

Painterly Rendering with Image Relighting-Based Faded Color

Jong-Hyun Kim*

*Associate Professor, College of Software and Convergence (Dept. of Design Technology), Inha University, Incheon, Korea

[Abstract]

This paper proposes a painterly rendering method that reflects 2D image relighting and color variation to visualize the effects of light on oil paintings. The proposed method relights a 2D image and uses the resulting shading values as weights to reflect corresponding changes in color. For the relighting process, the 2D image is approximated as a 3D surface, and normals are estimated. The angle between these normals and the specified light source is used to compute shading values. Users can freely set the position of the light source, and the resulting color changes under varying lighting conditions are derived with reference to prior research. Unlike conventional local-based painterly rendering methods that produce flat and overly simplified results, the proposed approach effectively captures faded colors and a sense of depth through image relighting. This allows for vivid, three-dimensional representations that closely resemble real oil paintings. Ultimately, the method contributes to the expressive depiction, prediction, and restoration of color changes in cultural and artistic works.

▶ **Key words:** Non-photorealistic rendering, Painterly rendering, Oil-painting rendering, Relighting, Discoloration, Faded color

[요 약]

본 논문에서는 빛의 영향에 따른 유화 변화를 보여주기 위해 2D 이미지 재조명과 색채 변화를 반영한 회화적 렌더링 방법을 제안한다. 제시하는 방법은 2D 이미지를 재조명하고, 그 음영 값을 가중치로 사용하여 색채 변화를 반영하는 렌더링 기법이다. 재조명의 경우, 2D 이미지를 3D 이미지로 근사 추정하고 법선을 결정하여 해당 값과 조명 위치 값 사이의 각을 음영 값으로 추출하여 적용한다. 사용자는 조명 위치를 지정할 수 있으며, 빛의 영향에 따른 색채 변화 결과는 기존 연구를 참조한다. 본 논문에서 제안하는 접근 방식은 기존의 로컬 기반 회화 렌더링 기법이 생성하는 단순하고 평면적인 결과와 달리, 이미지 재조명을 통해 빛바랜 색상과 깊이감을 효과적으로 반영함으로써, 현실에 존재하는 유화 작품과 유사한 생동감 있는 입체적 표현을 가능하게 한다. 이를 통해 문화예술작품의 표현과 색채 변화 예측 및 복원에 기여할 수 있다.

▶ **주제어:** 비사실적 렌더링, 회화적 렌더링, 유화 렌더링, 재조명, 변색, 빛바랜 색

-
- First Author: Jong-Hyun Kim, Corresponding Author: Jong-Hyun Kim
 - *Jong-Hyun Kim (jonghyunkim@inha.ac.kr), College of Software and Convergence (Dept. of Design Technology), Inha University
 - Received: 2025. 02. 25, Revised: 2025. 04. 17, Accepted: 2025. 04. 17.

I. Introduction

본 연구의 목적은 사용자 상호작용을 통해 2D 이미지를 재조명하고, 빛의 영향에 따른 색채 변화를 반영한 렌더링 결과를 제공하는 것이다. 기존 비사실적 렌더링 기법 중 하나인 로컬 이미지 근사 기반 자동 회화적 렌더링은 캔버스에 스트로크를 합성하는 방식을 사용하며, 색상 차이 이미지와 이미지 모멘트를 활용해 스트로크의 배치, 크기, 방향을 자동으로 결정한다[1]. 이 기법은 주로 유화 이미지 렌더링에 활용되며, 본 연구에서는 해당 알고리즘을 기반으로 회화적 렌더링을 구성하였다. 그러나 기존 방법이 2D 이미지만을 다루는 것과 달리, 본 연구에서는 이미지 재조명을 위해 3D 정보를 추가적으로 활용한다. 또한, 스트로크 색상을 결정하는 방식에서도 차이를 둔다.

회화적 렌더링은 회화 도구를 사용하는 아날로그 방식을 모방한 시각 미디어 수준의 예술적 표현 방식이다. 가장 일반적으로 많이 사용하는 접근법은 저수준 이미지 특성에 기반하여 로컬 자동 브러시 스트로크 배치를 사용하는 SBR(Stroke-based rendering)이 있다[2]. 회화적 렌더링은 가장 널리 사용하는 NPR(Non-photorealistic rendering) 기법 중 하나이며 많은 애플리케이션에서 사용되고 있다[3].

이전의 대부분의 접근법은 결과가 입력 이미지에 충분히 근접할 때까지 텍스처가 있는 브러시 스트로크 지오메트리를 연속적으로 렌더링하고 혼합하여 회화적 스타일화를 수행한다[4,5]. 이러한 접근방식의 한계는 다음과 같다.

- 원본 이미지로부터 브러시의 방향만을 계산할 뿐 색상은 원본색상을 사용하기 때문에 미디어아트 측면에서는 항상 원본 형태로밖에 표현할 수 없는 한계가 있다.
- 사진은 오래될수록 빛바랜 색이 표현되기 때문에 시물레이션과 같은 동적으로 변하는 특성이 존재하지만, 기존 접근법은 원본 스타일만을 고려하기 때문에 미디어아트 측면에서의 표현적 한계가 있을 수밖에 없다.

