Multi-Layer Lighting-Aware Style Transfer with Guidance Map Transformation

Muhammad Arief
Department of Media Science, Tokyo University of Technology, Hachioji, Tokyo, Japan
mail.muhammadarief@gmail.com (corresponding author)

Hideki Todo
Department of Computer Science, Faculty of Engineering, Takushoku University, Tokyo, Japan
htodo@cs.takushoku-u.ac.jp

Koji Mikami
Department of Media Science, Tokyo University of Technology, Hachioji, Tokyo, Japan
mikami@stf.teu.ac.jp

Mochamad Hariadi
Department of Computer Engineering, Institut Teknologi Sepuluh Nopember Surabaya, Indonesia
mochar@its.ac.id

Received: 19 July 2025 | Revised: 26 August 2025 | Accepted: 6 September 2025

DOI: https://doi.org/10.48084/etasr.13520

ABSTRACT

The present study addresses the challenge of animating high-frequency stylized details without introducing deforming artifacts or losing lighting impressions. While Lit-Sphere reliably produces coherent low-frequency shading, it often distorts the fine patterns in 3D animations. Conversely, StyleBlit can transfer high-frequency style elements but fails to animate the dynamic lighting effects. The proposed hybrid pipeline combines Lit-Sphere for low-frequency shading with a patch-based transfer inspired by StyleBlit. To achieve the lighting control, common normal maps are employed as guidance, enabling practical lighting control through normal map transformations. Additionally, local editing, posterization, and normal map smoothing are investigated to expand the range of achievable animation effects.

Keywords-rendering; non-photorealistic rendering; style transfer; lit-sphere; patch-based transfer

I. INTRODUCTION

Animating expressive styles from 2D textures poses a significant challenge: point-wise transfer often deforms high-frequency patterns, while patch-based techniques preserve these details but lack control over dynamic lighting. This is a critical limitation, as lighting is a decisive driver of visual impact, capable of influencing emotional perception [1]. Existing stylized shading techniques reflect this trade-off. Lit-Sphere [2] produces coherent low-frequency shading but distorts fine details in animation due to its static mapping, while StyleBlit [3] excels at high-frequency pattern transfer but cannot simulate dynamic lighting, as highlights and shadows remain statically glued to the view-dependent normal orientation. The proposed approach mitigates the limitations of previous techniques by introducing dynamic lighting effects while preserving the high-frequency patterns created by the artist in the original 2D texture. Specifically, normal map rotations are applied to simulate directional lighting changes, which allows for more flexible and accurate control over the stylized shading, ensuring that even the most detailed aspects of the 2D texture are animated smoothly.

Several approaches have been employed to enhance transfer outcomes and visual consistency, particularly in animated sequences. Local editing enables precise brightness adjustment in specific normal map regions, enhancing artistic versatility. Posterization employs K-means clustering to segment the normal map into flat regions, producing a more artistic, 'low-polygon' appearance with diminished shading. Gaussian filtering reduces noise and softens lighting transitions by altering normal map distribution, ensuring seamless and coherent lighting and texture across animation frames.

The key contributions of the present work are: dynamic lighting control via normal map transformations, artifact-free preservation of high-frequency patterns, and integration of local editing, posterization, and smoothing for visual coherence and flexibility in animations. A fundamental aspect of non-photorealistic rendering is cartoon shading, which employs a straightforward 1D texture lookup for style conversion. This

1D texture reference has enabled the development of various style extensions, including illustrative rendering [4] and dynamic stylized shading primitives [5]. The X-Toon method further expanded this approach by incorporating secondary attribute effects into a 2D texture reference. In the context of brush stroke styles, early example-based stylization techniques for 2D texture representation [6, 7] were proposed to model brush stroke effects using samples derived from painted images.

Deep learning style transfer [8-11] utilizes pretrained neural networks to extract abstract features from an example image, meticulously blending them onto another image. This method offers immense flexibility and control but requires significant computation due to the iterative optimization needed to match style image feature representations. In contrast, example-based methods [3, 12-14] avoid the need for pretrained models by directly copying and manipulating surface details from example 2D texture images, applying style with speed and specificity. This is particularly advantageous for matching precise style features, as 2D image textures offer rich detail for artists to incorporate. Moreover, artists often already have a foundational guidance map, such as a normal map, and they may want to explore how a specific style can be accurately transferred into an animation, ensuring that the style aligns precisely with the existing guidance.

Deep learning methods, when trained on extensive datasets, may effectively relight novel objects. Authors in [15] employ deep learning to generate authentic shadow effects in hand-drawn sketches, enhancing depth and perception in 2D images. RGB geometry is employed to transform the historical brushstrokes of artists into lighting effects through an innovative technique. Authors in [16] stated that this method employs fundamental color gradations to stylize lighting effects, resulting in a cohesive image. This approach emphasizes stylized illumination rather than intricate detail. Unlike deep learning approaches focusing on stylized illumination, the present work produces continuous, artistic lighting while preserving intricate high-frequency details. This study's guided patch-based synthesis uses straightforward computation to enhance the coherence and detail of styled lighting effects, distinguishing it from other solutions.

Example-based methods like Lit-Sphere (also known as MatCap [2]) are fundamental for creating stylized 3D scenes. Artists draw expressive shading on a spherical texture, which is then transferred to the 3D model via normal correspondence. However, simple normal correspondence leads to a static lighting appearance. To address this limitation, the Lit-Sphere extension [17] introduced a light-oriented texture projection mechanism . However, Lit-Sphere's parametric nature results in deforming the artifacts on high-frequency details, as shown in Figure 1. In the Lit-Sphere approach, each pixel in the target image is directly mapped to a corresponding pixel in the style image based on the guidance images. This is a one-to-one mapping, where the color value of each target pixel is determined by its counterpart in the style image.

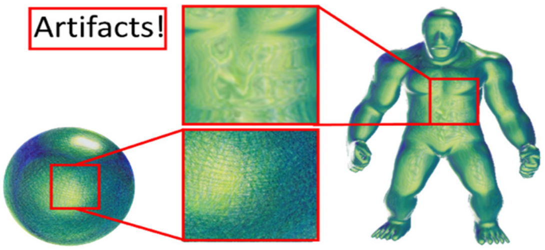

Fig. 1. The limitation of Lit-Sphere on transferring high-frequency details.

