Fast Two-Step Histogram-Based Image Segmentation

Damir Krstinić[⋆], Ana Kuzmanić Skelin[‡], Ivan Slapničar[††]

Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture

University of Split

R. Boškovića b.b, 21000 Split, Croatia

[⋆]damir.krstinic@fesb.hr, Department of Mathematics,

[‡]akuzmani@fesb.hr, Laboratory for Biomechanics and Automatic Control Systems,

[††]ivan.slapnicar@fesb.hr, Department of Mathematics

Abstract

We propose a novel image segmentation technique based on the non-parametric clustering procedure in the discretized color space. The discrete probability density function is estimated in two steps. Multidimensional color histogram is created, which is afterwards used to acquire final density estimate using the variable kernel density estimation technique. Segmentation is obtained by mapping revealed range domain clusters to the spatial image domain. The proposed method is highly efficient, running in time linear to the number of the image pixels with low constant factors. The output of the algorithm can be accommodated for a particular application to simplify the integration with other image processing techniques. Quantitative evaluation on a standard test dataset proves that the quality of the segmentations provided by the proposed method is comparable to the quality of the segmentations generated by other widely adopted low-level segmentation techniques, while running times are several times faster.

1 Introduction

Segmentation of a complex scene into perceptually logical homogeneous regions is a fundamental step in the process of understanding visual information. This low-level task is often the first step in complex vision systems

where the accuracy of the final scene interpretation strongly depends on the quality of provided segmentation. Partitioning an image into non-overlapping regions can be based on different homogeneity criteria, such as gray level, color or texture. Regardless of the attributes used, for an image segmentation algorithm to be broadly useful, the method should be non-parametric in the sense that it should not rely upon a priori knowledge, like number of segments or implicitly assumed shape [1]. The objective of a low-level segmentation should not aim to produce a final segmentation. Instead, it should use low-level attributes to suggest candidate partitions, while higher-level knowledge can be used to select among these for the further processing [2]. Externally, the algorithm should be operated with a small set of intuitively clear controlling parameters. These should be used to control low-level processing, based on task specific interpretations derived at higher levels [3]. Real time processing enforces constraints for the segmentation algorithm, making a computational complexity a central issue. The efficiency of the existing algorithms is still far from other low-level procedures such as edge detection [4]. Our motivation was to provide a computationally efficient low-level segmentation tool that can produce high quality segmentations.

The proposed segmentation algorithm relies on clustering of pixels in the feature space spanned by color coordinates. Clusters are represented by hills in the multidimensional color histogram estimated in two steps. Initial density is estimated by counting pixels which populate each cell in the discretized color space. Afterwards variable kernel density estimation [5] is applied to compensate the effect of scale variations in the input data. Revealed clusters of pixels are mapped to image segments, spatially continuous regions in the image.

The rest of the paper is organized as follows. The overview of the related work is given in Section 2. In Section 3 important concepts of density-based clustering and kernel density estimation are reviewed. In Section 4 we propose the general grid-based clustering technique. The image segmentation algorithm based on the technique introduced in Section 4 is proposed in Section 5, followed by evaluation results in Section 6. Conclusion is given in Section 7.

2 Related work

Although the research in this area has led to many different image segmentation techniques presented in the literature [6], no single algorithm can be considered good for all applications and all images [7]. Segmentation methods used for color images can be divided into two main categories: feature-space based techniques (clustering methods [8, 9, 10] and histogram multi-thresholding [7, 11, 12]) and image-domain based tech-

niques. The latter are further divided into pixel-similarity based algorithms [13, 14, 15] and pixel-difference based algorithms [2, 4, 16]. Image-domain based techniques exploit the pixel context interaction which increase computational complexity. The most common feature-space techniques gained their popularity through the adaptive k-means algorithm [17, 18]. Most of these methods are parameterized with number of expected clusters. Number of clusters is determined automatically in the Mean Shift algorithm [1] using nonparametric density estimation and gradient guided hill-climbing procedure.