일반적으로 이미지가 오래되면 빛바랜 식으로 나타나는 이유는 여러 가지 요인에 기인한다.

- 화학적 변화 : 사진 필름은 일반적으로 염화은 결정을 포함하는 에멀전(Emulsion)으로 만들어진다. 시간이 지남에 따라 이 염화은은 분해되거나 변질될 수 있으며, 이로 인해 이미지의 색상이 변할 수 있다.
- 노출과 퇴색 : 빛에 노출되는 것은 사진의 퇴색을 가속화시킬 수 있다. 자외선(UV)과 가시광선은 사진 내의 색소 분자를 분해하여 색이 바래고 흐려지는 원인이 된다.
- 열과 습도 : 높은 온도와 습도는 사진의 퇴색을 촉진시

킬 수 있다. 열은 색소 분자의 분해를 가속화시키고, 습도는 필름의 물리적 변화를 초래할 수 있다.

- 산소 : 산소는 필름의 색소와 상호작용하여 산화 반응을 일으킬 수 있다. 이는 색소의 분해와 퇴색을 초래할 수 있다.
- 오염 물질 : 공기 중의 오염물질이나 보관된 환경의 오염물질은 사진에 영향을 줄 수 있다. 예를 들어, 대기 중의 황이나 질소 산화물 등은 필름의 화학 반응을 유발하여 퇴색을 가속화시킬 수 있다.

이러한 요인들이 복합적으로 작용함에 따라, 시간이 경과하면서 사진의 색상은 점차 퇴색하게 된다.

위 문제를 해결하기 위해 본 논문에서는 "열에 의해 노출된 정도"를 조명 방정식(Lighting equation)으로 근사하여 빛바랜 색을 계산하고 이를 통해 회화적 렌더링을 할 수 있는 새로운 프레임워크를 제시한다. 이미지 재조명을 위해 2D 이미지를 3D로 근사 추정하여 법선 벡터를 결정한다[6]. 이후, 결정된 법선과 사용자가 지정한 조명 위치 벡터 사이의 각도를 계산하고, 이를 음영 값으로 변환하여 2D 이미지의 조명을 조정한다. 스트로크 색상 결정에는 기존 유화의 색채 변화를 연구한 결과를 반영한다. 이를 위해 유기안료와 무기안료의 내광성을 실험하고, 분광광도계를 활용한 측색을 통해 정량화된 데이터를 CIELAB 색 공간으로 변환한 연구를 참고한다[7]. 최종적으로, CIELAB 색채 변화 데이터를 RGB로 변환한 후, 재조명을 통해 측정된 음영 값을 가중치로 반영하여 보다 정확한 색상 변화를 표현하고자 한다.

II. Related Work

수채화나 유화 등 다양한 스타일로 표현되는 회화적 렌더링에는 여러 가지 접근 방식이 존재한다. Hedge et al.[3]과 Nolte et al.[8]는 이러한 접근 방식을 구현할 수 있는 프레임워크를 제안했다. 또한, Hertzmann[9]과 Nolte et al.[8]는 회화적 렌더링 알고리즘을 스트로크 배치 방식에 따라 분류했다. 일반적으로 SBR 문제는 원본 이미지에 대한 오류를 최소화하는 관점에서 접근하거나, "greedy" 휴리스틱을 따르는 방식으로 해결할 수 있다.

오류 최소화를 기반으로 하는 접근 방식에는 Brute-force 또는 Random search[10-12], 경사 하강(Gradient Descent)[13-15]을 활용한 최적화 반복 과정이 포함되거나, 지도 학습(Supervised)[16,17] 또는 심층 강화 학습[18,19]을 통해 심층 신경망을 학습시키는 방법이

있다. 이러한 방식은 반복 횟수가 많고 실행 시간이 길어 계산 비용이 높을 뿐만 아니라, 원본 영상에만 의존하기 때문에 빛의 노출량이나 강도에 따른 색 바램과 같은 효과를 표현하는 데 한계가 있다.