Fig. 2. To avoid these deforming artifacts, guided patch-based style transfer techniques [3,12] are employed. StyLit [12] utilizes a set of illumination guidance maps to achieve lighting changes, while StyleBlit offers faster transfer mechanisms but sacrifices some lighting control features, as displayed in Figure 2. Lighting control with high-frequency details is key to achieving stylized 3D shading. The current study expands upon the guided patch-based style transfer concept from StyleBlit, drawing from the principles of light-oriented Lit-Sphere [17]. The limitation of StyleBlit lies in its inability to adjust the style to simulate lighting effects dynamically. It cannot animate the style exemplar to reflect lighting changes, such as a moving light source shifting bright and dark regions. Consequently, bright regions highlighted with red ellipses remain fixed relative to the viewer, preventing the perception of dynamic illumination. The right image shows a bright green patch representing a typical bright region on the character.

Fig. 3. StyleBlit's inability to dynamically adjust the style to simulate lighting effects. The right image shows a bright green patch representing a typical bright region on the character.

In the original implementation of StyleBlit, the style transfer process occurs in 3D space, where the style and normal map are intrinsically tied to the 3D spatial location and geometry of the object. Each normal vector is specifically oriented based on its position on the object surface. Any attempt to rotate the normal map within this 3D context can significantly disrupt the process, as the spatial relationship between the normal map and object surface is altered, leading to misalignment and inconsistencies. The clustering process, which also operates within this 3D space, relies on the fixed spatial relationships of the object surface. Therefore, rotating the object or its normal map can break these clusters apart, leading to visual artifacts and diminishing the effectiveness of the style transfer. In contrast, this study's implementation works in image space, treating the normal map as a 2D projection, detached from complex 3D spatial relationships.

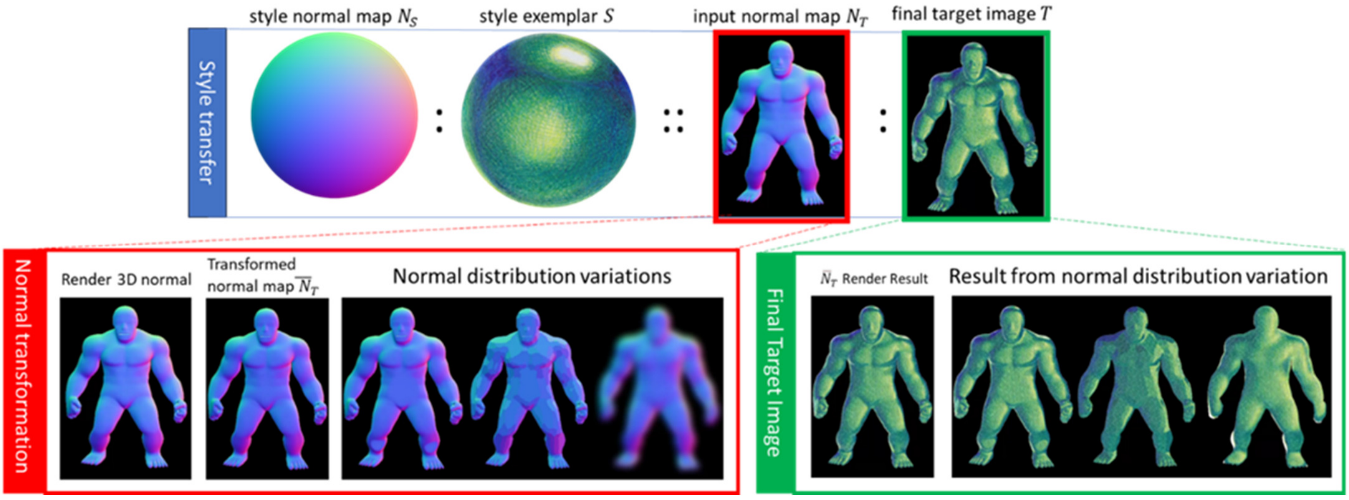

Fig. 4. Proposed method pipeline.

This approach simplifies manipulations, like rotation, making it easier to adjust the normal map without disrupting the overall shading. StyleBlit, as originally proposed in [3], implements a view-normal–guided patch transfer entirely on the GPU. For each fragment, the method encodes the view-space normal into a lookup coordinate via a precomputed normal-to-UV table, samples the style exemplar, and applies splatting with vote-based smoothing. This design excels at preserving detail. However, StyleBlit rendering pipeline deliberately disables all lighting computations and never evaluates a dot product between surface normal and light direction, thus every highlight and shadow baked into the exemplar remains fixed to its original normal orientation. Consequently, the stylization cannot respond to changes in illumination direction, producing a non-adaptive appearance conflicting with the perceptual expectations of dynamic shading.

In contrast, the proposed framework augments the patch-based transfer paradigm with an explicit simulation of light motion. For each output frame, this study applies a smoothly varying rotation to the normal map guidance, emulating a moving light source. This temporally evolving normal field drives a two-stage transfer: a base-layer pass capturing coarse illumination gradients and a detail-layer clustering pass sampling strokes according to the updated normal map. Since the normal map guidance changes frame by frame, the proposed method produces stylized outputs with coherently shifting highlights and shadows. To clarify this distinction, Table I summarizes the key differences between the proposed method and existing techniques.

II. RESEARCH METHODOLOGY

Figure 3 displays a three-dimensional model, a two-dimensional style texture, and a normal map. The normal vector can be calculated directly from the 3D surface, or a normal map can be projected using the Lit-Sphere approach [2,17]. Once the normal map is acquired, 2D images are generated for subsequent processing and analysis. The color distribution of the normal map is subsequently altered employing methods, such as K-means clustering for posterization, Gaussian filtering for smoothing, and surface normal vector transformation [18] for localized value modifications. The guidance normal map is animated by applying a rotation matrix to the normal vector encoded in the 2D image as a transformed guidance map. Finally, the stylized appearance is synthesized using guided patch-based style transfer.

TABLE I. COMPARISON OF EXISTING METHODS WITH THE PROPOSED METHOD

Description	[2]	[3]	Proposed
Low-frequency preservation	✓	✓	✓
High-frequency preservation	×	✓	✓
Lighting Interaction	×	×	✓
Post-Stylization Effects	×	×	✓
Easy Style Transfer to Various Shapes	✓	✓	✓

A. *Normal Map Distribution Variation*

This study manipulates the distribution of the rotated normal map, $\tilde{\mathbf{N}}_T$, through transformations, like rotation, smoothing, and posterization, to simulate dynamic lighting and preserve high-frequency details, thereby ensuring visual coherence and enabling localized edits.

1) *Local Editing Normal Map*

Local editing uses region control [18] to modify the z coordinates of normals, creating brightness changes. At this point, pseudo point light effects are added to brighten specific locations.

2) *Posterization Normal Map*

For posterization, this work investigates how segmented normal maps affect style transfer, and simulates low-polygon effects in the output images. The normal map is segmented using K-means clustering, being divided into regions, each with a unique normal vector.

