Performance Evaluation of Preprocessing in Color Image Segmentation

Henryk Palus
Silesian University of Technology, Gliwice, POLAND

In this article we address the problem of performance of preprocessing before color image segmentation. The main goal of preprocessing is noise removal. Our interests are limited to nonlinear color filters working in the spatial domain. Most often comparing such filters is based on calculation of different quality factors (e.g. PSNR, NCD etc.). The main idea of this article is to use an evaluation function, coming from research on segmentation, to evaluate the performance of preprocessing. The experiments were realized using both original and noisy images corrupted by Gaussian and impulsive noise.

Journal of Imaging Science and Technology 49: 583–587 (2005)

Introduction

Because image acquisition devices (scanners, cameras etc.) are sources of noise, it is important to have a preprocessing stage in the chain of color image processing. In the case of noisy images different filters can be applied as preprocessing algorithms for color image segmentation. The performance of the filters can be evaluated visually or quantitatively using, e.g., peak signal-to-noise ratio PSNR or NCD (normalized color difference) values. The other possibility is to evaluate segmented images. The application of such preprocessing algorithms may significantly improve segmentation results and therefore their effects on the segmentation results are studied. We use five types of nonlinear color filters, two different segmentation techniques and one postprocessing algorithm. The performance of preprocessing is checked by using the group of ten popular color images in their original and noisy versions. A special quality function is applied for evaluating the performance of preprocessing.

This article is organized as follows. In the next section, a short overview of filters used in the study as preprocessing algorithms is presented. The following section describes two color image segmentation techniques (mean shift and region growing without seeds). Then, a technique optionally used for postprocessing is presented. The next section treats quantitative evaluation of image segmentation results. The last section contains the results of experiments. A short discussion is given at the end of article.

Original manuscript received July 26, 2004

Corresponding Author: H. Palus, hpalus@polsl.pl

Preprocessing Algorithms

Different color filters have been developed to suppress noise and preserve edges. The filters achieve a goal that the image is more homogeneous whereas the edges are still preserved. It is very important to preserve the edges and corners. Unfortunately, most commonly used linear smoothing filters smooth images but at the same time blur the edges. The best performance in preprocessing is generally obtained with nonlinear filters. Nonlinear filters preserve edges and details and remove Gaussian and impulsive noise. For our research we have chosen following five nonlinear color filters:

1. SNN (Symmetric Nearest Neighbor Filter) described by Pietikainen and Harwood,[1]
2. KuNa (Kuwahara-Nagao Filter) proposed in the seventies,[2,3]
3. PGF (Peer Group Filtering) presented by Deng et al.,[4]
4. DPA (Digital Paths Approach) suggested by Szczepanski, et al.[5]
5. VMF (Vector Median Filter) developed by Astola et al.[6]

We have used most typical versions of filters working with 3×3 masks. We have limited the number of iterations for each filter to only one.

In the case of SNN filter the neighbors of the central pixel in a window are considered as four pairs of symmetric pixels: N-S, W-E, NW-SE and NE-SW. The filter's mask is shown in Fig. 1. For each pair the pixel closest in color to the central pixel is selected. The colors of these four selected pixels are averaged and the mean color value is a new color for central pixel.

The construction of Kuwahara–Nagao filter is similar to that of SNN. The 3×3 mask is splitted into four 2×2 slightly overlapping windows with the mask's central pixel as a common part (Fig. 2). For each window the sum of variances of color components is calculated. The mean color value of the window with minimal sum of variances (maximal homogeneous region) is used as the output value of the central pixel. The Ku-Na filter needs more computation time than the SNN filter.

