GLOPaint: History-Normalized Loss Balancing for Resource-Efficient Graph-Based High-Resolution Image Inpainting

Anonymous authors
Paper under double-blind review

Abstract

High-resolution image inpainting requires restoring missing regions with high reconstruction fidelity, perceptual consistency, and structural coherence while remaining practical in model size, memory usage, and inference latency. We propose **GLOPaint**, a lightweight graph-based framework for efficient high-resolution image inpainting. GLOPaint combines edge-aware convolutional feature extraction with graph attention reasoning at the bottleneck, allowing compact appearance features and predicted boundary cues to interact through a feature-space graph rather than fixed spatial proximity. This design supports structure-aware information exchange between semantically related regions while avoiding dense full-resolution graph construction. To stabilize training with heterogeneous objectives, we further introduce **Historical Moving Average (HMA)** weighting, a history-normalized loss balancing mechanism that adjusts each objective's dynamic weight according to its own optimization trajectory rather than relying on fixed coefficients or direct cross-objective normalization. Experiments on Places2 and CelebA-HQ at 1024×1024 resolution show that GLOPaint achieves the lowest LPIPS on both datasets and competitive PSNR among the compared high-resolution baselines. With 13.6M parameters and around 2.9 ms inference time per image, GLOPaint offers a favorable quality–efficiency trade-off for deployment-oriented high-resolution inpainting, particularly in settings where training data and GPU resources are constrained. These results suggest that compact feature-space graph reasoning, combined with history-normalized objective balancing, provides an effective and computationally practical direction for structure-aware image completion at high resolution.

1 Introduction

High-resolution image inpainting aims to restore missing image regions with visually plausible, semantically consistent, and structurally coherent content. It is an important problem in image restoration and editing, with applications in object removal, content completion, and visual enhancement. The task is inherently ill-posed, since multiple valid completions may exist for the same missing region. At high resolution, this ambiguity becomes more challenging: small local artifacts become more visible, long-range contextual dependencies are harder to model, and practical deployment requires careful control of memory usage, model size, training cost, and inference latency (Yi et al., 2020; Liu et al., 2023).

Recent progress in image inpainting has been driven by architectural advances for local feature extraction, contextual modeling, structural guidance, and generative restoration. Mask-aware and gated convolutional methods improve reconstruction under irregular missing regions (Liu et al., 2018; Yu et al., 2019), while structure-guided approaches use contour or edge priors to encourage boundary-consistent completion (Nazeri et al., 2019). Later methods extended inpainting toward larger-context reasoning through transformer-based architectures, Fourier-domain processing, and diffusion-based generative priors, improving structural coherence and perceptual fidelity (Li et al., 2022; Ko & Kim, 2023; Suvorov et al., 2022; Lugmayr et al., 2022). More recent work has further explored stronger guidance mechanisms (Ju et al., 2024; Zhuang et al., 2023) and faster sampling procedures (Kawar et al., 2022). These developments have substantially improved image completion quality, but they also raise an important practical question: how can high-resolution

inpainting remain usable when training data, GPU capacity, memory, model size, and inference budget are limited (Sargsyan et al., 2023)?

This question is particularly relevant for small-scale and low-resource settings. In this work, we use low-resource to refer to settings where the amount of training data and the available GPU resources for training and inference are constrained. Under such conditions, the objective is not only to improve restoration metrics, but also to make the model trainable and deployable within practical computational limits. Many high-resolution inpainting methods rely on staged or coarse-to-fine designs to combine global structure with local detail refinement (Yi et al., 2020; Zhang et al., 2025). Although such designs can improve visual quality, they may also increase architectural complexity, training cost, and inference overhead. Transformer-based and diffusion-based methods provide stronger contextual modeling and perceptual restoration quality (Li et al., 2022; Ko & Kim, 2023; Lugmayr et al., 2022), but their memory, runtime, and optimization costs can become significant at high resolution. These limitations motivate compact architectures that can preserve local reconstruction fidelity, support broader contextual reasoning, and remain practical under limited data and GPU resources.

In addition to architectural efficiency, high-resolution inpainting also requires stable optimization across heterogeneous objectives. Inpainting models are commonly trained using multiple loss terms that capture different reconstruction goals, such as pixel-level fidelity, perceptual consistency, and structural preservation. These objectives can have different numerical scales and convergence behaviors, and their relative importance may change during training. Fixed loss coefficients can therefore lead to suboptimal optimization, where one objective dominates the shared update while others receive insufficient training signal. Prior work on adaptive loss balancing and auxiliary reweighting suggests that more flexible weighting can improve optimization and reduce manual coefficient tuning (Kendall et al., 2018; Chen et al., 2018; Hui et al., 2021). This motivates treating loss balancing as an explicit component of the high-resolution inpainting framework, especially when extensive hyperparameter search or repeated large-scale training is impractical.

Graph-based reasoning offers a natural way to model relational structure between image regions. By encoding interactions through learned connectivity, graph neural networks can propagate contextual information across structurally or semantically related regions (Veličković et al., 2018; Brody et al., 2022). This property is useful for inpainting, where missing regions may need information from visually related areas that are spatially distant. However, applying graph reasoning directly at full image resolution is computationally expensive. Moreover, graph-based inpainting depends on how relational reasoning is integrated with strong local feature extraction and structural guidance. Although recent work suggests that graph representations can benefit image completion (Verma et al., 2024), compact graph reasoning for efficient high-resolution inpainting remains comparatively underexplored.

Motivated by these challenges, we propose **GLOPaint**: Graph Learning with History-Normalized Loss Optimization for High-Resolution Image Inpainting. GLOPaint is designed to improve restoration quality while also making high-resolution inpainting more practical in small-scale settings with limited training data and GPU resources. The framework combines compact convolutional representation learning, boundary-aware structural priors, and feature-space graph attention. Instead of applying relational reasoning densely at the pixel level, GLOPaint performs graph attention over bottleneck-level feature representations. This allows visually related regions to exchange appearance and structural information beyond fixed spatial neighborhoods while keeping the graph reasoning stage computationally compact for both training and inference.

To coordinate heterogeneous training objectives, we further introduce **Historical Moving Average (HMA)** weighting, a history-normalized loss balancing mechanism. HMA assigns each objective a bounded dynamic weight by comparing its current loss with its own historical baseline. This avoids direct cross-objective normalization between losses with different numerical scales and reduces reliance on fixed loss coefficients. The mechanism complements the architecture-level efficiency of GLOPaint by stabilizing multi-objective optimization during training, which is particularly useful when data, GPU resources, or opportunities for extensive hyperparameter tuning are limited.

The primary contributions of this work are summarized as follows:

- **Feature-space graph attention for high-resolution inpainting.** We introduce GLOPaint, a lightweight graph-based inpainting framework that combines compact convolutional feature extraction, boundary-aware structural priors, and graph attention reasoning. The proposed design performs relational reasoning at the bottleneck level, allowing visually related regions to exchange appearance and structural information while avoiding dense full-resolution graph operations.
- **History-normalized loss optimization via HMA.** We introduce Historical Moving Average (HMA) weighting for multi-objective inpainting training. HMA compares each objective with its own historical baseline and updates a bounded dynamic weight according to its relative optimization trajectory. This reduces dependence on fixed coefficient tuning and supports joint optimization across reconstruction, perceptual, and structural objectives with different numerical ranges.
- **High-resolution evaluation with perceptual and efficiency analysis.** We evaluate GLOPaint against representative high-resolution inpainting baselines using reconstruction, perceptual, and efficiency-oriented metrics. The results show that GLOPaint achieves strong perceptual quality and competitive reconstruction fidelity while obtaining the lowest GFLOPs and fastest estimated inference time among the compared methods, with a moderate parameter count. These results support the goal of making high-resolution inpainting more usable under limited training data and GPU resources, where both training feasibility and inference efficiency are important practical constraints.

2 Related Work

GLOPaint is positioned at the intersection of two research directions: (i) structural guidance and objective balancing for boundary-faithful image inpainting, and (ii) efficient high-resolution completion through compact contextual reasoning. These directions are closely related to the central goal of this work: improving high-resolution inpainting quality while keeping training and inference practical under limited data and GPU resources.

2.1 Structural Guidance and Multi Objective Optimization

Boundary faithful inpainting requires structural cues to remain accessible throughout the reconstruction process. Mask aware convolutional methods improve local recovery under irregular missing regions (**?**Yu et al., 2019). Structure guided approaches further show that explicit edge or contour priors can improve geometric coherence in completed regions (Nazeri et al., 2019; Dong et al., 2022). At high resolution, even small boundary inconsistencies can become visually noticeable. This makes structural guidance particularly important for reliable inpainting.

Structural objectives are commonly optimized together with pixel level reconstruction and perceptual losses. These objectives differ substantially in numerical scale and convergence behavior. Fixed loss coefficients can therefore overemphasize one objective while limiting the contribution of others. This form of task interference has been widely studied in multi task learning (Vandenhende et al., 2021). Adaptive weighting methods address this issue through uncertainty estimation (Kendall et al., 2018), gradient normalization (Chen et al., 2018), auxiliary reweighting (Hui et al., 2021), cross objective normalization (Liu et al., 2019), and gradient manipulation (Yu et al., 2020; Liu et al., 2021). However, many of these methods assume homogeneous output heads, shared gradient access across all objectives, or computational budgets that may be difficult to maintain with frozen subnetworks and single GPU high resolution training. Adaptive balancing of heterogeneous objectives therefore remains underexplored in graph based high resolution inpainting, where structural, perceptual, and reconstruction signals must be jointly coordinated under practical resource constraints.