3) Smoothing Normal Map

Smoothing also alters the distribution of the normal map, affecting the behavior of the patch-based style transfer process. The current work investigates the extreme values of smoothing the guidance map. By applying a Gaussian blur filter with a large kernel size, a very smooth guidance map can be created, reducing the number of unique regions.

B. Normal Map Rotation for Lighting

This process is crucial for integrating lighting control with Lit-Sphere and patch-based style transfer techniques. The original normal map $\mathbf{N}_T(\mathbf{p})$ transforms achieved using a rotation matrix, resulting in a transformed normal map $\widetilde{\mathbf{N}}_T(\mathbf{p})$. This rotation matrix is constructed through common Euler rotations in xy order without the z-axis, influenced by the angles θ and ϕ, respectively. The resulting rotated normal $\widetilde{\mathbf{N}}_T$ is obtained by:

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_y(\phi)\mathbf{R}_x(\theta) \\ \widetilde{\mathbf{N}}_T(\mathbf{p}) &= \mathbf{R}\mathbf{N}_T(\mathbf{p}) \end{aligned} \tag{1}$$

In the proposed method, the shared rotation matrix $\mathbf{R} = \mathbf{R}_y(\phi)\mathbf{R}_x(\theta)$ is applied to each normal orientation $\widetilde{\mathbf{N}}_T(\mathbf{p})$ across all image pixels.

C. Multi-Layer Style Transfer

With the given rotated normal map $\widetilde{\mathbf{N}}_T$, the guided patch-based style transfer is applied in a manner like StyleBlit [3]. This transfer is performed separately for the low-frequency base layer and the high-frequency detail layer, as illustrated in Figure 4.

$$\begin{aligned} B_S, D_S &= \text{separateBaseDetail}(S) \\ B_T &= \text{baseTransfer}\left(B_S, \mathbf{N}_S, \widetilde{\mathbf{N}}_T\right) \\ D_T &= \text{detailTransfer}\left(D_S, \mathbf{N}_S, \widetilde{\mathbf{N}}_T\right) \\ T &= B_T + D_T \end{aligned} \tag{2}$$

where S is the style exemplar, B_S is the style base layer, D_S is the style detail layer, B_T represents the target base layer, D_T is the target detail layer, T is the final target image, N_S is the style normal map and N_T is the input normal map.

1) Separate Base Layer and Detail Layer

The Gaussian filter is used to create the smooth base layer and extract the detail layer as the difference.

$$\begin{aligned} B_S(\mathbf{p}) &= \text{gaussianFilter}(S, \mathbf{p}, \sigma) \\ D_S(\mathbf{p}) &= S(\mathbf{p}) - B_S(\mathbf{p}) \end{aligned} \tag{3}$$

where σ is a parameter in the Gaussian filter that controls the width of the kernel, determining the level of smoothing applied. It influences the separation of the image into base and detail layers by adjusting the balance between low-frequency (base) and high-frequency (detail) components.

a) Base Layer Transfer

To make a correspondence between the target location $\mathbf{p}$ and style location $\mathbf{u}$ for the base layer, Lit-Sphere mapping [2] for $\widetilde{\mathbf{N}}_T$ is employed:

$$\begin{aligned} \mathbf{u}_q &= \text{litsphereCoords}(\widetilde{\mathbf{N}}_T(\mathbf{p})) \\ &= ((\widetilde{\mathbf{N}}_{Tx}(\mathbf{p}) + 1)/2, (1 - \widetilde{\mathbf{N}}_{Ty}(\mathbf{p}))/2) \\ B_T(\mathbf{p}) &= B_S(\mathbf{u}) \end{aligned} \tag{4}$$

where $\mathbf{u} \in [0, 1]$ denotes the coordinates in the style base layer, obtained by converting the target guide normal $\widetilde{\mathbf{N}}_T(\mathbf{p})$ to pick corresponding face orientations on the sphere. Lit-Sphere mapping is efficient because it does not require an argmin search to determine normal correspondences.

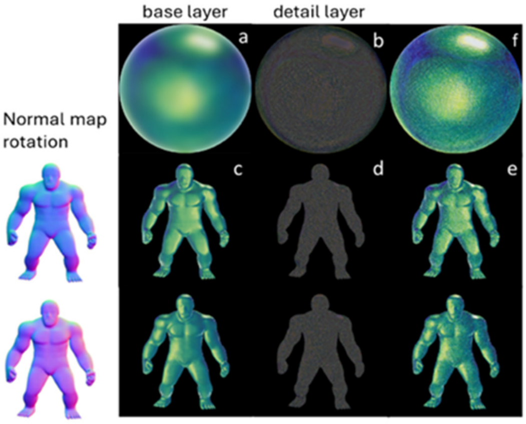

Fig. 5. Multi-layer style transfer approach: Decomposing the style exemplar (f) into base (a) and detail (b), applying the Lit-Sphere for the base (c), the proposed patch-based on the detail (d), and finally, composing the result to preserve both smoothness and finally, composing the result to preserve both smoothness and high-frequency details (e).

b) Detail Layer Transfer

Unlike StyleBlit's original hierarchical grid-based correspondence, this works' implementation explicitly creates cluster regions based on the input normal map $\widetilde{\mathbf{N}}_T$. A straightforward K-means clustering approach is deployed for a set of normal and location features:

$$\{\widetilde{\mathbf{N}}_{ci}\}, \{\mathbf{q}_{ci}\} = \text{kmeansClustering}\left(\widetilde{\mathbf{N}}_T, \mathbf{p}, w_p, k\right) \tag{5}$$

where the clustering is applied to a feature vector $\mathbf{F}(\mathbf{p}) = (\widetilde{\mathbf{N}}_T(\mathbf{p}), w_p(\mathbf{p}))$ to obtain k cluster centers for guidance normal map $\{\widetilde{\mathbf{N}}_{ci}\}$ and locations $\{\mathbf{q}_{ci}\}$. The parameter w_P in the detail layer transfer controls the weighting of the spatial coordinates relative to normal features during k-means clustering.