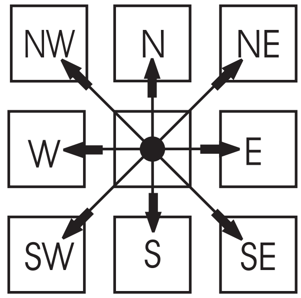


Figure 1. Mask for the SNN filter

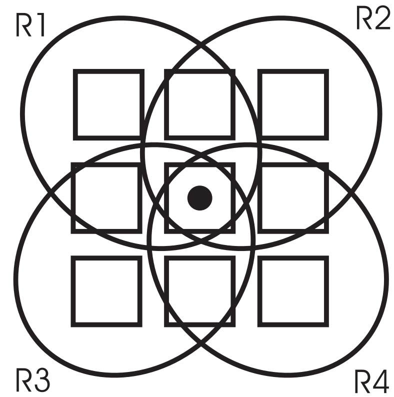


Figure 2. Mask for the Kuwahara-Nagao filter

For each pixel the PGF filter finds a group of neighbors (*peer group* members) based on its color similarity and replaces this pixel with its weighted mean color value. The differences in color between pixels in the mask are used for identifying noisy pixels. These pixels do not take part in determination of the peer group size. The main advantage of PGF filter is that it determines the corrupted pixels and does not modify non-corrupted pixels. Besides the size of filtering window (3×3 or 5×5), the method needs additionally two other parameters: δ^2 used for definition of Gaussian weights in the averaging process, and α used during removal of impulsive noise pixels. Averaging over the peer group instead of the entire mask allows avoiding edge blurring.

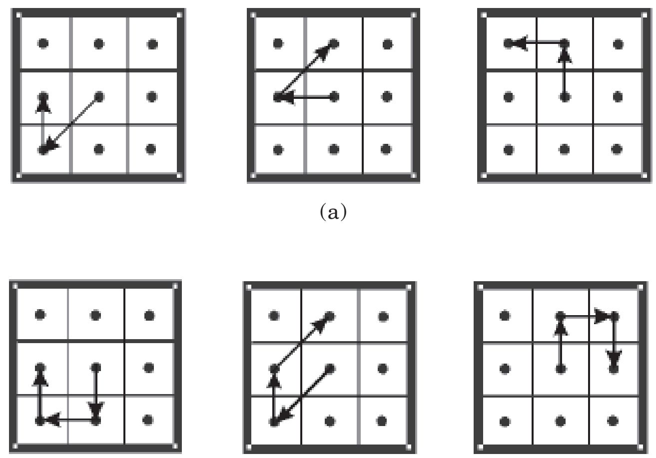
The DPA filter is based on a general concept of digital paths in filtration window and the connection costs defined over digital paths. A digital path, a sequence of neighboring pixels, models a random walk of virtual particle on the two-dimensional image lattice (Fig. 3). The connection cost is a measure of dissimilarity between color image pixels which form the path. This measure is based on the exponential function with a smoothing parameter. The new color for the central pixel of the window is calculated as weighted arithmetical mean of colors in the window with exclusion of the central pixel. The weights are defined using all digital paths going from the central pixel and crossing its nearest pixels. The filter is robust to improper values of its parameters. For further details of DPA filter and formulae see Szczepanski et al.[5]

The VMF filter[6] is probably the most popular filtration technique from those considered in this article. It is an extension of the scalar median filter which orders the *RGB* color vectors. Each pixel from the mask (3×3) defines one *RGB* vector. For each vector the distances to all of eight vectors from the mask are calculated and the sum of distances is computed as a cumulative distance. Finally, the output of the VMF filter is the *RGB* color vector with a minimal cumulative distance. Euclidean metric can be used for distance calculations.

Most of these edge-preserving smoothing filters lead, in opinions presented in the literature, to good segmentation results.

Segmentation Techniques

The goal of color image segmentation is to identify homogeneous regions in color image that represent objects or meaningful parts of objects present in a scene. Techniques for image segmentation can be most often classified into following categories: pixel-based techniques, region-based techniques, edge-based techniques, physics-based techniques and hybrid techniques.[7,8] We have chosen two segmentation techniques: one pixel-based and one region-based technique. The pixel-based technique is a clustering technique based on the mean shift idea, which has been first used for clustering tasks by Cheng.[9] Colors of the image pixels create a multivariate probability distribution. The local modes of its probability density function correspond to cluster centers. Let $\{X_i\}_{i=1..n}$ is n arbitrary set of n points in a d-dimensional Euclidean space R^d. There is possible to define the density estimate in x point using a kernel $K(x)$ and window radius h:



(a)

(b)

Figure 3. Digital paths connecting two pixels in the window 3×3: (a) length of 2, (b) length of 3.

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right). \qquad (1)$$

If the kernel $K(x)$ is differentiable, then the estimate of density gradient is equal to the gradient of the kernel density estimate:

$$\hat{\nabla} f(x) \equiv \nabla \hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} \nabla K\left(\frac{x - X_i}{h}\right) \qquad (2)$$

Using the Epanechnikov kernel we obtain the final formula for the density gradient:

$$\nabla\hat{f}(x) = \hat{f}(x)\,\frac{d+2}{h^2}\,M_h(x) \tag{3}$$

where

$$M_h(x) = \frac{1}{n_x}\sum_{X_i \in S_h(x)} (X_i - x) \tag{4}$$

M_h is called the sample mean shift, $S_h(x)$ is a sphere with radius h centered on x and containing n_x points. The main idea of the mean shift procedure is to calculate the mean shift vector and shift iteratively a fixed size window (a sphere of defined radius) to the mean of the points within it. It estimates the gradient of density function. The center of the sphere is then placed at this mean, and the algorithm is iterated until convergence. The shifts are in the direction of a maximum of density function and they are large in low density regions and small near local maxima. The procedure is guaranteed to converge and does not need any knowledge about the number or the shape of clusters. The radius of sphere is used as the sole parameter of procedure.