2.2 Efficient High-Resolution Inpainting and Graph-Based Reasoning

High-resolution image inpainting requires broad contextual reasoning while keeping memory usage, model size, training cost, and inference latency within practical limits. Convolutional and frequency-domain methods address large-mask completion through contextual aggregation, multi-scale modeling, and Fourier-domain processing (Yi et al., 2020; Zeng et al., 2021; Suvorov et al., 2022). Coordinate-based and patch-aware

methods further reduce resolution-dependent computational cost by changing how image representations are parameterized or processed (Liu et al., 2023; Zhang et al., 2025). Transformer, diffusion, and iterative restoration methods improve contextual coherence and perceptual quality through attention-based modeling or repeated refinement strategies (Li et al., 2022; Ko & Kim, 2023; Lugmayr et al., 2022; Kawar et al., 2022; Delbracio & Milanfar, 2023). Although these methods have advanced restoration quality, their architectural complexity, memory demand, or iterative inference cost can limit their suitability for small-scale settings with constrained GPU resources.

Efficiency-focused inpainting methods directly study this quality–efficiency trade-off for mobile or edge deployment (Sargsyan et al., 2023; Sanchez et al., 2025). GLOPaint follows this deployment-oriented motivation, but approaches efficiency from a different angle: it restricts relational reasoning to compact bottleneck representations rather than applying expensive global reasoning over full-resolution spatial grids. Dense convolutional operations are used for local feature extraction, while the graph module is confined to a compressed feature space where non-local information exchange is more computationally manageable.

Graph neural networks provide a natural mechanism for relational reasoning over structured representations. Graph Attention Networks enable attention-weighted neighborhood aggregation, while GATv2 improves the expressiveness of the original attention formulation (Veličković et al., 2018; Brody et al., 2022). For inpainting, graph-based reasoning is attractive because missing regions may require information from semantically related but spatially distant areas. Recent graph-based inpainting work further suggests that relational representations can benefit image completion (Verma et al., 2024). However, applying graph reasoning directly over pixels or dense high-resolution grids is computationally expensive, making it less suitable for high-resolution settings with constrained GPU resources.

A practical alternative is to perform graph reasoning over compact feature representations, where relational connectivity can support long-range information exchange without the expense of full-resolution graph construction. GLOPaint adopts this direction by constructing a feature-space adaptive graph at the bottleneck resolution. Each bottleneck spatial location is treated as a node, and edges are formed based on feature similarity rather than fixed spatial proximity. Compared with existing graph-based inpainting work, GLOPaint couples compact graph attention with boundary-aware structural priors and history-normalized objective balancing. This combination is designed to support structure-aware restoration while maintaining practical training and inference costs at high resolution.

Overall, prior work has made substantial progress in structural guidance, contextual modeling, generative restoration, graph-based reasoning, and efficient inference. Yet these directions are often studied separately. GLOPaint addresses this gap by combining boundary-aware feature extraction, compact feature-space graph attention, and history-normalized loss balancing within a single high-resolution inpainting framework.

3 Methodology

GLOPaint addresses resource-efficient high-resolution image inpainting through three sequential processing stages, illustrated in Figure 1. First, a structural feature extractor, comprising a sequential ensemble of a bottleneck encoder and a dilated encoder, produces compact appearance features, while a dedicated edge prediction network generates structural boundary priors. Second, the extracted features and edge priors are projected into a graph-compatible representation and supplied as contextual input to a Graph Neural Network (GNN) with Graph Attention (GAT) support. This network performs non-local structural reasoning over a feature-space adaptive graph constructed at the bottleneck resolution. Third, a lightweight convolutional decoder reconstructs the full-resolution image from the propagated graph node features.

The problem formulation and masking strategy are introduced in Section 3.1. The structural feature extractor and edge prediction network are described in Section 3.2.1. The graph construction and GNN reasoning modules are detailed in Section 3.2.2, and the Historical Moving Average (HMA) optimization mechanism is presented in Section 3.3. HMA is the primary optimization contribution of this work: it provides a generalized, scale-invariant loss-balancing strategy that stabilizes joint optimization across any number of heterogeneous objectives with arbitrary numerical ranges.

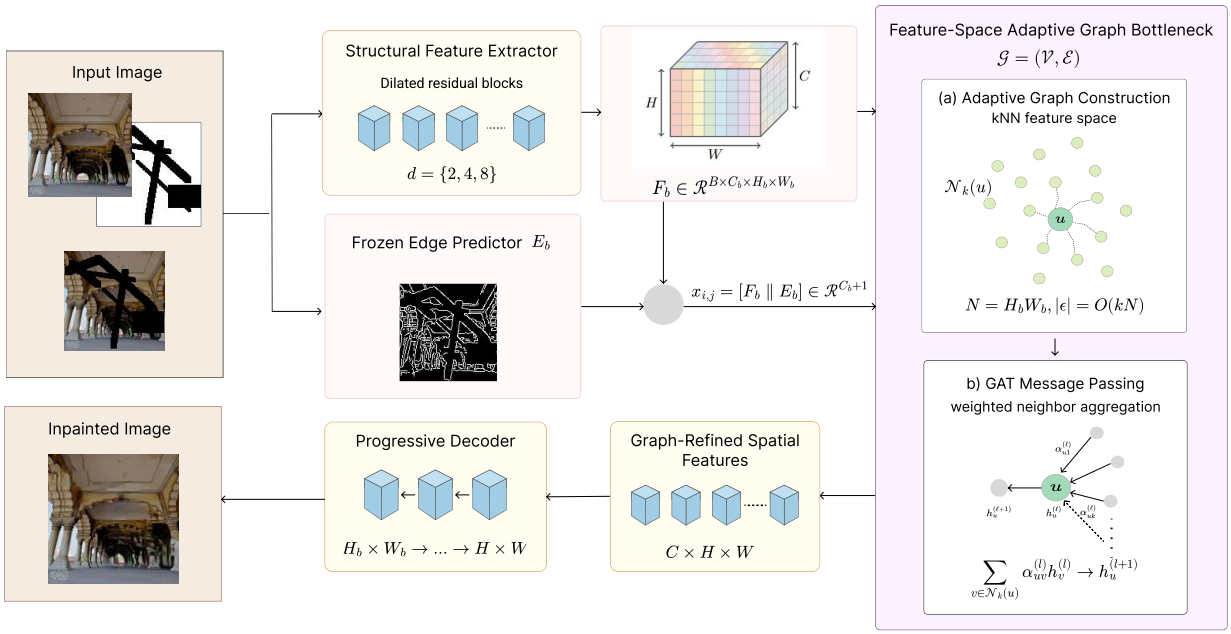


Figure 1: Overall architecture of the proposed GLOPaint framework for high-resolution image inpainting. The masked image and binary mask are first processed by an edge-aware structural feature extractor, which produces compact bottleneck features and structural edge cues. These representations are then projected into a feature-space adaptive graph bottleneck, where graph attention-based message passing performs structure-aware feature propagation. Finally, the propagated graph features are reshaped back into a tensor representation and decoded to reconstruct the completed image.

3.1 Problem Formulation

Given a batch of high-resolution images $\mathbf{I} \in \mathbb{R}^{B \times 3 \times H \times W}$ and binary masks $\mathbf{M} \in \{0, 1\}^{B \times 1 \times H \times W}$, where $\mathbf{M} = 1$ denotes valid pixels and $\mathbf{M} = 0$ denotes missing regions, the masked observation is formed as

$$\mathbf{I}_m = \mathbf{I} \odot \mathbf{M}, \tag{1}$$

where $\odot$ denotes channel-wise Hadamard multiplication with the binary mask broadcast across RGB channels. The goal is to learn an inpainting function f_θ that takes the masked observation and its binary mask as input and predicts a completed image $\hat{\mathbf{I}} = f_\theta(\mathbf{I}_m, \mathbf{M})$ whose missing region is visually plausible and structurally consistent with the surrounding content. The model receives the concatenated input tensor $\mathbf{X}_{\text{in}} = [\mathbf{I}_m, \mathbf{M}] \in \mathbb{R}^{B \times 4 \times H \times W}$, where the four channels correspond to the masked RGB image and the binary mask.

To prevent the model from overfitting to fixed geometric occlusion patterns, we adopt a stochastic masking strategy during training. At each iteration, a target occlusion ratio r is sampled, and irregular masks are generated by compositing randomized geometric primitives, including lines, circles of varying radius, and rectangles. This process continues until the realized ratio $\hat{r}$ approximately matches r. The resulting stochastic mask distribution reduces memorization of spatially regular missing regions and encourages generalization across diverse irregular masks. This is particularly important at high resolution, where mask diversity strongly affects the robustness of image reconstruction.

3.2 Hybrid Convolutional-GNN Architecture

The core architectural hypothesis of GLOPaint is that local texture extraction is most effectively handled by dense convolutional operations, whereas longer-range structure-aware reasoning is better addressed through graph-based message passing over a compact bottleneck representation. Rather than applying the GNN

directly to the raw masked input, GLOPaint first extracts rich appearance features and structural boundary priors using dedicated convolutional networks. These extracted priors serve as contextual input to the GNN, enabling it to operate over semantically enriched node representations rather than learning low-level structural context from scratch. The following subsections describe the convolutional feature extraction stage, followed by graph construction and graph-based reasoning.

3.2.1 Structural Feature Extraction

The input tensor $\mathbf{X}_{\text{in}}$ is processed by a structural feature extractor comprising two sequential components ensembled within a single convolutional model. The first component is a strided bottleneck encoder that progressively downsamples the spatial dimensions of the input, yielding an intermediate feature map $F \in \mathbb{R}^{B\times C\times H_s\times W_s}$ that captures compact appearance representations. The second component is a dilated encoder applied directly to F, with a dilation schedule $d \in \{2, 4, 8\}$, which expands the receptive field without further reducing spatial resolution. The combined sequential operation produces a bottleneck feature tensor $F_b \in \mathbb{R}^{B\times C_b\times H_b\times W_b}$ encoding multi-scale contextual representations at the bottleneck resolution. By ensembling these two components, the feature extractor controls the granularity and volume of node-level information supplied to the GNN, ensuring a steady and dimensionally compatible flow of feature representations without overwhelming the graph reasoning stage.