휴리스틱(Heuristics) 기반 접근 방식은 가장자리 감지나 영역 분할을 통해 얻은 이미지 콘텐츠 표현을 활용하여, 단일 패스에서 직접 스트로크 배치를 계산한다. 일반적으로 저수준의 예지는 이미지의 그래데이션 정보를 기반으로 하며, 이는 스트로크 배치를 안내하는 데 사용된다. Haeberli[20]의 초기 연구에서는 사용자가 직접 스트로크를 배치하거나, Litwinowicz[21]와 같이 정규 격자에 배치한 후 저수준 예지를 따라 클리핑하고 방향을 지정할 수 있도록 했다. Hertzmann[20]은 로컬 그래데이션의 직교 방향을 따라 B-스플라인을 활용해 더 긴 곡선 스트로크를 사용하는 방식을 제안했다. 페인팅 프로세스는 대규모 그래데이션을 고려하여 여러 개의 레이어를 통합하는 방식으로 진행된다. 먼저 넓은 스트로크를 사용해 큰 영역을 채운 후, 세부적인 디테일을 보완하기 위해 더 작고 얇은 획을 위에 덧그리는 방식이다. 여러 접근 방식이 이러한 레이어링 기법을 공유하지만, 가장자리를 배치하는 방식에는 차이가 있다[22-25]. 또한, 인간의 그림 그리기 방식을 보다 정교하게 모방하기 위해 의미론적으로 더 풍부한 상위 레벨 정보를 통합할 수도 있다. 이를 통해 추상화 수준을 조절하고, 객체의 경계를 따라 스트로크를 배치하기 위해 세그멘테이션 기법이 활용되기도 한다[26,27]. Lindemeier et al.[27]은 입력 이미지를 여러 영역으로 계층적으로 분할한 뒤, 이를 다시 레이어별로 나누어 거친 부분에서 세밀한 부분까지 점진적으로 표현하는 방식을 제안했다. Collomosse et al.[28]은 계산된 이미지 중요도를 활용하여 더 중요한 영역에는 작은 획을 배치하는 기법을 제시했으며, Wexler et al.[5]은 중요도 샘플링을 적용하여 얼굴 렌드마크를 따라 디테일을 배치하는 방식을 제안했다.

회화적 렌더링 프로세스를 신속하게 처리하기 위해 다양한 전략이 제안되었다. 예를 들어, Hertzmann[29]은 새로운 프레임에서 변화가 큰 캔버스 영역에만 스트로크를 배치하는 방식으로 동영상에 회화적 렌더링을 적용했다. Mukundan et al.[30]은 모바일 기기에서 회화적 렌더링을 효율적으로 수행하기 위해 인덱스 테이블 기반의 빠른 영역 라벨링 접근법을 제안했으며, Fischer et al.[31]은 증강 현실 애플리케이션을 위한 고속 점묘 렌더링 필터를 개발했다.

다양한 접근 방식을 통해 회화적 렌더링에 인터랙티브 제어 메커니즘을 추가한 연구도 있다. 이러한 메커니즘에

는 핑거페인팅을 위한 손 제스처[32] 또는 터치 기반 컨트롤[33]의 통합, 사용자가 대화형 스트로크 프로세스를 구현하여 스트로크 스타일을 대화형으로 수정하거나[34] 브러시와 캔버스 간의 상호작용을 시뮬레이션하는[35] 방식 등이 있다. Benedetti et al.[36]은 초보 사용자를 위한 대화형 페인팅 애플리케이션을 위한 여러 가지 디자인 가이드라인을 제시하고, 이러한 기준에 따라 사용자가 캔버스에 획을 그리면 시스템이 참조 이미지를 통해 지침을 제공하는 페인팅 시스템을 개발했다. 이 시스템을 사용하면 초보 사용자도 쉽게 회화적인 이미지를 만들 수 있으며, 동시에 창의적인 유연성을 발휘하여 완성도 높은 결과물을 얻을 수 있다.

III. The Proposed Scheme

1. Painterly Rendering

회화적 렌더링 알고리즘의 기초로는 색상 차이 이미지와 이미지 모멘트가 있다. 색상 차이 이미지의 경우 다음 수식으로 정의된다 (수식 1과 2 참조).

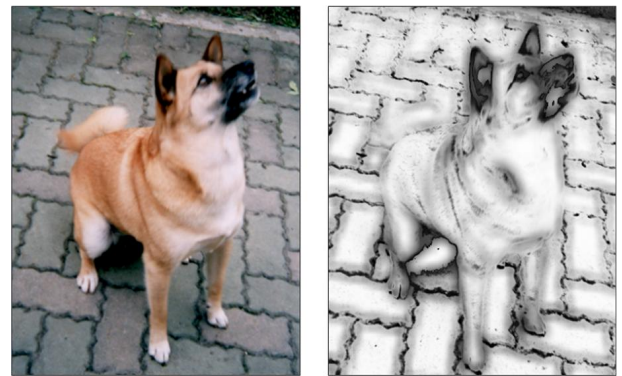


Fig. 1. Stroke area image computed from the input image.

$$I(x,y) = f(d(C, C_w(x,y))) \quad (1)$$

$$f(d) = \begin{cases} (1 - (d/d_0)^2)^2 & d \in [0, d_0] \\ 0 & d \in [d_0, \infty] \end{cases} \quad (2)$$

여기서 $I(x,y)$ 는 입력 이미지에서 잘라 온 로컬 픽셀을 의미하며, 함수 $f(d) \in [0,1]$ 의 계산으로 결정된다. $d(C_1, C_2)$ 는 CIE L^*, u^*, v^* 색상 공간에서 C_1 과 C_2 의 유클리드 거리를 계산한 것이다. 이때 C_1 에 해당하는 C 는 입력 이미지에서 잘라 온 로컬에 해당하는 중심 색상이자 스트로크의 임시 색상을 의미한다. 향후 이 스트로크의 색

상을 빛바랜 색상으로 변경할 것이다. C_2 에 해당하는 $C_w(x,y)$ 는 입력 이미지에서 잘라 온 로컬 이미지의 픽셀 들을 의미한다. d_0 의 값은 유클리드 거리를 통해 계산된 색상 차이 정도를 어느 범위까지 유사한 색상으로 인식할 지를 결정한다 (수식 3 참조).

$$d = \sqrt{(L-L')^2 + (u-u')^2 + (v-v')^2} \quad (3)$$

이렇게 계산된 스트로크 영역이미지는 Fig. 1과 같다. 그림에서 보듯이 색상 차이를 기반으로 부드러운 가중치 형태로 만들어 진 것을 볼 수 있다.

