Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

TriMagic: Single-Image 3D Generation Integrating Semantic-Role Multi-Token Conditioning and Dynamic Pseudo-Multi-View Supervision

Min Pang
North University of China
Xiaokang Cui
North University of China
Shichao Jiao
North University of China
Yajun Li
North University of China
Fengguang Xiong
hopenxfg@nuc.edu.cn
North University of China

Article

Keywords: digital heritage display, single-image 3D generation, semantic-role multi-token, dynamic pseudo-multi-view supervision, Stable Diffusion, Zero123++

Posted Date: November 9th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11148280/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

TriMagic: Single-Image 3D Generation Integrating Semantic-Role Multi-Token Conditioning and Dynamic Pseudo-Multi-View Supervision

Min Pang[1†], Xiaokang Cui[1†], Shichao Jiao[1], Yajun Li[1], Fengguang Xiong[1*]

[1*]School of Computer Science and Technology, North University of China, No. 3 Xueyuan Road, Taiyuan, 030051, Shanxi, China.

*Corresponding author(s). E-mail(s): hopenxfg@nuc.edu.cn;
Contributing authors: Pmin@nuc.edu.cn; cxk_20010120@163.com;
†These authors contributed equally to this work.

Abstract

Generating 3D models from single images supports digital heritage displays, but unseen regions are unconstrained, causing Janus artifacts, cross-view conflicts, and inconsistent textures. We propose TriMagic, a single-image 3D generation framework combining semantic-role multi-tokens with dynamic pseudo-multi-view supervision. Three learnable tokens, guided toward shape, color, and texture by role-specific prompts, contrastive alignment, and residual diversity constraints, form complementary instance representations. Zero123++ provides pseudo views; against their generation errors, empirical view weights, three-stage scheduling after warm-up, and separate absolute weighting of real and pseudo views refine the generation loss. Anchored dual-view sampling pairs the real reference with one random pseudo view in each pseudo-supervision update, while a Stable Diffusion prior and multi-reference-weighted Zero123 guidance constrain novel views. On NeRF4, RealFusion15, and our Artifact, PSNR improves over Magic123 from 24.62/19.50/27.58 to 24.75/20.01/28.99 dB; Artifact LPIPS rises from 0.03 to 0.04, with gains in semantic similarity and reference-view fidelity.

Keywords: digital heritage display;single-image 3D generation;semantic-role multi-token;dynamic pseudo-multi-view supervision;Stable Diffusion; Zero123++;

1 Introduction

Single-image 3D generation creates 3D content from existing 2D images and can serve online museum displays, e-commerce, AR interaction, and virtual scene construction. Conventional manual modeling, laser scanning, and photogrammetry require consideration of accuracy, acquisition conditions, surface properties, and budget[1]. Automatically generating 3D models with high visual quality and multiview consistency from a single image offers a technical approach to lowering the barriers to content creation.

This problem has particular practical value in cultural heritage digitization. Digitizing cultural heritage objects supports professional tasks such as heritage information preservation, archaeological research, and long-term archive management, while also serving digital museum displays, educational dissemination, and interactive public experiences. Digitization for scientific archiving and conservation research requires verifiable records of geometry and appearance, for which photogrammetry and 3D scanning are established and important approaches[1]. This study considers the input condition in which only a single collection catalog image, exhibition photograph, or historical archival image is available. In this setting, the absence of complete multiview geometric constraints and observations of unseen regions makes it difficult to directly meet the data requirements of conventional multiview reconstruction. Generating visually plausible 3D models from a single image thus offers a technical approach to using existing collection images for digital display. We focus on preserving instance appearance and the visual quality of generated models. Unseen regions are inferred from generative priors, and their geometric authenticity cannot be guaranteed by a single input image.

Recent advances in diffusion models and neural 3D representations have driven rapid progress in single-image 3D generation. Existing methods typically use the input image to provide constraints from real observations and employ text priors or view-conditioned diffusion models to generate information about unobserved views, which is then used to optimize NeRFs, meshes, or other 3D representations[2, 3]. By transforming 2D image generation into a 3D optimization problem jointly driven by real observations and generative priors, these methods have made substantial progress in model completeness and visual quality. Nevertheless, a single 2D image inherently cannot provide complete information about an object's spatial structure. Models must rely on prior knowledge learned from training data to infer the sides, rear surface, and occluded regions. Although this generative "hallucination" process can supplement missing information, it may also introduce discrepancies between the generated result and the real object, including geometric errors, texture drift, and cross-view inconsistency. For objects with distinct instance attributes, such as cultural heritage objects and artworks, generated results must not only conform to category characteristics but also preserve the decorative pattern layout, color distribution, material appearance, and local structural details of the specific object. Improving instance preservation in single-image 3D generation is therefore an important challenge when applying these methods to complex objects.

This challenge primarily concerns supervision reliability and the representation of instance conditioning. First, view-conditioned diffusion priors and their generated

views[3, 4] are not real observations. For asymmetric structures, complex decorative patterns, or unusual materials, generative models may incorrectly replicate features of the input view in unseen regions, producing Janus artifacts, local structural conflicts, and texture inconsistency. If generated views and the real input are treated as equivalent supervision, errors may accumulate during optimization. Their roles therefore need to be distinguished, and the contribution of pseudo-views needs to be controlled dynamically. Second, natural language cannot easily describe fine-grained instance attributes precisely. Although textual inversion can introduce object information through learnable tokens, a single token must simultaneously represent attributes such as shape, color, and texture[5]. The lack of a clear division of roles may affect the preservation of complex details. We therefore consider both the controlled use of generated supervision and the structured organization of instance conditioning.

To address these issues, we propose TriMagic, which combines semantic-role multi-token conditioning and dynamic pseudo-multiview supervision within coarse-to-fine 3D optimization driven by diffusion priors. The former constructs complementary instance conditioning using shape, color, and texture roles. The latter uses views generated by Zero123++ as weak supervision, controls their contribution through dynamic weighting and anchored dual-view sampling, and continuously retains constraints from the real input.

The main contributions of this study are as follows:

(1) We propose semantic-role multi-token conditioning for single-image 3D optimization. Starting from a shared object semantic anchor, role-specific prompts, role contrastive alignment of attribute residuals, and diversity constraints organize shape, color, and texture information to provide complementary instance conditioning for random novel views.

(2) We develop a dynamic pseudo-multiview supervision mechanism that combines empirical base weights for views, weight scheduling after warm-up, absolute weighting of real and pseudo views without a shared denominator, and dual-view sampling that includes the real reference image in every pseudo-supervision update. This mechanism controls the supervisory contribution of generated views while retaining the constraints from the real reference image.

(3) We integrate this instance conditioning and supervision mechanism into TriMagic, a framework based on the coarse-to-fine optimization workflow of Magic123[3]. Comparative and ablation analyses on NeRF4, RealFusion15, and Artifact examine semantic similarity, reconstruction quality at the reference view, and generated rear-view appearance, providing a method and experimental evidence for the digital display of cultural heritage objects.

2 Related Work

2.1 Text-Driven 3D Generation and Personalized Conditioning Representations

Text-driven 3D generation typically uses pretrained text-to-image diffusion models to provide 2D visual priors. Poole et al. proposed DreamFusion[6], which applies diffusion

model gradients to NeRF optimization through score distillation sampling, alleviating the shortage of paired text–3D data. However, it still has limitations, including time-consuming optimization, unstable geometry, and inconsistent appearance across views. Lin et al. proposed Magic3D[7], which uses a coarse-to-fine strategy from a low-resolution neural field to a high-resolution textured mesh to improve generation resolution and optimization efficiency. Nevertheless, these methods primarily rely on text descriptions of category, material, and style, making it difficult to precisely constrain the decorative pattern layout and local form of a specific cultural heritage object.

To improve the ability of diffusion models to represent specific instances, Gal et al. proposed textual inversion, which learns new concept embeddings to introduce a reference object into a frozen pretrained diffusion model[5]. Ruiz et al. proposed DreamBooth, which binds a unique identifier to a specific object and fine-tunes the diffusion model for subject-driven image generation[8]. These studies established a foundation for instance personalization, but single-token conditioning does not itself prescribe distinct roles for shape, color, and texture.

Attribute-based conditioning representations have also been studied. ProSpect learns multiple sets of text embeddings across diffusion denoising stages to control attributes such as material, style, and layout[9]. MATTE jointly considers network layers and denoising time steps, learns separate tokens for color, object, layout, and style, and introduces attribute constraints[10]. ColorPeel learns color concepts using samples of multiple geometric shapes in a specified color, color/shape tokens, and cross-attention alignment[11]. Thus, neither the use of multiple tokens nor the division of attributes is unique to this study. Our specific design keeps the diffusion backbone frozen, uses a shared object word embedding as an anchor, applies role contrastive alignment and diversity constraints to the attribute residuals of three tokens, and uses the joint conditioning for single-image 3D optimization. This design differs from mechanisms that allocate conditioning by network layer or denoising stage and does not require additional training images combining colors and shapes. The current experiments examine the effects of our modules but do not directly compare the above attribute inversion methods. We therefore do not claim superiority over these methods or strict attribute disentanglement on this basis.

2.2 Single-View Image-Driven 3D Generation

Single-view 3D generation is underdetermined because observations of the sides, rear surface, and occluded regions are unavailable. Early methods rely on category-level shape priors. For example, Pixel2Mesh[12] progressively deforms an initial mesh through graph convolutions to predict a triangular mesh from a single image. Its predictions for unseen regions depend on the shape distributions of the training categories, limiting generalization to out-of-distribution forms and instance-specific details.

With advances in diffusion models, view-conditioned generation has become an important means of supplementing information missing from a single view. Liu et al. proposed Zero-1-to-3[4], which introduces relative camera pose conditioning into a pretrained image diffusion model to generate novel images at specified views. However, the target views are relatively independent, making them susceptible to

texture drift, changes in local form, and cross-view inconsistency. To address this issue, Liu et al. proposed SyncDreamer[13], which improves consistency among generated views through cross-view feature interactions. Shi et al. subsequently proposed Zero123++[14], which uses a fixed camera layout and joint multiview generation to obtain images with explicit view relationships. These methods improve coordination among generated views, but their primary goal is to generate visually plausible novel views. Cross-view consistency does not imply consistency with the input instance, and the generated content may still contain conjectured structures and texture deviations in unseen regions.

For 3D optimization driven by diffusion priors, Qian et al. proposed Magic123, which jointly applies a text-to-image diffusion prior and the Zero-1-to-3 view-conditioned prior to 3D optimization, combining textual inversion, supervision from the real reference view, and a coarse-to-fine optimization workflow from NeRF to a DMTet mesh[3]. We adopt this backbone and further develop role-specific instance conditioning and a supervision mechanism for the pseudo-reference set. Thus, the combination of 2D/3D priors, the coarse-to-fine transition between representations, and the retention of real reference-view supervision are all established foundations. Our improvements focus on the organization of conditioning, pseudo-view weighting, and sampling.