The second technique, proposed by the present author, is based on the concept of region growing without seeds as needed to start the segmentation process.[10] At the beginning of the algorithm each pixel has its own label (one-pixel regions). The concept of 4-connectedness is used for its computational simplicity. For the region growing process the centroid linkage strategy is used. This strategy includes a pixel in the region if it is 4-connected to this region and has a color value in the specified range from the mean color of an already constructed region. After inclusion of a pixel the region's mean color is updated. For this updating recurrent formulae are used. Two simple raster scans of the color pixels are employed: from left to right and from top to bottom. The version of the algorithm used here, works in the *RGB* color space. The segmentation results are strongly determined by a tuning parameter: threshold d which limits the value of the homogeneity criterion. The homogeneity criterion is represented by the Euclidean distance between *RGB* components of the color of the merged pixel and *RGB* components of the mean color of a growing region. The practical result of the proposed technique is a set of regions described by mean colors, sizes and lists of pixels contained in these regions. This segmentation technique is relatively fast and effective: a PC with a 3.2 GHz Intel processor segments a color image (512×512 pixels) in 0.1 s.

Postprocessing Algorithm

The segmented image can be further postprocessed, e.g., by removing small regions that are usually not significant for the further stages of image processing.[11] Postprocessing needs an additional pass from the top left corner of the image to the bottom right corner, whose aim is to remove the regions which consist of a number of pixels smaller than a certain threshold. During this merging process each region with a number of pixels below a specified threshold A is merged into a region with a larger area and the nearest in the sense of the color distance. After the merging, a new mean color of the region is calculated, and the labels of pixels belonging to a region are modified. The pre-selected size of the removed regions A functions as a control parameter.

Performance Evaluation of Preprocessing

The simple approach to measure the quality of an image segmentation is a visual inspection. Some researchers claim that the segmentation performance must be evaluated in the context of a defined task. However, in the literature a few methods of quantitative evaluation of image segmentation results exist, which, in the absence of general image segmentation theory, are very useful in practical applications. Zhang[12] has presented 10 recent evaluation methods for image segmentation from two categories: "goodness" methods and "discrepancy" methods. One of the first category methods has been proposed by Borsotti et al.[13] They proposed and tested following quality function $Q(I)$:

$$Q(I) = \frac{1}{10000\,(N \cdot M)}\sqrt{R}\sum_{i=1}^{R}\left[\frac{e_i^2}{1+\log A_i} + \left(\frac{R(A_i)}{A_i}\right)^2\right] \tag{5}$$

where: I is the segmented image, $N \times M$, size of the image, R, the number of regions in the segmented image, A_i, the area of pixels of the ith region, and e_i the color error of region I and $R(A_i)$ is the number of regions with the area equal to A_i. We have chosen this function because it has been constructed on the basis of the qualitative criteria for good image segmentation.

The pre-factor in Eq. (5) is a normalization factor, the next term penalizes results with too many regions (oversegmentation), while the term in the brackets penalizes the results with inhomogeneous regions. The right element of the term in the brackets is scaled by the area factor because the color error is higher for large regions. The color error in *RGB* space is calculated as the sum of the Euclidean distances between color components of pixels of the region and components of an average color which is an attribute of this region in the segmented image. More detailed information on the idea behind building this function may be found in the article by Borsotti et al.[13]

The evaluation function $Q(I)$ matches well with the visual judgement. The idea of using this function can be formulated as follows: the lower the value of $Q(I)$, the better the segmentation result. In the paper by Climent et al.,[14] $Q(I)$ was included in the segmentation algorithm based on graph minimization, i.e., the authors used the evaluation function for segmenting rather than for evaluating the segmentation results. Their algorithm, which does not require control parameters, finds the segmented image adequate to a minimal value of $Q(I)$. By using typical test images the authors showed that this segmentation algorithm had generated segmented images, which had a considerably lower value of the function $Q(I)$ than the other algorithms evaluated in the Borsotti's study.[13] We have used $Q(I)$ for performing experimental investigations which are described below.