이미지 모멘트의 경우 아래와 같은 방법으로 정리된다 (수식 4 참조).

$$M_{lm} = \sum_x \sum_y x^l y^m I(x,y) \quad (4)$$

여기서 이미지 모멘트는 0차, 1차, 2차 등이 존재하며, n 차 이미지 모멘트라고 부른다. $n = l + m$ 를 만족하며, 0차는 M_{00} , 1차는 M_{10} 와 M_{01} , 2차는 M_{20} , M_{02} 와 M_{11} 로 구분하며, 자세한 수식은 다음과 같다 (수식 5 참조).

$$\begin{aligned} M_{00} &= A = \iint_I b(x,y) dx dy \\ \frac{M_{10}}{M_{00}} &= \frac{M_{01}}{M_{00}} = \bar{x} = \frac{1}{A} \iint_I x b(x,y) dx dy, \bar{y} = \frac{1}{A} \iint_I y b(x,y) dx dy \\ \frac{M_{20}}{M_{00}} &= \frac{M_{02}}{M_{00}} = \frac{M_{11}}{M_{00}} = \iint_I r^2 b(x,y) dx dy \end{aligned} \quad (5)$$

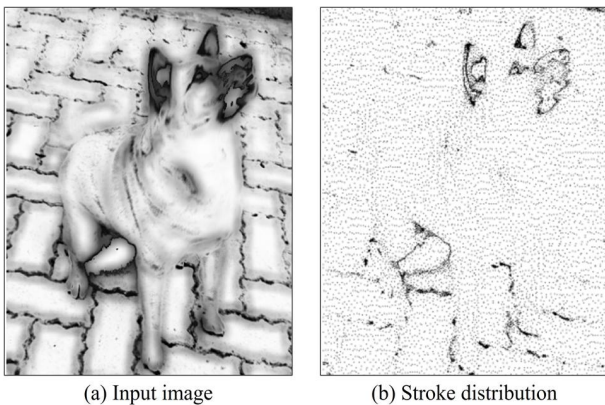


Fig. 2. Stroke distribution image computed from the stroke area estimation image.

회화적 렌더링을 계산하기 위해서는 먼저 붓칠을 할 범위 추정 후 붓칠 할 위치의 분포를 결정해야 한다 (Fig. 2 참조). Fig. 2a는 붓칠 할 범위를 추정한 결과이다. 범위

추정은 캔버스에 위치별로 배치할 스트로크의 밀도를 파악할 수 있게 한다. 여기서 밀도는 이미지의 세부 사항을 표현하기 위해서 어느 부분을 촘촘하게 혹은 듚성듬성하게 붓칠을 할지 알려준다. Fig. 2a는 앞에서 설명한 색상 차이 이미지를 통해 계산할 수 있으며, 검정색에 가까울수록 해당 픽셀의 색상이 주변 픽셀과 색상 차이가 크다는 것을 의미한다 (Fig. 1b 참조). 즉, 주변 픽셀과 색상 차이가 작을수록 밝게 표현되고 차이가 클수록 어둡게 표현된다. Fig. 2b는 붓칠 할 위치의 분포를 계산하기 위해 표시한 결과이며, Fig. 2a를 기반으로 생성한다. 어두운 부분을 더 촘촘하게 붓칠 할 위치로 선정하기 위해서 지역성을 보존해주는 힐베르트 공간 채우기 곡선으로 공간을 채운다. 곡선을 따라 이동하며 픽셀들을 누적하다 임계값에 도달하면 붓칠 할 위치로 선택한다. 이렇게 선택한 위치를 이미지로 표시한 것이 Fig. 2b이다.

앞의 과정을 통해 스트로크들의 붓칠 할 위치가 결정되었다면 이제는 스트로크의 속성을 결정해야 한다 (Fig. 3 참조).

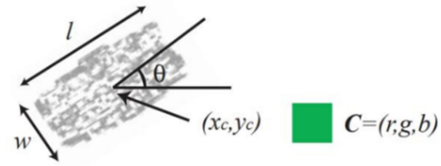


Fig. 3. Stroke attributes.