The dynamic use of priors has also been studied. Consistent123 first uses a 3D prior to optimize geometry, determines the stage transition through CLIP-based state detection, and then gradually increases the influence of the 2D prior[15]. After shaping the geometry, DreamCraft3D uses multiview renderings to fine-tune a personalized diffusion model and alternately updates the generative prior and 3D content to enhance texture[16]. TriMagic keeps the pretrained diffusion backbone frozen and primarily schedules the explicit supervision losses of a fixed generated reference set. Their contribution is controlled through empirical view weights, scheduling based on training progress, absolute weighting of real and pseudo views without a shared denominator, and retention of the real anchor in pseudo-supervision updates. This mechanism differs from adjusting the relative contributions of 2D/3D diffusion priors or alternately fine-tuning a diffusion model. The current experiments do not directly compare these scheduling methods, and a difference in mechanism does not imply a performance advantage.

2.3 Multiview Image-Driven 3D Reconstruction

Multiview reconstruction uses real images to establish geometric correspondences across views. The incremental SfM pipeline in COLMAP[17] improves the robustness of reconstruction from unordered images through feature matching, image registration, and triangulation. Combined with multiview stereo, it can recover relatively accurate structures when view coverage is sufficient, image overlap is good, and features are stable. Reflective surfaces, weak textures, and repetitive decorative patterns may impair matching.

Neural representation methods have further improved multiview scene modeling and novel-view synthesis. Mildenhall et al. proposed NeRF[18], which uses a continuous volumetric radiance field and differentiable volume rendering to optimize a

scene representation from multiple images with known camera poses. Niemeyer et al. proposed RegNeRF[19], which improves novel-view synthesis under sparse-view conditions by imposing regularization on unobserved views. Kerbl et al. proposed 3D Gaussian Splatting[20], which uses an explicit 3D Gaussian representation for high-quality real-time rendering, improving the efficiency of multiview scene optimization and rendering. These methods can obtain relatively stable 3D representations when multiview input and camera pose requirements are met, but their performance still depends on the view coverage and observation quality of real images.

A single collection image generally does not meet these data requirements. Although generated pseudo-multiview images can supplement camera-space cues, they cannot replace real observations. We therefore use pseudo-views with weak supervision weights that depend on the view and vary with optimization progress, combining them with structured instance conditioning for 3D optimization.

3 Methods

3.1 Network Framework

To address the limited representation of instance features in a single image and the lack of observational constraints for unseen regions, we develop the TriMagic framework, as shown in Figure 1. The framework adopts the coarse-to-fine 3D optimization workflow of Magic123[3], from NeRF[18] to DMTet[21], and introduces semantic-role multi-token conditioning and dynamic pseudo-multiview supervision. Input preparation comprises instance token learning, pseudo-reference view generation with Zero123++, and preprocessing of real and pseudo-reference images, producing instance text conditioning and supervision for color, masks, relative depth, and normals. The first stage optimizes the main geometry and low-resolution appearance using a NeRF representation. The second stage converts this representation into a DMTet mesh and refines high-resolution geometry and texture. Both stages distinguish between supervision at known views and guidance at random novel views. The former controls the contribution of pseudo views through dynamic weighting and retention of the real anchor, whereas the latter jointly uses instance semantic conditioning in Stable Diffusion and the view-conditioned prior of Zero123 with multiple references. The following sections describe input preprocessing, multi-token conditioning learning, pseudo-view weight design, and the two optimization branches in turn.

During input preprocessing, we first use CarveKit to remove the image background and obtain a foreground RGBA image. RGBA comprises RGB color channels and an additional alpha transparency channel; we extract a binary foreground mask from the alpha channel. We then use MiDaS[22] to estimate relative depth and estimate surface normals using pretrained weights from Omnidata[23] in a Dense Prediction Transformer[24]. Relative depth constrains the front-to-back relationships of the object surface, whereas normals provide information about local geometric orientations. Both are geometric priors estimated by monocular models rather than measured geometric ground truth. We apply the same preprocessing to the real reference image and the six pseudo-views generated by Zero123++, forming a fixed supervision set for known views, $\mathcal{V} = \{\mathrm{I}_i, \mathrm{M}_i, \mathrm{D}_i, \mathrm{N}_i\}_{\mathrm{i}=0}^{6}$, where I_0 denotes the real reference view and the

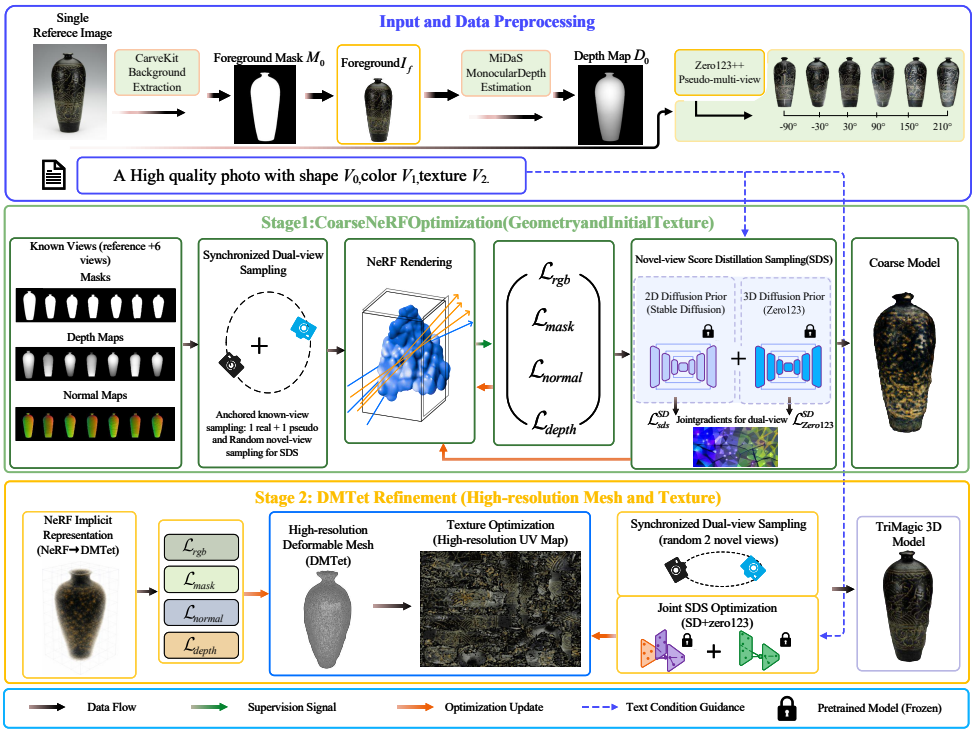


Fig. 1 Overall TriMagic framework. Input preparation and preprocessing: learn role-based multi-token conditioning, and generate and preprocess pseudo-reference images. Stage 1: optimize the main geometry and low-resolution appearance using a NeRF representation. Stage 2: convert to a DMTet mesh and refine high-resolution geometry and texture. Both stages combine weighted supervision at known views with diffusion guidance at random novel views.

views indexed by $i = 1, ..., 6$ are the pseudo-views generated by Zero123++. $\mathrm{I}_i, \mathrm{M}_i, \mathrm{D}_i$ and N_i denote the RGB image, foreground mask, relative depth, and surface normals of the ith known view, respectively, with its camera parameters denoted by π_i.

3.2 Semantic-Role-Guided Multi-Token Textual Inversion

Conventional single-token textual inversion[5] optimizes the embedding of a newly added token to represent the instance features of a reference object within a frozen text-to-image diffusion model. For objects with complex contours, nonuniform color distributions, and high-frequency surface patterns, a single token must simultaneously represent shape, color, and texture. Without an explicit division of attributes, it may struggle to preserve local details. Increasing the number of tokens can expand the representational capacity for instance features, but a shared image reconstruction objective does not specify which attributes each token should represent. Role swapping, semantic mixing, or representational redundancy may therefore still occur among tokens. We use the instance features represented by these tokens as text conditioning for the diffusion model, so both the division of attributes and the construction of conditioning must be specified.

To this end, we propose multi-token textual inversion based on semantic role contrastive alignment. The method uses role-specific text prompts to provide a structured semantic context for different tokens and further establishes explicit token–role contrastive constraints in the word embedding space of the frozen text encoder. These constraints guide different tokens to develop dominant semantic responses in the predefined shape, color, and texture directions, respectively. We also impose diversity regularization on attribute residuals relative to the object initialization word to suppress information redundancy among role tokens while retaining the object identity semantics shared by the different tokens. The overall workflow, shown in Figure 2, includes role-specific prompts and token initialization, image latent encoding and noise addition, noise prediction by the frozen diffusion model, and joint optimization under textual inversion, role contrastive alignment, and residual diversity constraints.

Semantic-role prompts and multi-token initialization. Given a single reference image I_0, we introduce three learnable tokens, denoted by $[v_0]$, $[v_1]$, and $[v_2]$, and assign them to the Shape, Color, and Texture roles, respectively, namely $[v_0] \rightarrow Shape, [v_1] \rightarrow Color, [v_2] \rightarrow Texture$

Here, $[v_0]$ models the contour, local structure, and visible form cues of the target object; $[v_1]$ describes its dominant colors, hues, and spatial color distribution; and $[v_2]$ captures surface decorative patterns, material appearance, and high-frequency local details. Let y denote the category or initialization word of the target object. The role-specific text prompt P_{role} is given in Equation (1):

$$P_{\text{role}} = \text{“a photo of a y with shape } [\text{v}_0], \text{ color } [\text{v}_1], \text{ texture } [\text{v}_2] \tag{1}$$

The role words “shape,” “color,” and “texture” provide distinct contextual semantics for the different tokens, enabling the frozen CLIP text encoder[25] to use its existing language knowledge to initially differentiate the directions of token updates. The token order remains fixed during training, and the three role tokens are not randomly permuted, maintaining a consistent correspondence between token indices and semantic roles.

