



Open Access

DOI: 10.3785/j.issn.1008-9497.2025.01.002

基于移动最小二乘的图像重着色

周家伟^{1,2}, 黄子康^{1,2}, 彭俊豪^{1,2}, 李盛善^{1,2}, 刘晓静^{1,2}, 杜正君^{1,2*}

(1. 青海大学 计算机与应用学院, 青海 西宁 810016; 2. 青海省智能计算与应用实验室, 青海 西宁 810016)

摘要: 近年来, 基于调色板的图像重着色技术在图形设计、照片编辑和广告制作等领域广受关注并得到应用, 还出现了一系列优化调色板提取和图像重着色效果的方法。但由于现有技术的编辑局部性, 用户在调整局部区域时常导致图像整体色调发生不可预知的变化。为此, 提出了一种新颖的基于移动最小二乘的图像重着色算法, 显著提升了颜色编辑的局部性和精准度。首先, 采用改进的 K -means 聚类算法提取输入图像的调色板。然后, 将图像及其调色板映射至 RGB 颜色空间, 并将其视作三维点云和控制点。最后, 将基于调色板的图像重着色任务转化为经典的点云变形问题。通过编辑调色板颜色精确控制图像的颜色变化, 并借助移动最小二乘算法优秀的局部性实现局部控制。算法允许用户通过简单编辑, 实时生成自然的、满足用户编辑意图的重着色结果。实验和用户调研结果表明, 相较于已有算法, 本文算法能得到更加自然、更符合用户预期的图像重着色效果。

关键词: 调色板; 重着色; 颜色编辑; 移动最小二乘; 局部性

中图分类号: TP 391

文献标志码: A

文章编号: 1008-9497(2025)01-010-12

ZHOU Jiawei^{1,2}, HUANG Zikang^{1,2}, PENG Junhao^{1,2}, LI Shengshan^{1,2}, LIU Xiaojing^{1,2}, DU Zhengjun^{1,2} (1. School of Computer Technology and Application, Qinghai University, Xining 810016, China; 2. Qinghai Provincial Laboratory for Intelligent Computing and Application, Xining 810016, China)

Palette-based image recoloring based on moving least squares. Journal of Zhejiang University (Science Edition), 2025, 52(1):10-21

Abstract: In recent years, palette-based image recoloring techniques have garnered widespread attention and application across various domains such as graphic design, photo editing, and advertising production. Researchers have proposed a series of methods to optimize palette extraction and image recoloring effects. However, the existing methods still suffer from poor editing locality, often leading to unpredictable changes in overall image tone even users adjust specific areas. Addressing this issue, this paper introduces a novel image recoloring algorithm based on moving least squares, significantly enhancing the locality and precision of color editing. Specifically, we first employ an improved K -means algorithm to extract the palette of the input image. Next, we map the image and its palette to the RGB color space, treating them as a three-dimensional point cloud and control points. Finally, we transform the palette based image recoloring task into a problem of classic point cloud deformation. The proposed approach enables users to precisely control color variations in the image by editing the palette colors, leveraging the excellent locality of the moving least squares algorithm to achieve effective local color editing. Overall, our method enables users to generate real-time, natural-looking recoloring results that align with their editing intentions through simple edits. Extensive experiments and user studies demonstrate that our approach can produce more natural and user-desired image recoloring effects compared to existing techniques.

Key Words: palette; recolor; color edit; moving least squares; locality

收稿日期: 2024-07-03; 修回日期: 2024-09-02; 接受日期: 2024-10-09; 出版日期: 2025-01-25.

基金项目: 青海省自然科学基金青年项目(2023-ZJ-951Q).

作者简介: 周家伟(1998—), ORCID: <https://orcid.org/0000-0002-6763-2892>, 男, 硕士研究生, 主要从事图像视频处理研究.

*通信作者, ORCID: <https://orcid.org/0000-0002-6763-2892>, E-mail: dzj@qhu.edu.cn.

在图像和视频处理中,颜色编辑是关键步骤之一。现有方法,如编辑传播方法^[1-5],通常需要用户提供密集的笔画,导致交互频繁。风格迁移方法^[6-8]虽然可自动将参考图像的风格映射至目标图像,但无法精准调整输入图像。而基于调色板的方法^[9-12]允许用户通过编辑少量代表性颜色调整图像的颜色,不仅交互方式简单还为用户提供了一定的编辑自由度。近年来,基于调色板的方法因其简单和高效广受学界关注。

在基于调色板的图像重着色算法中,调色板被定义为能反映输入图像主要色彩分布的代表性颜色。WANG等^[12]通过计算图像在RGB空间的凸包构建调色板。首先将图像投影至RGB空间,并将其视为RGB空间的三维点云。然后构建点云的凸包,并将凸包顶点作为调色板颜色。在提取调色板(凸包) P 后,输入图像中任意像素的颜色 I_i 均可通过广义重心坐标插值表示为调色板颜色(凸包顶点) P_j 的凸组合,即 $I_i = \sum_j W_{ij} P_j$,其中 W_{ij} 为 I_i 相对于 P_j 的混合权重。假设编辑后的调色板为 P'_j ,那么编辑后的颜色 I_i 可表示为 $I'_i = \sum_j W_{ij} P'_j$ 。对所有像素点执行此操作,便可得到重着色后的图像。值得注意的是,混合权重 W_{ij} 可预先计算,且在后续的重着色过程中保持不变。

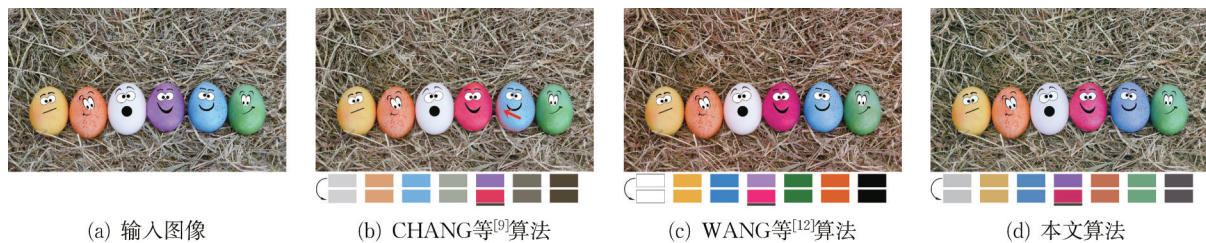


图1 重着色示例

Fig.1 Recoloring examples

为解决上述问题,提出了一种新颖的基于移动最小二乘的图像重着色算法,包含调色板提取和颜色映射两个阶段。在调色板提取阶段,基于CHANG等^[9]的算法,用改进的K-means聚类算法提取调色板,使得提取的调色板颜色具有足够的代表性,从而保证后续颜色编辑的直观和便捷。在颜色映射阶段,将颜色映射任务形式化为基于移动最小二乘的变形问题,在颜色编辑过程中,实现了更好的局部控制。与现有算法相比,本文算法(图1第4列)更符合用户意图,且更有效地保留了其他部分的颜色。