Concurrently, a dedicated edge prediction network Φ_{edge} processes the masked input and maps it to a pixel-wise edge probability map $\mathbf{E} \in [0, 1]^{B\times 1\times H\times W}$, representing predicted object contours and boundary structures within the image. The edge network is pretrained independently under a structural edge objective $\mathcal{L}$edge, computed between the predicted edge probability map and a Canny-derived edge target from the ground-truth image. Its parameters are subsequently frozen during inpainting training, so the predicted edge map serves as a fixed structural prior for graph construction. When $\mathcal{L}$edge is included among the inpainting objectives, it supervises structural boundary consistency of the generated output rather than updating the frozen edge prediction network. The relative weight of this objective during training is balanced adaptively by the HMA mechanism described in Section 3.3.

3.2.2 Graph Construction and Reasoning

Using the bottleneck feature tensor F_b from Section 3.2.1, we construct a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ over the bottleneck spatial lattice for each image. Each spatial coordinate (i, j) in the lattice is treated as a distinct node $v_{i,j} \in \mathcal{V}$, yielding $N = H_b W_b$ nodes per image. The edge probability map $\mathbf{E}$ is downsampled to the bottleneck resolution to produce $\mathbf{E}_b \in \mathbb{R}^{B\times 1\times H_b\times W_b}$, and the node feature at position (i, j) is formed by concatenating the appearance and boundary representations:

$$\mathbf{x}_{i,j} = \left[F_{b,:,i,j} \parallel \mathbf{E}_{b,:,i,j}\right] \in \mathbb{R}^{C_b+1} \tag{2}$$

Feature-Space Adaptive Graph. Rather than connecting nodes using fixed spatial proximity, GLOPaint constructs the edge set $\mathcal{E}$ through k-nearest-neighbor connectivity in the bottleneck feature space. For each node $v_{i,j}$, let

$$\mathbf{n}_{i,j} = \frac{F_{b,:,i,j}}{\|F_{b,:,i,j}\|_2} \tag{3}$$

denote its ℓ_2-normalized feature vector. The cosine similarity between any two nodes u and v is

$$s(u, v) = \mathbf{n}_u^\top \mathbf{n}_v \in [-1, 1] \tag{4}$$

Each node is then connected to its k most similar neighbors under this metric, excluding self-loops:

$$\mathcal{N}_k(u) = \arg\operatorname*{top\text{-}k}_{v\neq u}\ s(u, v), \tag{5}$$

and the edge set is defined as $\mathcal{E} = \{(u, v) : v \in \mathcal{N}_k(u)\}$, symmetrized to form an undirected graph. The cosine similarity score $s(u, v)$ is retained as a scalar edge attribute and supplied to the GAT layers as an input edge feature. Specifically, this scalar attribute is incorporated into the attention computation so that attention

logits can be informed directly by inter-node feature affinity. All pairwise similarities are computed through batched matrix multiplication on the GPU, giving $O(N^2)$ similarity computation followed by $O(N_k)$ edge retention. At bottleneck resolutions, this process is substantially faster than constructing a graph over the original full-resolution pixel space.

This feature-driven topology deliberately departs from fixed spatial adjacency in order to capture long-range structural dependencies. Natural scenes often exhibit non-local structural coherence: repeating textures, symmetrical boundaries, and occluded background regions may be separated by large spatial distances. By dynamically constructing the graph in feature space, GLOPaint bypasses this spatial bottleneck. Semantically similar regions can be connected through direct, single-hop edges, enabling the GAT to aggregate and propagate texture and boundary priors across the image independent of physical distance.

Graph Attention Reasoning. The GNN operates over the constructed graph using Graph Attention layers (Veličković et al., 2018; Brody et al., 2022) for attention-based aggregation. Each GAT layer updates node representations by computing learned attention coefficients over neighboring nodes and aggregating their features accordingly, enabling selective weighting of structural and appearance information during message passing. Each attention head produces an output of dimension d_h. Outputs from all H heads are concatenated and projected back to the node feature dimension via a 1×1 convolution. A gated residual connection, controlled by a learned sigmoid scalar per node, modulates the contribution of the graph-refined features relative to the original bottleneck representation. This design reduces the risk that the graph signal destabilizes early training, when node features may not yet be semantically meaningful.

Stacking L GAT layers supports non-local information flow through multi-hop propagation. After L layers of message passing, each node effectively aggregates structural and appearance cues from neighbors up to L hops away in the feature-space graph. This allows deeply masked regions to receive signals from visually related elements beyond their immediate spatial boundaries. The propagated node features are then reshaped into a spatial tensor $\mathbf{Z} \in \mathbb{R}^{B \times C_g \times H_b \times W_b}$, where C_g denotes the GNN output feature dimension, and passed to the convolutional decoder.

3.3 Dynamic Multi-Objective Optimization

Training GLOPaint requires the joint optimization of heterogeneous objectives that capture different aspects of the inpainting problem. These objectives naturally operate at different numerical scales and exhibit distinct optimization dynamics. Some losses may converge rapidly, others may change more gradually, and their gradient magnitudes can differ by orders of magnitude during training. Under static loss weighting, such discrepancies can cause one or more objectives to dominate the shared parameter update, leaving the remaining objectives with insufficient gradient influence.

A common remedy is to normalize each objective by a fixed constant so that all losses lie on a comparable scale. This strategy, however, can conflate numerical magnitude with optimization difficulty. For example, a VGG-based perceptual loss may carry a large numerical magnitude while following a relatively smooth convergence trajectory, whereas a structural boundary objective may be numerically smaller but exhibit greater volatility during training. Fixed normalization can map both to a similar range, but the resulting weights do not reflect their distinct convergence behaviours. Consequently, the variance that carries information about genuine optimization difficulty may be suppressed, making it harder for the network to distinguish between an objective that is safely converging and one that is actively struggling.

To preserve this relative optimization signal, we introduce Historical Moving Average (HMA) weighting. HMA is a generalized, scale-invariant mechanism that adjusts the weight of each objective according to its own historical trajectory rather than through cross-objective normalization. The mechanism is independent of any specific loss function and can be applied to objective sets of varying size and composition. In this work, we validate its robustness by jointly optimizing three differentiable per-batch objectives spanning widely different numerical ranges: reconstruction, perceptual, and structural edge penalties.

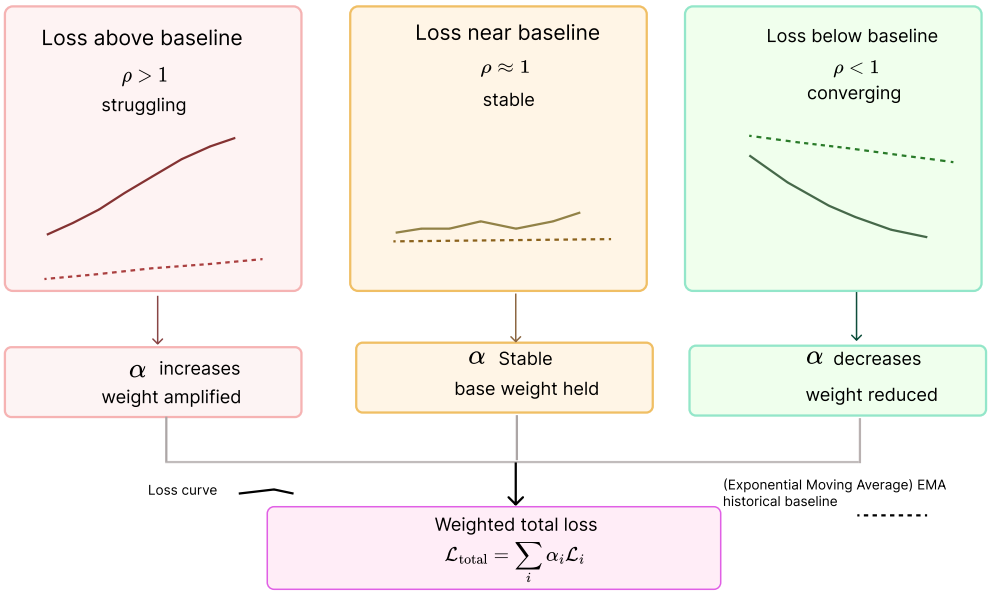


Figure 2: HMA adjusts each objective's dynamic weight $\alpha_i^{(t)}$ by comparing the current loss $\mathcal{L}_i^{(t)}$ against its own exponential moving average baseline $\bar{\mathcal{L}}_i^{(t-1)}$. When $\rho_i^{(t)} > 1$, the objective is above its historical baseline and receives an amplified weight, preventing starvation; when $\rho_i^{(t)} \approx 1$, the weight remains at its base value; and when $\rho_i^{(t)} < 1$, the objective is converging and its weight is reduced, preventing dominance. The adapted weights combine to form $\mathcal{L}_{\text{total}}$ shown in equation 6.

HMA operates over a general set of K differentiable per-batch objectives $\mathcal{T} = \{1, 2, \ldots, K\}$. The composite training objective is:

$$\mathcal{L}_{\text{total}}(\theta, t) = \sum_{i \in \mathcal{T}} \alpha_i^{(t)} \, \mathcal{L}_i(\theta, t), \tag{6}$$

where θ denotes the shared network parameters and $\alpha_i^{(t)}$ is the dynamic weight assigned to objective i at training step t. In this work, the specific objectives mapped to $\mathcal{T}$ comprise pixel-level reconstruction ($\mathcal{L}_{\text{rec}}$), and perceptual consistency ($\mathcal{L}_{\text{perc}}$) though the mechanism extends naturally to any arbitrarily composed objective set.

3.3.1 Gradient Dominance and Starvation

To understand the failure modes of standard joint optimization, consider the baseline in which the objective weights are fixed to static constants, $\alpha_i^{(t)} = c_i$. The parameter update at each step is governed by:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \sum_{i \in \mathcal{T}} c_i \, \mathbf{G}_i^{(t)}, \qquad \mathbf{G}_i^{(t)} = \nabla_\theta \mathcal{L}_i(\theta, t). \tag{7}$$

where η is the learning rate and $\mathbf{G}_i^{(t)}$ represents the task-specific gradient.