Histogram-based approaches relay on the estimation of the discrete density in the designated color space where the clusters are represented by hills in the histogram. These methods are not parameterized with the a priori knowledge about number of clusters or their shape. The result depends on the discretization resolution of the feature space and the quality of provided segmentations can degrade if scale variations are present in the input data. To overcome this drawback, the Hill-manipulation algorithm [19] adaptively refines histogram resolution in the regions of higher density. Beside abovementioned, methods exist that combine information contained in different attributes or apply postprocessing to improve the quality of the segmentation acquired by color quantization [20, 21].

3 Density based clustering

Broad category of feature space analysis techniques rely on the estimation of the probability density function (PDF) of the data set. Estimated density reveals patterns in data distribution where dense regions correspond to clusters of the data separated by regions of low density or noise. Clustering is based on determining local maxima (modes) of PDF and associated basins of attraction. The main concept is based on the density attractor notion [22].

Definition 1 (Density attractor) *A point $x^* \in \mathbb{R}^d$ is a density attractor of the probability density function $f : \mathbb{R}^d \to \mathbb{R}$ if x^* is a local maximum of f, that is, if there exists $\epsilon > 0$ such that*

$$0 < d(x, x^*) < \epsilon \Rightarrow f(x) < f(x^*), \tag{1}$$

where $d(x, x^*)$ is the distance in $\mathbb{R}^d$.

Each density attractor of PDF delineates associated basin of attraction, a set of points $x \in \mathbb{R}^d$ for which hill-climbing procedure started at x converges to x^*. Hill-climbing procedure can be guided by a gradient of PDF [23, 24] or step-wise [22].

Definition 2 (Strong density attractor) *Strong density attractor x^* is density attractor satisfying*

$$f(x^*) \geq \xi, \tag{2}$$

where ξ is a noise level.

Definition 2 defines significant density attractors with regard to the uniformly distributed noise and non-specific data samples, called outliers. Due to noise and outliers, PDF can exhibit local variations with the density low compared to the modes corresponding to significant clusters. Implicitly defined number of clusters corresponds to the number of modes above noise level.

Kernel density estimation [25] is a PDF estimation method based on the concept that the density function at a continuity point can be estimated using the sample observation that falls within a region around that point.

Let $D = \{x_1, ..., x_N\}$ be a set of N data samples in d-dimensional vector space $x_i \in \mathbb{R}^d$, and $K : \mathbb{R}^d \to \mathbb{R}, K(x) \geq 0$, a radially symmetric kernel satisfying

$$\int_{\mathbb{R}^d} K(x)dx = 1 \tag{3}$$

The inherent distribution $f(x)$ of the data set D at $x \in \mathbb{R}^d$ can be estimated as the average of the identically scaled kernels centered at each sample:

$$f^D(x) = \frac{1}{Nh^d} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right), \tag{4}$$

where bandwidth h is the scaling factor.

The shape of the PDF estimate is strongly influenced by the bandwidth h, which defines the scale of observation. Larger values of h result in smoother density estimate, while for smaller values the contribution of each sample to overall density has the emphasized local character, resulting in density estimate revealing details on a finer scale. The fixed bandwidth h, constant across $x \in \mathbb{R}^d$, affects the estimation performance when the data exhibit local scale variations. Frequently more smoothing is needed in the tails of the distribution, while less smoothing is needed in the dense regions.

Underlying data distribution can be more accurately described using the variable bandwidth kernel. By selecting a different bandwidth $h = h(x_i)$ for each sample x_i, the sample point density estimator [5, 26] is defined:

$$f_s^D(x) = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h(x_i)^d} K\left(\frac{x - x_i}{h(x_i)}\right) \tag{5}$$

Improvements can be obtained by setting the bandwidth $h(x_i)$ to be reciprocal to the square root [27] of the density $f^D(x_i)$

$$h(x_i) = h_0 \sqrt{\frac{\lambda}{f^D(x_i)}}, \tag{6}$$

where h_0 is the fixed bandwidth and λ is a proportionality constant. Since $f^D(x)$ is unknown density to be estimated, the practical approach [26] is to use an initial pilot estimate $\tilde{f}^D(x)$ and to take proportionality constant λ as the geometric mean of all $\left\{\tilde{f}^D(x_i)\right\}_{i=1,\ldots,N}$.