Results of Experiments

Figure 4 shows the reduced versions of 10 popular test images used during this work. In order to evaluate the performance of the color filters, an experiment on original (relatively noise-free) images has been carried out. For each segmented image, the $Q(I)$ value was calculated and averaged for the whole experiment. Average $Q(I)$ values for each segmentation technique were normalized for comparison purposes. It means that the $Q(I)$ value for the segmented and evaluated image (without filtration) has been assumed to be 100%. Results indi-

Figure 4. Popular test images used in the experiments: Airplane, Baboon, Girl, Lena, Peppers, Sailboats, Lighthouse, Motocross, Parrots and Hats.

cate that for the region-based segmentation, the preprocessing has very limited impact on the segmentation results. In the best case, the DPA filter reduces the *Q(I)* value to 70% (Fig. 5). On the other hand, in the case of the mean shift segmentation, the sense of preprocessing is more important. In this case the best filters (Kuwahara-Nagao, DPA and VMF) enable reduction to 15%.

In the next experiment noisy color images were used. Original images were corrupted with mixed Gaussian additive ($\sigma = 7.5$) and impulsive noise ($p = 0.1$ and $p1 = p2 = p3 = 0.02$). Details of the application of this three-variate impulsive noise model in color image processing are presented in the color image processing handbook.[15] The results from Fig.6 show that in this case all filters are very useful and can be used to enhance the segmentation results for both tested segmentation techniques. The biggest enhancement is noticeable after application of the DPA and VMF filters.

In the majority of cases investigated, the absolute values of *Q(I)* were smaller for the region-based segmentation. It means that this technique is better than the mean shift technique. The noisy images after segmentation can be improved by using the postprocessing algorithm which has been described above.

Figure 7 presents adequate quality function values. The results have changed drastically: the absolute *Q(I)* values decrease as a consequence of postprocessing but the results, in relation to the case of segmentation without filtration, are not satisfactory. The filtration requires a computation time and in this case its

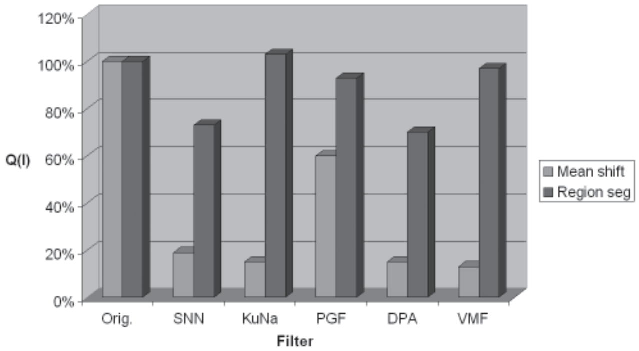


Figure 5. Average *Q(I)* function values for original images

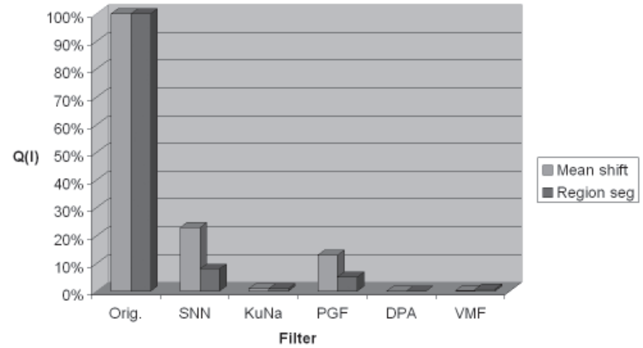


Figure 6. Average *Q(I)* function values for noisy images

effectiveness is not great. It means that if postprocessing is used, the preprocessing is useless.

During an additional experiment two 320 × 200 synthetic color images were used as ground truth. Excluding the background one image has contained five regions and the other one – three. The same mixed Gaussian and impulsive noises have been superimposed. In the case of unfiltered processing, the noise dominated in the segmented images: the typical number of regions for both segmentation techniques was equal to 1900. In the case of filtered processing, the smallest numbers of regions, close to six and four regions respectively, have been obtained with the VMF and DPA filters. The worst filter (SNN) has increased the input value of the number of regions.