When $\|\mathbf{G}_j^{(t)}\|_2 \gg \|\mathbf{G}_i^{(t)}\|_2$ for some objective j, the shared update is dominated by that objective, potentially sacrificing the optimization trajectory of all remaining tasks. Conversely, an objective that inherently yields massive gradients may continue to dominate the parameter update long after it has converged, progressively starving the tasks that actually require a gradient signal. The adaptive weighting mechanism formulated in Section 3.3.2 is explicitly designed to neutralize these twin issues of *dominance* and *starvation*.

3.3.2 HMA Formulation

The instantaneous value of an objective is most informative when interpreted relative to that objective's own optimization history. For each objective $i \in \mathcal{T}$, we maintain an exponentially smoothed Historical Moving

Average of its loss trajectory:

$$\bar{\mathcal{L}}_i^{(t)} = \beta\, \bar{\mathcal{L}}_i^{(t-1)} + (1-\beta)\, \mathcal{L}_i^{(t)}, \qquad \beta \in [0, 1). \tag{8}$$

A lower β makes the baseline more responsive to recent batch fluctuations, whereas a higher β yields a more stable long-term approximation of convergence progress. In this work, we set $\beta = 0.99$. The history-normalized ratio is then defined as:

$$\rho_i^{(t)} = \frac{\mathcal{L}_i^{(t)}}{\bar{\mathcal{L}}_i^{(t-1)} + \varepsilon}, \qquad \varepsilon > 0. \tag{9}$$

The additive stabilizer ε is necessary in two regimes. At the beginning of training, initializing a moving average to zero would cause the ratio $\rho_i^{(t)}$ to become undefined. To achieve numerical stability at the beginning of training, the historical baseline is explicitly initialized to the first observed batch loss ($\bar{\mathcal{L}}_i^{(0)} = \mathcal{L}_i^{(0)}$), and a warmup period T_{warm} is applied to allow the EMA to accumulate a reliable trajectory. However, the additive stabilizer ε remains strictly necessary for late-stage training. As objectives start to converge successfully, their losses naturally approach zero ($\bar{\mathcal{L}}_i^{(t-1)} \to 0$). Without ε, residual batch-level noise in the numerator $\mathcal{L}_i^{(t)}$ would be divided by a near-zero denominator, artificially inflating $\rho_i^{(t)}$ and triggering spurious weight amplification for a task that has already converged. The stabilizer ε safely suppresses this failure mode.

The ratio $\rho_i^{(t)}$ acts as a self-referential difficulty signal. A value of $\rho_i^{(t)} > 1$ indicates that the objective is currently above its historical baseline and therefore requires increased optimization attention relative to its recent trajectory. Conversely, $\rho_i^{(t)} < 1$ indicates that the objective is below its historical baseline and is progressing relative to its recent trajectory. The unconstrained dynamic weight is computed from this ratio:

$$\hat{\alpha}_i^{(t)} = \alpha_i^{(\text{base})} \left(\rho_i^{(t)}\right)^{\gamma_i}, \qquad \gamma_i > 0, \tag{10}$$

where $\alpha_i^{(\text{base})}$ is the base weight for objective i and $\gamma_i > 0$ controls the aggressiveness of modulation for that objective. Since γ_i appears as an exponent, it controls the sensitivity of the dynamic weight to deviations between the current loss and its historical baseline. Although the formulation allows objective-specific modulation exponents, we use a shared value $\gamma_i = \gamma = 2.0$ for all objectives in our experiments, as reported in Section 4.2.

To prevent unbounded amplification during volatile early training phases, the dynamically computed weight is clipped within a predefined operational envelope:

$$\tilde{\alpha}_i^{(t)} = \max\Big(\alpha_{\min},\, \min\Big(\alpha_{\max},\, \hat{\alpha}_i^{(t)}\Big)\Big). \tag{11}$$

Scale-Preserving Normalization. A final failure mode arises when the clipped weights are summed directly to form the total loss. On datasets with high inter-batch scene variance, the raw loss values of individual objectives can fluctuate substantially across batches, causing the EMA baseline to follow a moving target. The resulting ratios $\rho_i^{(t)}$ may oscillate, and the modulation exponent can amplify those oscillations. As a result, the uncontrolled weighted sum may expand or collapse depending on which objective dominates a given batch.

To prevent this instability, we apply softmax normalization over the clipped weights and rescale by the sum of base weights:

$$\alpha_i^{(t)} = \frac{\exp\Big(\tilde{\alpha}_i^{(t)}\Big)}{\sum_{j\in\mathcal{T}} \exp\Big(\tilde{\alpha}_j^{(t)}\Big)} \cdot \sum_{j\in\mathcal{T}} \alpha_j^{(\text{base})}. \tag{12}$$

This two-step normalization decouples the relative priority signal from the absolute scale of the total loss. The softmax ensures that the weight vector lies on a bounded simplex, preventing any single objective from driving the total loss to diverge through weight amplification alone. The subsequent rescaling by $\sum_j \alpha_j^{(\text{base})}$

restores the aggregate loss magnitude specified by the initial base weights. This keeps the optimizer's effective learning rate consistent across training. Priority ordering is preserved: if $\tilde{\alpha}_i^{(t)} > \tilde{\alpha}_j^{(t)}$, then $\alpha_i^{(t)} > \alpha_j^{(t)}$.

The complete algorithm, summarized in Figure 2, requires minimal computational overhead. In implementation, HMA maintains one scalar state $\bar{\mathcal{L}}i^{(t-1)}$ per objective and updates $\alpha_i^{(t)}$ before forming the total loss in equation 6. A linear warmup period of T_{warm} steps is applied at the start of training, during which $\alpha_i^{(t)} = \alpha_i^{(\text{base})}$. This delay prevents the ratio from overreacting to uninformative early-loss fluctuations before the EMA baselines have accumulated sufficient history. The entire mechanism is confined to the training objective and does not alter the network architecture or inference procedure.

3.3.3 Optimization Dynamics

The HMA update rule provides an interpretable and self-correcting balancing mechanism. When an objective is above its historical baseline ($\rho_i^{(t)} > 1$), its clipped weight is increased, directing more optimization attention toward that objective and reducing the risk of starvation. When an objective falls below its historical baseline ($\rho_i^{(t)} < 1$), its weight is reduced, limiting its ability to dominate the shared update after it has entered a stable convergence phase. When the objective stabilizes around its baseline ($\rho_i^{(t)} \to 1$), the weight returns toward its base value. This behavior applies independently and symmetrically to every objective in $\mathcal{T}$, regardless of absolute numerical range, acting as a self-correcting negative feedback loop that maintains dynamic equilibrium across the full set of training objectives.

Crucially, because the scale factor of any objective largely cancels out in the numerator and denominator of $\rho_i^{(t)}$, two objectives with the same relative optimization trajectory but substantially different absolute scales yield comparable history-normalized ratios. No cross-objective normalization is performed before computing these ratios, and the magnitude variance that encodes genuine differences in optimization difficulty is preserved rather than collapsed. The two-step normalization in equation 12 then controls the total loss magnitude without destroying this per-objective relative priority. This mitigates instability on high-variance scene datasets while retaining the scale-invariant behavior of the HMA mechanism.

3.3.4 Scale Invariance and Bounded Influence

By construction, $\tilde{\alpha}_i^{(t)} \in [\alpha_{\min}, \alpha_{\max}]$ for all t and all i, as a direct consequence of the clipping operation in equation 11. Because the softmax in equation 12 is a strictly monotone, bounded transformation, the final weights $\alpha_i^{(t)}$ inherit bounded behavior from the clipped weights. This leads to the following proposition.

Proposition *(Bounded and History-Normalized Behavior of HMA). Let $\mathcal{L}_i(\theta)$ be bounded for all $i \in \mathcal{T}$, and let $0 < \alpha_{\min} < \alpha_{\max}$. The dynamic weight $\alpha_i^{(t)}$ defined by the HMA update rule satisfies:* ***(A)*** *bounded loss-weight influence, mitigating weight-induced dominance;* ***(B)*** *adaptive amplification of objectives whose losses increase relative to their own histories, and suppression of objectives whose losses decrease relative to their own histories; and* ***(C)*** *history-normalized, ratio-based difficulty estimation across objectives with heterogeneous loss magnitudes, which modulates their dynamic weights.*

Proof.

(A) Bounded gradient contribution. Since $\tilde{\alpha}_i^{(t)} \in [\alpha_{\min}, \alpha_{\max}]$ by equation 11, and the softmax in equation 12 maps bounded inputs to a bounded simplex, the final weight satisfies $\alpha_i^{(t)} \in (0, \alpha'_{\max}]$ for a finite constant $\alpha'_{\max}$ that depends only on $\alpha_{\max}$, $|\mathcal{T}|$, and $\sum_j \alpha_j^{(\text{base})}$. Consequently, the weight-scaled gradient contribution of any objective satisfies:

$$\left\|\alpha_i^{(t)} \mathbf{G}_i^{(t)}\right\|_2 \leq \alpha'_{\max} \left\|\mathbf{G}_i^{(t)}\right\|_2, \tag{13}$$

for all t and all $i \in \mathcal{T}$. No dynamic weight can grow without bounds, so weight-induced gradient dominance is directly mitigated. HMA bounds the scalar weight multiplier; the raw gradient magnitude $\|\mathbf{G}_i^{(t)}\|_2$ may still be large and is not directly bounded by this mechanism.

(B) Adaptive amplification and suppression. We analyze the two primary failure modes of multi-objective optimization separately.

