A Background Removal Algorithm using the Variable Order $n \times m$ Dimensional Vector

Jeong-In Park, Jin-Tak Choi

1, 2 Dept. of computer science Eng., Incheon National University, Korea.

jipark@atec.kr, choi@incheon.ac.kr

Abstract. The code book algorithm is known to use the color similarity such that it removes by clustering like colors, however, it is disadvantageous that larger the code size or the objective image, slower the removal process. To improve the situation, this study suggests the variable order $n \times m$ dimensional vector. Since the vectors are applied to the reduced objective image to remove the background in this algorithm, the speed which has been the shortcoming of the code book is improved. Also, removal accuracy is superior to code book algorithm by 30% of accuracy result. It is confirmed that the removal of the background in complex color images where the background was difficult to distinguish is successful as a result of application of the proposed algorithm to the actual objective image.

Keywords: code book, variable order $n \times m$ dimensional vector, linear interpolation

1 Introduction

The new algorithm has similar background removal process as the existing code book algorithm in that it also uses the pixel mapping, but it's accuracy is further enhanced by using the variable order $n \times m$ dimensional vector to remove the background[1].

The code book algorithm slows down considerably through the LBG and K-means algorithm, however, the newly suggested algorithm improves the speed problem significantly by the application of the variable order $n \times m$ dimensional vector after the image division and reduction[2].

Generally, the background removal of gray images such as X-rays or CT images is quite successful with the code book. But if the code book is used on the complex color images or images with ambiguous background distinction, only the images that are similar with the learned image will be partially removed.

On the other hand, since the extent of background removal is decided when the values of $n$ and $m$ are established in the beginning of the new algorithm, the removal in dimensional layer is possible (not limited removal from the specified area) so that the dimensional layer background removal is easily done even on the images where the background is difficult to be distinguished[3].
2 Derivation of improvements in Code Book algorithm

The our new algorithm is proposed to resolve the speed problem in Code Book and to effectively remove the background. The logic is similar to each other in that it makes and uses the representative n×m dimensional vector to compare with each cell of the object image as the Code Book algorithm. However, the proposed algorithm, in terms of speed, improved significantly compared to the Code Book algorithm since it applies the n x m vector to the reduced or compressed image that is filled with the average values between adjacent pixels. We were able to get 30% improved results through the actual experiments[4].

Furthermore, if you want to modify the Code Book after it is already made, it needs to derive the Code Book from the basis image that has a different color similarity all over again. The proposed algorithm on the other hand, can be applied to the various order of vectors to the object image since the variable order n×m dimensional vector can always be created with simple input of the value of n and m dynamically. Table1 below shows the organized results derived with such improvements.

Table 1. Comparison of the proposed algorithm with Code Book

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Feature Comparison</th>
</tr>
</thead>
</table>
| Code Book[5]                     | - After dividing code book image with 4×4 square, creating first 16 pixels by calculating each mean of each 1~16th pixels.  
|                                  | - Then, creating 16 pixels by ±5% of each pixel.  
|                                  | - Removing pixels by adapting all pixels of basis image at every pixel in all completed code book list. |
| variable dimensional vector      | - After replaced all pixel value of origin image to image with 3×3 each 3 pixel mean, dividing it into vectorDegree×vectorTrimPer square.  
|                                  | - Reduction that image by calculating minimum pixel.  
|                                  | - Adapting all pixel of variable dimensional vector to every pixel of reduced image. |

3 The proposed algorithm

Once the variable order n×m dimensional vector is generated, it is used to apply to the real object image to remove the background. It operates with the following algorithm and an operating mechanism of the proposed algorithm is as follows:
Step 1. Define vectorDegree and vectorTrimPer according to the values of n and m input.
Step 2. Process the value of n entered to \(((n / \text{vectorDegree}) \times 2) + 1\), and get the \(n \times m\) dimensional vectors (width in terms of shapes) and implement the \(n \times m\) dimensional vector.
Step 3. Invert the sign of each channel values of RGB.
Step 4. Compute an average of 3 nearby pixels horizontally and vertically in the original image and save.
Step 5. Divide the results obtained from Step 4 into the areas of a rectangle as the same width and height of vectorDegree, and save the smallest image to make the reduced image.
Step 6. Apply the \(n \times m\) dimensional vector made in Step 2 to the reduced image of Step 5. (Add the value of vector with the biggest value difference in pixels between the reduced image and the value of vector, and save the smallest value.)
Step 7. Fill the pixels using linear interpolation while enlarging the reduced image that the \(n \times m\) dimensional vector applied to its original size.
Step 8. Invert the sign of each channel values of RGB once again.

Fig. 1. Building process of the \(n \times m\) dimensional vector

4 Test and evaluation of the proposed algorithm

4.1 Adaptation to real image

The proposed algorithm does not remove the actual background; rather it re-generates the filtered replica after replicating the background. This may look like the background is actually removed after it is applied to the image, but it is reconfigured with white color lines which are smoothly processed while retaining the background. In addition, it reduces noise and gives the effect of smooth texture. Figure 2 (a)(c)(d) below shows the result of applying the proposed algorithm to the complex color images.
Figure 2 (b) has many objects to be extracted such as the noise, but the desired object is rather well separated for its background and foreground are relatively clear. And Figure 2 (a)(c)(d), compare to (b), is the result of application to a very complex color image, but its background and foreground were well separated as well.

5 Conclusion

In this chapter, the new algorithm was proposed to remove the background of an image. While Code Book is slow in speed and removes the color based on the learned image as the basis image, the proposed algorithm eliminates the need for a separate basis image and maps the n×m dimensional vector from the n and m values onto the object images to make it multi-dimensional.

The proposed algorithm was able to improve the speed since it applies the vector after reducing the object image. The background removal is possible since it uses the variable n and m values applied to its own image.

The accuracy measurement result after removing the image shows about 30% improvement compared to Code Book algorithm and the improvement was also confirmed in removing the background in complex color images with ambiguous distinction, in addition to gray scale.

References