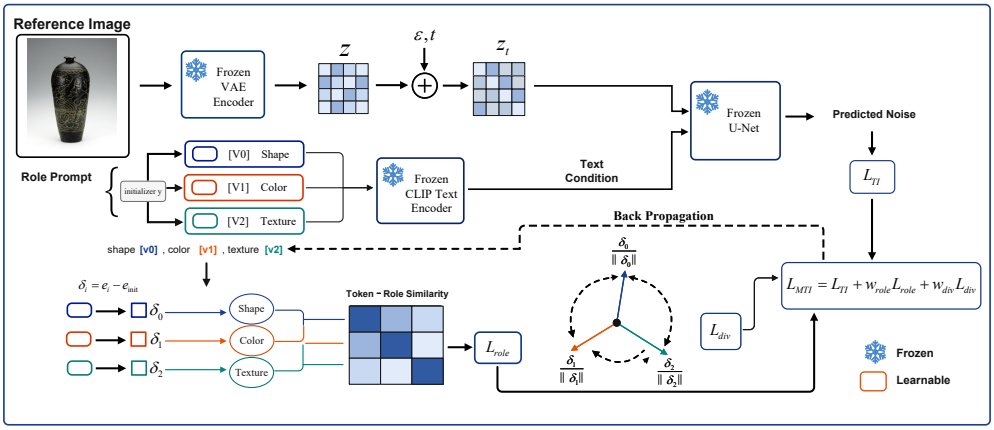


Fig. 2 Multi-token textual inversion workflow based on semantic role contrastive alignment.

In Figure 2, the reference image is encoded by the frozen VAE and noise is added. The role-specific text conditioning is used in U-Net noise prediction, and only the newly added token embeddings are updated. The initialization and joint loss are described below.

Let e_{init} denote the embedding of the object initialization word y obtained from the frozen word embedding layer. If the initialization word is split into multiple subwords, their mean embedding is used as e_{init}. All three learnable tokens are initialized from this object semantic anchor, with small independent Gaussian perturbations added, as shown in Equation (2):

$$e_i^{(0)} = e_{\text{init}} + \eta_i, \quad \eta_i \sim \mathcal{N}\left(0, \sigma^2 I\right), \quad i \in \{0, 1, 2\} \tag{2}$$

The shared initialization preserves object category semantics, whereas the independent perturbations break the initial symmetry among tokens. Instance features are subsequently learned through the image denoising objective.

Multi-token textual inversion objective. We freeze the VAE, U-Net, and text encoder backbone parameters of the pretrained latent diffusion model[26] and update only the three newly added token embeddings. First, the frozen VAE encoder $\mathcal{E}$ maps the reference image I to the latent variable $z = \mathcal{E}(I)$. At a randomly sampled diffusion time step t, Gaussian noise $\epsilon \sim \mathcal{N}(0, I)$ is added to obtain the noisy latent variable $z_t = \sqrt{\bar{\alpha}_t} z + \sqrt{1 - \bar{\alpha}_t}\epsilon$, where $\bar{\alpha}_t$ denotes the cumulative noise coefficient determined by the predefined diffusion noise scheduler. The role-specific prompt P_{role} is converted into text conditioning by the frozen text encoder $\mathcal{C}$. Under this conditioning, the pretrained U-Net predicts the noise added to the latent variable. The basic reconstruction loss for multi-token textual inversion is given in Equation (3):

$$\mathcal{L}_{\text{TI}} = \mathbb{E}_{I,t,\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t, t, \mathcal{C}(P_{\text{role}}))\|_2^2\right] \tag{3}$$

Here, ϵ_θ denotes the diffusion denoising network with frozen parameters. Equation (3) requires the text conditioning jointly formed by the three tokens to recover the instance features of the reference image, thereby learning instance information shared by the multiple tokens.

However, Eq. (3) constrains only the joint generative capacity of the three tokens and does not explicitly specify the visual role each token should assume. Even when role words are included in the prompt, the optimizer may still distribute color, texture, and shape information arbitrarily among the tokens. We therefore introduce a role contrastive alignment loss to establish an explicit correspondence between token indices and semantic roles in the word embedding space.

Role contrastive alignment loss. To address the ambiguity in attribute assignment, we add a role contrastive classification constraint to the basic reconstruction objective, directing the optimization of each token's attribute residual toward its corresponding role. Let the frozen CLIP word embedding layer be $\mathcal{W}(\cdot)$. For the three role names Shape, Color, and Texture, we define the role sequence $r_0 = \text{shape}, r_1 = \text{color}, r_2 = \text{texture}$. We first extract the role word embeddings, averaging over subwords when a role name contains multiple subwords. We then remove the component shared by the three embeddings to construct the normalized role directions in Eq. (4):

$$\mathbf{a}_j = \mathrm{Norm}\left[\mathcal{W}(r_j) - \frac{1}{K}\sum_{m=0}^{K-1}\mathcal{W}(r_m)\right], \quad K = 3 \tag{4}$$

Here, $Norm(x) = x/\|x\|_2$. Centering removes the embedding component shared by the different role words, allowing a_j to more specifically represent the relative semantic differences among the shape, color, and texture roles.

Directly pulling the full embedding e_i toward a generic role word may weaken instance semantics. We therefore decompose each token into a shared object anchor and an attribute residual δ_i, such that $e_i = e_{\text{init}} + \delta_i$. While retaining the shared object anchor, we apply the role alignment constraint only to the normalized attribute residual $\bar{\boldsymbol{\delta}}_i = \mathrm{Norm}(\boldsymbol{\delta}_i)$.

The semantic similarity between token i and role j is given by the dot product $s_{ij} = \bar{\boldsymbol{\delta}}_i^{\mathsf{T}}\mathbf{a}_j$. Arranging all s_{ij} by token index and role index yields the K×K token–role similarity matrix $S = [s_{ij}]_{i,j=0}^{K-1}$, whose rows correspond to $[v_0]$, $[v_1]$, and $[v_2]$, and whose columns correspond to Shape, Color, and Texture, respectively. To make each token more similar to its target role than to the other, non-target roles, we adopt the temperature-scaled contrastive classification objective in Eq. (5):

$$\mathcal{L}_{\text{role}} = -\frac{1}{K}\sum_{i=0}^{K-1}\log\frac{\exp(s_{ii}/\tau)}{\sum_{j=0}^{K-1}\exp(s_{ij}/\tau)} \tag{5}$$

Here, τ is the temperature coefficient. This loss increases the similarity between a token and its target role while reducing its similarity to non-target roles. For example, the objective for the token $[v_1]$ associated with the Color role is to make its similarities satisfy $s_{11} > s_{10}, s_{11} > s_{12}$.

Attribute residual diversity regularization. Beyond role-specific prompts, the role contrastive loss explicitly establishes a one-to-one correspondence between tokens and roles. However, after role alignment has specified the dominant semantics of each token, redundant encoding of instance attributes across tokens still needs to be suppressed. We therefore apply the diversity regularization in Eq. (6) to the normalized residuals $\bar{\boldsymbol{\delta}}_i$:

$$\mathcal{L}_{\text{div}} = \frac{1}{K(K-1)}\sum_{i=0}^{K-1}\sum_{\substack{j=0\\ j\neq i}}^{K-1}\left(\bar{\boldsymbol{\delta}}_i^{\mathsf{T}}\bar{\boldsymbol{\delta}}_j\right)^2 \tag{6}$$

Equation (6) reduces the absolute cosine similarity between attribute residuals, encouraging complementary directions. To retain the necessary shared object semantics, this constraint acts only on residuals relative to e_{init}, rather than directly on the full embeddings e_i.

Progressive joint optimization. Role contrastive alignment specifies attribute assignment, while diversity regularization suppresses representation overlap. The latter alone cannot assign roles, and the former alone may retain redundant instance information; we therefore use both constraints jointly. To allow the tokens to learn instance features first and then gradually introduce the role and diversity constraints,

we use a progressive weight schedule. The weight curves are controlled by the normalized optimization progress $q = n/N$, where n is the current optimization step and N is the total number of optimization steps. Given the maximum weight λ, starting progress q_0, and ramp-up interval Δq, the linear ramp-up function ω is defined in Eq. (7):

$$\omega\left(q;\lambda,q_0,\Delta q\right)=\lambda\cdot clip\left(\frac{q-q_0}{\Delta q},0,1\right) \tag{7}$$

Here, q denotes textual inversion training progress, λ denotes the maximum weight magnitude, and ω denotes the actual weight as it varies with progress. For the role contrastive loss and residual diversity loss, we use the parameters $(\lambda_{\text{role}}, q_{\text{role}}, \Delta q_{\text{role}})$ and $(\lambda_{\text{div}}, q_{\text{div}}, \Delta q_{\text{div}})$ in Eq. (7), respectively, to obtain their dynamic weights $\omega_{\text{role}}(q)$ and $\omega_{\text{div}}(q)$. Applying these two dynamic weights to the role contrastive and residual diversity losses, respectively, and combining them with the textual inversion loss yields Eq. (8):

$$\mathcal{L}_{\text{MTI}}=\mathcal{L}_{\text{TI}}+\omega_{\text{role}}(q)\mathcal{L}_{\text{role}}+\omega_{\text{div}}(q)\mathcal{L}_{\text{div}} \tag{8}$$

Here, $\mathcal{L}_{\text{TI}}$ learns overall instance features under the joint condition formed by the three tokens, $\mathcal{L}_{\text{role}}$ establishes an explicit semantic correspondence between token indices and Shape, Color, and Texture, and $\mathcal{L}_{\text{div}}$ reduces representation redundancy among the attribute residuals. During training, the pretrained VAE, U-Net, and text encoder backbone remain frozen, and only the word embedding parameters of $[v_0]$, $[v_1]$, and $[v_2]$ are updated.

We do not assume that shape, color, and texture are strictly independent; the tokens may share instance identity and necessary information across attributes. The joint constraints aim to form complementary conditions dominated by their respective roles. These conditions are subsequently used in the Stable Diffusion text branch to constrain the semantics and appearance of random novel views.

3.3 Dynamically weighted pseudo-multiview supervision strategy

Multi-token conditioning cannot provide direct observations of unseen regions, so we introduce pseudo-views generated by Zero123++[14] as auxiliary cues. To account for generation errors, empirical view weights set their relative strengths, a training-progress schedule controls their contributions at different stages, and absolute weighting by group preserves the independent coefficient of the real-view term. Anchored sampling is also used to retain the real reference constraint.

3.3.1 Dynamic pseudo-view weights

Given the reference image I_0, Zero123++ generates six pseudo-views with azimuth angles of -90°, -30°, 30°, 90°, 150°, and 210°. Considering that the reliability of pseudo-views varies across camera directions and supervision modalities, we set CSV base weights b_i^m for different views and supervision modalities, where $m \in \{rgb, depth, normal, 3D\}$. The base weight of the real reference view is fixed at 1.

Pseudo views receive small absolute weights: the side views assigned higher empirical weights receive 0.05, and the remaining views receive 0.01. This design expresses an empirical assumption about the reliability of different pseudo-views, without treating generated views as equivalent to real observations. These values are empirical supervision weights preset according to viewpoint, rather than confidence scores predicted from image content or calibrated probabilistically; we control the influence of pseudo-views by adjusting their contributions to optimization.