主要贡献:提出了一种新颖的基于调色板和移

动最小二乘法

尽管基于凸包^[10-12]的方法为图像重着色提供了强大的基础,但往往对重着色特定物体或区域内的无关物造成不必要影响,即颜色编辑局部性较差,通常“局部性”一词出现在网格变形领域,指一个特定的控制点对其邻近区域的形变影响较大,对较远区域的影响较小。优良的局部性可确保用户对局部细节的精确控制,并保持整体网格形状的稳定性和一致性。类似地,在图像编辑领域,一个具有良好编辑局部性的重着色方法应能精确按照用户意图对指定目标区域进行重着色,并尽可能减少对图像其他无关区域颜色的干扰和影响。

现有方法多在重着色特定物体或局部区域产生意外的全局颜色变化。如用户希望将图中紫色的彩蛋变成红色,现有方法常常出现颜色溢出(图1第2列)或背景颜色显著变化(图1第3列)等问题。造成编辑局部性差的主要原因有:(1)提取的图像调色板要么缺乏代表性,要么重着色算法缺乏局部性考虑,某些代表性颜色可能并未包含在调色板中,因此在对图像中相近颜色的物体进行编辑时会变得烦琐且不直观。(2)混合权重的求解结果对颜色编辑的效果也有重要影响,如果混合权重不够稀疏,修改单个调色板颜色可能导致大范围颜色发生变化,影响颜色编辑的局部性和准确性。

动最小二乘法的图像重着色算法。将输入图像及其对应的调色板颜色巧妙地视为RGB三维空间中的点云与控制点,进而运用移动最小二乘法实现了更精准的颜色编辑。定性和定量实验均表明,本文算法在实践中具有明显优势。

1 相关工作

首先回顾与本文算法相关的基于调色板的、基于编辑传播的和基于风格迁移的颜色编辑算法。

1.1 基于调色板的颜色编辑算法

近年来,基于调色板的颜色编辑技术越来越受

关注,因其简单且高效,被广泛应用于图像和视频编辑领域,通常包含调色板提取和颜色映射 2 个步骤,前者用于提取一组代表性颜色,以表示输入图像的颜色分布,后者将用户对调色板的编辑映射至整张图像。

调色板提取,早期大多采用数据驱动的方法^[13-15],从输入图像中提取调色板或颜色主题。CHANG 等^[9]和 ZHANG 等^[16]采用 K -means 聚类获取图像调色板。TAN 等^[10-11]和 WANG 等^[12]则提出将图像在 RGB 空间的简化凸包作为图像调色板。近期,又出现了几种新的调色板提取算法,ZHANG 等^[17]利用盲色分离模型同时从输入图像中提取图像调色板并计算混合权重;AKSOY 等^[18]将输入图像分割成一组区域,并将这些区域的平均颜色作为图像调色板;DIVERDI 等^[19]提出了用于数字绘画的调色板,可方便地合成新的颜色。

颜色映射是将调色板的颜色变化精准地映射至输入图像,实现对颜色的精准调整。此过程不仅要求颜色映射函数能精准捕捉调色板的颜色变化,而且需确保映射结果能最大程度地满足用户的编辑意图。CHANG 等^[9]在 lab 颜色空间,分别进行了亮度(l)映射和色度(ab)映射,对于亮度映射,按照亮度值对所有调色板颜色进行排序,并强制要求编辑后的调色板颜色亮度保持不变,以确保图像编辑前后亮度一致;对于色度映射,将图像中任意像素的颜色表示为调色板颜色变化量的线性组合,通过修改调色板的颜色驱动图像颜色的变化。与以往的工作不同,TAN 等^[10-11]和 WANG 等^[12]用 RGB 空间中的 3D 凸包表示调色板,进一步通过广义重心坐标插值的方式将每个像素 I_i 自然地表示为调色板颜色 P_j (凸包顶点)的凸组合,即 $I_i = \sum_j W_{ij} P_j$,用户可通过修改调色板颜色调整图像颜色。

最近,CHO 等^[20]提出了“PaletteNet”,利用神经网络预测输入图像变化后的颜色以匹配用户提供的目标调色板。AKIMOTO 等^[21]用 U-Net 估计给定图像及其对应调色板的混合权重。然而,基于深度学习的重着色算法要求调色板颜色的数量是固定的,且难以对图像颜色进行精确编辑。

上述方法虽然可实现直观、高效的颜色编辑,但在重着色局部区域易导致意外的全局颜色变化。针对此问题,本文通过改进的 K -means 聚类提取调色板,将颜色映射构建为基于移动最小二乘的变形问题,从而显著减小在编辑重着色局部区域时对其他部分的影响。

1.2 基于编辑传播的颜色编辑

编辑传播^[2-3,22-24]是一种广为人知的颜色编辑技术,通常要求用户在图像上添加一些稀疏的笔画,以指示颜色、亮度、对比度等的变化,然后将其自动传播至图像的其余部分。

LEVIN 等^[22]首次将编辑传播应用于彩色化任务,并取得了令人满意的结果。首先将带有笔画的灰度图像转换至 YUV 色彩空间,并假设具有相似强度(Y)的像素被赋予相似的颜色;然后,将彩色化任务转化为构建一个二次能量优化问题,通过求解线性方程组获得像素的 UV 色度值;最后将亮度通道和色度通道相融合得到彩色图像。AN 等^[3]通过计算所有像素对之间的相似性,实现了长距离编辑传播。以上算法通常计算成本高昂,需要求解大规模线性方程组。为解决此问题,XU 等^[23]提出了一种高效的基于相似性的编辑传播算法,通过使用 5D 空间 RGBXY 中的 k -d 树,显著减少了计算和存储开销。后来,LI 等^[24]提出了对图像和视频进行稀疏编辑的即时传播算法。在上述研究中,编辑传播被构建为在高维特征空间中的插值问题,从而极大地提高了求解速度,降低了内存需求,并且也能对高分辨率图像进行实时反馈。尽管 XU 等^[23]和 LI 等^[24]的算法在节省时间和空间方面取得了较大进展,但需要提供精细的笔画,加重了用户的使用负担。

随着深度学习的发展,ENDO 等^[25]提出了一种用于编辑传播的神经网络 DeepProp,利用卷积神经网络(CNN)提取高维特征进行编辑传播。GUI 等^[26]采用端到端的全卷积网络,将编辑传播转化为多元分类问题。