Case 1 (Starvation prevention). Suppose task i is struggling such that its instantaneous loss exceeds its historical baseline: $\mathcal{L}_i^{(t)} > \bar{\mathcal{L}}_i^{(t-1)}$. By definition of equation 9, $\rho_i^{(t)} > 1$. Since $\gamma_i > 0$, raising $\rho_i^{(t)} > 1$ to the power γ_i preserves the inequality, giving $(\rho_i^{(t)})^{\gamma_i} > 1$, and therefore:

$$\hat{\alpha}_i^{(t)} = \alpha_i^{(\text{base})} \left(\rho_i^{(t)}\right)^{\gamma_i} > \alpha_i^{(\text{base})}. \tag{14}$$

As $\mathcal{L}_i^{(t)}$ continues to increase relative to its history, $\hat{\alpha}_i^{(t)}$ increases monotonically until clipped at $\alpha_{\max}$. This raises the objective's pre-normalized priority signal within the shared update, so an objective whose loss rises above its own historical baseline receives increased weighting, subject to clipping and final normalization.

Case 2 (Dominance prevention). Suppose task i converges such that its instantaneous loss falls below its historical baseline: $\mathcal{L}_i^{(t)} < \bar{\mathcal{L}}_i^{(t-1)}$. Then $\rho_i^{(t)} < 1$, and since $\gamma_i > 0$:

$$\hat{\alpha}_i^{(t)} = \alpha_i^{(\text{base})} \left(\rho_i^{(t)}\right)^{\gamma_i} < \alpha_i^{(\text{base})}. \tag{15}$$

While $\mathcal{L}_i^{(t)}$ remains below its historical baseline, $\hat{\alpha}_i^{(t)}$ is reduced below its base value, with the clipped weight remaining lower-bounded by $\alpha_{\min}$. Under this condition, the effective gradient contribution $\alpha_i^{(t)} \|\mathbf{G}_i^{(t)}\|_2$ is reduced through the lower dynamic weight. As optimization approaches a stationary regime and $\|\mathbf{G}_i^{(t)}\|_2$ decreases, the contribution is further reduced by the shrinking gradient magnitude. This helps reduce the risk that a converged objective continues to dominate the shared parameter update of the remaining, unresolved tasks.

(C) Scale invariance. Let two objectives follow identical relative optimization trajectories but differ in absolute magnitude by a constant factor $c > 0$, such that $\mathcal{L}_j^{(t)} = c\,\mathcal{L}_i^{(t)}$ for all t. Since the HMA state update in equation 8 is a linear operation, their historical moving averages scale identically: $\bar{\mathcal{L}}_j^{(t-1)} = c\,\bar{\mathcal{L}}_i^{(t-1)}$. The history-normalized ratio for objective j is therefore:

$$\rho_j^{(t)} = \frac{c\,\mathcal{L}_i^{(t)}}{c\,\bar{\mathcal{L}}_i^{(t-1)} + \varepsilon} = \frac{\mathcal{L}_i^{(t)}}{\bar{\mathcal{L}}_i^{(t-1)} + \frac{\varepsilon}{c}}. \tag{16}$$

For any strictly positive baseline $\bar{\mathcal{L}}_i^{(t-1)} > 0$, as the stabilizer constant $\varepsilon \to 0^+$, the term $\frac{\varepsilon}{c}$ becomes negligible. Consequently, the limit yields:

$$\rho_j^{(t)} \approx \frac{\mathcal{L}_i^{(t)}}{\bar{\mathcal{L}}_i^{(t-1)}} \approx \rho_i^{(t)} \tag{17}$$

Thus, the invariance is established at the level of the history-normalized difficulty signal ρ. This ratio then drives the multiplicative modulation of each objective's dynamic weight through equation 10. When the same exponent γ is used, the multiplicative modulation relative to each objective's base weight is approximately equivalent in practice when ε is small relative to the historical baseline. The final absolute weights may still differ because they retain task-specific base coefficients and are subsequently normalized by equation 12. Therefore, this result should be interpreted as scale-invariant ratio-based weight modulation rather than equality of final absolute weights.

The modulation exponent γ_i governs the aggressiveness of the weight amplification. Although the formulation permits objective-specific modulation exponents, we use a shared value $\gamma_i = \gamma = 2.0$ for all objectives in our experiments. This shared setting keeps the HMA configuration simple while providing stable modulation across the reconstruction, perceptual, and structural objectives, as reported in Section 4.2.

4 Empirical Setup

We evaluate GLOPaint in a high-resolution inpainting setting at 1024×1024 resolution. The experimental design is intended to assess both restoration quality and computational practicality under a controlled irregular-mask protocol. We first describe the datasets and masking procedure, then provide the implementation details, including architecture, training configuration, and the hyperparameters used for HMA-based objective balancing

4.1 Datasets and Masking Protocol

We evaluate GLOPaint on two high-resolution datasets covering natural scene and face inpainting settings. Places2 (Zhou et al., 2018) is used to assess generalization on complex scene-centric images, where a 50K subset is prepared at 1024×1024 resolution. CelebA-HQ (Karras et al., 2018) is used for structured face completion, using the full 30K image set at 1024×1024 resolution. Each dataset is partitioned into 70%/20%/10% splits for training, validation, and testing, respectively. This partition is used to provide a sufficiently large validation set for model selection and hyperparameter tuning under the high-resolution setting.

To improve robustness to different occlusion patterns, irregular masks are generated during training using random combinations of geometric primitives, including lines, circles, and rectangles. The mask ratio is uniformly sampled from $r \in [0.20, 0.50]$, matching the evaluation protocol used in the quantitative comparison. This exposes the model to varying levels of structural degradation and reduces overfitting to a fixed mask shape or occlusion pattern.

4.2 Implementation Details

This subsection summarizes the implementation configuration used in all experiments, including the architectural settings, graph reasoning parameters, training protocol, and hardware setup.

Architecture. GLOPaint is implemented in PyTorch. The convolutional encoder processes the concatenated masked image and binary mask, $\mathbf{X}_{\text{in}} \in \mathbb{R}^{B \times 4 \times H \times W}$, through progressive downsampling and dilated residual blocks. The spatial dimensions are reduced by a factor of 8, mapping the input to a compressed bottleneck representation with hidden channel dimension $C = 256$. The structural edge prior is incorporated at the graph-node level by concatenating the downsampled edge map $\mathbf{E}_b$ with the bottleneck features, as described in Section 3.2.2. Receptive-field expansion is further supported within the bottleneck using a sequence of residual blocks with symmetric dilations $d \in 2, 4, 8$.

Relational reasoning is performed using $L = 2$ Graph Attention (GAT) layers with $H = 4$ attention heads. Rather than relying on a fixed spatial grid, the GAT module dynamically connects each node to its $K = 8$ nearest neighbors in feature space, enabling adaptive long-range semantic routing. The graph-refined features are then fused with the original convolutional bottleneck through a learned sigmoid gating mechanism. Finally, a progressive convolutional decoder, using spatial skip connections from the encoder hierarchy, maps the features back to the full-resolution output. Instance normalization is applied across convolutional stages to mitigate mask-induced feature distribution shifts.

Training. Training proceeds in two stages to improve structural stability. First, the lightweight edge predictor is pretrained for 50 epochs using a spatially weighted Binary Cross-Entropy loss that prioritizes valid regions and mask boundaries. After pretraining, its weights are frozen. Second, the full generative pipeline is trained for 100 epochs with an effective batch size of 24. Optimization is performed using AdamW ($\beta_1 = 0.5$, $\beta_2 = 0.999$) with a weight decay of 1×10^{-4}. The learning rate is decayed using a linear scheduler toward the end of training. The HMA-specific dynamic weighting parameters are reported in Section 4.3. All experiments are conducted on a single NVIDIA RTX 5090 GPU using mixed-precision training (AMP) for memory efficiency.

4.3 HMA Hyperparameters

The HMA weighting mechanism is applied to balance the primary generator objectives: pixel-level reconstruction $\mathcal{L}_{\text{rec}}$ and perceptual consistency $\mathcal{L}_{\text{perc}}$. In L_1-dominant optimization, the network often regresses to a smooth, low-frequency approximation. To mitigate this, the base weights are deliberately biased to enforce high-fidelity texture generation: $\alpha_{\text{rec}}^{(\text{base})} = 0.2$ and $\alpha_{\text{perc}}^{(\text{base})} = 1.0$. The exponential smoothing parameter is set to $\beta = 0.99$.

The modulation exponent controls the sensitivity of the dynamic weight to deviations of the current loss from its historical baseline. These exponents are tuned to reflect the convergence dynamics of each task, sharing a value of $\gamma = 2.0$ for both reconstruction and perceptual metrics. When an objective diverges above its baseline, its weight is proportionally amplified, whereas converging objectives are suppressed. To enforce the intended perception-distortion tradeoff, dynamic weights are strictly bounded using objective-specific clipping constraints. The reconstruction weight is bounded to $\alpha_{\text{rec}} \in [0.1, 1.0]$, preventing it from dominating late-stage training, whereas the perceptual weight is permitted to scale within $\alpha_{\text{perc}} \in [0.5, 3.0]$.

Finally, a linear warmup of $T_{\text{warm}} = 500$ steps is applied at the start of training, during which all objectives retain their base weights. The HMA state is initialized using the first observed loss value for each objective, $\bar{\mathcal{L}}_i^{(0)} = \mathcal{L}_i^{(0)}$, to avoid unstable ratio estimates at the beginning of training.

5 Results and Analysis

This section evaluates GLOPaint in terms of reconstruction quality, perceptual similarity, structural consistency, and computational efficiency. We compare the proposed framework with representative high-resolution inpainting baselines under the same irregular-mask evaluation protocol.

5.1 Baselines

We compare GLOPaint with representative high-resolution image inpainting baselines that are suitable for assessing the quality–efficiency focus of this work. The comparison jointly considers reconstruction quality, perceptual fidelity, model size, and inference cost. Specifically, we evaluate against LaMa-Fourier (Suvorov et al., 2022), HiFill/CRA (Yi et al., 2020), and CoordFill (Liu et al., 2023). These methods provide relevant comparisons for irregular-mask high-resolution image completion and are compatible with the evaluation protocol used for GLOPaint.