3.3.2 Three-stage dynamic weight schedule after warm-up

Using only fixed low weights would leave pseudo-views with insufficient contributions throughout training, whereas fixed high weights would amplify generation errors before the structure stabilizes. Let $q = t/T$ denote normalized training progress. We use a scheduling coefficient $c(q)$ that first suppresses, then strengthens, and finally decays pseudo-view supervision, as defined in Eq. (9):

$$c(q) = \begin{cases} 0, & q \le 0.02 \\ \dfrac{1}{2}\left[1 - \cos\left(\dfrac{\pi(q-0.02)}{0.28}\right)\right], & 0.02 < q < 0.30 \\ 1, & 0.30 \le q \le 0.80 \\ 1 - \dfrac{0.85(q-0.80)}{0.20}, & 0.80 < q \le 1 \end{cases} \tag{9}$$

Equation (9) sets the first 2% of training as a warm-up interval with zero pseudo-view weights, followed by three stages: ramp-up, hold, and decay. Specifically, cosine ramp-up smoothly introduces pseudo-views during the 2%–30% interval; full pseudo-multiview constraints are maintained during the 30%–80% interval; and the coefficient gradually decays to 0.15 over the final 20%. This reduces the strength of pseudo-view supervision in the known-view branch during late training while retaining nonzero pseudo-view constraints. The effective weight of pseudo-view i for modality m is $w_i^m(q) = b_i^m c(q)$, whereas the effective weight of the real reference image remains $w_0^m = 1$.

3.3.3 Absolute weighting that preserves the real anchor

If real- and pseudo-view losses are jointly averaged by the sum of weights or the number of views, the coefficient of the real reference term changes with the aggregation setting. If the pseudo-view group is separately normalized by its sum of weights, the common absolute scale of the pseudo-view weights is canceled. We divide the sampled set into a real-view group $k_t^R = \{0\}$ and a pseudo-view group $k_t^P = \{j_t\}$ and apply absolute weighting to their losses without a shared denominator. Let $\ell_{t,i}^m$ denote the single-view modality loss for view i at iteration t, whose calculation is described in Section 3.4.1. The objective aggregated across the two groups is given in Eq. (10):

$$L_t^m = \ell_{t,0}^m + w_{j_t}^m(q_t)\, \ell_{t,j_t}^m \tag{10}$$

Here, $\ell_{t,0}^m$ is the real-view loss with unit weight, and $w_{j_t}^m(q_t)\, \ell_{t,j_t}^m$ is the weak supervision loss from the sampled pseudo-view. Neither term is normalized by the number

of views or the sum of weights, preserving the unit coefficient of the real-view term and the 0.01/0.05 scale of the pseudo-view weights. This controls loss contributions but does not guarantee that gradient directions are free of conflict.

3.4 Joint optimization with pseudo-multiview supervision

TriMagic combines supervision at discrete known views with guidance at continuously sampled random novel views. The former computes pixel and geometric losses on the real image and one randomly selected pseudo-view, while the latter uses the complete reference set to fuse Zero123 view conditions.

3.4.1 Multimodal supervision at known views

During the coarse NeRF modeling and DMTet refinement stages, we apply multimodal supervision to the real reference view and the sampled pseudo-reference view. The color, mask, and monocular geometric prior constraints at the real reference view follow the basic approach of existing single-image 3D optimization methods[3, 16]; we extend this approach with weighting and sampling of the pseudo-view group. When dual-view supervision is triggered at iteration t, $j_t \sim \mathrm{Uniform}(\{1, \ldots, 6\})$ is sampled from the pseudo-view group, and the active index set is defined as $K_t = \{0, j_t\}$. For any $i \in K_t$, under camera π_i, the model renders $\hat{I}_{t,i}$, $\hat{M}_{t,i}$,i, $\hat{D}_{t,i}$, and $\hat{N}_{t,i}$ and computes single-view losses against I_i, M_i, D_i, and N_i.

Color and silhouette supervision. Let O_i denote the valid color region obtained by processing the edges of the foreground mask M_i. For an active view $i \in K_t$, the foreground RGB loss is defined in Eq. (11):

$$\ell_{t,i}^{\mathrm{rgb}} = \mathrm{MSE}(O_i \odot \hat{I}_{t,i},\, O_i \odot I_i) \tag{11}$$

Here, $\odot$ denotes element-wise multiplication, and MSE denotes mean squared error. A mask regression loss $\ell_{t,i}^{\mathrm{mask}} = \mathrm{MSE}(\hat{M}_{t,i},\, M_i)$ is also applied over the entire image to align the predicted foreground mask with the ground-truth mask pixel by pixel. For both RGB and mask losses, single-view terms are first calculated for each view and then aggregated into view-level losses using the absolute weighting for dual views anchored to the real reference in Eq. (10).

Surface normal loss. Let Ω_i denote the set of valid foreground pixels within the target mask. Predicted and target normals are mapped from [0,1] to [-1,1] and normalized to unit vectors. Their cosine similarity $\langle \cdot, \cdot \rangle$ measures normal consistency, giving the loss in Eq. (12):

$$\ell_{t,i}^{\mathrm{normal}} = \frac{1}{2}\left[1 - \frac{1}{|\Omega_i|}\sum_{p \in \Omega_i} \langle \hat{N}_{t,i,p},\, N_{i,p} \rangle\right] \tag{12}$$

Relative depth loss. To avoid the effects of differences in monocular depth scale and shift, we construct a loss within the target foreground Ω_i using the Pearson correlation coefficient $\rho(\,\cdot\,,\,\cdot\,)$ between the predicted and target depths. This measure is invariant to depth shifts and positive scaling, and the loss is defined in Eq. (13):

$$\ell_{t,i}^{\text{depth}} = \frac{1}{2}[1 - \rho(\hat{D}_{t,i}|_{\Omega_i},\, D_i|_{\Omega_i})] \tag{13}$$

Joint supervision objective. The four types of single-view losses are aggregated according to Eq. (10) into the view-level losses L_t^{rgb}, L_t^{mask}, L_t^{normal}, and L_t^{depth}, which are then combined into the total known-view supervision in Eq. (14):

$$\mathcal{L}_{\text{known}} = \lambda_{\text{rgb}} L_t^{\text{rgb}} + \lambda_{\text{mask}} L_t^{\text{mask}} + s(t)\lambda_{\text{normal}} L_t^{\text{normal}} + \lambda_{\text{depth}} L_t^{\text{depth}} \tag{14}$$

Here, λ_{rgb}, λ_{mask}, λ_{depth}, and λ_{normal} control the color, mask, depth, and normal losses, respectively; $s(t) = t/T$ increases linearly from 0 to 1 during training, gradually strengthening the normal constraint.

3.4.2 Novel-view SDS and dynamic multi-reference guidance

Random novel views jointly use the Stable Diffusion (SD)[26] and Zero123[4] priors. SD constrains semantics and appearance through view-dependent text conditions. Zero123 fuses noise predictions according to the angular differences between the target view and the seven reference views, limits the total weight of the pseudo-reference group, and uses the angular weight of the real reference to adjust the magnitude of the 3D SDS gradient.

View-dependent 2D guidance based on score distillation sampling. For target camera j in a novel-view training batch, the model first obtains the image I_j through differentiable rendering and uses the variational autoencoder E to map it to the latent variable $z_j = E(I_j)$. A diffusion timestep t_j and Gaussian noise ϵ_j are then sampled independently, and the noisy latent variable $z_{t_j,j}$ is constructed using the DDPM forward process.

Rather than using a fixed text condition, we linearly interpolate between front-, side-, and rear-view text embeddings according to the azimuth of the target camera to obtain the text condition y_j corresponding to the target viewing direction. We predict noise using classifier-free guidance (CFG)[27], namely $\hat{\epsilon}_j^{\text{SD}} = \epsilon_\phi\left(z_{t_j,j}, \varnothing, t_j\right) + s_{2\text{D}}\left[\epsilon_\phi\left(z_{t_j,j}, y_j, t_j\right) - \epsilon_\phi\left(z_{t_j,j}, \varnothing, t_j\right)\right]$, where ϵ_ϕ is the SD U-Net noise prediction function, $\varnothing$ is the unconditional text embedding, and $s_{2\text{D}}$ is the 2D CFG scale. Following SDS[6], the SD branch injects the following optimization gradient into the latent space: $\nabla_{z_j}\mathcal{L}_{\text{SDS}}^{\text{2D}} = \lambda_{2\text{D}}\left(1 - \bar{\alpha}_{t_j}\right)\left(\hat{\epsilon}_j^{\text{SD}} - \epsilon_j\right)$.

Multi-reference 3D guidance with dynamic weighting based on angular differences. Zero123 fuses view conditions according to the relationships between the target and reference cameras. Let the angular difference between target j and reference i be $\Delta\theta_{j,i}$. All seven references are used here for novel-view noise fusion, unlike the dual-view loss at known views.

Let $a_i(q_t)$ denote the effective Zero123 base weight of reference view i at the current training progress q_t. Combining the reference base weight with exponential angular decay yields the initial reference weight for target view j, as defined in Eq. (15):

$$q_{j,i} = \frac{a_i(q_t)exp(-\Delta\theta_{j,i}/\tau)}{\sum_{k=0}^{6} a_k(q_t)exp(-\Delta\theta_{j,k}/\tau) + \varepsilon_0} \tag{15}$$

Here, τ controls angular decay, and ε_0 prevents the denominator from being zero. The weights reflect both the empirical base weights and the proximity of the camera directions.

To prevent the larger number of pseudo-reference views from suppressing the real reference view during fusion, we further limit the total weight of the pseudo-reference group. Based on the minimum angular difference between the target view and the real reference set $\mathcal{P}$, we linearly interpolate between the near-view upper bound c_{near} and the far-view upper bound c_{far} to determine the upper bound on the total pseudo-view weight c_j. Let $Q_j^{\mathcal{P}}$ be the sum of the initial pseudo-reference weights. When $Q_j^{\mathcal{P}}$ exceeds c_j, all pseudo-view weights are uniformly scaled by the proportional factor η_j; otherwise, $\eta_j = 1$. The remaining weight is redistributed according to the original proportions within the real-view group. The final weights $w_{j,i}$ are defined in Eq. (16):

$$w_{j,i} = \begin{cases} \eta_j q_{j,i}, & i \in k_t^P \\ \dfrac{q_{j,i}}{\sum_{k \in R} q_{j,k} + \varepsilon_0} \left(1 - \eta_j Q_j^{\mathcal{P}}\right), & i \in k_t^R \end{cases} \tag{16}$$

This adjustment preserves relative weights within each group while limiting the total weight of the pseudo-view group. The pseudo-view group receives a lower share when the target is close to the real reference direction and a moderately larger share when it is farther away. Because the denominator for the real-view group in Eq. (16) contains a stabilization term, the total weight is approximately 1 only when that term is negligible; the weights are not strictly normalized.