编辑传播算法通常需要较多的用户交互,而基于调色板的算法可通过编辑一组代表性颜色调整图像色彩,更容易理解,使用更方便。

1.2 基于风格迁移的颜色编辑算法

风格迁移是另一种经典的颜色编辑技术,通过将参考图像的颜色和纹理特征转移至目标图像,实现全局颜色调整。REINHARD 等^[6]通过优化参考图像和输入图像的颜色分布差异得到输出图像。NEUMANN 等^[27]使用一种基于 3D 直方图匹配的方法进行颜色映射,即便参考图像和目标图像之间的对应关系较弱也能得到自然的颜色编辑结果。PITIE 等^[7]设计了一种连续变换算法,将维特征分布映射到另一个维空间,成功实现不同内容的两个图像之间的颜色迁移。但这些算法只考虑了低层次的视觉特征,因此当参考图像和目标图像差异较大时,可能会产生明显的伪影。近年来,深度学习算法

在图像风格迁移领域取得了显著进展,主要包括基于CNN的风格迁移算法^[8,28]和基于GAN的风格迁移算法^[29-30]。这些算法均参考图像和目标图像语义特征的对应关系进行迁移。然而,基于深度学习的算法虽然强大,但只根据参考图像进行颜色迁移,无法根据用户的意图操控局部细节特征,编辑自由度非常有限。

编辑传播算法通常需要频繁的用户交互,而风格迁移算法只能提供有限的颜色变化控制。基于调色板的图像编辑在用户交互代价和控制自由度之间实现了很好的平衡。本文在调色板基础上,利用移动最小二乘优良的局部性增强图像颜色编辑的直观性和局部性,取得了良好的效果。

2 算法

如图2所示,算法主要包括调色板提取和颜色映射两阶段。调色板提取阶段,首先,给定输入图像(第1列),将其投影至RGB空间,并将输入图像视为一组点云(第2列);然后,用改进的K-means聚类算法提取对应的调色板(第3列);一旦获得调色板,用户就可通过修改调色板颜色(第4列)调整图像的颜色。颜色映射阶段,根据调色板颜色的变化调整点输入图像对应的点云形状并映射至输出图像。最后两列显示的是形变后的点云和生成的图像。

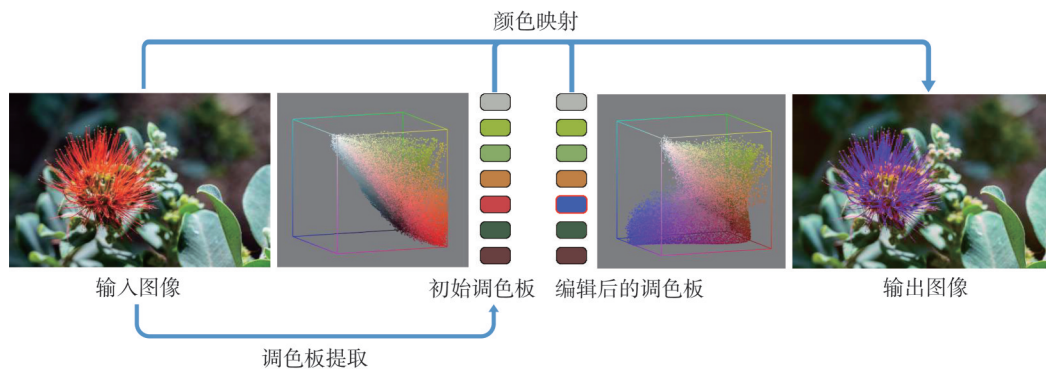


图2 算法流程

Fig.2 Pipeline of our approach

2.1 调色板提取

调色板提取的目标是获得一组代表性颜色以反映输入图像的主要颜色分布。通常应满足3个条件:(1)调色板中的颜色应具有较高的区分度,确保调色板颜色较低的冗余度;(2)调色板颜色应具有较高的代表性,方便用户理解和执行局部编辑;(3)为方便操作,调色板应包含较少的颜色,因为调色板颜色越多编辑工作越烦琐。为此,提出了一种用改进的K-means聚类算法提取符合上述条件的高质量调色板。

首先将输入图像 I 的所有像素投影至RGB空间,使颜色为 (r, g, b) 的像素对应于RGB空间中的点 $p(r, g, b)$ 。因此,可将输入图像自然地视为RGB空间的一个点云。在包含数百万个像素的图像上进行K-means聚类非常耗时,为降低计算成本,对输入图像进行采样,对采样点进行K-means聚类,以提升聚类的计算效率,即将RGB空间 $[0, 255]^3$ 均匀地划分为 $16 \times 16 \times 16$ 个网格,对每个非空网格(空网格表示无像素),将网格内所有像素颜色的平均值

作为该网格的采样点,采样后,最多只有4 096个采样点参与聚类,使参与聚类的数据量与图像分辨率无关,从而提高计算效率。然后,使用改进的K-means算法对采样点进行聚类,获得输入图像的调色板。

初始中心的选择对K-means聚类结果有重要影响。随机初始化中心通常会减慢收敛速度,并可能得到局部最优的结果。鉴于初始中心之间的代表性和差异性,采用类似于K-means++的算法选择初始中心^[31]。假设采样点集合为 $S = \{s_i\}$,为每个采样点 s_i 分配一个显著性权重 n_i 。最初, n_i 等于 s_i 所在区间的像素数量。在每次迭代中,选择具有最大显著性权重的采样点并将其添加至初始中心集合,然后更新剩余采样点的显著性权重。具体地,如果 s_i 被选为初始中心之一,则任意候选样本点 s_j 的显著性权重 n_j 被更新为

$$n_j = [1 - \exp(-d_{ij}^2/\sigma^2)]n_j, \quad (1)$$

其中, d_{ij} 表示 s_i 和 s_j 之间的欧氏距离, σ 设置为0.5。每次迭代,接近 s_i 的采样点的显著性权重大幅衰减,即远离当前已确定聚类中心的采样点更有可能在后

续过程中被选中,这符合初始中心选择的代表性要求。重复上述迭代过程,直到所有 k 个初始中心被选中,迭代停止。

由于 lab 颜色空间更接近人类视觉感知,因此先将采样点以及选定的初始均值从 RGB 空间转换到 lab 空间再进行 K -means 聚类。聚类结束后,再将得到的 k 个聚类中心重新转换回 RGB 空间,以方便后续的颜色映射。根据经验,将 k 设置为 7,以平衡调色板的代表性和用户操作的复杂性。与 CHANG 等^[9]的区别:由于本文算法的颜色映射过程在 RGB 空间进行,因此在 RGB 空间计算采样点,然后投影至 lab 空间聚类,以便更好地描述图像

像素颜色在 RGB 空间的分布。固定最暗颜色可能会扰乱聚类结果,进而损害调色板颜色的代表性,因此,本文算法仅在聚类时直接对 lab 空间采样点进行聚类分析,以生成分布更均匀且具有代表性的调色板颜色。将本文算法和另外两种现有算法生成的调色板进行了比较,结果如图 3 所示。可知本文算法的调色板比文献[9]和文献[12]的调色板区分度更好。以第 1 列为例,现有算法生成的调色板包含相似的蓝色,并未捕捉到花朵的紫色。本文算法生成的调色板成功地提取了输入图像的主要颜色,且调色板颜色具有更高的区分度和代表性。