스트로크의 속성은 크기에 해당하는 w 과 l , 중심에 해당하는 x_c 와 y_c , 그리고 X 축을 기준으로 회전 방향에 해당하는 θ , 마지막으로 임시 색상에 해당하는 C 가 있다. 해당 속성들은 색상 차이 이미지(Fig. 1b 참조)와 이미지 모멘트를 활용해 계산되며 스트로크 색상 외의 속성들은 아래 수식에 따라 계산한다 (수식 6~9 참조).

$$\{x_c, y_c\} = \left\{ \frac{M_{10}}{M_{00}}, \frac{M_{01}}{M_{00}} \right\} \quad (6)$$

$$\theta = \frac{\tan^{-1}\left(\frac{b}{a-c}\right)}{2} \quad (7)$$

$$\begin{aligned} w &= \sqrt{6(a+c - \sqrt{b^2 + (a-c)^2})} \\ l &= \sqrt{6(a+c + \sqrt{b^2 + (a-c)^2})} \end{aligned} \quad (8)$$

$$a = \frac{M_{20}}{M_{00}} - x_c^2 \quad (9)$$

$$b = 2\left(\frac{M_{11}}{M_{00}} - x_c y_c\right)$$

$$c = \frac{M_{02}}{M_{00}} - y_c^2$$

Fig. 4는 앞에서 계산한 스트로크의 속성들을 통해 브러시를 정의하고 이용하여 캔버스에 칠하는 결과를 보여주는 그림이다.



Fig. 4. Process of painting with a brush.

2. Relighting and Faded Color

본 논문에서는 이미지로부터 법선벡터를 추출하기 위해 Bednarik et al.이 제안한 학습 기법을 이용한다[37]. 이로부터 계산된 법선 맵은 다음과 같다 (Fig. 5 참조). 이렇게 계산된 법선 맵을 이용하여 본 논문에서는 재조명을 계산한다. 법선 맵을 3차원 벡터로 변환한 뒤, 사용자가 지정한 조명 사이의 음영을 계산한다. 이렇게 계산된 음영의 강도를 이용하여 원본 색상을 강조한다.



Fig. 5. Extracted normal map.

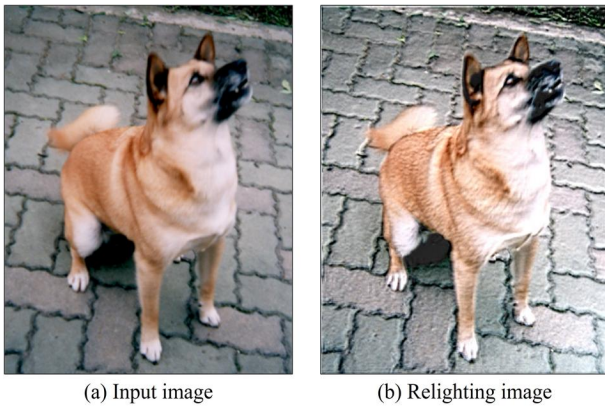
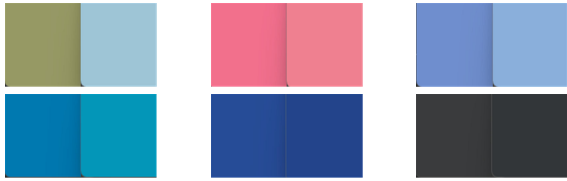


Fig. 6. Relighting image computed from the input image.

본 논문에서는 이 음영의 강도에 따라 빛바랜 정도를 제어한다. 스트로크의 색상 결정은 Lee의 연구를 참고하여 재조명에 필요한 음영을 결정하도록 한다[38]. Lee의 방법에서 제시된 CIELAB 색상 공간을 본 연구의 목적에 맞게 활용하기 위해서 색채의 시각적 확인이 가능해야한다.

Table 1. Faded color table (left color : original color, right color : faded color).



본 논문에서는 원본 색상과 색상 차이가 가장 작은 빛바랜 색을 선택했으며, 이 과정에서 색상 차이는 유클리드 거리를 통해 계산한다. 스트로크 색상 결정 과정을 정리하면 다음과 같다. 임의 색상 C 의 RGB값과 오차가 가장 적은 자외선 노출 전 RGB 데이터를 찾는다. 해당 데이터의 자외선 노출 후 RGB 데이터를 가져오고 이 값에 음영을 가중치로 반영한다.

스트로크를 칠하는 순서는 속성인 (w, l) 가 큰 것에서 작은 것 순서로 캔버스에 그리도록 정렬한다. 그리고 정렬한 스트로크 순서대로 각각의 스트로크의 속성들을 참고하여 칠 한다.

IV. Experiment and Results

본 연구의 결과들을 만들기 위해 실험한 환경은 Intel Core i7-7700K CPU, 32GB RAM, Geforce GTX 1080Ti GPU가 탑재된 컴퓨터를 이용하였다.

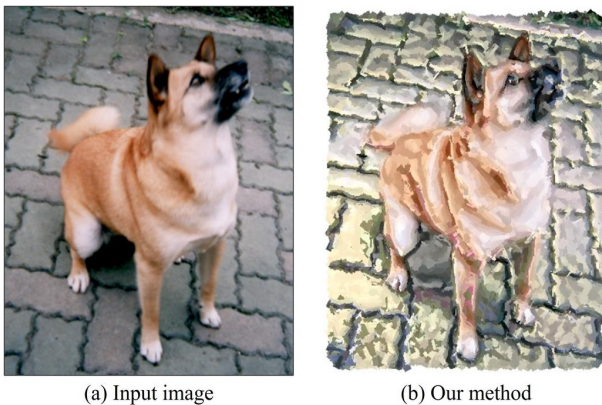


Fig. 7. Faded color based painterly rendering with our method.

