SPARSE EDIT PROPAGATION FOR HIGH RESOLUTION IMAGE USING SUPPORT VECTOR MACHINES

Changjae Oh*, Seungchul Ryu*, Youngjung Kim*, Jihyun Kim†, Taewoong Park†, and Kwanghoon Sohn*

*School of Electrical and Electronic Engineering, Yonsei University, Seoul, Korea
†Creative Innovation Center, LG Electronics, Seoul, Korea

ABSTRACT

In this paper, we formulate image edit propagation as a task of machine learning to handle a high resolution image efficiently. Conventional graph-based methods solve the edit propagation by minimizing an energy function which considers the relationship between a reference pixel and its spatially neighboring ones. It is becoming a time-consuming and memory-requiring task due to the increase of the image size. Inspired by the observation that similar features get analogous edits, the edit propagation is casted as a classification problem using support vector machines in the feature space. A classifier is trained with initial sparse edits given by user interaction, and then the rest of the features are classified and manipulated. In experiments, the proposed method is applied to an image recoloring to verify the performance. Experimental results show that the proposed method gives competitive editing results comparing to other state-of-the-art methods.

Index Terms—Edit propagation, support vector machines, image recoloring

1. INTRODUCTION

Image editing has been widely used for various applications to manipulate original images. Since it is hard to reflect user’s intention with an unsupervised manner, many of recent works have exploited an interactive way to solve the problem, e.g., image segmentation [1, 2], matting [3, 4], and colorization [5, 6, 7]. However, it has shown limitations in practical usage since the user interaction might become complicated to achieve the image editing effectively.

Edit propagation simplifies the task of image editing in which only sparse edits are interacted to an image [8, 9, 10, 11, 12, 13, 14]. The sparse edits are then propagated to the nearby regions by considering their appearance. In this scheme, only a few edits are required since the initial edits are automatically propagated to neighbors until the problem converges to the optimal solution.

Conventionally, many researches have solved the edit propagation as a graph-based optimization where the initial edits are propagated to their spatially adjacent neighborhoods during an optimization [8, 9, 10, 11]. The edits can be effectively propagated to long range since global structures are captured in the optimization. However, noting that the graph-based optimization heavily depends on the data size, it is becoming a time-consuming and memory-requiring task since the resolution of an image is getting higher.

In order to address this problem, we present an edit propagation which handles the problem in a feature space using support vector machines (SVM). The main contributions of the paper are described as follows. First, the edit propagation is reformulated as a task of machine learning to handle the features effectively. Under the observation that similar features get analogous edits, we recast the problem as an edit classification by leveraging SVM to generate an optimal classifier. Second, the image editing is directly performed to the feature space. The features are directly manipulated in the feature space, which makes the computation of the edit classification invariant to the image size. Lastly, in experiments, we present an image recoloring as an application to demonstrate that the proposed method handles a high resolution image effectively.

The rest of this paper is presented as follows. In section 2, we describe previous works in edit propagation. Section 3 presents the proposed edit propagation using SVM in the feature space. The experimental results are shown in section 4. Finally, in section 5, we conclude the paper with a brief summary.

2. RELATED WORKS

Various studies have been proposed to enhance the accuracy and complexity of edit propagation [8, 9, 10, 11, 12, 13, 14]. An et al. employed all-pair constraints to enable long-range propagation [8]. Farbman et al. presented a diffusion-based distance metric for computing the affinity between two image pixels [9]. For efficient implementation, both of these methods accelerated the distance computation from sparsely sampled pixel pairs with the Nyström method [15]. Chen et al. proposed a manifold-preserving propagation to avoid color blending during edit propagation [10]. In [11], Xu et al. presented a sparse control model which resolves the ambiguity and possible conflict by employing Gaussian mixture model.
Since these researches rely on the graph-based optimization, the accuracy of the image editing shows promising results. However, they suffer from handling a high resolution image since solving a large linear system causes the expensive computational cost and memory requirement.

Several approaches have been proposed to reduce the computational cost and memory requirement of edit propagation while not sacrificing the performance [12, 13, 14]. Xu et al. clustered appearance values to achieve the accelerated affinity computation and edit propagation [12]. Li et al. addressed the edit propagation as a function interpolation where initial edits are simply interpolated to rest of the image [13]. Since these methods find approximated solutions for fast computation, the artifacts might be incurred during edit propagation. Recently, Chen et al. employed dictionary learning to obtain a compact set of representative samples in an image, which achieves significant reduction of memory requirement while preserving the performance [14].