图 3 不同算法生成的调色板比较

Fig.3 Comparison of color palettes generated by different approaches

一旦提取了输入图像的调色板,用户就可通过修改调色板调整图像的颜色。

2.2 颜色映射

颜色映射的目标是将编辑后的调色板映射至输入图像。关于图像重着色效果的评估,除是否满足编辑意图外,还需注重颜色编辑的局部性,即在修改调色板颜色时,仅变化重着色图像中相关像素,不会产生全局的颜色变化。虽然现有方法可生成自然而直观的重着色图像,但通常不具备良好的编辑局部性。为解决此问题,受图像和网格变形技术的启发,将移动最小二乘法^[32-33]引入至颜色映射,以实现良好的编辑局部性。

首先给出此问题的形式化定义。给定输入图像 I ,提取的调色板为 $C = \{C_1, C_2, \dots, C_k\}$,修改后的调色板为 $C' = \{C'_1, C'_2, \dots, C'_k\}$ 。通过颜色映射函数 f 计算重着色图像 I' 。值得注意的是,本文将颜色映射视为刚体形状变形问题,即使用控制点(调色板颜色)将投影到 RGB 的点云(输入图像)映射至变形后的位置(重着色图像),将输入图像的每个像素 p 转换为 p' ,即 $p' = f(p)$ 。为实现令人满意的颜色映射效果, f 需满足:

(i) 平滑性约束:在输入图像中将颜色相似的像素映射至相似的颜色,即 $\lim_{p \rightarrow q} f(p) = f(q)$ 。

(ii) 编辑约束:将颜色为原调色板颜色的像素精确映射至编辑后的调色板颜色,即 $p = C_i \Rightarrow f(p) = C'_i$,以满足用户的编辑意图。

像素点 p 的变换定义为

$$f_p(x) = Rx + T, \quad (2)$$

其中, x 为自变量, R 为 3×3 的正交矩阵, T 为平移矩阵。此变换在网格变形中被称为刚体变形,具有非常优异的形变局部性。

根据约束(ii),所有原调色板的颜色在变换后应尽可能接近编辑后的调色板颜色。因此,将能量函数定义为

$$E = \sum_i^k w_i |f_p(C_i) - C'_i|^2, \quad (3)$$

其中, w_i 为权重参数,

$$w_i = 1/|C_i - p|^4. \quad (4)$$

研究目标是通过优化式(3)求解式(2)中的正交矩阵 R 和平移矩阵 T ,并通过 $p' = f_p(p) = Rp + T$ 得到像素 p 的重着色结果。为简化方程,将式(2)代入

式(3),并令式(3)的偏导数为0,得到 $T = \overline{C}' - R\overline{C}$,其中,

$$\overline{C} = \frac{\sum_i^k \omega_i C_i}{\sum_i^k \omega_i}, \quad \overline{C}' = \frac{\sum_i^k \omega_i C'_i}{\sum_i^k \omega_i}, \quad (5)$$

消除 T 后,式(2)即为 $f_p(x) = Rx + \overline{C}' - R\overline{C}$,代入式(3),有

$$E = \sum_i^k |R\sqrt{\omega_i}\widehat{C}_i - \sqrt{\omega_i}\widehat{C}'_i|^2, \quad (6)$$

其中, $\widehat{C}_i = C_i - \overline{C}$, $\widehat{C}'_i = C'_i - \overline{C}'$ 。现设

$$\begin{cases} A = \sqrt{\omega_i}(\widehat{C}_1, \widehat{C}_2, \dots, \widehat{C}_k), \\ B = \sqrt{\omega_i}(\widehat{C}'_1, \widehat{C}'_2, \dots, \widehat{C}'_k), \end{cases} \quad (7)$$

代入式(6),得到

$$\begin{aligned} E &= \|RA - B\|_F^2 = \\ &= \text{tr}((RA - B)^T(RA - B)) = \\ &= \text{tr}(A^T A) + \text{tr}(B^T B) - 2\text{tr}(B^T RA), \end{aligned} \quad (8)$$

由于 $\text{tr}(A^T A)$ 和 $\text{tr}(B^T B)$ 是确定的,要最小化能量函数,需使 $\text{tr}(B^T RA)$ 达到最大,根据迹的性质, $\text{tr}(B^T RA) = \text{tr}(RAB^T)$,然后对 AB^T 执行奇异值分解,使 $AB^T = SVD^T$,其中 S 和 D 都为正交矩阵。可得到 $\text{tr}(RAB^T) = \text{tr}(RSVD^T) = \text{tr}(S^T R^T DV)$ 。由正交矩阵的性质,当 $R = DS^T$ 时,得到 $\text{tr}(B^T RA)$ 的最大值,从而得到最优能量函数解决方案。

求得 R 和 T 后,式(2)可表示为 $f_p(x) =$

$DS^T(x - \overline{C}) + \overline{C}'$ 。计算每个像素的旋转矩阵 R 和平移矩阵 T ,从而修改所有图像像素的颜色。然而,一张图像通常包含数百万像素,为每个像素计算矩阵是非常费时的,为加速计算,将RGB空间划分为 $16 \times 16 \times 16$ 个方格,仅计算方格顶点的颜色映射结果,最后通过三线性插值计算方格内像素的重着色结果。

3 实验

利用C++语言编程实现,实验平台为Intel Core i5-12500 H, 16 GB 内存,操作系统为Windows 11。首先展示本文算法生成的重着色结果,然后验证调色板大小 k ,并将本文算法与2种先进的基于调色板的重着色算法进行比较,最后通过用户调研进一步验证本文算法的有效性。

3.1 重着色结果展示

图4展示了本文算法生成的重着色结果。示例中,输入图像位于第1列,第2~4列为3个不同编辑目标对应的重着色结果。在每个重着色结果下方给出了原调色板和编辑后的调色板。在“森林”示例中,用户通过改变树叶的颜色实现不同季节的风格。在“树木”示例中,用户对背景和前景颜色进行了不同的修改。在“轿车”示例中,用户根据个人喜好改变了汽车的颜色。结果表明,在不同类型样例中,本文算法均实现了自然的重着色结果。