Fig. 7은 본 논문에서 제안하는 방법을 이용하여 빛바랜 색을 결정하고 이를 브러시의 색상으로 결정하여 회화렌더링을 계산한 결과이다. 이 과정에서 빛바랜 색만 표현할 뿐만 아니라 재조명을 동시에 계산하기 때문에 전체적인 사진의 밝기 또한 달라진 것을 볼 수 있다.

Fig. 8은 좀 더 다양한 사진에서 실험한 결과이다. 그림

에서 보듯이 다양한 장면에서 빛바랜 색을 표현하여 회화적 느낌을 잘 표현했다.



Fig. 8. Painterly rendering results experimented on a wider variety of images.

Fig. 9는 기존 로컬 기반 회화 렌더링 알고리즘인 스트로크 렌더링 기법과 본 논문에서 제안한 기법을 직접적으로 비교한 결과이다. 이를 통해 본 논문에서 제안한 재조명 기반의 색상 변화 반영 기법이 단순히 스타일만의 변화가 아닌, 양감과 생동감 표현에서의 명확한 차별성을 가질 수 있음을 실험적으로 보여주고 있다 (Fig. 9b 참조).

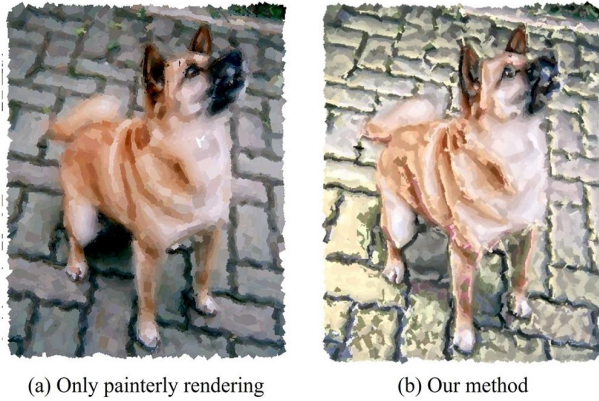


Fig. 9. Comparison between traditional local-based painterly rendering and our method.

V. Conclusion

본 논문에서는 2D 이미지의 재조명에 따른 색채 변화를 반영한 회화적 렌더링 기법을 새롭게 제안하였다. 기존 연구에서 다루지 않았던 빛의 영향을 고려하여 알고리즘을 개선하였으며, 이를 통해 회화적 렌더링을 보다 예술적인 관점에서 접근하고자 하였다. 또한, 기존 렌더링 결과와의 비교를 통해 차이를 확인하고, 추가적인 이미지 테스트를 수행하여 빛에 따른 색채 변화를 반영한 렌더링 기법의 성능을 입증하였다.

회화적 렌더링에 대한 다양한 연구가 기존에 수행되었으며, 특히 스트로크 배치 및 색상 결정과 관련된 알고리즘은 충분히 정립되어 있는 상태이다. 본 연구에서는 이러한 기초 기술을 기반으로 삼되, 기존 연구를 단순히 조합하는 것에 그치지 않고, 다음과 같은 명확한 차별성과 기술적 확장을 제안하였다.

첫째, 스트로크 색상 결정 과정에서 사용된 Lee의 색상 변화 연구는 자외선 노출 전후의 색상 데이터를 제시한 실험 연구이며, 본 논문에서는 해당 데이터를 재조명 기반 회화 렌더링 프레임워크 내에서 실질적으로 응용하였다. 특히 CIELAB 색공간 기반 색상 차이 계산 \rightarrow RGB 변환 \rightarrow 음영 가중치 적용이라는 일련의 연산 과정을 통해 실제 색상 변화를 반영하는 스트로크 색상 매핑 알고리즘을 구

현하였다.

둘째, 기존 회화적 렌더링 기법은 대부분 정적인 색상 기반의 스트로크 스타일링에 집중되어 있으며, 시간에 따른 재조명이나 색상 변화 요소를 고려하지 않았다. 반면, 본 연구는 2D 이미지의 3D 근사, 법선 추정, 사용자 지정 조명 위치 기반 음영 계산, 그에 따른 색상 변화 시뮬레이션이라는 물리 기반 조명 반응 모델을 도입함으로써, 동적이고 생동감 있는 양감 표현을 가능하게 하였다.

셋째, 연구의 최종 목적은 단순한 시각화 기법 제안에 그치지 않고, 문화예술 작품의 색상 변화 예측 및 복원에 적용 가능한 실제 응용 시스템을 위한 기반 기술 개발에 있다. 따라서 본 연구는 기존 알고리즘을 통합적으로 구조화하여 예술적 표현뿐 아니라 보존과 복원이라는 목적성 있는 응용 방향으로 기술을 확장한 점에서 독창적 의의가 있다.

본 연구에서 제안한 알고리즘은 문화예술 작품의 색채 변화 예측 및 복원에 기여할 뿐만 아니라, 2D 가상 예술 작품의 제작 가능성을 확장하는 데에도 활용될 수 있다. 다만, 색채 변화는 화학적 조성과 제작 기술에 따라 다르게 나타나므로, 본 알고리즘이 예술 작품의 예측 및 복원에 효과적으로 활용되기 위해서는 다양한 안료 조합에 대한 색채 변화 연구가 선행되어야 한다. 또한, 자외선이나 LED 등 다양한 광원의 영향을 고려한 색채 변화 연구도 필요하다.