Guided patch-based transfer is then applied using this cluster information. For a location $\mathbf{p}$ belonging to cluster ci (i.e., with center location $\mathbf{q} := \mathbf{q}_{ci}$ and center guide normal $\widetilde{\mathbf{N}}_q := \widetilde{\mathbf{N}}_{ci}$), this study determines the location $\mathbf{u}_q$ in style exemplar S with Lit-Sphere coordinates, as described in (4). $\mathbf{u}_q$ is used as the correspondence center and $\mathbf{u}$ is calculated by being offset based on the difference between the target location $\mathbf{p}$ and the cluster center location $\mathbf{q}$:

$$\begin{aligned} \mathbf{u}_q &= \text{litsphereCoords}(\widetilde{\mathbf{N}}_q) \\ \mathbf{u} &= \left[\mathbf{u}_q + \alpha(\mathrm{p} - \mathrm{q})\right] \\ D_T(\mathbf{p}) &= D_S(\mathbf{u}) \end{aligned} \tag{6}$$

where the function $\lfloor x \rceil$ clamps the values within the range $[0, 1]$ to prevent the lookups outside this range. The target detail layer $D_T(\mathbf{p})$ is transferred from the style detail layer $D_S(\mathbf{u})$ with correspondence between $\mathbf{p}$ and $\mathbf{u}$. The final image is obtained as $T(\mathbf{p}) = B_T(\mathbf{p}) + D_T(\mathbf{p})$.

III. RESULTS

The proposed method was implemented in MATLAB for multi-layer style transfer, posterization, and smoothing of the normal map. The latter was initially prepared in Autodesk Maya, which provided a convenient tool for local editing. The present sudy defined the number of target frames and set the rotation angles for each frame.

The keyframes were evenly spaced across the sequence using a linear distribution method, ensuring uniform intervals between them. These rotation angles were smoothly interpolated between the keyframes using spline techniques, resulting in a seamless transition across the frames. In the presented multi-layer style transfer approach, modifications to the guidance map, such as rotating the normal map, directly influence the lighting effects and texture assignments from the source to the target. This is achieved through a two-phase process involving both the base layer and the detail layer. For the base layer, the guidance map affects how lighting is mapped from the style base layer to the target. Specifically, the guidance map serves to compute target coordinate mappings, modifying how the base lighting patterns and shading are transferred. The patch-based transfer step for detail layers employs the guidance map to determine clustering assignments and feature points. The guide texture influences the clustering centers and feature computations, determining the sampled and transmitted regions of the source detail layer. Consequently, texture changes influence the grouping assignments, guiding attributes, and detail layer transfer outcomes.

The current work consistently uses σ=0.03 (3% of image width) for separating base layer, k=200, w_p=4.0 for clustering, and α=1.0 for detail layer transfer in all results. The proposed system operates primarily in an offline rendering environment. The pipeline consists of normal map rotation, multi-layer style transfer, normal map smoothing, and posterization, all executed on a frame-by-frame basis, where each rendered 2D image is processed independently across all animation frames. In the local transformation stage, processing is carried out within a 3D environment. A normal map is first rendered as a 2D image from the 3D scene; then a rotation matrix is applied to simulate lighting variations.

By rotating the normal map, multiple target normal frames can be generated from a single input, to produce a stylized animation sequence. To evaluate the computational performance, the proposed method processes 32 frames to generate a 3-second animation, requiring 16 s of total computation time. The baseline result of the proposed method is demonstrated in Figure 5(a). We will further discuss the behavior of the transfer results effects of local editing, Figure 6(a), smoothing (Figure 6(b) and posterization Figure 6(c) on the transfer results. To see the entire animation, please refer to the supplemental video.

Figure 5 presents a comprehensive comparison of the proposed method with baseline techniques and style variations. Figure 5(a) shows that normal map rotation produces temporally coherent results in both low- and high-frequency layers. Figure 5(b) highlights how the proosed method significantly reduces artifacts compared to Lit-Sphere when applied to detailed regions like the chest and leg. Figure 5(c) demonstrates the flexibility of the proposed pipeline, as it can adapt to diverse artistic media, preserving style integrity in stroke-based and texture-rich exemplars. Figure 5(b) compares this study's results with the Lit-Sphere approach. Artifacts visible in the baseline output illustrate the limitations of static spherical texture mapping in preserving high-frequency details. In contrast, the second row illustrates the effectiveness of multi-layer style transfer applied to rotated normal maps, demonstrating improved high-frequency detail preservation. As illustrated in Figure 5(c), the proposed system can generate diverse shading styles using various media, such as stroke effects, traditional hatching tones, and dot patterns. The system maintains stable motion even with different high-frequency characteristics. Compared to state-of-the-art guided patch-based transfer techniques for 3D scenes [3,12], the proposed method produces visually comparable results with additional lighting features. Artists can utilize the introduced method by preparing a single spherical map for the style exemplar, encompassing high-frequency patterns like brush strokes. The proposed method effectively transfers both low and high-frequency components and offers lighting control through normal rotation. Moreover, the addition of simple post-processing operations, such as posterization, Gaussian smoothing, and localized normal-map transformations, provides flexible modulation of the output and enables a broad spectrum of the stylistic intents to be achieved.

A. Normal Map Distribution Control

1) Local Editing Normal Map

Figure 6(a) demonstrates the local editing of the normal map. The transformed normal map in the leg area results in increased brightness in the low-frequency layer, while high-frequency patterns are also varied by referencing the brighter samples without any deforming artifacts. In this scenario, the modified normal distributions can be applied to the illumination change, similar to other parts.

2) Posterization Normal Map

In the posterization case, as depicted in Figure 6(c), the normal map is segmented into flat regions, resulting in uniform lighting in the base layer. Meanwhile, detail transfer animates high-frequency patterns, like texture translation, due to the shifts in cluster center reference points. The synthesized patterns are preserved without deforming artifacts through patch sampling.

3) Smoothing Normal Map

In contrast, smoothing (Figure 6(b)) tends to blur surface details, resulting in rounder and larger clusters. Concerning the high-frequency elements, their primary characteristic is texture translation, which is similar to what happens in posterization, even though the normal distributions are quite different.