LaMa-Fourier (Suvorov et al., 2022) is included as a strong convolutional inpainting baseline with effective long-range modeling. HiFill/CRA (Yi et al., 2020) is included as a high-resolution baseline designed for plausible completion under challenging missing regions. CoordFill (Liu et al., 2023) is included as a recent coordinate-aware method relevant to efficient high-resolution image completion. For each baseline, we use the official public implementation and released pretrained checkpoint where available, and evaluate all methods under the same masking protocol and quantitative metrics.

Transformer and diffusion-based inpainting methods are important directions in the broader literature. However, they follow substantially different computational assumptions from the lightweight high-resolution setting studied here. Transformer-based models often rely on attention-heavy global reasoning, whereas diffusion-based models generally require iterative sampling at inference time. Therefore, our main quantitative comparison focuses on baselines with more comparable quality–efficiency assumptions.

5.2 Evaluation Metrics

We evaluate all methods using four complementary full-reference metrics on Places2 (Zhou et al., 2018) and CelebA-HQ (Karras et al., 2018) at 1024×1024 resolution: L1 reconstruction error, Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) (Wang et al., 2004), and Learned Perceptual Image Patch Similarity (LPIPS) (Zhang et al., 2018). These metrics capture different aspects of reconstruction quality. L1 measures pixel-level reconstruction error, PSNR measures signal-level fidelity, SSIM evaluates structural similarity between the restored and reference images, and LPIPS measures perceptual similarity in

a learned feature space. Together, they provide a balanced quantitative evaluation of high-resolution image inpainting across both scene-centric and face-centric settings.

L1 reconstruction error measures the mean absolute pixel-wise difference between the inpainted output and the ground-truth image. It provides a direct estimate of reconstruction accuracy and is useful for assessing low-level consistency, particularly around masked regions and object boundaries. Lower L1 values indicate better pixel-level reconstruction.

Peak Signal-to-Noise Ratio (PSNR) measures full-reference reconstruction fidelity between the inpainted output and the ground-truth image. A higher PSNR indicates lower pixel-level distortion and stronger numerical similarity to the reference image. Although PSNR may not fully capture perceptual quality in ill-posed restoration tasks, it remains a standard measure of reconstruction fidelity.

Structural Similarity Index Measure (SSIM) evaluates the structural similarity between the restored image and the ground-truth image by considering luminance, contrast, and structural information. Compared with purely pixel-wise error measures, SSIM better reflects local structural consistency, which is important for image inpainting where boundary continuity and texture structure affect visual quality. Higher SSIM values indicate stronger structural agreement with the reference image.

Learned Perceptual Image Patch Similarity (LPIPS) measures perceptual similarity using distances in a learned deep feature space. Since image inpainting is inherently ill-posed and multiple visually plausible completions may exist for the same missing region, LPIPS complements pixel-wise and structural metrics by better reflecting perceptual agreement with the reference image. Lower LPIPS values indicate stronger perceptual similarity.

Overall, lower L1 and LPIPS values indicate better reconstruction accuracy and perceptual similarity, respectively, while higher PSNR and SSIM values indicate stronger full-reference fidelity and structural consistency.

5.3 Quantitative Results

Table 1 reports the quantitative comparison at 1024×1024 resolution on Places2 (Zhou et al., 2018) and CelebA-HQ (Karras et al., 2018). The evaluation uses irregular masks covering 20%–50% of the image area and compares performance across both scene-centric and face-centric inpainting settings. For the baselines, we use the official public implementations and released pretrained checkpoints when available, and evaluate all methods under the same masking protocol and quantitative metrics.

On Places2 (Zhou et al., 2018), GLOPaint achieves competitive L1 reconstruction error while obtaining the highest PSNR and the lowest LPIPS among the compared methods. Although CoordFill reports a slightly lower L1 value, GLOPaint shows stronger full-reference fidelity and perceptual similarity (higher PSNR, lower LPIPS) while maintaining comparable SSIM. These results suggest that the proposed edge- and feature-aware graph formulation supports visually consistent restoration in complex natural scenes.

On CelebA-HQ (Karras et al., 2018), GLOPaint achieves competitive PSNR and obtains the lowest LPIPS among the compared baselines. Although CoordFill reports lower L1 and higher SSIM values, GLOPaint improves perceptual similarity, suggesting that the proposed framework produces visually consistent face completions with strong perceptual agreement. This is important in the structured face-completion setting, where facial coherence, boundary continuity, and local appearance consistency strongly affect visual quality.

The efficiency results further support the compact and computationally efficient design of the proposed framework. As shown in Table 2, GLOPaint uses 13.6M parameters, giving it a moderate model size while remaining smaller than the larger baselines. GFLOPs are included to compare computational cost beyond parameter count, since models with similar parameter sizes can still differ in forward-pass complexity. GLOPaint requires 236.53 GFLOPs, which is the lowest among the compared methods. It also achieves the fastest inference time, with 2.9 ms/image compared with 49 ms, 5 ms, and 3.6 ms for the baseline methods.

Overall, these results suggest that GLOPaint provides an effective quality–efficiency trade-off for 1024×1024 image inpainting under a compact architectural design. The proposed method achieves strong reconstruction and perceptual restoration quality while maintaining a moderate parameter count and reporting the lowest GFLOPs and fastest inference time among the compared high-resolution baselines. Under the fixed training

Table 1: Quantitative comparison of GLOPaint with LaMa-Fourier, HiFill/CRA, and CoordFill for 1024 × 1024 image inpainting on Places2 and CelebA-HQ. We report L1 reconstruction error, PSNR, SSIM, and LPIPS. Higher PSNR and SSIM indicate better reconstruction fidelity and structural similarity, while lower L1 and LPIPS indicate better reconstruction accuracy and perceptual quality.

Method	**Places2** (1024 × 1024)				**CelebA-HQ** (1024 × 1024)			
	L1 ↓	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**	**L1 ↓**	**PSNR ↑**	**SSIM ↑**	**LPIPS ↓**
LaMa-Fourier (**?**)	–	26.154	**0.924**	0.076	–	26.098	0.924	0.080
HiFill / CRA (Yi et al., 2020)	0.052	23.456	0.894	0.096	–	–	–	–
CoordFill (Liu et al., 2023)	**0.021**	26.322	0.920	0.075	**0.003**	**28.756**	**0.934**	0.065
GLOPaint	0.022	**26.803**	0.8959	**0.068**	0.012	27.372	0.8978	**0.0593**

Table 2: Computational efficiency comparison of GLOPaint with LaMa-Fourier, HiFill/CRA, and CoordFill for 1024 × 1024 image inpainting. We report parameter count, GFLOPs, and inference time per image. Parameter count reflects model size, GFLOPs measure computational complexity for a single 1024 × 1024 forward pass, and inference time is reported in milliseconds per image. Lower values indicate better efficiency.

Method	**# Params ↓**	**GFLOPs ↓**	**Inference time ↓**
	M	**GFLOPs**	**ms/image**
LaMa-Fourier (**?**)	27	512.24	49
HiFill / CRA (Yi et al., 2020)	**2.7**	288.48	5
CoordFill (Liu et al., 2023)	39.6	242.31	3.6
GLOPaint	13.6	236.53	2.9

protocol, GLOPaint maintains competitive restoration quality on the 50K Places2 subset and the 30K CelebA-HQ image set across both scene-centric and face-centric settings. These findings indicate that the proposed architecture offers a practical balance between restoration performance and computational efficiency without increasing model complexity.

5.4 Qualitative Results

Figures 3 and 4 present qualitative comparisons with LaMa-Fourier (**?**), HiFill/CRA (Yi et al., 2020), and CoordFill (Liu et al., 2023) on Places2 (Zhou et al., 2018) and CelebA-HQ (Karras et al., 2018) at 1024×1024 resolution. All methods are evaluated under the same masked-input and irregular-mask protocol used in the quantitative evaluation.

In the Places2 examples, the compared baselines produce plausible high-resolution completions, but visible differences remain in texture-rich regions, object boundaries, and larger missing areas. HiFill/CRA can produce local texture inconsistencies near masked boundaries, while LaMa-Fourier tends to smooth fine details in texture-dense restored regions. CoordFill produces visually sharp completions in several cases, although some outputs show reduced structural continuity around boundary-sensitive regions. In comparison, GLOPaint produces restored regions with more consistent appearance continuity and fewer visible boundary discontinuities in the shown examples.

In the CelebA-HQ examples, GLOPaint better preserves facial contour continuity and local appearance cues across the completed regions. Compared with the baselines, the restored facial regions show smoother boundary transitions and fewer visible artifacts around mask edges in the shown examples. These observations suggest that the edge- and feature-aware graph formulation supports structure-aware restoration in high-resolution face completion, including cases with larger masked regions.

Overall, the qualitative results are consistent with the quantitative trends. GLOPaint does not dominate every visual case, but it provides competitive high-resolution restoration quality while maintaining a compact and efficient architecture.