3. PROPOSED METHOD

3.1. Background and motivation

Conventional approaches have addressed edit propagation problem by the manner of graph-based optimization [8, 9, 10, 11]. For every pixel index $i$ in an input image, the edit propagation computes values of output editing $e_i$, which minimizes an energy function as follows:

$$E = \sum_i \sum_{j \in \Omega} z_{ij} (e_i - g_j)^2 + \lambda \sum_i \sum_{j \in N_i} z_{ij} (e_i - e_j)^2,$$

where $g_j$ is an initial seed located at pixel $j$. With a given set of initial edits $\Omega$, the initial edit of each pixel index $i$ is propagated to its neighbors $N_i$ by considering the affinity between $i$ and $j$, i.e. $z_{ij}$, which is defined as follows:

$$z_{ij} = \exp \left\{ -\| f_i - f_j \|_2^2 \right\}$$

$$= \exp \left\{ -\| c_i - c_j \|_2^2 / \sigma_c^2 - \| p_i - p_j \|_2^2 / \sigma_p^2 \right\},$$

where $f_i = (c_i / \sigma_c, p_i / \sigma_p)$ is a feature vector of $I$ ($p_i$) represented by color $c_i$ and position $p_i$. Effects of color and spatial similarities are controlled by $\sigma_c$ and $\sigma_p$, respectively.

Since the conventional approaches have aimed at propagating initial edits based on the spatially neighboring pixels with respect to their color and spatial affinities [8, 9, 10, 11, 12], they cannot avoid the increase of complexity as the image size becomes larger. Although there have been approaches to alleviate the computational costs, the solutions are still affected by the image size [8, 9, 12], or the qualitative performance becomes degraded [13].

3.2. Sparse edit propagation using SVM

We propose to handle the edit propagation in a feature space that is invariant to the image size while maintaining the performance of the image editing. Under the observation that similar features get analogous edits, we recast the edit propagation as an edit classification by leveraging a framework of machine learning. Machine learning has been actively employed to estimate a decision boundary which determines true labels from the inputs, and it has shown deep connections with image processing, computer vision, and graphics. One of the advantages in using a machine learning technique is that it is rather flexible to train features without any definition of neighborhood connections. Thus, it is efficient to handle arbitrary feature space while the conventional graph-based methods require additional definition to neighbors [8, 9, 10, 11, 12].

The proposed method employs SVM to handle the edit classification effectively. SVM is one of the most famous machine learning techniques in solving classification problems [16]. The goal of SVM is to find a classifier which has the minimum error on the test samples. This corresponds to computing the hyper plane which maximizes the margin between training samples.

The overview of the proposed method is presented in Fig. 1. Let us denote an original feature space $f = \{f_1, f_2, \ldots, f_N\}$ where $N$ is the size of $f$. It can be converted to a feature space which is formulated by an arbitrary function $\psi (f) = \{\psi_1^f, \psi_2^f, \ldots, \psi_M^f\}$ having the size of $M$. Note
Algorithm 1 Image Recoloring using SVM

Let \( \psi(f_i) = (\psi_{cb}(f_i), \psi_{cr}(f_i)) \) where chromatic components on pixel \( i \) are used, and \( \hat{\psi}(f_i) \) as target color.

1. Feature representation: \( \psi(f) = \{ \psi_1^f, \psi_2^f, ..., \psi_M^f \} \)
2. Train \( \psi(g) \) based on (3)
3. for each feature \( \psi_i^f \) do
   4. if \( \text{sgn} \left( w^T \phi(\psi_i^f) + b \right) \) then
   5. \( \hat{\psi}(f_i) \leftarrow \psi(f_i) \)
   6. end if
   7. end for
8. for each pixel \( i \) do
   9. \( e_i \leftarrow M(\psi(f_i)) \)
10. end for

that \( \psi(\cdot) \) can be designed in various manners by the purpose of a feature description. Then, the initial edits \( g = \{g_1, g_2, ..., g_{N_g} \} \) are represented in the feature space as \( \psi(g) = \{ \psi_{cb}^g, \psi_{cr}^g, ..., \psi_{cb}^g \} \), where \( N_g \) denotes the total size of initial edits and \( M_k \) is that of initial edits in the converted feature space.

By reformulating the edit propagation as the edit classification using SVM, the initial edits can be viewed as a set of labeled training samples \( (\psi_i^g, y_i) \), where \( y_i \) is a label assigning 1 for positive samples and 0 for negative ones. The training samples are then used to compute a hyper plane \( w, b \), and slack variable \( \xi_i \) to minimize an objective function as follows:

\[
\min_{w, b, \xi} \frac{1}{2} w^T w + \lambda \sum_{i=1}^{M_k} \xi_i \quad \text{s.t.} \quad y_i \left( w^T \phi(\psi_i^g) + b \right) \geq 1 - \xi_i, \quad \xi_i \geq 0,
\]

where \( \phi(\psi_i^g) \) denotes a transformation function and \( \lambda \) controls the trade-off between error and margin. In (3), it can be formulated as the method of Lagrange multipliers, which is solved by the quadratic programming to its dual form [16].