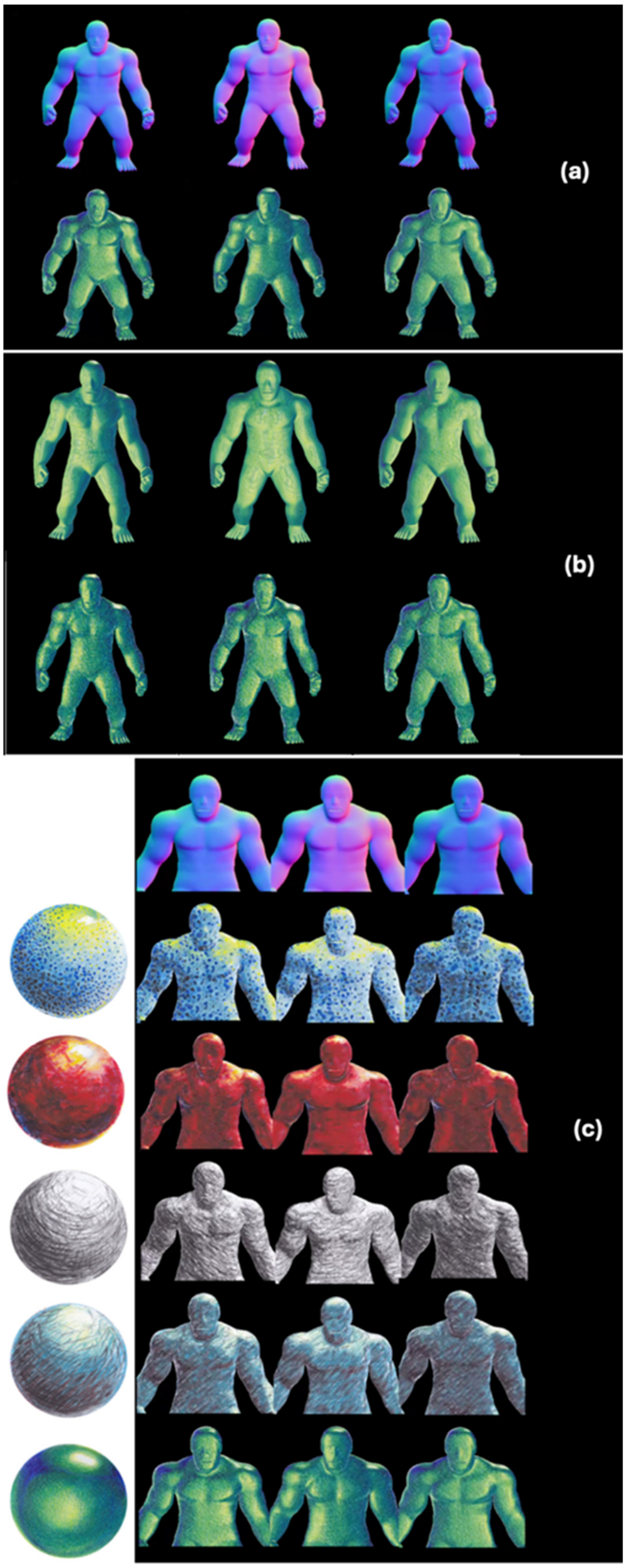

Fig. 6. (a) Baseline result using rotated normal map showing temporal coherence in stylized animation. (b) Comparison of Lit-Sphere and our method. (c) Stylization results using different exemplars.

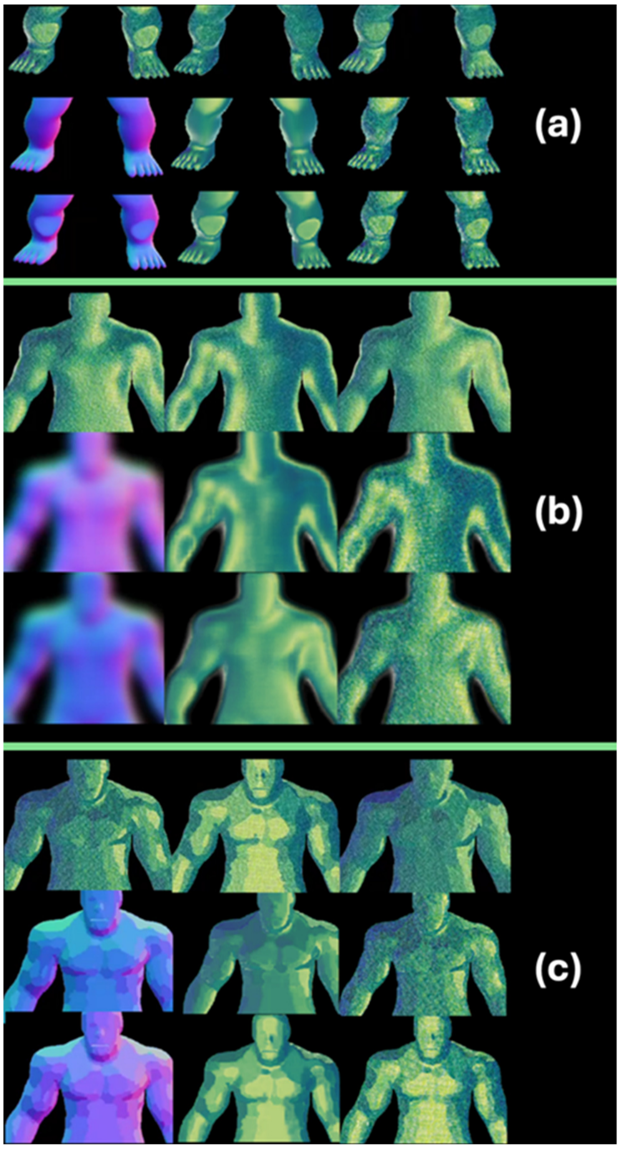

Fig. 7. (a) Local editing of the normal map brightens targeted regions, enhancing localized shading and detail variation. (b) Smoothing with Gaussian filtering creates soft lighting in the base layer and maintains temporal coherence. (c) Posterization segments the normal map into flat regions, producing graphic-style shading with animated detail transfer.

The pseudocode and experimental parameter table play a crucial role in clarifying the structure of the animation pipeline.

The pseudocode outlines the main procedural stages: normal map transformation, base-detail separation, clustering, and style transfer, offering a modular overview of the implementation. Table II summarizes the key experimental parameters used throughout the animation pipeline. The parameters include the number of frames, duration, normal map rotation angles, and core transfer settings, such as Gaussian smoothing (σ), patch scale (α), and number of clusters (k), for detail layer transfer. These values were selected based on preliminary tuning to balance the visual quality and computational efficiency.