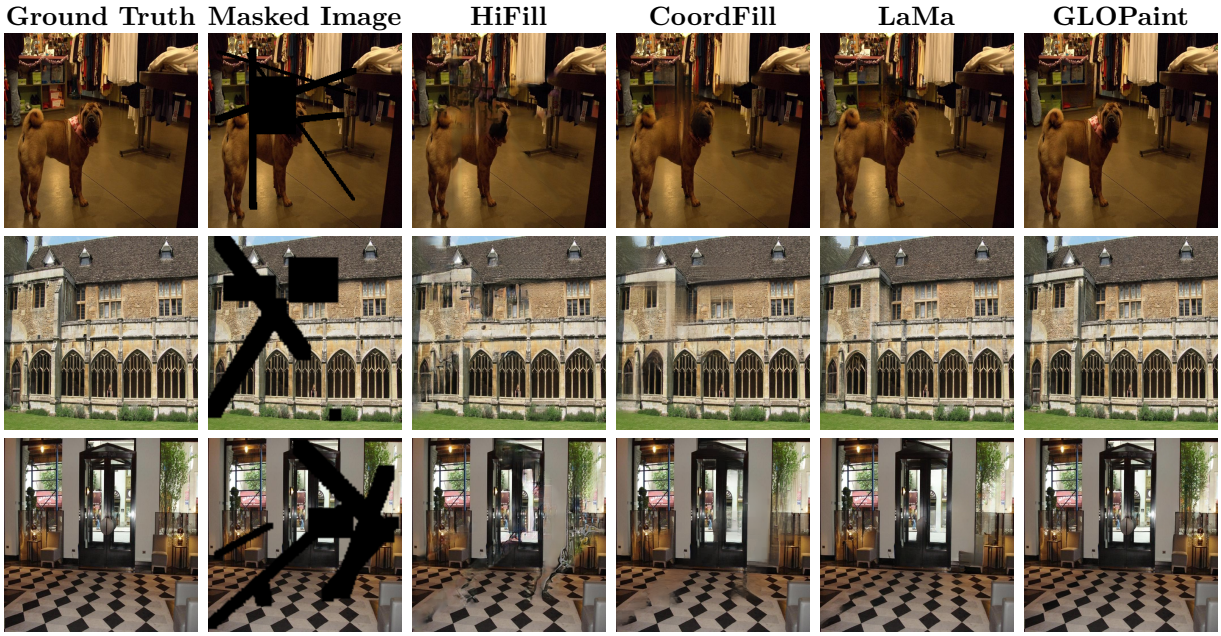


Figure 3: Qualitative comparison on Places2 (Zhou et al., 2018) scene image inpainting at 1024×1024 resolution. We compare the inpainting results of HiFill/CRA (Yi et al., 2020), CoordFill (Liu et al., 2023), LaMa (Suvorov et al., 2022), and the proposed GLOPaint model against the corresponding ground truth and masked input. GLOPaint produces more visually coherent completions, with smoother transitions around masked regions and better preservation of surrounding scene structure and texture.

6 Ablation Study

We evaluate the contribution of HMA weighting through a controlled comparison against fixed static loss weights. Both variants use the same architecture, loss terms, training schedule, mask generation strategy, and base coefficients $\alpha_i^{(\text{base})}$; the only difference is the loss-weighting rule. This design isolates the effect of history-normalized adaptive weighting from other architectural and training factors. We analyze the resulting training dynamics, validation behavior, and visual boundary consistency in Sections 6.1–6.3.

6.1 Loss Trajectory Analysis

Figure 5 compares the training trajectories of $\mathcal{L}$rec and $\mathcal{L}$perc under static weighting and HMA. The faint traces show raw per-step values, while the bold curves show exponentially smoothed trends for readability. Under static weighting, the reconstruction objective dominates the optimization process. With base weight $\alpha_{\text{rec}}^{(\text{base})} = 10.0$ compared with $\alpha_{\text{perc}}^{(\text{base})} = 1.0$, the reconstruction loss receives substantially greater influence in the shared update. As a result, $\mathcal{L}_{\text{perc}}$ shows high-variance fluctuations and limited downward progress during the early-to-mid training phase. Under HMA, both objectives exhibit more consistent downward trends. This suggests that per-objective re-weighting mitigates the imbalance by adapting each objective's coefficient relative to its own historical trajectory.

6.2 Adaptive Weight Dynamics and Convergence

Figure 6 relates the HMA coefficient dynamics to downstream validation fidelity. The plotted coefficients illustrate how HMA adjusts the relative influence of the training objectives during optimization. The α trajectories in Figure 6(a) are consistent with the intended behavior of HMA. After warmup, α_{rec} settles into an operating range of approximately 0.55–0.85 under $\beta = 0.99$ and $[\alpha_{\min}, \alpha_{\max}] = [0.1, 10.0]$. It exhibits transient spikes on batches where $\mathcal{L}_{\text{rec}}$ exceeds its historical average, and then recedes as the EMA baseline catches up. α_{perc} remains stable near its base value throughout training, reflecting the increasing relative

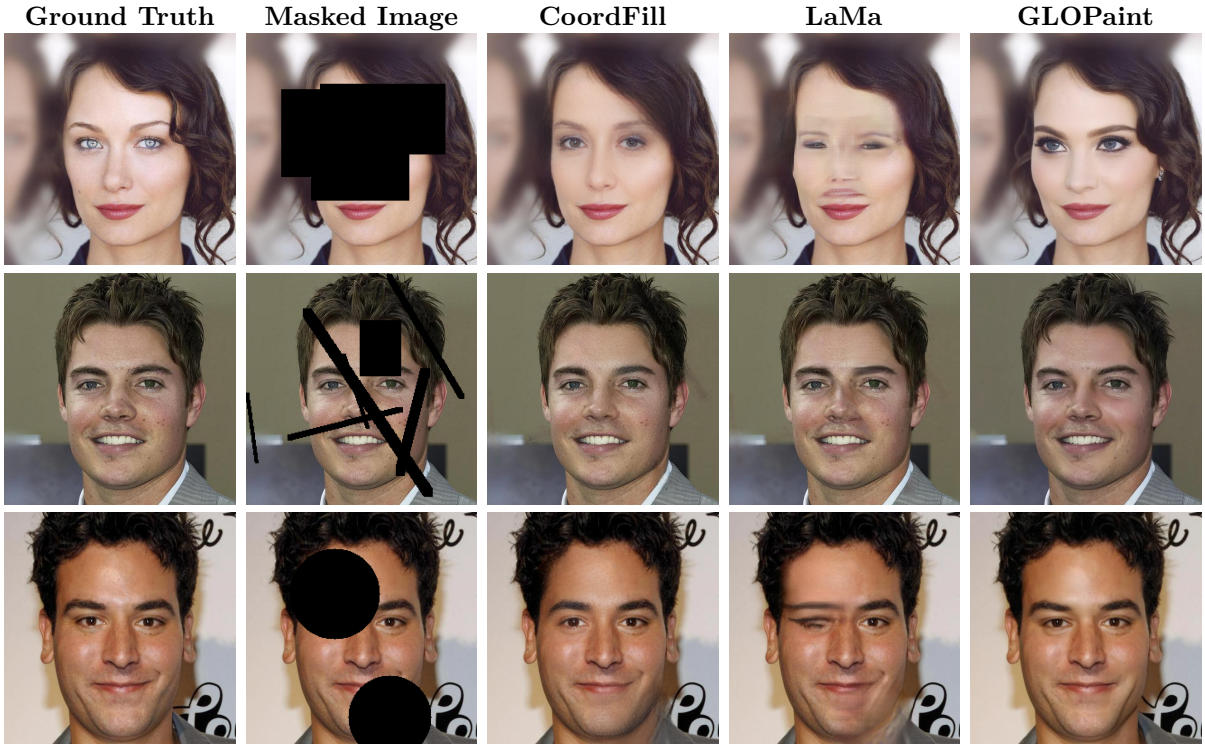
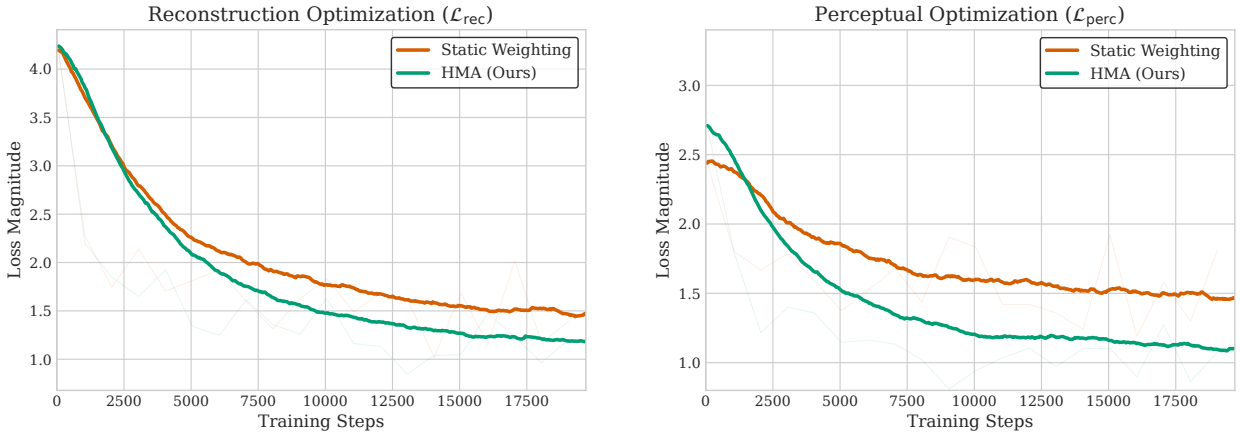


Figure 4: Qualitative comparison on CelebA-HQ (Karras et al., 2018) at 1024 × 1024 resolution. We compare the inpainting results of CoordFill (Liu et al., 2023), LaMa (Suvorov et al., 2022), and the proposed GLOPaint model against the corresponding ground truth and masked input. GLOPaint produces more structurally coherent and visually consistent restorations, especially around facial details and mask boundaries.