4 Grid-based clustering with adaptive kernel density estimation

Grid-based clustering methods discretize the data domain into a finite number of cells, forming a multidimensional grid structure on which the operations are performed. These techniques are typically fast since they depend on the number of cells in each dimension of the quantized space rather than the number of actual data objects [28, p. 243].

We propose a new two-step technique for the discrete density estimation. Firstly, initial histogram is created by counting input data samples which populate each cell of the discretized domain. Secondly, the sample point density estimator (5) is applied to gain final density estimate, where initial density estimate is used to compute variable bandwidth (6). We will now proceed with the introduction of the general grid-based clustering technique.

4.1 Discrete density estimation

The process of the estimation of the discrete density of the data set $D = \{x_1, ..., x_N\} \in \mathbb{R}^d$ starts with domain discretization. Bounding d-dimensional hyperrectangle is determined and the domain is partitioned into d-dimensional hypercubes with side length σ. The density at some point x is strongly influenced by a limited set of samples near that point, thus density can be approximated with local density function $\hat{f}^D(x)$. We adopt the sample point density estimation technique (5) where each sample is assigned the variable bandwidth (6), resulting in the adaptive local density function:

$$\hat{f}^D(x) = \frac{1}{N} \sum_{x_l \in \mathcal{N}_l(x)} \frac{1}{h(x_l)^d} K\left(\frac{x - x_l}{h(x_l)}\right) \tag{7}$$

The variable bandwidth is defined by:

$$h(x_l) = h_0 \sqrt{\frac{\lambda}{\tilde{f}^D(x_l)}}, \tag{8}$$

where $\tilde{f}^D$ is the initial density estimate obtained by counting data samples which populate each cell of the multidimensional histogram. Proportionality constant λ is set to the geometric mean of $\tilde{f}^D$ [26]. Adaptive neighborhood for each sample x_l is defined by:

$$\mathcal{N}_l(x) = \{x_l : d(x_l, x) \leq \tau(x_l)\sigma\} \tag{9}$$

where $\tau(x_l)$ is the variable proportionality factor. According to (6) we define $\tau(x_l)$ with:

$$\tau(x_l) = \tau_0 \sqrt{\frac{\lambda}{\tilde{f}^D(x_l)}}, \tag{10}$$

where τ_0 is the fixed proportionality factor. The cell width σ is computed from the fixed bandwidth h_0 by:

$$\sigma = \frac{h_0}{\rho} \tag{11}$$

where ρ is the discretization parameter. Setting the bandwidth to be multiple of the cell width intuitively prescribes a set of neighboring cells in which contribution of each sample should be accounted for.

The introduced PDF estimation technique requires two passes through the data to compute the initial and the final density estimate. Efficiency of this procedure can be increased such that it requires one pass while preserving the accuracy. Let

$$Z(x, x_j) = \frac{1}{h(x_j)^d} K\left(\frac{x - x_j}{h(x_j)}\right) \tag{12}$$

be the contribution of the sample x_j at point x. Inserting (12) into (7) yields

$$\hat{f}^D(x) = \frac{1}{N} \sum_{x_l \in \mathcal{N}_l(x)} Z(x, x_l) \tag{13}$$

Let c_u denote the spatial coordinates of the center of the u-th histogram cell. The contribution of all samples which populate the v-th cell to the density of the u-th cell is approximately

$$\hat{Z}(c_u, c_v) = \tilde{f}^D(c_v) \frac{1}{h(c_v)^d} K\left(\frac{c_u - c_v}{h(c_v)}\right) \tag{14}$$

The contribution is accounted for as if all samples were located in the center of the cell. Asset of the adaptively scaled kernel centered at the v-th cell is multiplied with the number of samples populating that cell. By inserting (14) into (13) we obtain:

$$\hat{f}^D(c_u) \approx \frac{1}{N} \sum_{c_l : d(c_u, c_l) \leq \tau(c_l)\sigma} \hat{Z}(c_u, c_l), \tag{15}$$

where $\hat{f}^D(c_u)$ is estimated discrete density of the u-th cell. The approximation in (14) computes the contribution of all samples which populate a single cell in one pass.