In each SDS update, Zero123 performs CFG noise prediction separately, conditioned on each reference image and its relative camera pose. It then uses $w_{j,i}$ to fuse the predictions from all reference views by weighted aggregation, as shown in Eq. (17):

$$\begin{aligned} \hat{\epsilon}_j^{\text{Zero123}} = \sum_{i=0}^{6} w_{j,i} \Big\{ & \epsilon_\phi(z_{t_j,j}, \varnothing, t_j) \\ & + s_{\text{3D}} \left[\epsilon_\phi(z_{t_j,j}, c_{j,i}, t_j) - \epsilon_\phi(z_{t_j,j}, \varnothing, t_j)\right] \Big\} \end{aligned} \tag{17}$$

Here, ϵ_ϕ is the Zero123 noise prediction function, $\varnothing$ is the unconditional input, $c_{j,i}$ consists of the features of reference image i and its relative pose with respect to target camera j, and s_{3D} is the 3D CFG scale.

In addition to dynamically fusing noise directions, we use the angular weight of the real reference view to adjust the gradient strength for each target view. The closer the real reference camera is to the target camera, the higher its angular weight. A minimum angular gradient scaling coefficient $r_{\min}$ is also set to prevent the guidance gradient from vanishing completely at distant views. Substituting this angular weight directly into the SDS gradient yields

$$\begin{aligned} \nabla_{z_j} \mathcal{L}_{\text{SDS}}^{\text{3D}} = \lambda_{\text{3D}} \Bigg[& \sum_{i \in \mathcal{R}} \frac{w_{j,i}}{\sum_{k \in \mathcal{R}} w_{j,k} + \varepsilon_0} \operatorname{clip}\left(1 - \frac{\Delta\theta_{j,i}}{180^\circ}, r_{\min}, 1\right) \Bigg] \\ & \cdot (1 - \bar{\alpha}_{t_j})(\hat{\epsilon}_j^{\text{Zero123}} - \epsilon_j) \end{aligned} \tag{18}$$

Here, λ_{3D} is the Zero123 base gradient scaling coefficient, $\bar{\alpha}_{t_j}$ is the cumulative signal retention coefficient at timestep t_j, and ϵ_j is the injected noise. The angular

weight in square brackets is determined by the real reference, while the overall update still depends on the fused noise residual.

This mechanism separately controls the fusion of multi-reference noise predictions and the strength of SDS updates. Both real and pseudo-reference images participate in noise prediction fusion, while the gradient scaling factor is calculated from the angular weights of the real reference views. Ignoring the small numerical deviation introduced by the stabilized denominator, pseudo-reference images do not directly determine this scaling factor. However, their noise predictions still alter the fused gradient, so this mechanism cannot guarantee that the final gradient norm remains unchanged or that errors are completely eliminated.

3.4.3 Anchored dual-view sampling

Anchored dual-view sampling applies only to the known-view supervision branch; the random novel-view SDS branch remains independent. During 3D optimization, known-view constraints are triggered at a specified iteration interval. At each trigger, TriMagic constructs the active set $K_t = \{0, j_t\}$, where index 0 denotes the real reference view and j_t is randomly sampled from the pseudo-view set. Color, mask, normal, and depth losses are first computed for each view and then aggregated by absolute weighting of the real and pseudo-view groups according to Eq. (10). Thus, the real-view term is always retained with unit weight, while pseudo-views supplement side- and rear-view constraints only through small dynamic weights.

The pseudo-view term follows the schedule in Section 3.3.2: its weight is 0 during warm-up and is dynamically adjusted thereafter. One of the six pseudo-views is randomly selected each time, allowing different reference directions to participate in optimization over successive updates.

The random novel-view branch independently samples diffusion timesteps and noise and simultaneously injects the 2D SDS gradient from the SD branch and the Zero123 gradient in Eq. (18). Anchored sampling does not change the SDS formulation or the novel-view batch size. Section 4.3.4 compares its endpoint metrics and runtime with those of the single-view setting.

4 Experiments

4.1 Datasets and evaluation metrics

Datasets. We evaluate TriMagic on NeRF4, RealFusion15, and our Artifact image collection of cultural heritage objects. All methods use a single reference image as input. NeRF4 is a subset of 4 objects selected by Magic123 from the synthetic NeRF data[3, 18] and is used to evaluate single-image 3D generation under controlled imaging conditions. RealFusion15 comprises the 15 real-object reference images released by RealFusion[2, 3], covering diverse geometric forms and appearance characteristics, and is used to evaluate generation from real-image inputs. The Artifact collection contains 378 images of cultural heritage objects provided by the administrative staff of Shanxi Museum, covering various types, including porcelain and jade objects. We selected several instances from each of the different types, for a total of 24 test instances.

Only one reference image is provided for each instance, with no ground-truth side- or rear-view annotations. Dataset statistics are presented in Table 1.

Table 1 Dataset statistics

Dataset	Data type	Original size	Test instances
NeRF4	Synthetic objects	4	4
RealFusion15	Real objects	15	15
Artifact	Cultural heritage objects	378 images	24

Evaluation metrics. We use CLIP-Similarity, PSNR, and LPIPS to evaluate object semantic consistency, pixel fidelity in the reference view, and differences in perceptual features, respectively. For images with an alpha channel, fully transparent pixels are replaced with a white background and the alpha channel is removed before evaluation. The input resolution for CLIP and LPIPS is 224×224, whereas PSNR is computed at a resolution of 800×800.

CLIP-Similarity. We use the CLIP-ViT-bigG/14 image encoder[25, 28] to extract normalized features from the input reference image and multiview renderings. By default, TriMagic generates 100 test renderings per object, keeping the camera polar angle and radius fixed while uniformly sampling azimuths from 0° to 356.4° at 3.6° intervals. CLIP evaluation uses all target renderings in the selected output directory, first averaging scores across views for each object and then averaging across objects. Each rendered view is compared with the same input reference image, so this metric does not directly measure geometric consistency between rendered views. For sample n, let the input image be I_n and rendered view v be $I_{n,v}$. The mean CLIP similarity is given in Eq. (19):

$$S_{\text{CLIP}} = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{V_n}\sum_{v=1}^{V_n}\frac{f(I_n)^{\mathsf{T}} f(I_{n,v})}{\|f(I_n)\|_2 \|f(I_{n,v})\|_2} \tag{19}$$

Here, N is the number of samples, V_n is the number of rendered views for sample n, and $f(\cdot)$ is the CLIP image encoder. A higher value indicates greater consistency between the rendered result and the input object in terms of semantics and overall appearance.

PSNR. PSNR evaluates pixel-level reconstruction quality at the input reference view. Let E_n denote the mean squared error between the input image and the reference-view rendering for sample n. The dataset-averaged PSNR is given in Eq. (20):

$$S_{\text{PSNR}} = \frac{1}{N}\sum_{n=1}^{N} 10\log_{10}\left(\frac{1}{E_n}\right) \tag{20}$$

Because all evaluation images are normalized to [0,1], the peak value of the dynamic range is set to 1. A higher PSNR indicates greater pixel-level reconstruction fidelity in the reference view.

LPIPS[29]. We use AlexNet as the feature network to compare the input image with the reference-view rendering in a multilayer perceptual feature space, as formulated in Eq. (21):

$$S_{\text{LPIPS}} = \frac{1}{N}\sum_{n=1}^{N}\sum_{l=1}^{L}\frac{1}{H_l W_l}\sum_{h,w}\left\|\mathbf{w}_l \odot \left(\hat{\mathbf{y}}^l_{n,h,w} - \mathbf{y}^l_{n,h,w}\right)\right\|_2^2 \tag{21}$$

Here, y^l and $\widehat{y}^l$ are the normalized features at layer l, and w_l denotes the learned channel weights. A lower LPIPS indicates greater similarity between the two images in perceptual features.

Because a single image provides no real observations of unseen regions, we also conduct a qualitative analysis of renderings from selected rear views and views sensitive to Janus artifacts to examine texture plausibility, obvious structural repetition, and local conflicts. This analysis concerns only the generated appearance in the displayed views.

Implementation details. TriMagic is implemented using PyTorch 2.11.0 and CUDA 12.6, and all experiments are performed on a single NVIDIA RTX PRO 6000 GPU. We use Stable Diffusion 2.1-base as the 2D text-conditioned diffusion prior, the pretrained Zero123 model 105000.ckpt as the view-conditioned diffusion prior, and Zero123++ v1.2 to generate 6 pseudo-reference images at fixed camera views for each input image. Textual inversion and coarse-to-fine 3D optimization are performed separately for each input instance. Throughout optimization, the pretrained parameters of all three diffusion models remain frozen; only the new token embeddings and the NeRF and DMTet 3D representations are updated. The main hyperparameters are listed in Table 2.

Table 2: Main training and optimization parameters of TriMagic

Module	**Parameter**	**Setting**
Textual inversion	Training steps / batch size	3000/16
	Learning rate	1 × 10-4
	Diffusion timestep	[0-1000)
	$\omega_{div}(q)$	q<0.2: 0; 0.2≤q<0.4: linear increase to 0.1; q≥0.4: fixed at 0.1; maximum magnitude λdiv=0.1
	Maximum role loss weight	0.03
	Role contrastive temperature	0.07
	Role weight schedule	0 at q=0; linear increase to 0.03 over the first 20% of progress, then held at 0.03
Coarse NeRF stage	Optimization steps / optimizer / learning rate	4000/Adam/1 × 10-3
	Training render resolution	128 × 128
	Normalized diffusion timestep range	[0.2,0.6]
	SDS guidance weight	$\lambda_{2D} = 1, \lambda_{3D} = 40$
	CFG guidance scale	$s_{2D} = 100, s_{3D} = 5$

Module	Parameter	Setting
DMTet	Optimization steps / optimizer / learning rate	4000/Adam/1 × 10-3
	Training render resolution	1024 × 1024
	Normalized diffusion timestep range	[0.02,0.50]
	SDS guidance weight	$\lambda_{2D} = 0.001$,$\lambda_{3D} = 0.01$
	CFG guidance scale	$s_{2D} = 100$,$s_{3D} = 5$
Known-view supervision	Color loss	$\lambda_{rgb} = 5$
	Mask, depth, and normal losses	$\lambda_{mask} = 0.01$,$\lambda_{depth} = 0.01$,$\lambda_{normal} = 0.01$
Multi-reference angular weighting	Angular decay scale	45°
	Near / far weight caps for the pseudo-reference group	0.35/0.55
	Minimum angular gradient scaling factor	0.20

4.2 Comparative experiments

To evaluate the overall performance of the proposed method, we compare it with Shap-E[30], 3DFuse[31], Make-It-3D[32], RealFusion[2], Zero-1-to-3[4], and Magic123[3] using the same input images and rendering viewpoints. Metrics for all methods are computed using the same evaluation procedure. The quantitative and qualitative results are presented in Table 3 and Figure 3, respectively.