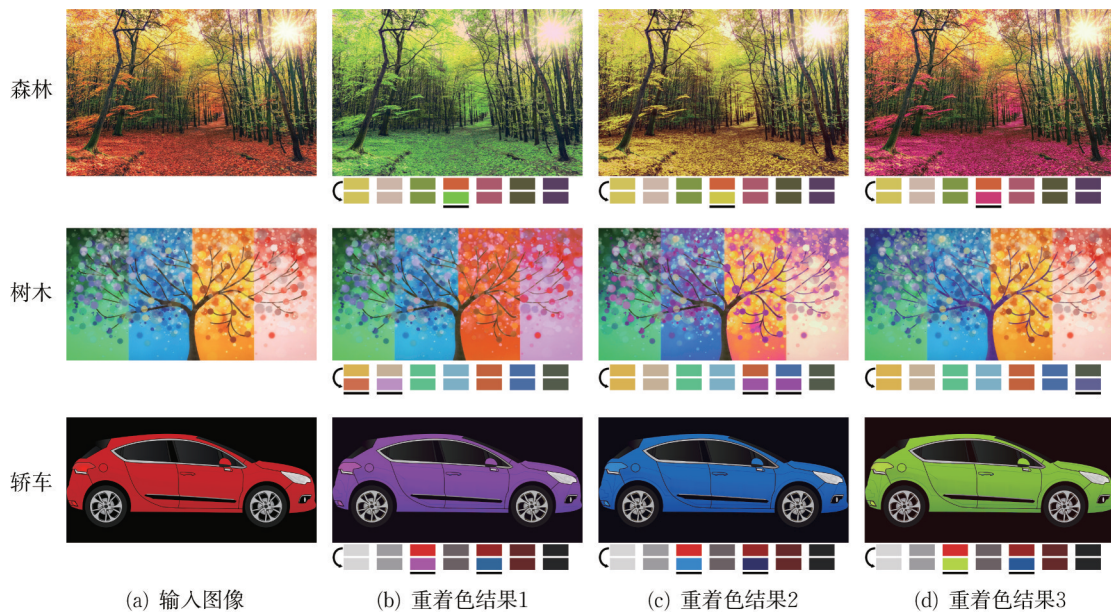


图4 本文算法生成的重着色结果

Fig4 Recoloring results generated by our approach

3.2 调色板大小验证

调色板应当包含适量的颜色,既不太多而缺乏

表现力,也不太多而影响编辑体验。如图5所示,通过3组示例评估颜色数量对调色板的影响。将每组

示例提取的不同数量颜色的调色板置于输入图像下方,以便观察和比较。在“小鹿”示例中,当 $k=5$ 时,调色板忽略了绿色部分,而当 $k=9$ 时,在调色板中出现了多个相似的蓝色和黄色。类似地,在“卧室”

示例中,当 $k=5$ 时,调色板未能捕捉到椅子的深灰色,而当 $k=9$ 时,调色板中包含了意外的紫色。因此,将 $k=7$ 设定为默认的调色板大小,以平衡调色板的代表性和用户操作的便捷性。



图 5 不同 R 值生成的调色板对比

Fig.5 Comparison of color palettes generated using different R values

为更好地解释调色板数对编辑效果的影响,给出了使用不同颜色数量的调色板进行重着色的示例(图 6),在此示例中,用户尝试将树叶的颜色从黄色修改为绿色。使用较小尺寸的调色板进行重着色图 6(b)往往会导致全局颜色变化,例如背景

颜色的意外变化;相反,使用较大尺寸的调色板(图 6(d))通常需要修改多个颜色才能生成理想的结果。相比之下,使用包含 7 种颜色的调色板对图像进行重着色,可实现自然的重着色结果,同时显著减少用户交互。

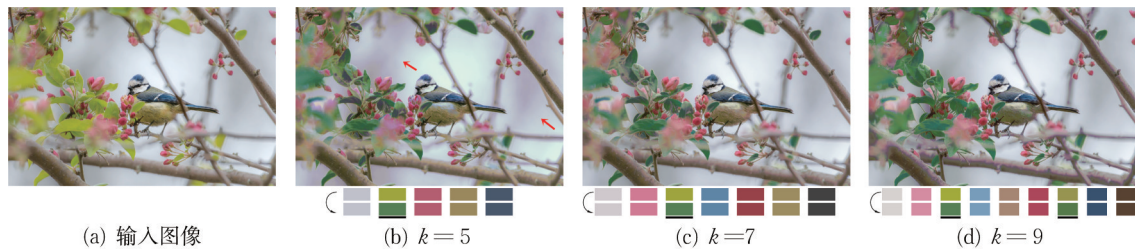


图 6 包含不同颜色数量的调色板对重着色的影响

Fig.6 Recoloring with different sized palettes

3.3 基于相同调色板的图像重着色效果比较

为验证本文基于移动最小二乘的重着色算法的有效性,与文献[9]基于聚类的重着色算法进行了定性比较。在实验过程中,2种算法均使用本文算法生成的调色板进行颜色映射,以更好地评估编辑结果。图 7 展示了 5 个示例的比较结果,其中,第 1 列为输入图像及其调色板,后两列为文献[9]算法和本文算法生成的重着色结果,且 2 种算法均使用相同的调色板变化(修改的颜色用下划线做标记)。在“鸢尾花”和“枝叶”示例中,编辑意图是将鸢尾花瓣和叶片的颜色调整为绿色,文献[9]算法意外地造成了背景的颜色变化。在“草原”和“花丛”示例中,编辑意图是将草和花的颜色调整为红色。文献[9]算法在重着色过程中产生了不自然的模糊效果。在“彩蛋”示例

中,编辑意图是将青色球体的颜色调整为蓝色。文献[9]算法产生了不期望的全局颜色变化,使其他物体的颜色也发生变化,如背景中的花、领结等。本文算法生成的结果更符合编辑意图,自然且无伪影;此外,图像细节得到了更好保留,并且对图像其他无关区域无影响。

3.4 图像重着色结果的定性和定量比较

将本文算法的重着色结果和当前较先进的 2 种算法^[9,12]进行了定性和定量的比较。定性比较主要关注:

- (1) 编辑意图:重着色结果是否达到用户意图;
- (2) 颜色编辑局部性:在编辑特定目标时,非兴趣物体或区域的颜色是否得以保留;
- (3) 重着色质量:重着色图像是否存在伪影。