The basic feature of all grid-based clustering techniques is a low computational complexity. However, discretization unavoidably introduces error and lowers the performance in disclosing clusters of different scales which are often present in the real data. Some techniques attempt to solve this problem by allowing arbitrary mesh refinement in densely populated areas [29, 30]. The resulting grid structure has complex neighboring relations between adjacent cells, which introduces computational overhead in the clustering procedure. Moreover, an additional pass through the data is often required to construct the adaptive grid.

The proposed technique is based on a single resolution grid, thus a simple hill-climbing procedure can efficiently detect modes and associated basins of attraction. The estimator is self-tuned to the local scale variations in the data set by using the variable bandwidth and adaptive neighbourhood. The density estimation algorithm requires one pass through the input data and an additional pass through populated cells, yielding computational complexity $O(N + rM)$, where N is cardinality of the data set, M is the number of populated cells and constant r, proportional to ρ^d, is the average number of neighboring cells in which kernel contribution is accounted for.

4.2 Clustering and noise level

Clustering is based on determining strong density attractors and the associated basins of attraction of the estimated discrete density. Subset of input samples pertaining to a basin of attraction of a strong attractor is assigned a unique cluster label. Samples pertaining to basins of attraction of attractors with the density below noise level ξ are labeled as noise. Step wise hill-climbing procedure started from populated cells (satisfying $\tilde{f}^D > 0$) has two stopping criteria:

1. Local maxima detected

   If the density $\hat{f}^D$ of the detected maxima is below noise level ξ, all cells on the path of the hill-climbing procedure are labeled as noise. Otherwise, a strong attractor is detected and a new label denoting new cluster is assigned to all cells on the path of the procedure.

2. Cell with already assigned label detected

   Continuation of the procedure from this point leads to an already detected maxima. The encountered label is assigned to all cells on the path of the hill-climbing procedure, regardless of representing noise or clustered data.

Clustering procedure stops when all populated cells are labeled. As prior knowledge would be required to anticipate absolute noise level, we define ξ relative to PDF by:

$$\xi = \varepsilon\Lambda, \tag{16}$$

where Λ is the geometric mean of $\hat{f}^D$ and ε is the algorithm parameter. Note that value Λ differs from the proportionality constant λ, defined as the geometric mean of the initial density estimate. The computational complexity of the hill-climbing procedure is $O(M)$.

5 Image segmentation algorithm

In this section image segmentation algorithm is introduced based on the clustering technique proposed in Section 4. The algorithm is based on the assumption that salient regions in an image are presented by dense clouds of points in the range domain, the feature space spanned by color coordinates. Revealed clusters of pixels are mapped to image segments, spatially continuous regions in the image.

Before describing the details of the algorithm, we comment the selection of the range domain. Since the use of the radially symmetric kernel relies on the Euclidean metric, the implemented color space should satisfy the assumption that the difference between two colors is proportional to the length of the straight line joining them. Although the proposed algorithm is not designed for any particular color space, the performance could degrade if the assumed Euclidean metric of the color space is not valid. Further discussion hereafter refers to the algorithm implementation in approximately uniform $L^*u^*v^*$ color space.

The Fast two-step histogram-based image segmentation algorithm (FHS) can be described in following steps:

1. Color space transformation

   Minimal bounding rectangle (MBR) of the input data $D = \{x_1, ..., x_N\}$ in the designated color space is determined, where $x_1, ..., x_N$ are image pixels represented by their color coordinates and N is the number of pixels. The computational complexity of this step is $O(N)$.

2. Domain discretization

   The range domain is partitioned into cells with side length $\sigma = h_0/\rho$, where fixed bandwidth h_0 and proportionality factor ρ are algorithm parameters.

3. Initial density estimation

   The discrete density $\tilde{f}^D$ is acquired by counting pixels which populate each histogram cell. Proportionality constant λ is computed as the geometric mean of $\tilde{f}^D$. The complexity of this step is $O(N)$.