현재 색채 변화 데이터가 부족하여 다양한 조합에 따른 색상의 변화를 완벽하게 반영하기는 어려운 한계가 있다. 향후에는 혼합색과 물질적 성질 변화에 따른 색채 영향을 반영할 수 있도록 데이터를 확보하고, 이를 통해 렌더링 결과의 사실성을 더욱 강화하고자 한다. 충분한 데이터가 축적된다면, 이를 기반으로 가상 예술 작품 제작이 가능하도록 렌더링 알고리즘을 보완할 계획이다.

ACKNOWLEDGEMENT

The National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. RS-2023-00254695, Contribution Rate : 50%). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.RS-2022-00155915, Artificial Intelligence Convergence Innovation Human Resources Development (Inha University), Contribution Rate : 50%)

REFERENCES

- [1] Shiraishi, Michio, and Yasushi Yamaguchi. "An algorithm for automatic painterly rendering based on local source image approximation." Proceedings of the 1st international symposium on Non-photorealistic animation and rendering. 2000. DOI: 10.1145/340916.340923
- [2] Kyprianidis, Jan Eric, et al. "State of the art": A taxonomy of artistic stylization techniques for images and video." IEEE transactions on visualization and computer graphics, vol. 19, no. 5, pp. 866-885, 2012. DOI: 10.1109/TVCG.2012.160
- [3] Hegde, Siddharth, Christos Gatzidis, and Feng Tian. "Painterly rendering techniques: a state-of-the-art review of current approaches." Computer Animation and Virtual Worlds, vol. 24, no. 1, pp. 43-64, 2013. DOI: 10.1002/cav.1435
- [4] Hertzmann, Aaron. "Painterly rendering with curved brush strokes of multiple sizes." Proceedings of the 25th annual conference on Computer graphics and interactive techniques. 1998. DOI: 10.1145/280814.280951
- [5] Wexler, Daniel, and Gilles Dezeustre. "Intelligent brush strokes." ACM Siggraph 2012 Talks. pp. 1, 2012. DOI: 10.1145/2343045.2343112
- [6] Khan, Salman, et al. "A guide to convolutional neural networks for computer vision.", pp. 1-207, 2018.
- [7] Huh, Soo-Jung, and Jin-Wan Park. "Analysis of the Oil painting for the painterly rendering-focusing on the 19C Impressionism painting." Proceedings of the Korea Contents Association Conference. The Korea Contents Association, pp. 259-264, 2006.
- [8] Nolte, Florian, Andrew Melnik, and Helge Ritter. "Stroke-based rendering: From heuristics to deep learning." arXiv preprint arXiv:2302.00595, 2022. DOI: 10.48550/arXiv.2302.00595
- [9] Hertzmann, Aaron. "A survey of stroke-based rendering." Institute of Electrical and Electronics Engineers, 2003. DOI: 10.1109/MCG.2003.1210867
- [10] Hertzmann, Aaron. "Paint by relaxation." Proceedings. Computer Graphics International 2001. IEEE, pp. 47-54, 2001. DOI: 10.1109/CGI.2001.934657
- [11] O'Donovan, Peter, and Aaron Hertzmann. "Anipaint: Interactive painterly animation from video." IEEE transactions on visualization and computer graphics, vol. 18, no. 3, pp. 475-487, 2011. DOI: 10.1109/TVCG.2011.51
- [12] Kang, Hyung W., Charles K. Chui, and Uday K. Chakraborty. "A unified scheme for adaptive stroke-based rendering." The Visual Computer, vol. 22, pp. 814-824, 2006. DOI: 10.1007/s00371-006-0066-7
- [13] Nakano, Reiichiro. "Neural painters: A learned differentiable constraint for generating brushstroke paintings." arXiv preprint arXiv:1904.08410, 2019. DOI: 10.48550/arXiv.1904.08410
- [14] Zou, Zhengxia, et al. "Stylized neural painting." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15689-15698, 2021. DOI: 10.1109/CVPR46437.2021.01543
- [15] Kotovenko, Dmytro, et al. "Rethinking style transfer: From pixels to parameterized brushstrokes." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12196-12205, 2021. DOI: 10.1109/CVPR46437.2021.01202
- [16] Zheng, Ningyuan, Yifan Jiang, and Dingjiang Huang. "Stroketnet: A neural painting environment." International Conference on Learning Representations. 2018.
- [17] Liu, Songhua, et al. "Paint transformer: Feed forward neural painting with stroke prediction." Proceedings of the IEEE/CVF international conference on computer vision. pp. 6598-6607, 2021. DOI: 10.1109/ICCV48922.2021.00653
- [18] Ganin, Yaroslav, et al. "Synthesizing programs for images using reinforced adversarial learning." International Conference on Machine Learning. PMLR, pp. 1666-1675, 2018. DOI: 10.48550/arXiv.1804.01118
- [19] Huang, Zhewei, Wen Heng, and Shuchang Zhou. "Learning to paint with model-based deep reinforcement learning." Proceedings of the IEEE/CVF international conference on computer vision. pp. 8709, 8718, 2019. DOI: 10.48550/arXiv.1903.04411
- [20] Haeberli, Paul. "Paint by numbers: Abstract image representations." Proceedings of the 17th annual conference on Computer graphics and interactive techniques. pp. 207-214, 1990.
- [21] Litwinowicz, Peter. "Processing images and video for an impressionist effect." Proceedings of the 24th annual conference on Computer graphics and interactive techniques. pp. 407-414, 1997. DOI: 10.1145/258734.25889
- [22] Hays, James, and Irfan Essa. "Image and video based painterly animation." Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering. pp. 113-120, 2004. DOI: 10.1145/987657.987676
- [23] Kovács, Levente, and Tamás Szirányi. "Painterly rendering controlled by multiscale image features." Proceedings of the 20th Spring Conference on Computer Graphics. pp. 177-184, 2004. DOI: 10.1145/1037210.1037237
- [24] Seo, SangHyun, JinWan Park, and KyungHyun Yoon. "A Painterly Rendering Based on Stroke Profile and Database." CAe. pp. 9-16, 2009.
- [25] Huang, Hua, Tian-Nan Fu, and Chen-Feng Li. "Painterly rendering with content-dependent natural paint strokes." The Visual Computer, vol. 27, pp. 861-871, 2011. DOI: 10.1007/s00371-011-0596-5
- [26] Zeng, Kun, et al. "From image parsing to painterly rendering." ACM Trans. Graph., vol. 29, no. 1, pp. 2, 2009. DOI: 10.1145/1640443.1640445
- [27] Lindemeier, Thomas, Marc Spicker, and Oliver Deussen. "Artistic

