to motion magnitude, sampling more densely in segments with rapid motion or complex edits and more sparsely in static segments [41]. In space, sampling can be guided by salience maps, edit masks, or gradient magnitudes, focusing computation on regions where inconsistencies are more likely to be perceptually salient. From a numerical analysis perspective, these strategies can be viewed as Monte Carlo approximations to integrals over space-time, with variance that depends on the sampling distribution.

When metrics are used to compare editing algorithms, numerical stability of the metric itself becomes important. Small perturbations of input sequences due to compression, resampling, or platform-specific processing should not cause large fluctuations in metric scores. This robustness can be quantified by Lipschitz-type bounds [42]. A metric J is Lipschitz continuous with constant L if

$$\big|J(V_1) - J(V_2)\big| \leq L\,\|V_1 - V_2\|, \text{[43]}$$

for an appropriate norm on sequences. For linear operators combined with bounded nonlinearities, such bounds can be derived from operator norms of the constituent matrices and derivatives of nonlinear functions. In learned metrics, explicit regularization or architectural constraints can be used to encourage small Lipschitz constants, improving robustness to minor perturbations.

Finally, integrating perceptual metrics into optimization loops for editing can require solving large-scale nonlinear optimization problems. In some scenarios, the editing operator $\mathcal{F}_\phi$ is itself defined implicitly through iterative procedures, such as solving a variational problem or running a recurrent network. In such cases, differentiating through $\mathcal{F}_\phi$ involves unrolling iterations or using implicit differentiation, each with its own numerical trade-offs. Memory and computational constraints may limit the extent to which perceptual metrics can be evaluated during optimization, motivating approximations, surrogate losses, or multistage schemes in which simpler metrics guide early iterations and more complex metrics refine later stages.

6. Evaluation Protocols and Experimental Considerations

The practical value of perceptual metrics for spatial and temporal consistency depends on how well they predict human judgments across diverse editing scenarios [44]. Designing evaluation protocols requires careful consideration of dataset construction, subjective testing methodology, statistical analysis, and the relationship between metric scores and perceptual scales. Because edited media encompass a wide variety of operations, content types, and viewing conditions, evaluation protocols must capture this diversity while remaining tractable [45].

Dataset construction begins with selecting source footage that spans different scene categories, including indoor and outdoor environments, natural and man-made structures, and varying levels of motion and complexity. Editing operations can then be applied to this footage using a range of algorithms and parameter settings, producing edited sequences with varying degrees and types of spatial and temporal inconsistency. It is useful to include both targeted artifacts, such as deliberate shadow mismatches or motion jitter, and naturally occurring artifacts from practical editing pipelines [46]. Annotated masks indicating edited regions, motion fields, and semantic segmentations can facilitate analysis of how metric performance varies across spatial locations and object categories.

Subjective evaluation protocols often rely on pairwise comparison or rating experiments. In pairwise comparisons, observers are shown two edited sequences side by side or in succession and asked which appears more spatially and temporally consistent relative to the underlying scene. This setup aligns well with the training paradigm for learning-based metrics and tends to produce reliable ordinal data. In rating experiments, observers assign scores to single sequences on a discrete or continuous scale representing perceived consistency or overall quality [47]. Rating experiments can be more efficient in terms of the number of sequences evaluated but may be more sensitive to individual differences in scale use.

Experimental design choices, such as randomized presentation order, viewing duration, and display calibration, influence the variability and bias in subjective data. For temporal artifacts, it is important that sequences can be replayed and that playback is smooth and synchronized, as timing jitter or dropped frames can confound judgments. Viewing distance and display size should be controlled to the extent possible, particularly when evaluating artifacts that depend on spatial resolution or when simulating specific usage scenarios, such as viewing on mobile devices versus large screens. When crowd-sourcing is used, additional quality-control mechanisms are needed to filter unreliable responses.

Once subjective data are collected, metric performance is typically assessed through correlation and classification measures [48]. Rank-based correlations between metric scores and mean opinion scores provide a measure of monotonic agreement, while linear correlations can indicate how well a metric captures absolute differences in perceived quality. For pairwise preference data, one can compute the probability that metric differences correctly predict human choices. Confidence intervals for these statistics can be estimated using bootstrap resampling, providing a sense of variability due to limited sample sizes. Care must be taken to avoid overfitting when metrics are learned on part of the data and evaluated on held-out sets with different content or editing operations.