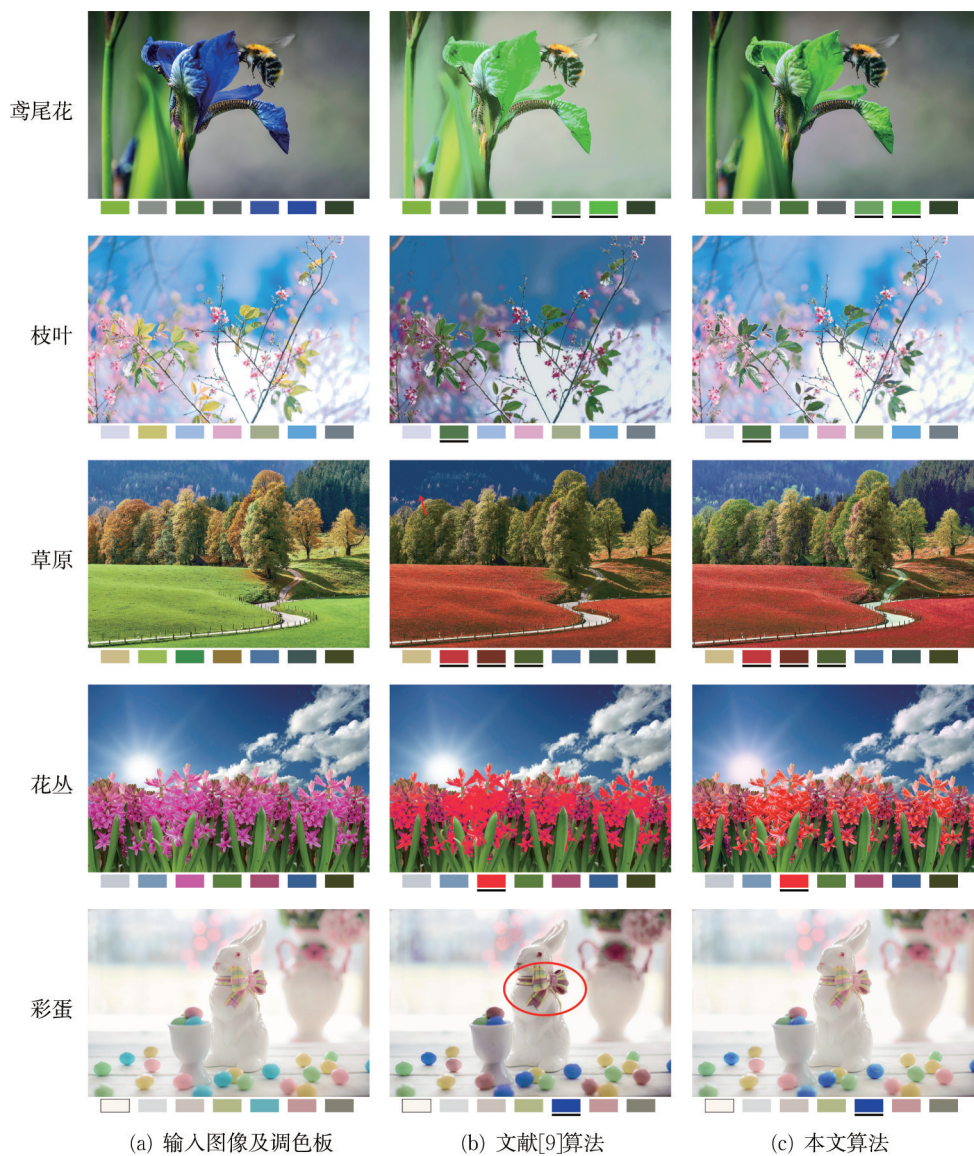


图7 不同算法在相同调色板下的重着色结果比较

Fig.7 Comparison of recoloring results of different algorithms using the same color palette

图8展示了本文算法与文献[9]算法和文献[12]算法的定性比较。在“女士”示例中,编辑意图是将衣服的颜色调整为紫色。虽然2种算法均能满足编辑意图,但文献[9]算法将人脸也着色为不自然的紫色,文献[12]算法则影响了背景颜色。在“壁画”示例中,编辑意图是将帽子的颜色编辑为蓝色,2种算法均令背景产生了意外变化。在“孔雀”示例中,编辑意图是将伪眼的颜色转变为绿色,2种算法均改变了羽毛的颜色。相比之下,本文算法实现了更好的颜色编辑局部性,只改变了编辑部分的颜色,没有改变其他物体的颜色。

除进行上述定性比较外,还将本文算法与2种算法进行了定量比较,结果如表1所示。首先邀请专业人员精心创建了每个示例的重着色目标,然后

使用不同算法对输入图像进行重着色编辑,并使重着色结果尽可能接近编辑目标。通过计算重着色图像与输入图像之间的均方误差(MSE)、峰值信噪比(PSNR)以及结构相似度(SSIM)评估图像的质量。在所有示例中,本文算法生成的图像均方误差均最小,说明本文算法更能满足用户的编辑意图。

3.5 算法的时间性能

表2展示了不同算法各示例在各阶段所花费的时间,包括分辨率、调色板提取时间和颜色映射所需时间。可知,本文算法能高效提取调色板,并能在重着色过程中实时反馈结果。

3.6 用户调研

为进一步评估本文算法的有效性,进行了用户调研实验。邀请21名用户评估3种算法生成的重着

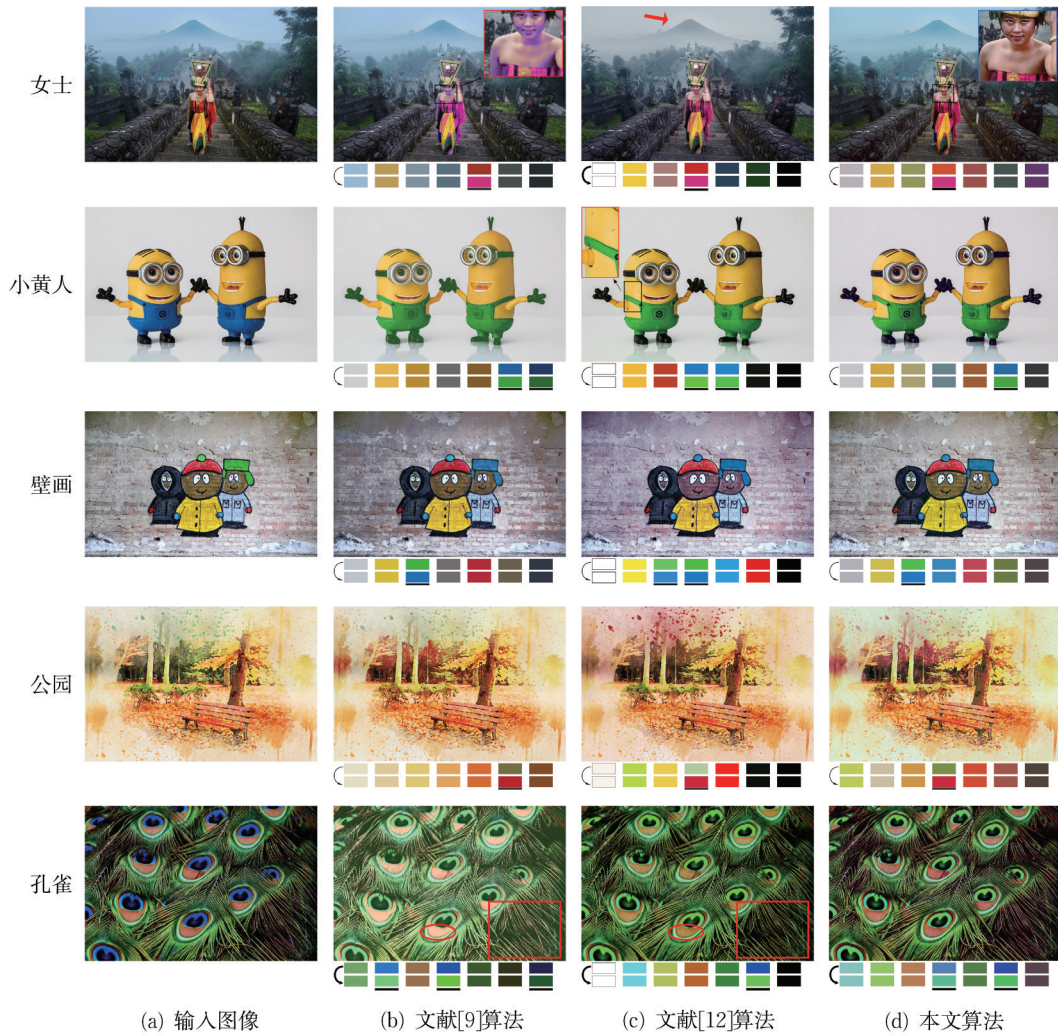


图 8 本文与其他重着色算法的定性比较

Fig.8 Qualitative comparison of ours with other recoloring algorithms

色结果。其中,80%的用户熟悉专业图像编辑软件,如 Adobe Photoshop 或 CorelDRAW。调研问卷主要包含 3 项任务:

任务 1 调色板评估。由 3 种算法生成的 3 个调色板随机排列,匿名展示给用户,让其选出最具代表性的调色板。

任务 2 重着色结果评估。向用户展示输入图像、编辑意图以及采用本文算法生成的重着色效果,邀请其对编辑结果从“非常不满意”到“非常满意”进行 5 个等级评分。

任务 3 编辑意图匹配度评估。向用户提供输入图像和编辑意图,同时匿名随机展示由 3 种算法生成的 3 个重着色结果,要求其选择与编辑意图最吻合的重着色结果。

对于任务 1,分别有 27%,23%,50%的用户认为文献[9]算法、文献[12]算法和本文算法生成的调色板更好。对于任务 2,超过 90%的用户对本文

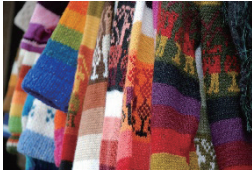
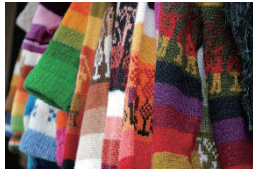


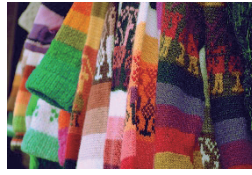

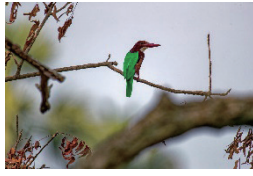
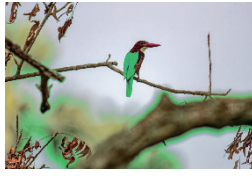
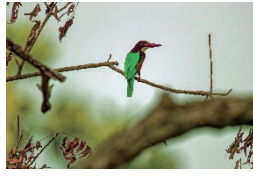
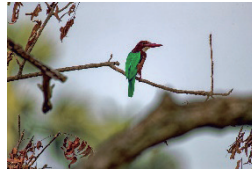










算法的重着色结果给出了满意或更高的评价。对于任务 3,在所有示例中,超过 70%的用户认为本文算法生成的结果最符合编辑意图。

4 结 论

为解决现有图像重着色算法编辑局部性差的问题,将移动最小二乘算法应用于基于调色板的图像重着色。对比实验和用户调研结果表明,本文算法在重着色编辑过程中实现了更好的局部性,并且产生的重着色结果更自然且无伪影。

尽管如此,本文算法仍然存在 2 个主要缺陷。首先,调色板尺寸是固定的,需用户手动选择,无法根据图像自适应确定。其次,具有相同颜色的不同物体无法分别进行重着色,例如,如果一幅图像中包含一个红色苹果和一面红色旗帜,用户无法同时将它们着色为不同颜色。因此,自适应估计调色板尺寸和实现

表1 本文算法与其他重着色算法的定量比较
Table 1 Quantitative comparison of ours with existing methods

输入图像	重着色目标	文献[9]算法	文献[12]算法	本文算法
				
		MSE=0.452 6 PSNR=28.53 SSIM=0.956 3	MSE=0.023 7 PSNR=30.32 SSIM= 0.973 8	MSE=0.007 2 PSNR=32.60 SSIM=0.891 7
				
		MSE=0.357 3 PSNR=21.80 SSIM=0.936 2	MSE=0.005 8 PSNR=29.73 SSIM=0.973 7	MSE=0.001 1 PSNR=42.19 SSIM=0.981 6
				
		MSE=0.202 3 PSNR=21.62 SSIM=0.866 1	MSE=0.003 0 PSNR=35.10 SSIM=0.962 9	MSE=0.001 0 PSNR=44.40 SSIM=0.972 0
				
		MSE=0.208 7 PSNR=24.64 SSIM=0.908 3	MSE=0.422 0 PSNR=30.84 SSIM=0.852 8	MSE=0.012 2 PSNR=35.39 SSIM=0.918 8

语义感知的图像重着色编辑是未来研究的方向。

参考文献 (References):

[1] LEVIN A, LISCHINSKI D, WEISS Y. Colorization using optimization [C]// **ACM SIGGRAPH 2004 Papers**. Los Angeles: ACM, 2004: 689-694. DOI: 10.1145/1186562.1015780

[2] PELLACINI F, LAWRENCE J. AppWand: Editing measured materials using appearance-driven optimization [C]// **ACM SIGGRAPH 2007 Papers**.

San Diego: ACM, 2007. DOI: 10.1145/1275808.1276444

[3] AN X B, PELLACINI F. AppProp: All-pairs appearance-space edit propagation [C]// **ACM SIGGRAPH 2008 Papers**. Los Angeles : ACM, 2008: 1-9. DOI:10.1145/1399504.1360639

[4] XU K, LI Y, JU T, et al. Efficient affinity-based edit propagation using k-d tree [C]// **ACM SIGGRAPH Asia 2009 Papers**. Yokohama: ACM, 2009: 1-6. DOI:10.1145/1661412.1618464

[5] LI Y, JU T, HU S M. Instant propagation of sparse edits on images and videos [J]. **Computer Graphics**

表 2 算法时间统计

Table 2 Time statistics of different algorithms

图像	分辨率	调色板提取时间/ms			颜色映射所需时间/ms		
		文献[9]算法	文献[12]算法	本文算法	文献[9]算法	文献[12]算法	本文算法
森林	1 280×853	21	6 525	20	3	3	4
树木	1 280×677	18	8 719	17	3	3	4
轿车	1 280×640	20	1 733	18	3	3	3
鸢尾花	1 280×853	22	11 817	21	3	3	4
枝叶	1 280×853	21	8 780	22	3	3	4
草原	1 280×853	22	21 039	20	3	3	4
花丛	1 280×853	22	19 002	21	3	3	4
彩蛋	1 280×853	23	12 967	21	3	3	4
女士	1 280×853	22	11 825	22	3	3	5
小黄人	1 280×853	28	10 316	36	3	3	5
壁画	640×401	8	5 231	7	3	3	2
公园	1 280×853	21	7 963	22	3	3	4
孔雀	1 280×854	23	9 827	28	3	3	5