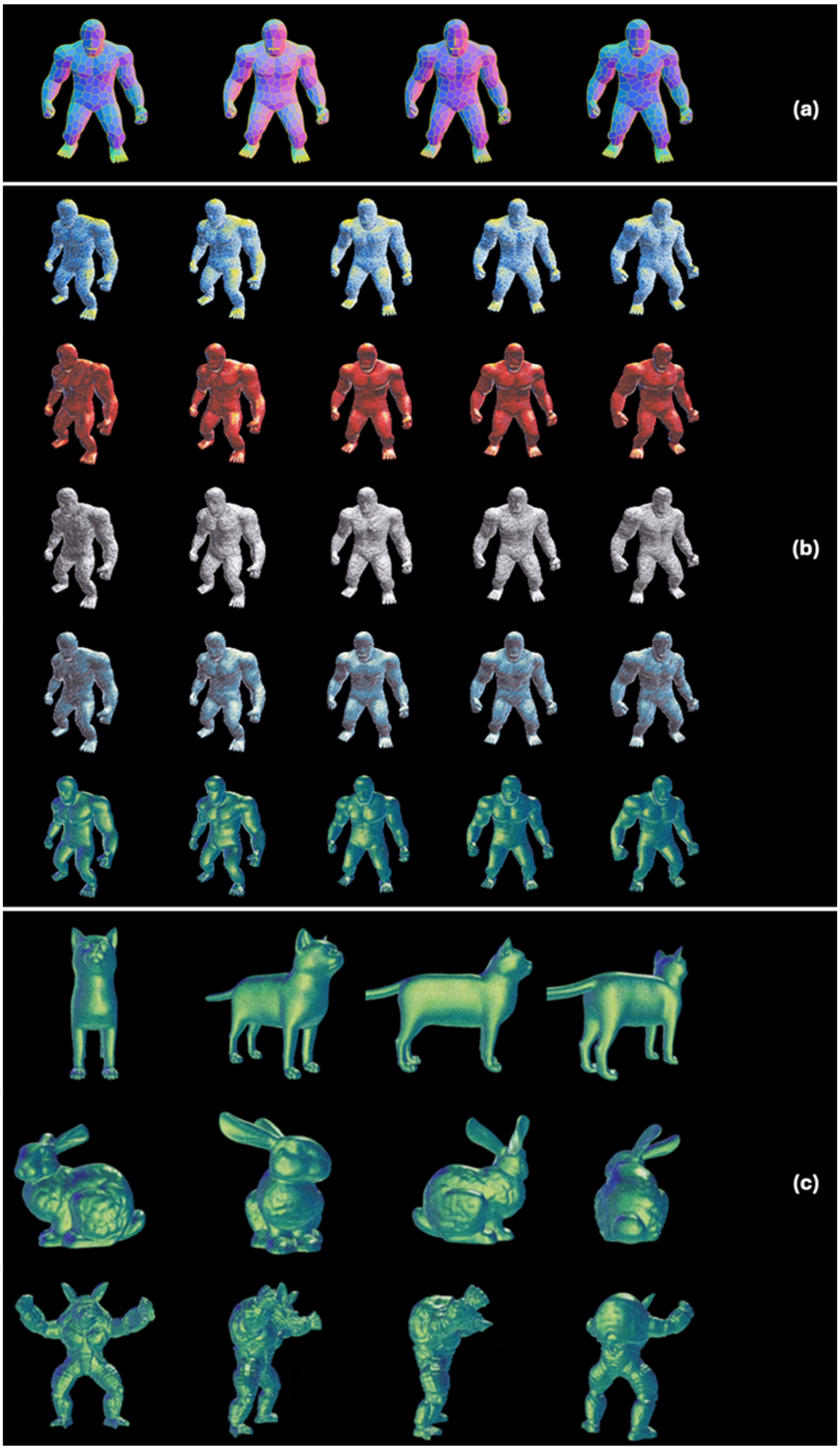

Fig. 8. (a) Cluster visualization from rotated normal maps across frames, showing stable boundaries for each segmented region. (b) Stylized animation under view transition, demonstrating temporally coherent detail layer movement. (c) Style transfer results across different shapes, illustrating consistent cluster-based detail alignment under dynamic perspectives.

```
Inputs: S,N0,F,keyframes,params{σ_s, k,w_p,α,edits}
Output: {T_f}

# Preprocess (once)
B_S = GAUSSIAN_BLUR(S,σ_s)
D_S = S - B_S
(Θ_f, Φ_f) = INTERPOLATE_ANGLES(keyframes)
for f = 1..F:
  R_f = RY(Φ_f[f]) * RX(Θ_f[f])
  N   = NORMALIZE(R_f * N0)
GUIDANCE EDITS (optional)
if edits.local_ops:
 N = LOCAL_EDIT(N, edits.local_ops);
 N = NORMALIZE(N)
if edits.posterize_K:
N = POSTERIZE_KMEANS(N, K=edits.posterize_K);
 N = NORMALIZE(N)
if edits.smooth_σ:
 N = GAUSSIAN_BLUR(N, edits.smooth_sigma);
 N = NORMALIZE(N)

#Base via Lit-Sphere
  for p in Ω:
   B_T[p] = SAMPLE(B_S, L(N[p]))
   #L: normal -> UV mapping
  end
#Detail via cluster-guided offset sampling
 for p in Ω:
 FEATURES[p] = CONCAT(N[p], w_p * POS(p))
 end
 (labels, centers) = KMEANS(FEATURES,k)
 #centers: n_c[c], q_c[c]

 for p in Ω:
  c  = labels[p]
  u  = CLAMP01(L(n_c[c])+α*(POS(p)-q_c[c]))
  D_T[p] = SAMPLE(D_S,u)
end
 T_f = B_T + D_T
end
return {T_f}
```

TABLE II. EXPERIMENTAL PARAMETERS

Parameter	Value
Number of frames	32
Frame duration	3 seconds total
Normal rotation	θ, ϕ evenly distributed
σ (Gaussian)	0.03
α (Patch Scale)	1.5
k (Clusters)	25

IV. DISCUSSION

Figure 8 illustrates the visual behavior of the proposed system under key parameter variations. Figure 8(a) shows the base layer result rendered using Lit-Sphere mapping, driving the low-frequency shading animation. Figure 8(b) demonstrates how varying σ influences base and detail layer separation; increasing σ results in smoother base layers and fewer deforming artifacts. Figure 8(c) explores the patch scale parameter α, showing how it affects the granularity of high-frequency transfer, with larger α producing denser, richer detail but potentially increasing misalignment artifacts.

A small σ=0.00002 results in high-frequency patterns in the base layer, potentially causing deformation during base transfer. Conversely, a large σ=0.2 includes base color transitions on the detail layer, potentially creating segmented transfer regions. At σ=0.03, high-frequency patterns are effectively removed from the base layer, producing stable final animations.

A. *Scale Control in Patch Transfer*

The parameter scales the pixel sampling densities in the style detail layer, proportional to the sampling space of the target image. Consequently, a lower patch scale parameter α=0.5 induces a sparser sampling of the style image. This sparse sampling can lead to less intricate stylization, leaving more target image regions unaltered by the style characteristics. This outcome may be desirable for preserving the target image's original texture details or for subtle style imposition. Conversely, a higher patch scale parameter α=6 yields a denser sampling pattern, intensifying the resulting high-frequency patterns. While this setting can contribute to a more richly stylized output, it also increases the risk of misalignment in patch sampling, especially when there is a significant difference in normal guidance. Thus, the patch scale parameter α plays a crucial role in shaping the perception of high-frequency patterns. However, careful consideration is needed to avoid using drastically different feature references, as it can lead to artifact issues. The chosen setting of α=1.5 balances style transfer fidelity and normal correspondence in the detail transfer process.