Evaluation protocols can be refined by analyzing performance in specific regimes [2]. For example, one can examine how metric sensitivity to temporal inconsistencies varies with motion magnitude, scene complexity, or the duration of sequences. Similarly, performance on spatial consistency can be studied separately for different semantic categories, such as faces, text, or natural textures, where perceptual priorities differ. This type of stratified analysis can reveal systematic biases, such as a metric that performs well for global color consistency but poorly for geometric alignment, and can suggest targeted improvements or combinations of metric components.

Another consideration is the calibration of metric scales. Raw metric outputs may live on arbitrary numeric ranges and may not be directly interpretable as perceptual distances or probabilities [49]. Calibration procedures, such as fitting nonlinear mappings between metric scores and mean opinion scores or between score differences and preference probabilities, can make outputs more interpretable and comparable across metrics. These mappings can also be used to define threshold values corresponding to specific perceptual criteria, such as the score above which edits are likely to be acceptable for a certain proportion of viewers.

Finally, evaluation protocols should consider robustness and generalization. Metrics trained or tuned on one dataset may degrade when applied to different content, editing styles, or viewing conditions. Cross-dataset evaluations, where metrics are trained on one distribution and tested on another, are informative about generalization. Stress tests that introduce unusual artifacts, extreme edits, or distortions outside the training distribution can reveal failure modes [50]. In practical deployment, metrics may need to operate under constraints such as limited resolution previews or compressed streams, so evaluations that account for these constraints are also relevant.

7. Conclusion

Perceptual metrics for evaluating spatial and temporal consistency in edited visual media play a central role in both assessing and guiding modern editing pipelines. Unlike traditional distortion metrics that focus primarily on fidelity to a fixed reference, these metrics must account for the intentional introduction of new content and the complex ways in which human observers judge plausibility. Spatial consistency involves geometric, photometric, and semantic alignment of edited regions with their surroundings, while temporal consistency involves coherent evolution of these properties across frames under constraints of motion perception and temporal integration.

The discussion above has outlined how edited sequences can be represented as spatiotemporal tensors and how spatial and temporal consistencies can be quantified using combinations of derivatives, features, and graph structures [51]. Linear-algebraic formulations, such as Gram matrices and graph Laplacians, provide interpretable measures of global and local coherence, while differential and tensorial viewpoints connect discrete metrics to continuous models of motion and illumination. Probabilistic formulations, particularly those based on pairwise comparisons, link metric outputs to human judgments and support the learning of parametric metrics that adapt to empirical data.

Learning-based approaches, typically implemented with neural networks that process spatiotemporal features, allow metrics to capture complex perceptual cues across scales and content types. Numerical analysis shows how the structure of these metrics influences gradient flow and optimization behavior when metrics are integrated into editing algorithms. Stability, robustness, and computational efficiency emerge as important design considerations, especially for high-resolution video applications where metrics must be evaluated repeatedly during iterative optimization [52].

Evaluation protocols grounded in subjective experiments provide the empirical basis for validating and comparing perceptual metrics. The design of datasets, annotation strategies, and statistical analyses affects how well metrics can be judged and improved. Stratified evaluations reveal strengths and weaknesses across editing operations, content categories, and motion regimes, guiding the refinement of metric architectures and training procedures. Calibration of metric scales enhances interpretability and supports practical decisions about acceptable levels of spatial and temporal inconsistency in different application contexts.

Future work in this area can explore richer perceptual models that integrate attention, task, and context, as well as metrics that explicitly account for uncertainty and variability across observers. As editing tools continue to evolve and generate increasingly complex modifications, the design and analysis of perceptual metrics for spatial and temporal consistency will remain a relevant topic, requiring a combined perspective from vision science, mathematics, machine learning, and practical evaluation methodology [53].

References

[1] M. P. Kumar*, J. Preethi, J. Sandhya, and I. S. Harshini, ‘‘Medical image forgery detection using cnn,’’ *International Journal of Innovative Technology and Exploring Engineering*, vol. 9, pp. 1325–1329, 4 2020.

[2] E. S. Lahemer and A. Rad, ‘‘Holoslam: a novel approach to virtual landmark-based slam for indoor environments,’’ *Complex & Intelligent Systems*, vol. 10, pp. 4175–4200, 3 2024.