Table 3 presents the quantitative results on the three datasets. On NeRF4, TriMagic increases CLIP-Similarity from 0.80 for Magic123 to 0.81 and PSNR from 24.62 to 24.75 dB (+0.13 dB), achieving the highest values in the table for both metrics, while both methods have an LPIPS of 0.03. On RealFusion15, CLIP-Similarity increases from 0.82 to 0.84 and PSNR from 19.50 to 20.01 dB (+0.51 dB), while LPIPS is 0.10 for both TriMagic and Magic123 at the precision displayed in the table. TriMagic achieves the highest CLIP-Similarity and ties with Make-It-3D for the highest PSNR. On Artifact, CLIP-Similarity increases from 0.71 to 0.75 when displayed to two decimal places, and PSNR increases from 27.58 to 28.99 dB (+1.41 dB), but LPIPS increases from 0.03 to 0.04, performing worse than Magic123. These differences are described using the displayed table values; identical displayed values do not imply that the unrounded results are exactly equal. Overall, TriMagic improves semantic similarity and pixel fidelity in the reference view, with a trade-off in the perceptual quality of cultural heritage object details.

Figure 3 shows the generated rear views of selected examples, allowing comparisons of overall object contours, color distributions, local textures, and obvious structural repetition. Some results from the comparison methods exhibit blurred textures, color shifts, or repeated components. In some examples, TriMagic preserves color and material characteristics that more closely resemble those of the input object, although fine ornamentation may still be smoothed or locally discontinuous. These observations are

Table 3 Quantitative comparison of single-image 3D generation methods on the NeRF4, RealFusion15, and Artifact datasets (CLIP-Similarity↑, PSNR (dB)↑, LPIPS↓)

Dataset	Metric	Shap-E	3DFuse	Make-It-3D	RealFusion	Zero-1-to-3	Magic123	Ours
NeRF4	CLIP-Similarity	0.60	0.60	0.78	0.38	0.62	0.80	0.81
	PSNR	0.99	5.86	19.20	15.37	23.96	24.62	24.75
	LPIPS	0.76	0.76	0.14	0.20	0.05	0.03	0.03
RealFusion15	CLIP-Similarity	0.59	0.62	0.83	0.67	0.75	0.82	0.84
	PSNR	1.23	18.87	20.01	16.87	19.49	19.50	20.01
	LPIPS	0.74	0.80	0.12	0.14	0.11	0.10	0.10
Artifact	CLIP-Similarity	0.55	0.54	0.61	0.56	0.68	0.71	0.75
	PSNR	1.15	13.90	23.10	22.40	26.70	27.58	28.99
	LPIPS	0.70	0.78	0.08	0.11	0.05	0.03	0.04

limited to the rear views shown and cannot verify consistency over a complete orbit or true rear-surface geometry. PSNR and LPIPS quantify reconstruction quality in the reference view.

4.3 Ablation studies

This section compares four configurations: Baseline uses a single token without pseudo views; "+Multi-token" adds only the module described in Section 3.2; "+Pseudo views" retains a single token and adds dynamic pseudo supervision; and Ours enables both. The inputs, random seeds, rendering viewpoints, training steps, and all other parameters are identical across the four configurations. Weight scheduling and sampling strategies are then compared on Artifact.

To clarify the correspondence between the ablation studies and the key modules, Table 4 summarizes the modules enabled in each experimental configuration. MT denotes the structure with multiple semantic tokens; Role, the role contrastive alignment constraint; Div, residual diversity regularization; PV, the Zero123++ pseudo-multiview prior; Dynamic, the three-stage dynamic pseudo-view weighting strategy after warm-up; Anchor, retention of the real reference view as an anchor during updates with pseudo supervision; and Dual, anchored dual-view sampling. A–D analyze the independent and combined effects of multiple semantic tokens and the pseudo-multiview prior. E evaluates the effectiveness of the role alignment constraint. F–G compare pseudo-view weighting strategies. H–I compare the recorded runtime and generation quality of different view sampling configurations.



Fig. 3 Comparison of the rear appearance of objects generated by each method. The first column shows the input reference images, and the remaining columns show the generated results.

4.3.1 Ablation study of textual inversion with multiple semantic tokens

The comparison between Baseline and "+Multi-token" examines the effect of the module without pseudo views, while the comparison between "+Pseudo views" and Ours examines its effect under the same pseudo supervision. "+Multi-token" includes the

Table 4 Correspondence between ablation configurations and key modules

ID	Configuration	MT	Role	Div	PV	Dynamic	Anchor	Dual
A	Baseline	×	×	×	×	×	×	×
B	+Multi-token	✓	✓	✓	×	×	×	×
C	+Pseudo views	×	×	×	✓	✓	✓	✓
D	Ours	✓	✓	✓	✓	✓	✓	✓
E	w/o Role Alignment	✓	×	✓	×	×	×	×
F	Fixed high	✓	✓	✓	✓	×	✓	✓
G	Fixed low	✓	✓	✓	✓	×	✓	✓
H	Ours-single	✓	✓	✓	✓	✓	×	×
I	Ours-dual	✓	✓	✓	✓	✓	✓	✓

Note: "✓" indicates that the corresponding module is enabled, and "×" indicates that it is disabled. Both Fixed high and Fixed low use fixed pseudo-view weights, so the Dynamic column is marked "×" for both; they differ in the value of the fixed weight. Anchor indicates whether the real reference view is used simultaneously in each update with pseudo supervision; it does not indicate that the model makes no use of the real input image.

three role tokens, role-specific prompts, role contrastive constraint, and residual diversity constraint described in Section 3.2. It is the complete module, rather than merely an increase in the number of parameters.

The comparison between Baseline and "+Multi-token" in Table 5 shows that PSNR on NeRF4 increases from 24.62 to 25.99 dB (+1.37 dB). On RealFusion15, CLIP-Similarity is 0.82 for both at two decimal places, PSNR increases from 19.50 to 19.63 dB (+0.13 dB), and LPIPS is 0.10 and 0.10, respectively. On Artifact, CLIP-Similarity increases from 0.71 to 0.75, but PSNR decreases from 27.58 to 27.09 dB (-0.49 dB) and LPIPS increases from 0.03 to 0.04. These results indicate that the benefits of the multi-token module depend on the dataset and metric and cannot be generalized as improvements in all reconstruction metrics. A further comparison between "+Pseudo views" and Ours shows that, at the displayed precision of two decimal places, their CLIP-Similarity values are identical on NeRF4 and RealFusion15, at 0.81 and 0.84, respectively, whereas the value on Artifact increases from 0.74 to 0.75. This comparison supports complementary gains from the complete module on some datasets, but does not show a visible improvement in CLIP-Similarity on all three datasets.

Figure 4 further shows that the single-token model is more prone to color shifts, blurred textures, and missing local details in unseen views. Multiple semantic tokens improve color continuity, material attributes, and overall contours in rear-facing regions. This module does not directly provide rear-view pixels. Instead, it uses shape, color, and texture conditioning to constrain the diffusion model's completion distribution for unknown regions, and can therefore complement an explicit pseudo-view prior.

To examine the effect of the role constraint on correspondence in the embedding space when the number of learnable tokens is held constant, we use token–role cosine similarity heatmaps to analyze the correspondence between individual tokens and predefined semantic roles. The rows represent token0, token1, and token2, and the



Fig. 4 Ablation results for the textual inversion module with multiple semantic tokens

columns represent shape, color, and texture. The color indicates the similarity between each token and the corresponding semantic role direction. The experiment keeps the number of learnable tokens, training data, initialization, training steps, and other loss terms identical, comparing only two settings with and without the role contrastive alignment loss. This comparison evaluates whether the loss can guide multiple tokens to develop the intended specialization in visual attributes.

As shown in Figure 5, without the role contrastive alignment loss, the similarity matrix does not exhibit a clear diagonal structure. The Top-1 matching rate between the three tokens and the predefined roles is only 30.6%, and the per-sample Perfect3/3 matching rate is 0.0%. This indicates that merely increasing the number of tokens and using diversity regularization can reduce representational redundancy among tokens, but cannot ensure that they correspond to specific visual attributes. After adding the role contrastive alignment loss, the similarities for token0–shape, token1–color, and token2–texture reach 0.272, 0.338, and 0.272, respectively, all substantially higher than those for the non-target roles in the same rows. The similarity matrix exhibits clear diagonal dominance, the role Top-1 matching rate increases to 93.1%, and the per-sample Perfect3/3 matching rate reaches 79.2%. These results indicate that the role contrastive alignment loss can effectively guide different tokens to emphasize shape, color, and texture semantics, respectively, forming complementary representations dominated by their roles in the text embedding space. Table 4 lists the module configurations, and Table 5 and Figure 4 show the 3D generation performance of the complete multi-token module. Figure 5 supports the effect of the role constraint on

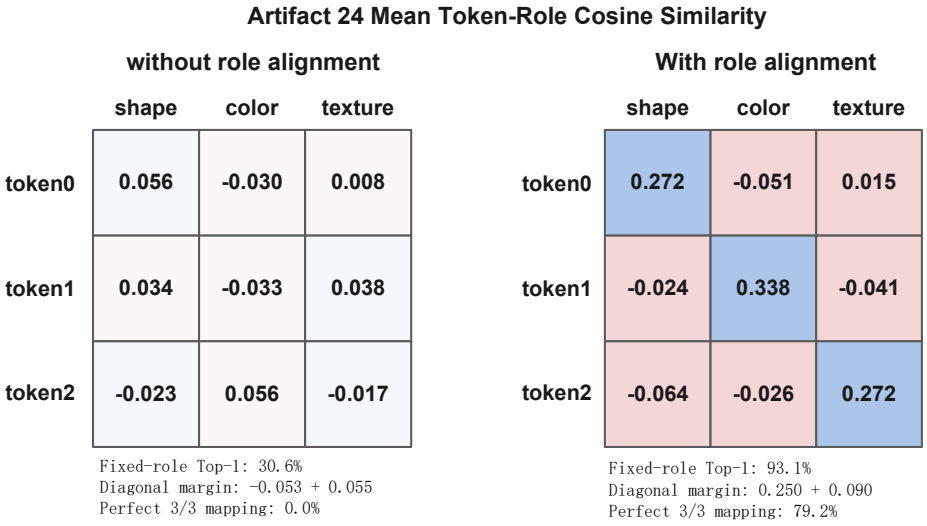


Fig. 5 Ablation results shown as role cosine similarity heatmaps. The matrices show the mean token–role cosine similarities across 24 test objects in the Artifact dataset. Rows represent the three learnable semantic tokens, and columns represent the semantic roles of shape, color, and texture. The left panel shows the results with the role alignment constraint removed, where no stable correspondence is formed between tokens and semantic roles. The right panel shows the results using the complete textual inversion strategy with multiple semantic tokens. Its diagonal entries are substantially higher than its off-diagonal entries, indicating that different tokens form differentiated semantic representations oriented toward shape, color, and texture attributes, respectively. The displayed values are cosine similarities averaged across the 24 objects.

correspondence in the embedding space, but cannot independently establish this loss's contribution to 3D generation metrics.