B. *Animation for View Transition*

Although the proposed method has been applied to static scenes, adapting it for view changes is straightforward. To produce the results shown in Figures 7 (b) and 7 (c), animated sequences of the target normal map were prepared and the style transfer was applied frame by frame. Compared to static scenes, this approach increases the risk of artifacts in cluster-based patch transfer. However, the detail transfer process remains effective even in complex scenes, preserving high-frequency pattern temporal coherence while maintaining lighting controllability.

C. *Visualization of Guidance Normal Map Cluster Stability*

Figure 7(a) visualizes the inter-frame stability of the guidance-normal map after applying rotations (θ, ϕ). For every frame, surface normals are clustered using K-means, and each pixel is then labeled with its nearest centroid's index. This produces a segmentation mask with distinct pseudo-colors, while cluster boundaries are highlighted in yellow for readability. The near-invariance of cluster geometry and pixel membership across the rotated sequence confirms the normal-space partitioning is robust to orientation changes. This robustness preserves the high-level structural cues, necessary for temporally coherent, lighting-aware style transfer, minimizing flicker and guaranteeing smooth visual continuity.

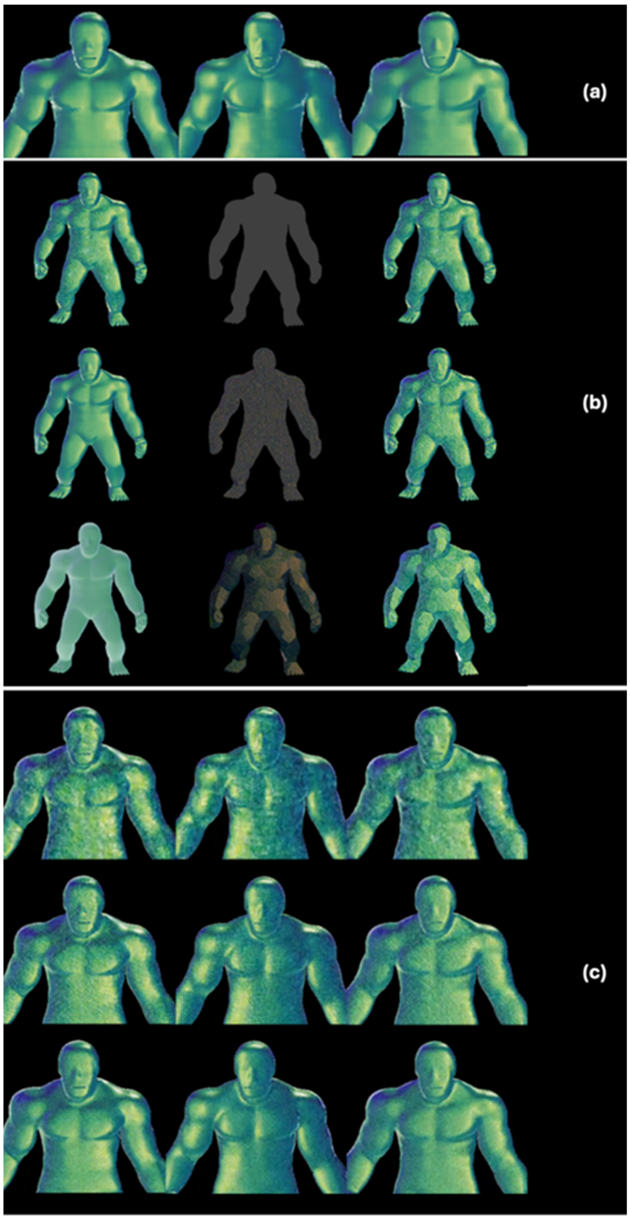

Fig. 9. (a) Base layer result using Lit-Sphere mapping. (b) Effect of varying the Gaussian filter parameter σ on base-detail layer separation. (c) Influence of patch scale parameter α on the density of transferred high-frequency details.

For a qualitative insight into the artistic effectiveness of the proposed method, an expert review evaluation involving five professional animators was conducted. Each participant has over 5 years of experience in 2D/3D animation and was asked to evaluate the stylized outputs based on three criteria: visual quality, lighting realism, and detail preservation. The evaluation was carried out using five representative sequences rendered with different styles and lighting scenarios. The scores were given on a scale from 1 (poor) to 10 (excellent). Additionally, participants provided brief qualitative feedback, as summarized in Table III. Overall, the method received favorable evaluations, particularly for preserving fine artistic details and simulating lighting-aware motion. The feedback also indicated opportunities for refinement in cluster alignment for highly structured styles.

TABLE III. ANIMATOR EVALUATION SUMMARY

Animator	Background	Visual Quality	Lighting Realism	Detail Preservation	Comments
A1	3D Game Artist	9	8	9	Smooth motion, excellent texture preservation
A2	VFX Supervisor	8	7	8	Good lighting mimicry, slightly coarse clusters
A3	Animation Lecturer	9	9	8	Great concept for teaching stylization
A4	Freelance Animator	8	8	9	High detail retention in motion
A5	Toon Shader Dev.	7	7	7	Minor misalignments in regular textures

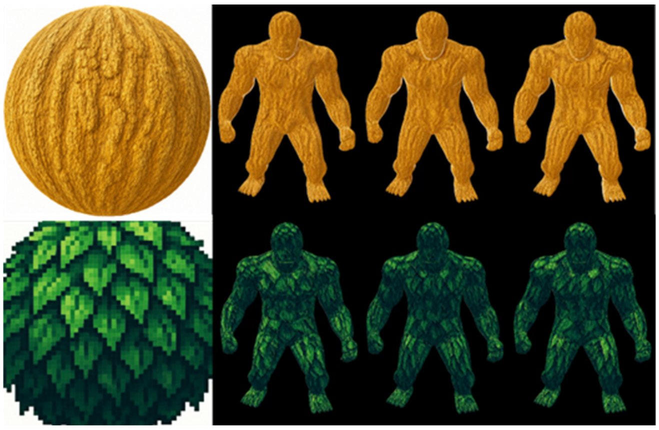

Fig. 10. Our method preserves the delicate structure of manually crafted pixel art, such as leaf textures, while also maintaining the integrity of high-level surface characteristics like those found in stone.

Figure 9 demonstrates that the proposed technique preserves fine-grained handcrafted pixel art (e.g., leaf venation) and maintains the integrity of higher-level surface characteristics (e.g., rough stone relief), enabling the animation of inherently stochastic textures, like leaves and bricks without perceptible seams. Figure 10 demonstrates that the clustering-based detail transfer can introduce visible misalignments when applied to (semi-)regular exemplars: the clustering process merges spatially distinct elements of repeated patterns, and floor/clamp operations, used to enforce integer coordinates and shift sampling positions by whole pixels. Furthermore, applying a density-scaled offset followed by another rounding step perturbs pixel locations even more, causing small quantization errors to accumulate and disrupt the precise alignment of regularly repeating structures.