[3] S.-H. Pan and S.-C. Wang, ‘‘Identifying vehicles dynamically on freeway cctv images through the yolo deep learning model,’’ *Sensors and Materials*, vol. 33, pp. 1517–, 5 2021.

[4] R. Doyle, ‘‘A digital language resource center,’’ *IALLT Journal of Language Learning Technologies*, vol. 32, pp. 17–26, 4 2000.

[5] S. S. Harsha, D. Agarwal, A. Revanur, and S. Agrawal, ‘‘Digital video editing based on a target digital image,’’ Aug. 21 2025. US Patent App. 18/583,067.

[6] K.-I. Yoon, T.-S. Jeong, S.-C. Kim, and S.-C. Lim, ‘‘Anonymizing at-home fitness: enhancing privacy and motivation with virtual reality and try-on.,’’ *Frontiers in public health*, vol. 11, pp. 1333776–, 12 2023.

[7] C. M. Horvath, T. Thomessen, and G. Sziebig, ‘‘Overview of modern teaching equipment that supports distant learning,’’ *Recent Innovations in Mechatronics*, vol. 5, 1 1970.

[8] Z. Zheng, N. Chen, J. Wu, Z. Xv, S. Liu, and Z. Luo, ‘‘Ew-yolov7: A lightweight and effective detection model for small defects in electrowetting display,’’ *Processes*, vol. 11, pp. 2037–2037, 7 2023.

[9] A. Pal, S. Gopi, and K. M. Lee, ‘‘Fintech agents: Technologies and theories,’’ *Electronics*, vol. 12, pp. 3301–3301, 7 2023.

[10] R. Alfaifi and A. M. Artoli, ‘‘Human action prediction with 3d-cnn,’’ *SN Computer Science*, vol. 1, pp. 1–15, 8 2020.

[11] J. Zhang, X. Wen, A. Cho, and M. Whang, ‘‘An empathy evaluation system using spectrogram image features of audio.,’’ *Sensors (Basel, Switzerland)*, vol. 21, pp. 7111–, 10 2021.

[12] E. de Souza e Silva, R. M. M. Leão, A. K. da Silva dos Santos, B. C. M. Netto, and J. A. G. Azevêdo, ‘‘Multimedia supporting tools for the cederj distance learning initiative applied to the computer systems course,’’ *Revista Brasileira de Aprendizagem Aberta e a Distância*, vol. 5, 5 2008.

[13] M. Jeong, M. Park, J.-Y. Nam, and B. C. Ko, ‘‘Light-weight student lstm for real-time wildfire smoke detection.,’’ *Sensors (Basel, Switzerland)*, vol. 20, pp. 5508–, 9 2020.

[14] P. K. D. Pramanik, S. Pal, and P. Choudhury, ‘‘Mobile crowd computing: potential, architecture, requirements, challenges, and applications,’’ *The Journal of Supercomputing*, vol. 80, pp. 2223–2318, 7 2023.

[15] B. Lafreniere, T. Grossman, J. Matejka, and G. Fitzmaurice, ‘‘Chi - investigating the feasibility of extracting tool demonstrations from in-situ video content,’’ in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 4007–4016, ACM, 4 2014.

[16] R. N. Abirami, P. M. D. R. Vincent, K. Srinivasan, U. Tariq, and C.-Y. Chang, ‘‘Deep cnn and deep gan in computational visual perception-driven image analysis,’’ *Complexity*, vol. 2021, pp. 1–30, 4 2021.

[17] S. Krusche, I. A. Naser, M. Bdiwi, and S. Ihlenfeldt, ‘‘A novel approach for automatic annotation of human actions in 3d point clouds for flexible collaborative tasks with industrial robots.,’’ *Frontiers in robotics and AI*, vol. 10, pp. 1028329–, 2 2023.

[18] T. Mori, M. Shimosaka, and T. Sato, *ISER - SVM-Based Human Action Recognition and Its Remarkable Motion Features Discovery Algorithm*, pp. 15–25. Germany: Springer Berlin Heidelberg, 3 2006.

[19] C. Porumb, S. Porumb, B. Orza, and A. Vlaicu, *Generic Framework for Collaborative Work Environments*. InTech, 3 2010.

[20] J. Kong and C. Wang, ‘‘Resolution enhancement for low-resolution text images using generative adversarial network,’’ *MATEC Web of Conferences*, vol. 246, pp. 03040–, 12 2018.