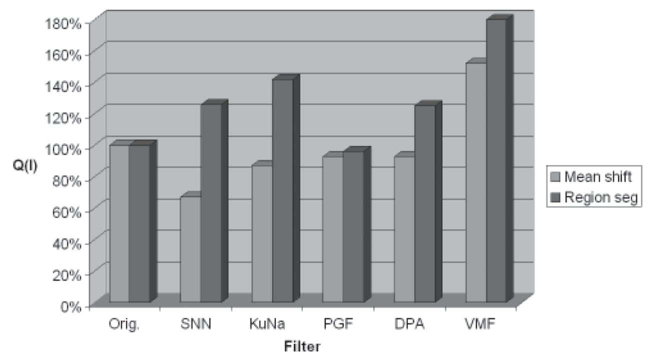


Figure 7. Average *Q(I)* function values for noisy images and postprocessing

Conclusion

We have shown that the quality function *Q(I)* can be used not only for comparing segmentation techniques but also for evaluating the performance of preprocessing in color image segmentation. The methodology for such evaluation is proposed. The performance of preprocessing depends on the method of color image segmentation: generally it is more effective for simpler pixel-based segmentation than region-based segmentation. Nonlinear filters, presented in the article, are very effective in the case of relative noisy images. The comparison of preprocessing filters shows that the DPA and VMF filters outperform other tested filters. When, however, postprocessing based on removing small regions is used, the application of preprocessing (filtering) can be useless.

References

1. M. Pietikainen and D. Harwood, "Segmentation of color images using edge-preserving filters", *Advances in Image Processing and Pattern Recognition*, V. Cappellini, R. Marconi, Eds.,(Elsevier, New York, 1986), pp.94-99.
2. M. Kuwahara, K. Hachimura, S. Eiho, and M. Kinoshita, "Processing of ri-angiocardiographic images", *Digital Processing of Biomedical Images*, K. Preston and M. Onoe, Eds., (Plenum Press, New York, 1976), pp.187- 202.
3. M. Nagao and T. Matsuyama, "Edge preserving smoothing", *Computer Graphics and Image Processing* **9,** 394 (1979).
4. Y. Deng, C. Kenney, M.S. Moore, and B.S. Manjunath, "Peer group filtering and perceptual color image quantization", *Proc. IEEE International Symposium on Circuits and Systems (ISCAS),* vol. 4, (IEEE Press, Piscataway, NY, 1999), pp. 21-24.
5. M. Szczepanski, B. Smolka, D. Slusarczyk, K.N. Plataniotis, and A.N. Venetsanopoulos, "Geodesic paths approach to color image enhancement", *Electronic Notes in Theoretical Computer Science*, **46,** (2001). Available at http://www.elsevier.nl/locate/entcs/volume46.html.
6. J. Astola, P. Haavisto and Y. Neuvo, "Vector median filters", *Proc. IEEE*, **78,** 678 (1990).
7. L. Lucchese, S.K. Mitra, "Color image segmentation: A state-of-the-art survey", *Proc. Indian National Science Academy (INSA-A),* **67**(2), 207–221 (2001).
8. H.D. Cheng, X.H. Jiang, Y. Sun, and J. Wang, "Color image segmentation: advances and prospects", *Pattern Recognition*, **34,** 2259 (2001).
9. Y. Cheng, "Mean shift, mode seeking and clustering", *IEEE Trans. Pattern Analysis and Machine Intelligence*, **17,** 790 (1995).
10. H. Palus, "Region-based color image segmentation: Control parameters and evaluation functions", *Proc. 1st European Conference on Color in Graphics, Imaging and Vision*, CGIV'2002, (IS&T, Springfield, VA, 2002), pp.259-262.
11. M. Sonka, V. Hlavac and R. Boyle, *Image Processing, Analysis, Analysis and and Machine Vision*, 2nd ed., (PWS Publishing, Pacific Grove, CA, 1998).
12. Y.J. Zhang, "A review of recent evaluation methods for image segmentation", *Proc. Sixth International Symposium on Signal Processing and its Applications (ISSPA2001),* vol. 1, (IEEE Press, Piscataway, NJ, 2001), pp.148-151.
13. M. Borsotti, P. Campadelli and R. Schettini, "Quantitative evaluation of color image segmentation results", *Pattern Recognition. Lett.* **19,** 741 (1998).
14. J. Climent, A. Grau, J. Verges, and A. Sanfeliu, "Image Segmentation Based on Graph Minimization", *Revista Electronica de Vision por Computador*, **4,** 2, (2000), http://revc.uab.es/revista/04/0402-art.pdf, in Spanish.
15. K.N. Plataniotis and A.N. Venetsanopoulos, *Color Image Processing and Applications*, (Springer Verlag, Berlin, 2000).