4. Adaptive density estimation

   The discrete density $\hat{f}^D$ is estimated according to (15) using the Epanechnikov kernel (17):

$$K_E(x) = \begin{cases} \frac{1}{2}c_d^{-1}(d+2)(1-\|x\|^2) & \text{for } \|x\| \leq 1 \\ 0 & \text{for } \|x\| > 1 \end{cases}, \tag{17}$$

   where c_d is equal to the volume of the d-dimensional hypersphere.

   As the influence of the Epanechnikov kernel centered at x_i vanishes for $d(x, x_i) > h(x_i)$, the contribution of all samples populating the v-th cell with center c_v is accounted for in the set of cells with center c satisfying $d(c, c_v) \leq h(c_v)$. The variable bandwidth $h(x)$ is computed (8) from the initial estimate $\tilde{f}^D$. The computational complexity of this step is $O(rM)$, where M is the number of populated cells and $r \propto \rho^3$.

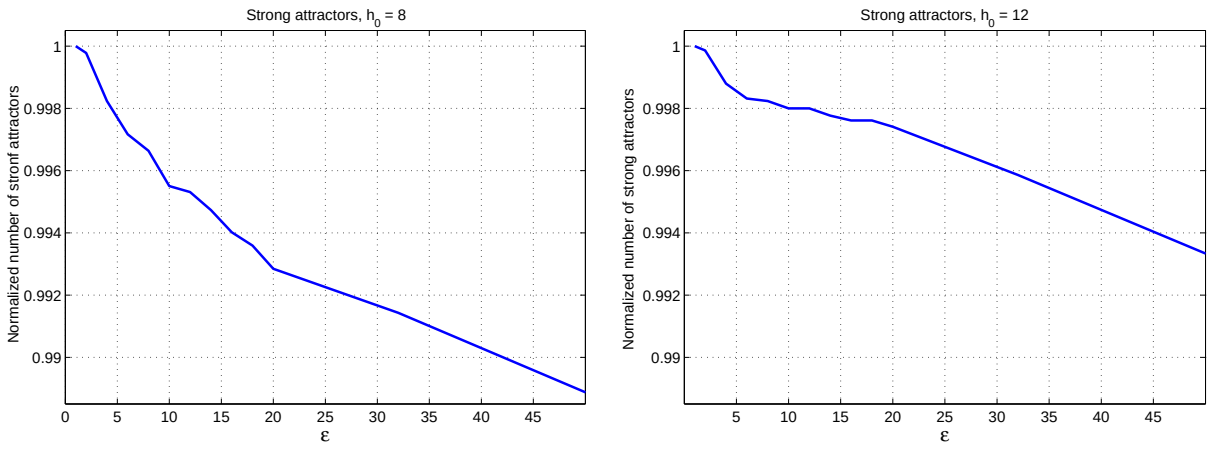
5. Range domain clustering

   The step wise hill-climbing procedure, outlined in Section 4.2, is started from cells satisfying $\tilde{f}^D > 0$. The hill-climbing procedure partitions the input data into clusters of pixels pertaining to basins of attraction determined by strong attractors satisfying $\hat{f}^D > \varepsilon\Lambda$ and noise, where Λ is the geometric mean of $\hat{f}^D$ and ε is the algorithm parameter. The computational complexity of this step is $O(M)$.

6. Mapping range domain clusters to the spatial image domain

   Spatial constraints of image regions are recovered using the region growing procedure. The growing process is continued as long as adjacent pixels belong to the same range domain cluster. The computational complexity of this step is $O(N)$.

The region growing mapping procedure is selected to evaluate the ability of the proposed histogram-based approach to produce quality segmentations. When implemented as a part of a complex vision system, the output of the algorithm can be accommodated for a particular application and the desired input for higher level processing modules. An alternative mapping procedure (step 6) could be based on the fuzzy region growing [31], with fuzzy segments which retain uncertainty for propagation to higher level processing, where more intelligent decisions can be made. Overall complexity of the FHS algorithm is $O(3N + (r+1)M)$.



(a) $h_0 = 8$ (b) $h_0 = 12$

Figure 1: Normalized number of strong attractors in the range domain with respect to the relative noise level ε. The results are given for two values of fixed bandwidth.