- Forum**, 2010, 29(7): 2049-2054. DOI: 10.1111/j.1467-8659.2010.01791.x
- [6] REINHARD E, ADHIKMIN M, GOOCH B, et al. Color transfer between images[J]. **IEEE Computer Graphics and Applications**, 2001, 21(5): 34-41. DOI: 10.1109/38.946629
- [7] PITIE F, KOKARAM A C, DAHYOT R. N -dimensional probability density function transfer and its application to colour transfer[J]. **IEEE Computer Society**, 2005, 2: 1434-1439. DOI: 10.119/ICCV.2005.166
- [8] GATYS L A, ECKER A S, BETHGE M. Image style transfer using convolutional neural networks[C]// **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. Las Vegas: IEEE, 2016: 2414-2423. DOI: 10.1109/CVPR.2016.265
- [9] CHANG H W, FRIED O, LIU Y M, et al. Palette-based photo recoloring[J]. **ACM Transactions on Graphics**, 2015, 34(4): 1-11. DOI: 10.1145/2766978
- [10] TAN J C, LIEN J M, GINGOLD Y. Decomposing images into layers via RGB-space geometry[J]. **ACM Transactions on Graphics**, 2017, 36(4): 1. DOI: 10.1145/3072959.3126798
- [11] TAN J C, ECHEVARRIA J, GINGOLD Y. Efficient palette-based decomposition and recoloring of images via RGBXY-space geometry[J]. **ACM Transactions on Graphics**, 2018, 37(6): 1-10. DOI: 10.1145/3272127.3275054
- [12] WANG Y L, LIU Y F, XU K. An improved geometric approach for palette-based image decomposition and recoloring[J]. **Computer Graphics Forum**, 2019, 38(7): 11-22. DOI: 10.1111/cgf.13812
- [13] O'DONOVAN P, AGARWALA A, HERTZMANN A. Color compatibility from large datasets[J]. **ACM Transactions on Graphics**, 2011, 30(4): 1-12. DOI: 10.1145/2010324.1964958
- [14] LIN S, HANRAHAN P. Modeling how people extract color themes from images[C]// **Proceedings of the SIGCHI Conference on Human Factors in Computing Systems**. Paris: ACM, 2013: 3101-3110. DOI: 10.1145/2470654.2466424
- [15] CAO Y, CHAN A B, LAU R W H. Mining probabilistic color palettes for summarizing color use in artwork collections[C]// **SIGGRAPH Asia 2017 Symposium on Visualization**. Bangkok: ACM, 2017: 1-8. DOI: 10.1145/3139295.3139296
- [16] ZHANG Q, XIAO C X, SUN H Q, et al. Palette-based image recoloring using color decomposition optimization[J]. **IEEE Transactions on Image Processing**, 2017, 26(4): 1952-1964. DOI: 10.1109/TIP.2017.2671779
- [17] ZHANG Q, NIE Y W, ZHU L, et al. A blind color separation model for faithful palette-based image recoloring[J]. **IEEE Transactions on Multimedia**, 2022, 24: 1545-1557. DOI: 10.1109/TMM.2021.3067463
- [18] AKSOY Y, AYDIN T O, SMOLI A, et al. Unmixing-based soft color segmentation for image manipulation[J]. **ACM Transactions on Graphics**, 2017, 36(2): 1-19. DOI: 10.1145/3002176

- [19] DIVERDI S , LU J , ECHEVARRIA J , et al. Generating playful palettes from images[J]. **Proceedings of the 8th ACM/Eurographics Expressive Symposium on Computational Aesthetics and Sketch Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering**. Genoa: Eurographic Association, 2019:69-78. DOI:10.23R/exp.20191078
- [20] CHO J, YUN S, LEE K, et al. PaletteNet: Image recolorization with given color palette [C]// **2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)**. Honolulu: IEEE, 2017. DOI:10.1109/cvprw.2017.143
- [21] AKIMOTO N, ZHU H C, JIN Y H, et al. **Fast Soft Color Segmentation**[Z/OL]. [2020-04-12]. <https://arxiv.org/abs/2004.08096v1>.
- [22] LEVIN A, LISCHINSKI D, WEISS Y. Colorization using optimization [J]. **ACM Transactions on Graphics**, 2004, 23 (3) : 689-694. DOI: 10.1145/1015706.1015780
- [23] XU K, LI Y, JU T, et al. Efficient affinity-based edit propagation using k-d tree [J]. **ACM Transactions on Graphics**, 2009, 28(5): 1-6. DOI: 10.1145/1618452.1618464
- [24] LI Y, JU T, HU S M. Instant propagation of sparse edits on images and videos[J]. **Computer Graphics Forum**, 2010, 29 (7) : 2049-2054. DOI: 10.1111/j.1467-8659.2010.01791.x
- [25] ENDO Y, IIZUKA S, KANAMORI Y, et al. DeepProp: Extracting deep features from a single image for edit propagation[J]. **Computer Graphics Forum**, 2016, 35(2): 189-201. DOI:10.1111/cgf. 12822
- [26] GUI Y, ZENG G. Joint learning of visual and spatial features for edit propagation from a single image[J]. **The Visual Computer**, 2020, 36(3): 469-482. DOI: 10.1007/s00371-019-01633-6
- [27] NEUMANN A, NEUMANN L. Color style transfer techniques using hue, lightness and saturation histogram matching [C]// **Eurographics Conference on Computational Aesthetics in Graphics**. Dublin: Eurographics Association, 2005. DOI: 10.2312/COMPAESTH/COMPAESTH05/111-122
- [28] ULYANOV D, LEBEDEV V, VEDALDI A, et al. Texture networks: Feed-forward synthesis of textures and stylized images[C]// **Proceeding of the 33rd International Conference on International Conference on Machine Learning**. ACM Digital Library, 2016, 48:1349-1357.
- [29] CHOI Y, CHOI M, KIM M, et al. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation[C]// **IEEE/CVF Conference on Computer Vision and Pattern Recognition**. Wellington: IEEE/CVF, 2018.
- [30] ZHU J Y, PARK T, ISOLA P, et al. **Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks**[Z/OL]. (2017-03-30) [2020-08-24]. <https://doi.org/10.48550/arXiv.1703.10593>.
- [31] ARTHUR D, VASSILVITSKII S. K -means++ : The advantages of careful seeding[J]. **Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms**, 2007(7/8/9): 1027-1035.
- [32] SCHAEFER S, MCPHAIL T, WARREN J. Image deformation using moving least squares [J]. **ACM Transactions on Graphics**, 2006, 25 (3) : 533-540. DOI:10.1145/1141911.1141920
- [33] ZHU Y C, GORTLER S. **3D Deformation Using Moving Least Squares**[Z/OL]. 2007. https://www.research.net/publication/242367766_3D_Deformation_Using_Moving_least_Squares.