4.3.2 Ablation study of the Zero123++ pseudo-multiview prior

The comparisons between Baseline and "+Pseudo views" and between "+Multi-token" and Ours examine the effects of the dynamic pseudo-multiview module under single-token and multi-token conditioning, respectively. This module includes the pseudo-reference set, dynamic weights, real anchor, and dual-view sampling described in Sections 3.3–3.4; therefore, its benefits cannot be attributed solely to the number of pseudo views.

The qualitative results are shown in Figure 6. Baseline lacks explicit constraints for side and rear views and is prone to copying frontal textures into unseen regions, producing Janus artifacts. Adding only pseudo-multiview supervision substantially reduces multiple heads and repeated frontal textures, but the rear-view textures in some examples remain affected by pseudo-view noise, with moderate detail quality. Ours uses both the camera-space constraints of pseudo views and the instance-specific semantic conditioning of multiple tokens, suppressing Janus artifacts while improving color, texture, and category characteristics in rear-facing regions.

In Table 5, compared with Baseline, "+Pseudo views" changes CLIP-Similarity on NeRF4 from 0.80 to 0.81 at two decimal places and increases PSNR from 24.62 to 24.70 dB (+0.08 dB). On RealFusion15, CLIP-Similarity increases from 0.82 to 0.84 and PSNR from 19.50 to 19.84 dB (+0.34 dB). On Artifact, CLIP-Similarity increases from 0.71 to 0.74 and PSNR from 27.58 to 28.43 dB (+0.85 dB), whereas LPIPS increases from 0.03 to 0.04. Together with the examples in Figure 6, these results support gains from the dynamic pseudo-multiview module in some semantic and reference-view reconstruction metrics and in rear-view visual appearance, while also indicating that local perceptual quality may deteriorate. They do not support a claim of improvement in all metrics.

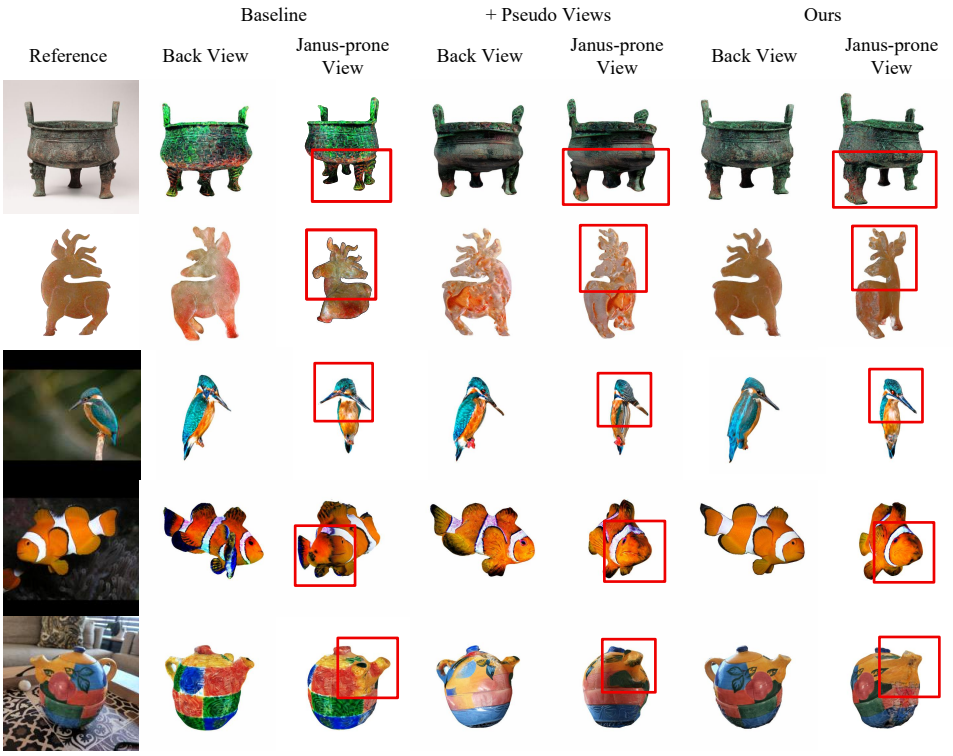


Fig. 6 Qualitative ablation results for dynamically weighted pseudo-multiview supervision (red boxes indicate Janus artifacts, repeated textures, or local structural conflicts)

4.3.3 Ablation study of pseudo-view weighting strategies

Because pseudo views are generated observations, their supervision strength directly determines the balance between the benefits of the prior and error accumulation. We compare fixed high weights, fixed low weights, and three-stage dynamic weights after warm-up on the Artifact dataset. The fixed high and low settings set the pseudo-view base weight to 1 and 0, respectively, representing two extreme settings of the

Table 5 Ablation results for multiple semantic tokens and dynamic pseudo-multiview supervision (CLIP-Similarity↑, PSNR (dB)↑, LPIPS↓)

Method	Multi-token	Pseudo views	NeRF4	RealFusion15	Artifact
			(CLIP-Similarity/ PSNR/LPIPS)	(CLIP-Similarity/ PSNR/LPIPS)	(CLIP-Similarity/ PSNR/LPIPS)
Baseline			0.80/24.62/0.03	0.82/19.50/0.10	0.71/27.58/0.03
+Multi-token	✓		0.80/25.99/0.03	0.82/19.63/0.10	0.75/27.09/0.04
+Pseudo views		✓	0.81/24.70/0.03	0.84/19.84/0.10	0.74/28.43/0.04
Ours	✓	✓	0.81/24.75/0.03	0.84/20.01/0.10	0.75/28.99/0.04

Analysis of the overall effects. Compared with "+Pseudo views," Ours changes CLIP-Similarity/ PSNR/LPIPS from 0.84/19.84/0.10 to 0.84/20.01/0.10 on RealFusion15 and from 0.74/28.43/0.04 to 0.75/28.99/0.04 on Artifact, with PSNR increases of 0.17 and 0.56 dB, respectively. Compared with "+Multi-token," Ours increases CLIP-Similarity from 0.82 to 0.84 and PSNR by 0.38 dB on RealFusion15, and increases PSNR by 1.90 dB on Artifact, while CLIP-Similarity and LPIPS on Artifact are identical at the precision displayed in the table. The results support complementary gains from the two types of conditioning on some datasets and metrics, but do not indicate that the combined method is optimal on every dataset and metric. For example, "+Multi-token" achieves a higher PSNR on NeRF4.

base weight. The dynamic strategy uses the empirical base weights and progress-based schedule described in Section 3.3. All three settings use the same pseudo views, sampling order, and training budget, changing only the pseudo-view weighting strategy.

Table 6 shows that, among the listed settings, fixed high weights yield lower CLIP-Similarity and PSNR and higher LPIPS. Fixed low weights achieve the lowest LPIPS of 0.038, but lower CLIP-Similarity and PSNR than the dynamic strategy. Dynamic weights achieve the highest CLIP-Similarity of 0.751 and PSNR of 28.99 dB, with an LPIPS of 0.042, slightly worse than that of fixed low weights. These final metrics indicate that the dynamic strategy achieves higher semantic similarity and pixel fidelity in the reference view in this comparison, with a trade-off in perceptual quality. They cannot independently quantify training stability or determine which specific type of cross-view conflict causes the degradation.

Table 6 Ablation results for pseudo-view weight scheduling strategies

Method	Pseudo-view weighting strategy	CLIP-Similarity	PSNR(dB)	LPIPS	Metric summary
Fixed high	Base weight fixed at 1	0.413	13.38	0.496	Worse on all three metrics
Fixed low	Base weight fixed at 0	0.728	26.80	**0.038**	Lowest LPIPS
Dynamic	Dynamic weights	**0.751**	**28.99**	0.042	Highest CLIP-Similarity and PSNR

4.3.4 Comparison of single-view sampling and anchored dual-view sampling

The reference-view sampling method determines which supervision is included in a single known-view update. We compare two known-view sampling strategies: single-view sampling and anchored dual-view sampling. The single-view configuration samples only one pseudo-reference image during a pseudo-supervision event. The dual-view configuration renders both the real reference view and one randomly selected pseudo view within the same loss computation. The latter does not increase the batch size of SDS at random novel views; instead, it ensures that every update with pseudo supervision retains the real input as an anchor.

In Table 7, the single-view configuration runs for 5000 steps in both the coarse NeRF stage and the DMTet refinement stage, while the anchored dual-view configuration runs for 4000 steps in each stage. The reported runtime is the total time for the coarse and refinement stages. The final metrics of the two configurations are similar: with dual-view sampling, CLIP-Similarity decreases from 0.756 to 0.751, PSNR increases from 28.56 to 28.99 dB (+0.43 dB), and LPIPS remains 0.042 for both. The total runtime across the two stages decreases from 36.85 min to 30.52 min, a reduction of 6.33 min, or approximately 17.18%, while the number of steps in each stage decreases by 20.00%. These results describe differences in metrics and runtime under the listed run configurations. Because the two configurations use different run budgets and no common convergence criterion is provided, these results alone cannot establish faster convergence or faster iterations with dual-view sampling. In addition, the comparison changes both real-anchor retention and the number of views during updates with pseudo supervision, so their independent effects cannot be separated.