Finally, the editing candidates are determined by classifying the test samples \( \psi(f) \) with the computed decision boundary from (3). The decision of a test sample \( \psi_i^f \), i.e., \( S(\psi_i^f) \), is performed as follows:

\[
S(\psi_i^f) = \text{sgn} \left( w^T \phi(\psi_i^f) + b \right),
\]

where \( \text{sgn}(\cdot) \) is the sign function which allocates 1 for positive decisions and 0 for negative ones. The classified samples from (4) are then manipulated by the target edits \( \hat{\psi}(f) \) as follows:

\[
M(\psi_i^f) = S(\hat{\psi}(f_i)) \cdot \hat{\psi}(f_i) + (1 - S(\hat{\psi}(f_i))) \cdot \psi(f_i),
\]

where \( M(\psi_i^f) \) is the editing result of \( \psi_i^f \) which is represented in the converted feature space \( \psi(f) \). In (5), the samples with positive decision are manipulated based on \( \hat{\psi}(f) \) while those with negative decision are remained as original. Finally, \( M(\psi_i^f) \) is converted to the original domain as follows:

\[
e_i = M(\psi(f_i)),
\]

where \( e_i \) indicates final editing results in the original domain. From (6), it can be seen that the editing results in \( \psi(f) \) are simply remapped to the original domain since the edit classification is already performed in \( \psi(f) \). Accordingly, it is an efficient solution to handle a high resolution image since the computation demanding task in conventional methods, i.e., the edit propagation (classification) process, is performed invariant to the image size.

4. APPLICATIONS

In order to demonstrate the performance of the proposed method, we presented an image recoloring in our framework as shown in Algorithm 1. \( Cb-Cr \) components of an input image, quantized to 8-bit integers [0 255], are exploited as a feature space \( \psi(f) \) to achieve recoloring of pixels with similar appearance effectively. As shown in Algorithm 1, the image recoloring is directly performed in the feature space by substituting the features of the positive samples to the target \( Cb-Cr \) components \( \psi(f) \).

To validate the performance of the proposed model, the illustrative representation and computational cost were presented. The experiments were performed on a PC with Intel Core i7 Quad processor (3.40GHz) and 12 GB memory. The proposed method was compared with the all-pairs appearance propagation (APP) [8], the manifold-preserving edit propagation (MPP) [10], the sparse control model (SCM) [11], and the instant propagation (IP) [13]. The parameters in each method were fixed to all dataset for fair evaluation. The illustrative results and the computational costs are presented in Fig. 2 and Table 1, respectively. In Table 1, we only measured the elapsed time in edit propagation by changing image resolution to 4M, 12M, and 20M. Note that all the methods were implemented without any hardware acceleration.

As shown in Fig. 2(b), the APP effectively handles sparse initial edits with long-range propagation since it captures all-pairwise for the edit propagation. However, the computation time is slow although the numerical approximation (Nyström extension) is applied as shown in Table 1. The MPP effectively encodes manifold structure in the edit propagation, which gives competitive performance as shown in Fig. 2(c). However, some artifacts are still remained when manifold
distance between positive and negative edits is small (first and fourth column). The MPP also shows high computational cost due to the additional computation in manifold embedding. In Fig. 2(d), the SCM shows promising results, but color blending is shown when color between positive and negative edits are similar (second and third column). It is also having high computational cost due to the iterative optimization. Contrary to the previous methods [8, 10, 11], the IP provides the results with low complexity as shown in Table 1, while the qualitative performance is degraded (Fig. 2(e)). It shows that the interpolation scheme extremely reduces the computation time, but shows the limitation to cover all the pixels. In the proposed method, both qualitative performance and computational cost are competitive to the other state-of-the-art methods. In addition, as shown in Table 1, the computational cost of the proposed method is invariant to the image size.

5. CONCLUSION

We have presented the edit propagation which formulates the problem via machine learning. SVM has employed to generate the classifier by training initial edits in the feature space, and rest of the features are classified to determine the editing region. The experimental results showed that the proposed method effectively applied to image recoloring compared to state-of-the-art methods. For further work, we will extend our approach to solve video editing problem via machine learning.
6. REFERENCES