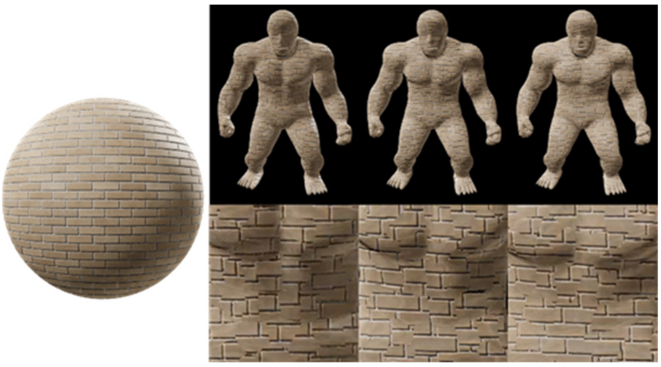

Fig. 11. Our method can cause visible misalignments in regular structures when applied to (semi) regular style exemplars.

V. CONCLUSION

The current study presented a method that integrates dynamic lighting control with guided patch-based style transfer by manipulating normal map distributions. This provides artists with a powerful tool for creating diverse stylistic outputs. While the proposed technique produces compelling results across various styles, its limitations include potential sampling artifacts from significant normal guidance disparities, requiring careful parameter tuning for stability. The method works well with stochastic, hand-drawn exemplars, where natural variation helps mask inconsistencies [19]. However, an improper balance between the base and detail layers can reveal visible seams or unintended low-polygon effects.

RESULT EXHIBITION

To view our results, please watch the video at https://intip.in/guidancemaplightingcontrol6

REFERENCES

[1] D. Prasetyo, N. Ramadhani, M. Hariadi, and I. R. Mutiaz, "Lighting Dynamics for Emotional Perception: A Technology-Driven Virtual Simulation Approach," *Engineering, Technology & Applied Science Research*, vol. 15, no. 2, pp. 22196–22202, Apr. 2025, https://doi.org/10.48084/etasr.10434.

[2] P.-P. J. Sloan, W. Martin, A. Gooch, and B. Gooch, "The lit sphere: a model for capturing NPR shading from art," in *Proceedings of Graphics Interface 2001*, CAN, June 2001, pp. 143–150

[3] D. Sýkora *et al.*, "StyleBlit: Fast Example-Based Stylization with Local Guidance," *Computer Graphics Forum*, vol. 38, no. 2, pp. 83–91, 2019, https://doi.org/10.1111/cgf.13621.

[4] J. Mitchell, M. Francke, and D. Eng, "Illustrative rendering in Team Fortress 2," in *Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*, New York, NY, USA, Aug. 2007, pp. 71–76, https://doi.org/10.1145/1274871.1274883.

[5] P. Barla, J. Thollot, and L. Markosian, "X-toon: an extended toon shader," in *Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, New York, NY, USA, June 2006, pp. 127–132, https://doi.org/10.1145/1124728.1124749.

[6] C.-R. Yan, M.-T. Chi, T.-Y. Lee, and W.-C. Lin, "Stylized Rendering Using Samples of a Painted Image," *IEEE Transactions on Visualization and Computer Graphics*, vol. 14, no. 2, pp. 468–480, Mar. 2008, https://doi.org/10.1109/TVCG.2007.70440.

[7] C. D. Kulla, J. D. Tucek, R. J. Bailey, and C. M. Grimm, "Using texture synthesis for non-photorealistic shading from paint samples," in *11th Pacific Conference onComputer Graphics and Applications, 2003. Proceedings.*, Oct. 2003, pp. 477–481, https://doi.org/10.1109/PCCGA.2003.1238298.

[8] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image Style Transfer Using Convolutional Neural Networks," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 2414–2423, https://doi.org/10.1109/CVPR.2016.265.

[9] J. Liao, Y. Yao, L. Yuan, G. Hua, and S. B. Kang, "Visual Attribute Transfer through Deep Image Analogy." arXiv, June 06, 2017, https://doi.org/10.48550/arXiv.1705.01088.

[10] H. Huang *et al.*, "Real-Time Neural Style Transfer for Videos," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 7044–7052, https://doi.org/10.1109/CVPR.2017.745.

[11] A. Šubrtová, M. Lukáč, J. Čech, D. Futschik, E. Shechtman, and D. Sýkora, "Diffusion Image Analogies," in *ACM SIGGRAPH 2023 Conference Proceedings*, New York, NY, USA, July 2023, pp. 1–10, https://doi.org/10.1145/3588432.3591558.

[12] J. Fišer *et al.*, "StyLit: illumination-guided example-based stylization of 3D renderings," *ACM Trans. Graph.*, vol. 35, no. 4, July 2016, Art. no. 92:1-92:11, https://doi.org/10.1145/2897824.2925948.

[13] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, New York, NY, USA, Aug. 2001, pp. 327–340, https://doi.org/10.1145/383259.383295.

[14] P. Bénard *et al.*, "Stylizing Animation By Example," *ACM Transactions on Graphics*, vol. 32, no. 4, 2013, Art. no. 119:1, https://doi.org/10.1145/2461912.2461929.

[15] Q. Zheng, Z. Li, and A. Bargteil, "Learning to Shadow Hand-drawn Sketches." arXiv, Apr. 02, 2020, https://doi.org/10.48550/arXiv.2002.11812.

[16] L. Zhang, E. Simo-Serra, Y. Ji, and C. Liu, "Generating Digital Painting Lighting Effects via RGB-space Geometry," *ACM Trans. Graph.*, vol. 39, no. 2, Feb. 2020, Art. no. 13:1-13:13, https://doi.org/10.1145/3372176.

[17] H. Todo, K. Anjyo, and S. Yokoyama, "Lit-Sphere extension for artistic rendering," *The Visual Computer*, vol. 29, no. 6, pp. 473–480, June 2013, https://doi.org/10.1007/s00371-013-0811-7.

[18] M. Arief, H. Todo, K. Mikami, and K. Kondo, "Region Control in Stylized Shading Using Radial Transformation within Texture Projection," *IIEEJ Transactions on Image Electronics and Visual Computing*, vol. 7, no. 1, pp. 36–45, 2019, https://doi.org/10.11371/tievciieej.7.1_36.

[19] M. Ashikhmin, "Synthesizing natural textures," in *Proceedings of the 2001 symposium on Interactive 3D graphics*, New York, NY, USA, Mar. 2001, pp. 217–226, https://doi.org/10.1145/364338.364405.