Table 7 Ablation results for anchored dual-view sampling

Method	Sampling strategy	CLIP-Similarity↑	PSNR(dB)↑	LPIPS↓	Steps per stage	Total runtime (min)
Ours-single	Single-view sampling	0.756	28.56	0.042	5000	**36.85**
Ours-dual	Dual-view sampling	0.751	28.99	0.042	4000	**30.52**

5 Conclusion

We present TriMagic, a single-image 3D generation framework that integrates multiple tokens with semantic roles and dynamic pseudo-multiview supervision. The multiple tokens organize shape, color, and texture into complementary instance-specific conditioning, providing fine-grained semantic and appearance constraints for random novel views. Zero123++ pseudo-multiview images provide generated cues for side and rear views. We control the contribution of pseudo supervision to optimization through

empirical view-dependent base weights, a ramp-up–hold–decay schedule after warm-up, and absolute weighting of separate real and pseudo-view groups. During updates with pseudo supervision, we use anchored dual-view sampling comprising the real reference image plus one randomly selected pseudo view. Stable Diffusion guidance and multi-reference Zero123 guidance at random novel views jointly cover camera directions beyond the discrete pseudo views. Compared with Magic123, TriMagic increases CLIP-Similarity on RealFusion15 from 0.82 to 0.84 and PSNR from 19.50 to 20.01 dB (+0.51 dB). On Artifact, PSNR increases from 27.58 to 28.99 dB (+1.41 dB), but LPIPS increases from 0.03 to 0.04 at two decimal places, indicating a trade-off in perceptual quality. Qualitative rear-view examples show that some structural repetition and texture problems are alleviated, but cannot demonstrate the recovery of true unseen geometry. In the sampling ablation, the dual-view configuration, with 4000 steps in each of the coarse and refinement stages, achieves final metrics similar to those of the single-view configuration with 5000 steps in each stage, and also has a lower total runtime across the two stages. This comparison does not yet constitute a speed evaluation under a common convergence criterion. These results provide experimental evidence to inform the generation of digital display content from a single image of a cultural heritage object. In application, information visible in the reference image must still be distinguished from unseen content inferred by generative priors.

This study has the following limitations. First, Zero123++ pseudo-multiview images are generated observations whose structure, textures, and occlusion relationships may be inconsistent with the target instance. The current base weights in the CSV file are preset empirically by view and cannot yet adaptively estimate uncertainty from the content of individual images. Second, dynamic scheduling can limit the overall strength of pseudo supervision, but cannot completely eliminate local conflicts among pseudo views. Texture seams, geometric misalignment, and deterioration in LPIPS may therefore still occur. Finally, single-image data lack real observations of unseen regions. Our evaluation mainly relies on reference-view metrics and displays of selected rear views, which are insufficient to rigorously establish the authenticity of rear-surface geometry. In addition, Artifact includes only 24 tested instances selected from different categories, so the conclusions cannot be directly generalized to all types of cultural heritage objects. We do not report variation across repeated runs, and the available means are insufficient to assess the statistical stability of small metric differences. The current application is therefore the generation of cultural heritage object appearance and digital displays conditioned on a single image; the generated results cannot yet serve as a basis for measurement-grade 3D records. Future work will investigate learnable confidence estimation based on content quality, occlusion, and cross-view consistency, introduce conflict detection and region-level weights, and evaluate the geometric and texture accuracy of unseen regions using data with real multiview or 3D scan ground truth.

Data availability

The Artifact dataset constructed for this study is publicly available, and access is provided.

Code availability

The custom code for this study is publicly available, and access is provided.

Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant No. 62272426), the open-competition project under the Shanxi Province Science and Technology Major Special Project (Grant No. 202201150401021), and the Shanxi Provincial Key R&D Program (Grant No. 202402020101001).

Author contributions

Min Pang contributed to the conceptualization and overall design of TriMagic, the design of the semantic-role multi-token conditioning strategy, and the drafting of the manuscript. Xiaokang Cui developed the algorithmic framework, implemented the semantic-role multi-token conditioning module, integrated the constituent modules, and implemented the overall experimental pipeline. Shichao Jiao contributed to the design and implementation of the dynamically weighted pseudo-multi-view supervision and anchored dual-view sampling mechanisms. Yajun Li established the experimental platform, conducted the comparative experiments, and reviewed and revised the manuscript. Fengguang Xiong curated the dataset, prepared the experimental data, coordinated the experimental work, and handled correspondence.

Competing interests

The authors declare no competing interests.

Statement on AI assistance

This study used ChatGPT to assist with translation and language polishing, and to check the logical and content consistency of the manuscript. The authors are responsible for the research judgments and the content of the manuscript.

References

[1] Remondino, F. Heritage Recording and 3D Modeling with Photogrammetry and 3D Scanning. *Remote Sens.* **3**, 1104–1138 (2011). URL https://doi.org/10.3390/rs3061104.

[2] Melas-Kyriazi, L., Laina, I., Rupprecht, C. & Vedaldi, A. *RealFusion: 360° Reconstruction of Any Object From a Single Image.* Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 8446–8455 (IEEE, 2023).

[3] Qian, G. *et al. Magic123: One Image to High-Quality 3D Object Generation Using Both 2D and 3D Diffusion Priors.* Proc. International Conference on Learning Representations (2024).

[4] Liu, R. *et al. Zero-1-to-3: Zero-shot One Image to 3D Object.* Proc. IEEE/CVF International Conference on Computer Vision, 9298–9309 (IEEE, 2023).

[5] Gal, R. *et al. An Image is Worth One Word: Personalizing Text-to-Image Generation using Textual Inversion.* Proc. International Conference on Learning Representations (2023).

[6] Poole, B., Jain, A., Barron, J. T. & Mildenhall, B. *DreamFusion: Text-to-3D using 2D Diffusion.* Proc. International Conference on Learning Representations (2023).

[7] Lin, C.-H. *et al. Magic3D: High-Resolution Text-to-3D Content Creation.* Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 300–309 (IEEE, 2023).

[8] Ruiz, N. *et al. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation.* Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 22500–22510 (IEEE, 2023).

[9] Zhang, Y. *et al.* ProSpect: Prompt Spectrum for Attribute-Aware Personalization of Diffusion Models. *ACM Trans. Graph.* **42**, 244 (2023). URL https://doi.org/10.1145/3618342.

[10] Agarwal, A., Karanam, S., Shukla, T. & Srinivasan, B. V. *An Image is Worth Multiple Words: Multi-attribute Inversion for Constrained Text-to-Image Synthesis.* Proc. Winter Conference on Applications of Computer Vision, 6053–6062 (IEEE, 2025).

[11] Butt, M. A., Wang, K., Vazquez-Corral, J. & van de Weijer, J. *ColorPeel: Color Prompt Learning with Diffusion Models via Color and Shape Disentanglement.* Computer Vision – ECCV 2024, 456–472 (Springer, 2025). URL https://doi.org/10.1007/978-3-031-72667-5_26.

[12] Wang, N. *et al. Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images.* Computer Vision – ECCV 2018, 55–71 (Springer, 2018). URL https://doi.org/10.1007/978-3-030-01252-6_4.

[13] Liu, Y. *et al. SyncDreamer: Generating Multiview-consistent Images from a Single-view Image.* Proc. International Conference on Learning Representations (2024).

[14] Shi, R. *et al.* Zero123++: a Single Image to Consistent Multi-view Diffusion Base Model (2023). Preprint at https://arxiv.org/abs/2310.15110.

[15] Lin, Y. *et al. Consistent123: One Image to Highly Consistent 3D Asset Using Case-Aware Diffusion Priors*. Proc. 32nd ACM International Conference on Multimedia, 6715–6724 (ACM, 2024). URL https://doi.org/10.1145/3664647.3680994.

[16] Sun, J. *et al. DreamCraft3D: Hierarchical 3D Generation with Bootstrapped Diffusion Prior*. Proc. International Conference on Learning Representations (2024).

[17] Schonberger, J. L. & Frahm, J.-M. *Structure-From-Motion Revisited*. Proc. IEEE Conference on Computer Vision and Pattern Recognition, 4104–4113 (IEEE, 2016).

[18] Mildenhall, B. *et al. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis*. Computer Vision – ECCV 2020, 405–421 (Springer, 2020). URL https://doi.org/10.1007/978-3-030-58452-8_24.

[19] Niemeyer, M. *et al. RegNeRF: Regularizing Neural Radiance Fields for View Synthesis from Sparse Inputs*. Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 5480–5490 (IEEE, 2022).

[20] Kerbl, B., Kopanas, G., Leimkühler, T. & Drettakis, G. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.* **42**, 139 (2023). URL https://doi.org/10.1145/3592433.

[21] Shen, T., Gao, J., Yin, K., Liu, M.-Y. & Fidler, S. *Deep Marching Tetrahedra: a Hybrid Representation for High-Resolution 3D Shape Synthesis*. Adv. Neural Inf. Process. Syst., Vol. 34, 6087–6101 (2021).

[22] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K. & Koltun, V. Towards Robust Monocular Depth Estimation: Mixing Datasets for Zero-shot Cross-dataset Transfer. *IEEE Trans. Pattern Anal. Mach. Intell.* **44**, 1623–1637 (2022). URL https://doi.org/10.1109/TPAMI.2020.3019967.

[23] Eftekhar, A., Sax, A., Malik, J. & Zamir, A. *Omnidata: A Scalable Pipeline for Making Multi-Task Mid-Level Vision Datasets From 3D Scans*. Proc. IEEE/CVF International Conference on Computer Vision, 10786–10796 (IEEE, 2021).

[24] Ranftl, R., Bochkovskiy, A. & Koltun, V. *Vision Transformers for Dense Prediction*. Proc. IEEE/CVF International Conference on Computer Vision, 12179–12188 (IEEE, 2021).

[25] Radford, A. *et al. Learning Transferable Visual Models From Natural Language Supervision*. Proc. 38th International Conference on Machine Learning, Vol. 139, 8748–8763 (PMLR, 2021).

[26] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. *High-Resolution Image Synthesis with Latent Diffusion Models*. Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 10684–10695 (IEEE, 2022).

[27] Ho, J. & Salimans, T. Classifier-Free Diffusion Guidance (2022). Preprint at https://arxiv.org/abs/2207.12598.

[28] Cherti, M. *et al. Reproducible scaling laws for contrastive language-image learning*. Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2818–2829 (IEEE, 2023).

[29] Zhang, R., Isola, P., Efros, A. A., Shechtman, E. & Wang, O. *The Unreasonable Effectiveness of Deep Features as a Perceptual Metric*. Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition, 586–595 (IEEE, 2018).

[30] Jun, H. & Nichol, A. Shap-E: Generating Conditional 3D Implicit Functions (2023). Preprint at https://arxiv.org/abs/2305.02463.

[31] Seo, J. *et al. Let 2D Diffusion Model Know 3D-Consistency for Robust Text-to-3D Generation*. Proc. International Conference on Learning Representations (2024).

[32] Tang, J. *et al. Make-It-3D: High-Fidelity 3D Creation from A Single Image with Diffusion Prior*. Proc. IEEE/CVF International Conference on Computer Vision, 22819–22829 (IEEE, 2023).