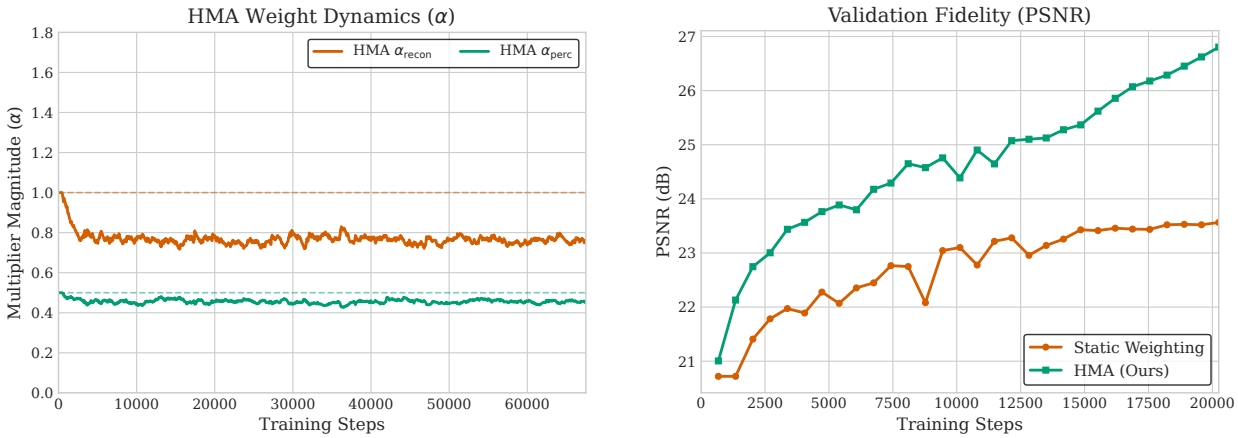
(a) Reconstruction Loss ($\mathcal{L}_{\text{rec}}$)

(b) Perceptual Loss ($\mathcal{L}_{\text{perc}}$)

Figure 5: Training trajectories under static weighting and HMA. Bold curves are exponentially smoothed; faint traces show raw per-step values. Under static weighting, the 10× higher base weight of $\mathcal{L}_{\text{rec}}$ suppresses the perceptual signal. HMA produces consistent downward trends in both objectives.

difficulty of the structural objective as overall reconstruction improves and the remaining error concentrates around boundary regions.

Figure 6(b) shows the corresponding validation PSNR trajectory. Static weighting produces pronounced oscillations and plateaus near 23.5639 dB, whereas HMA converges to 26.8031 dB with reduced variance, yielding a gain of +3.23 dB. This trend is consistent with improved coordination among the training objectives under HMA.



(a) HMA weight dynamics (α_{rec} and α_{perc}).

(b) Validation PSNR

Figure 6: Mechanism and outcome of HMA adaptation. **(a)** Adaptive HMA coefficients for the two reported objectives, α_{rec} and α_{perc}. α_{rec} exhibits a spike-and-recede pattern on difficult batches, while α_{perc} remains comparatively stable during training. **(b)** Validation PSNR trajectory under static weighting and HMA. HMA shows smoother convergence behavior and reduced oscillation compared with static weighting.

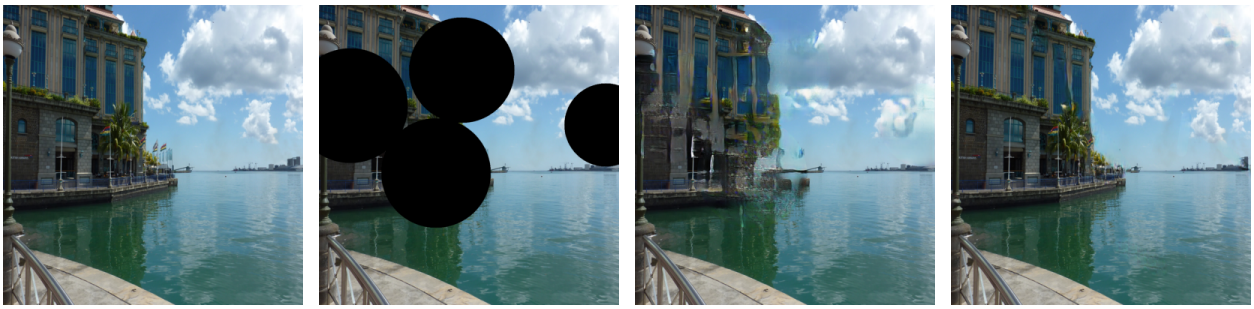
6.3 Qualitative Boundary Consistency

We further examine whether the improved loss coordination under HMA is reflected in boundary-consistent visual reconstruction. Figure 7 shows a representative high-resolution inpainting example using the same edge–feature graph architecture. Under static weighting, the reconstruction objective can dominate the shared update, reducing the effective contribution of perceptual and structural signals. This leads to completions with weaker boundary definition and reduced structural coherence. In contrast, HMA coordinates the reconstruction, perceptual, and structural objectives more effectively, helping the model align texture restoration with the surrounding boundary structure. The qualitative example suggests that the improved visual consistency is associated with the optimization balance maintained by HMA across the training objectives.

7 Conclusions

In this paper, we introduced GLOPaint, a lightweight hybrid framework for efficient high-resolution image inpainting. The framework addresses two closely related challenges: preserving structural coherence in missing regions and keeping training and inference practical under constrained computational resources. GLOPaint combines edge-aware convolutional feature extraction with graph attention reasoning at a compact bottleneck, enabling appearance features and predicted boundary cues to interact through a feature-space graph rather than dense full-resolution relational modeling. This design supports structure-aware information exchange between visually related regions while maintaining a compact computational footprint.

The effectiveness of GLOPaint is further supported by Historical Moving Average (HMA) loss weighting, a history-normalized optimization mechanism for balancing heterogeneous training objectives. By adapting each objective's weight according to its own recent optimization trajectory, HMA reduces dependence on fixed loss coefficients and helps prevent high-magnitude reconstruction losses from suppressing perceptual and structural signals. Experiments on Places2 and CelebA-HQ at 1024×1024 resolution show that GLOPaint achieves strong perceptual quality, competitive reconstruction fidelity, and favorable efficiency compared



(a) Original image (b) Masked image (c) Static weight inpainting (d) Inpainting with HMA

Figure 7: Qualitative comparison between static weight and HMA-based inpainting under the same edge-prior and feature-based graph architecture. The static-weighting variant produces a less coherent completion, while the HMA-trained model shows improved visual consistency.

with representative high-resolution inpainting baselines. The framework maintains a compact parameter budget and low inference latency, offering a practical quality–efficiency trade-off for deployment-oriented high-resolution inpainting, particularly in settings where training data and GPU resources are limited.

GLOPaint also suggests several directions for future work. Since graph reasoning is performed over compressed bottleneck features, the synthesis of extremely fine local textures may remain limited by the capacity of the convolutional decoder. Future extensions could explore learned or content-adaptive graph topologies beyond fixed k-NN connectivity, allowing more flexible long-range correspondence. Another promising direction is to make HMA more autonomous by adapting its clipping bounds or modulation strength during training. Extending the framework to temporally consistent video inpainting and edge-device deployment would further test the generality of compact feature-space graph reasoning for efficient structure-aware generative restoration.

References

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? In *International Conference on Learning Representations (ICLR)*, 2022.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. Proceedings of the International Conference on Machine Learning (ICML), 2018.

Mauricio Delbracio and Peyman Milanfar. Inversion by direct iteration: An alternative to denoising diffusion for image restoration. *Transactions on Machine Learning Research*, 2023.

Qiaole Dong, Chenjie Cao, and Yanwei Fu. Incremental transformer structure enhanced image inpainting with masking positional encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11358–11368, 2022.

Siqi Hui, Sanping Zhou, Ye Deng, Wenli Huang, and Jinjun Wang. Auxiliary loss reweighting for image inpainting. *arXiv preprint arXiv:2111.07279*, 2021.

Xuan Ju, Xian Liu, Xintao Wang, Yuxuan Bian, Ying Shan, and Qiang Xu. Brushnet: A plug-and-play image inpainting model with decomposed dual-branch diffusion. In *Computer Vision – ECCV 2024*, Lecture Notes in Computer Science, pp. 150–168, 2024.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018.

Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 23593–23606, 2022.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Keunsoo Ko and Chang-Su Kim. Continuously masked transformer for image inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13169–13178, 2023.

Wenbo Li, Zhe Lin, Kun Zhou, Lu Qi, Yi Wang, and Jiaya Jia. Mat: Mask-aware transformer for large hole image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10758–10768, 2022.

Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, Andrew Tao, and Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Shikun Liu, Edward Johns, and Andrew J. Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Weihuang Liu, Xiaodong Cun, Chi-Man Pun, Menghan Xia, Yong Zhang, and Jue Wang. Coordfill: Efficient high-resolution image inpainting via parameterized coordinate querying. *arXiv preprint arXiv:2303.08524*, 2023.

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11461–11471, 2022.

Kamyar Nazeri, Eric Ng, Tony Joseph, Faisal Qureshi, and Mehran Ebrahimi. Edgeconnect: Generative image inpainting with adversarial edge learning. *arXiv preprint arXiv:1901.00212*, 2019.

Marcelo Sanchez, Gil Triginer, Ignacio Sarasua, Lara Raad, and Coloma Ballester. Rethined: A new benchmark and baseline for real-time high-resolution image inpainting on edge devices, 2025.

Andranik Sargsyan, Shant Navasardyan, Xingqian Xu, and Humphrey Shi. Mi-gan: A simple baseline for image inpainting on mobile devices. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7335–7345, 2023.

Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2149–2159, 2022.

Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3614–3633, 2021.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *International Conference on Learning Representations (ICLR)*, 2018.

Shashikant Verma, Aman Sharma, Roopa Sheshadri, and Shanmuganathan Raman. Graphfill: Deep image inpainting using graphs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4996–5006, 2024.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. doi: 10.1109/TIP.2003.819861.

Zili Yi, Qiang Tang, Shekoofeh Azizi, Daesik Jang, and Zhan Xu. Contextual residual aggregation for ultra high-resolution image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7508–7517, 2020.

Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S. Huang. Free-form image inpainting with gated convolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4471–4480, 2019.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Yanhong Zeng, Jianlong Fu, Hongyang Chao, and Baining Guo. Aggregated contextual transformations for high-resolution image inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

Jianhui Zhang, Sheng Cheng, Qirui Sun, Jia Liu, Luyang Wang, Chaoyu Feng, Chen Fang, Lei Lei, Jue Wang, and Shuaicheng Liu. Ultra high-resolution image inpainting with patch-based content consistency adapter. *arXiv preprint arXiv:2510.13419*, 2025.

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 (CVPR)*, pp. 586–595, 2018.

Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(6): 1452–1464, 2018. doi: 10.1109/TPAMI.2017.2723009.

Junhao Zhuang, Yanhong Zeng, Wenran Liu, Chun Yuan, and Kai Chen. A task is worth one word: Learning with task prompts for high-quality versatile image inpainting, 2023.