[21] M. Liu and J. Bu, ‘‘Deep integration of physical health education based on intelligent communication technology.,’’ *Journal of healthcare engineering*, vol. 2021, pp. 4323043–6, 9 2021.

[22] S. Changder, A. Majumder, and S. Kundu, ‘‘An improved location mapping and cover synthesis based data hiding by model sssp problem generation,’’ *Multimedia Tools and Applications*, vol. 82, pp. 29255–29281, 2 2023.

[23] B. Gan, V. Chang, G. Wang, X. Pan, G. Wang, N. Zou, and F. Feng, ‘‘Design and implementation of intel-sponsored real-time multiview face detection system,’’ in *Computer Science & Information Technology (CS & IT)*, pp. 37–47, Academy & Industry Research Collaboration Center (AIRCC), 5 2015.

[24] S.-M. Hu, T. Chen, K. Xu, M.-M. Cheng, and R. R. Martin, ‘‘Internet visual media processing: a survey with graphics and vision applications,’’ *The Visual Computer*, vol. 29, pp. 393–405, 3 2013.

[25] R. Munoz, R. Olivares, C. Taramasco, R. Villarroel, R. Soto, M. F. Alonso-Sánchez, E. Merino, and V. H. C. de Albuquerque, ‘‘A new eeg software that supports emotion recognition by using an autonomous approach,’’ *Neural Computing and Applications*, vol. 32, pp. 11111–11127, 12 2018.

[26] W. Paier, A. Hilsmann, and P. Eisert, ‘‘Interactive facial animation with deep neural networks,’’ *IET Computer Vision*, vol. 14, pp. 359–369, 8 2020.

[27] V. K. Chauhan, J. Zhou, P. Lu, S. Molaei, and D. A. Clifton, ‘‘A brief review of hypernetworks in deep learning,’’ *Artificial Intelligence Review*, vol. 57, 8 2024.

[28] Q. Gao and X. Wu, ‘‘Real-time deep image retouching based on learnt semantics dependent global transforms,’’ *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 30, pp. 7378–7390, 8 2021.

[29] R. Raman, V. K. Nair, P. Nedungadi, I. Ray, and K. Achuthan, ‘‘Darkweb research: Past, present, and future trends and mapping to sustainable development goals.,’’ *Heliyon*, vol. 9, pp. e22269–e22269, 11 2023.

[30] S. Z. Maw, T. T. Zin, P. Tin, I. Kobayashi, and Y. Horii, ‘‘An absorbing markov chain model to predict dairy cow calving time.,’’ *Sensors (Basel, Switzerland)*, vol. 21, pp. 6490–, 9 2021.

[31] K. Kadam, S. Ahirrao, and K. Kotecha, ‘‘Multiple image splicing dataset (misd): A dataset for multiple splicing,’’ *Data*, vol. 6, pp. 102–, 9 2021.

[32] U. Uche-Ike and L. Angrave, ‘‘Improving access to engineering education: Unlocking text and table data in images and videos,’’ in *2023 IL-IN Section Conference Proceedings*, ASEE Conferences, 2 2024.

[33] K. Maksymenko, A. K. Clarke, I. M. Guerra, S. Deslauriers-Gauthier, and D. Farina, ‘‘A myoelectric digital twin for fast and realistic modelling in deep learning.,’’ *Nature communications*, vol. 14, pp. 1600–, 3 2023.

[34] Y. Wang, Q. Zhang, G.-G. Wang, and H. Cheng, ‘‘The application of evolutionary computation in generative adversarial networks (gans): a systematic literature survey,’’ *Artificial Intelligence Review*, vol. 57, 6 2024.

[35] A. Malik, H. Lhachemi, and R. Shorten, ‘‘A cyber-physical system to design 3d models using mixed reality technologies and deep learning for additive manufacturing.,’’ *PloS one*, vol. 18, pp. e0289207–e0289207, 7 2023.

[36] M. M, M. Mahanta, and P. R. K. Gupta, ‘‘Arya: A progeniture intelligent assistant for superior user experience,’’ *EPRA International Journal of Research & Development (IJRD)*, pp. 386–390, 4 2020.