- composition for painterly rendering.", 2016. DOI:10.2312/vmv.20161350
- [28] Collomosse, J. P., and P. M. Hall. "Painterly rendering using image salience." *Proceedings 20th Eurographics UK Conference*. IEEE, pp. 122-128, 2002. DOI:10.1109/EGUK.2002.1011281
- [29] Hertzmann, Aaron, and Ken Perlin. "Painterly rendering for video and interaction." *Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*. pp. 7-12, 2000. DOI: 10.1145/340916.340917
- [30] Mukundan, Ramakrishnan, and Charles Han. "A Fast Algorithm for Painterly Rendering on Mobile Devices.", 2008.
- [31] Fischer, Jan, Dirk Bartz, and Wolfgang Straßer. "Artistic reality: Fast brush stroke stylization for augmented reality." *Proceedings of the ACM symposium on Virtual reality software and technology*, pp. 155-158, 2005. DOI: 10.1145/1101616.110164
- [32] Grubert, Jens, Sheelagh Cappendale, and Tobias Isenberg. "Interactive stroke-based NPR using hand postures on large displays." *EPRINTS-BOOK-TITLE*. University of Groningen, Johann Bernoulli Institute for Mathematics and Computer Science, 2008. DOI:10.2312/egs.20081030
- [33] Kang, Dongwann, and Kyunghyun Yoon. "Interactive painterly rendering for mobile devices." *Entertainment Computing-ICEC 2015: 14th International Conference, ICEC 2015, Trondheim, Norway, September 29-October 2, 2015, Proceedings 14*. Springer International Publishing, pp. 445-450, 2015. DOI : 10.1007/978-3-319-24589-8_38
- [34] Zhao, Mingtian, and Song-Chun Zhu. "Customizing painterly rendering styles using stroke processes." *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on non-photorealistic animation and rendering*. pp. 137-146, 2011. DOI: 10.1145/2024676.202469
- [35] Baxter, William V., and Ming C. Lin. "A versatile interactive 3D brush model." *12th Pacific Conference on Computer Graphics and Applications, 2004. PG 2004. Proceedings.. IEEE*, pp. 319-328, 2004. DOI: 10.1109/PCCGA.2004.1348363
- [36] Benedetti, Luca, et al. "Painting with Bob: assisted creativity for novices." *Proceedings of the 27th annual ACM symposium on User interface software and technology*. pp. 419-428, 2014. DOI: 10.1145/2642918.2647415
- [37] Bednarik, Jan, Pascal Fua, and Mathieu Salzmann. "Learning to reconstruct texture-less deformable surfaces from a single view." *2018 international conference on 3d vision (3DV)*. IEEE, pp. 606-615, 2018. DOI: 10.1109/3DV.2018.00075
- [38] Hun-hyung Lee. "A Study on the Oil Color's Light Stability and Color Change Resulting from Ultraviolet Rays". *Journal of Korean Society of Color Studies*. pp.106-116. 2013.

Authors



Jong-Hyun Kim received the B.A. degree in the Department of Digital Contents at Sejong University in 2008. He received M.S. and Ph.D. degrees in the Department of Computer Science and Engineering at Korea University,

in 2010 and 2016. Prof. Kim is an Associate Professor in the College of Software and Convergence (Dept. of Design Technology) in Inha University. His current research interests include fluid animation and virtual reality.