[37] G. Bao, X. Wang, R. Xu, C. Loh, O. D. Adeyinka, D. A. Pieris, S. Cherepanoff, G. Gracie, M. Lee, K. L. McDonald, A. K. Nowak, R. B. Banati, M. E. Buckland, and M. B. Graeber, ‘‘Pathofusion: An open-source ai framework for recognition of pathomorphological features and mapping of immunohistochemical data.,’’ *Cancers*, vol. 13, pp. 617–, 2 2021.

[38] N. Kaur, N. Jindal, and K. Singh, ‘‘Passive image forgery detection techniques: A review, challenges, and future directions,’’ *Wireless Personal Communications*, vol. 134, pp. 1491–1529, 4 2024.

[39] C. Ulloa, D. M. Ballesteros, and D. Renza, ‘‘Video forensics: Identifying colorized images using deep learning,’’ *Applied Sciences*, vol. 11, pp. 476–, 1 2021.

[40] G. Touya, X. Zhang, and I. Lokhat, ‘‘Is deep learning the new agent for map generalization,’’ *International Journal of Cartography*, vol. 5, pp. 142–157, 5 2019.

[41] J. Cho, J. Kang, Y. Song, S. Lee, and J. Yeon, ‘‘Innovative imaging and analysis techniques for quantifying spalling repair materials in concrete pavements,’’ *Sustainability*, vol. 16, pp. 112–112, 12 2023.

[42] W. Xue, Z. Feng, C. Xu, Z. Meng, and C. Zhang, ‘‘Adaptive object tracking via multi-angle analysis collaboration.,’’ *Sensors (Basel, Switzerland)*, vol. 18, pp. 3606–, 10 2018.

[43] M. R. Bueno, C. Estrela, J. M. Granjeiro, M. R. de Araújo Estrela, B. Azevedo, and A. R. Diogenes, ‘‘Cone-beam computed tomography cinematic rendering: clinical, teaching and research applications,’’ *Brazilian oral research*, vol. 35, pp. 1–13, 2 2021.

[44] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, ‘‘Learning-based view synthesis for light field cameras,’’ *ACM Transactions on Graphics*, vol. 35, pp. 193–10, 11 2016.

[45] S. S. Harsha, A. Revanur, D. Agarwal, and S. Agrawal, ‘‘Genvideo: One-shot target-image and shape aware video editing using t2i diffusion models,’’ in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7559–7568, 2024.

[46] A. C. Cruz, A. Luvisi, L. D. Bellis, and Y. Ampatzidis, ‘‘X-fido: An effective application for detecting olive quick decline syndrome with deep learning and data fusion.,’’ *Frontiers in plant science*, vol. 8, pp. 1741–1741, 10 2017.

[47] A. Pavel, C. Reed, B. Hartmann, and M. Agrawala, ‘‘Uist - video digests: a browsable, skimmable format for informational lecture videos,’’ in *Proceedings of the 27th annual ACM symposium on User interface software and technology*, pp. 573–582, ACM, 10 2014.

[48] null null, ‘‘D3.8 interactive authoring toolkit integration with miragexr,’’ 2 2022.

[49] D. Bogaevskiy, S. Ezhov, P. Fedoseev, D. Butusov, D. H. Elkamchouchi, M. S. Alqahtani, M. Abbas, B. O. Soufiene, and D. Kaplun, ‘‘Key sets analysis of compiler vector options for h.264 video compression algorithms implemented on the mips simd architecture,’’ 6 2023.

[50] H. Utz, U. Kaufmann, G. Mayer, and G. K. Kraetzschmar, ‘‘Vip - a framework-based approach to robot vision,’’ *International Journal of Advanced Robotic Systems*, vol. 3, pp. 12–, 3 2006.

[51] D. Khosrowi, F. Finn, and E. Clark, ‘‘Engaging the many-hands problem of generative-ai outputs: a framework for attributing credit,’’ *AI and Ethics*, vol. 5, pp. 4495–4513, 3 2024.

[52] N. Patwardhan, S. Marrone, and C. Sansone, ‘‘Transformers in the real world: A survey on nlp applications,’’ *Information*, vol. 14, pp. 242–242, 4 2023.

[53] J. A. Omiye, H. Gui, R. Daneshjou, Z. R. Cai, and V. Muralidharan, ‘‘Principles, applications, and future of artificial intelligence in dermatology.,’’ *Frontiers in medicine*, vol. 10, pp. 1278232–, 10 2023.