The algorithm is driven by three controlling parameters: fixed bandwidth h_0, discretization granularity ρ and relative noise level ε. Noise level, separating statistically important clusters from the noise and outliers, is given by (16) relatively to the geometric mean of the estimated density. In Fig. 1 number of clusters (strong attractors) in the range domain with respect to the relative noise level ε is shown, normalized by the number of clusters disclosed for $\varepsilon = 1$. The results, shown for two values of h_0, present the average number of clusters for all images in the publicly available Berkeley Segmentation Dataset [32] collection. In both cases number of clusters for $\varepsilon = 40$ is above 99% of number of clusters disclosed for $\varepsilon = 1$, suggesting low susceptibility of the algorithm with respect to this parameter. As shown in Fig. 1, the resulting number of clusters retains temporal stability starting at $\varepsilon = 9$, thus we fix relative noise level to this value.

Factor ρ determines discretization granularity of the color space. This parameter prescribes the complexity of the density estimation procedure $O(rM)$, as constant $r \propto \rho^3$ directly depends on the average number of neighboring cells in the discretized domain in which kernel contribution is accounted for. We fix the discretization granularity to $\rho = 3$. This value represents a compromise between the efficiency and the quality of the provided segmentations. For greater ρ space and time complexity of the algorithm rapidly grows, without prominent advances in the quality of generated segmentations.

After fixing parameters ε and ρ, the algorithm is driven with the single parameter h_0 which defines the scale of observation in the range domain. The additional parameter, often used in the segmentation algorithms, can be the minimum segment size.

6 Results

The evaluation of the FHS algorithm is divided in two subsections. The segmentations provided by the proposed method are subjectively evaluated and compared to the segmentations generated by other well known segmentation algorithms. Results are given in subsection 6.1.

While the subjective evaluation can demonstrate the ability of the algorithm to produce quality segmentations on some images and give coarse insight into the algorithm properties, if the algorithm is going to be used in an automated system, the objective numerical results on large dataset are desired [33]. The extensive experimental evaluation is performed and results are given in subsection 6.2.

6.1 Subjective evaluation

FHS technique was applied to the images from the Berkeley segmentation Dataset (BSD) [32] and other images collected by authors. The segmentations provided by the FHS algorithm are shown in Fig. 2. For larger value of the parameter h_0 less detailed segmentations are obtained. Variable bandwidth ensures that details in different scales are perceived for both values of h_0. Segmented areas in Fig. 2(g) and Fig. 2(h) are shown with segments smaller than the predefined size merged with the most similar neighbour.

The segmentations obtained by the proposed algorithm are compared to the segmentations generated by other well known segmentation algorithms in Fig. 3. The methods used in the comparison include two color-based techniques, namely the Mean Shift segmentation [1] and the Efficient Graph-Based segmentation [4]. Results are also compared with the JSEG algorithm [20] which incorporates texture information in segmentation process and the Gradient Network Method (GNM) [21] which generates final segmentation based on color gradients. The results demonstrate the ability of all algorithms to produce correct segmentations. The FHS algorithm can provide eligible segmentations for an ample parameter range, with details on finer scale perceived for smaller values of h_0. Smaller h_0 should be used if the FHS algorithm is going to be implemented as the color quantization preprocessing step of a segmentation technique which builds up final segmentation upon preprocessed over-segmented image [21].

6.2 Experimental evaluation

The FHS method is evaluated with respect to the quality of provided segmentation and efficiency of the algorithm. Our aim is to show that the proposed method is comparable in quality with other widely adopted

low-level segmentation techniques, while running times are several times faster. Proposed method is compared with two existing segmentation methods based on same attributes. In the Mean Shift (MS) segmentation algorithm [1], widely adopted in the vision community, MS procedure [23, 24] is implemented in the five-dimensional joint spatial-range domain. The multivariate kernel is defined as the product of two radially symmetric kernels controlled by the spatial bandwidth h_s and the range bandwidth h_r. MS algorithm is integrated into the Edge Detection and Image Segmentation (EDISON) system [34], with source code available at `http://www.caip.rutgers.edu/riul/`.

Efficient Graph-Based Image Segmentation algorithm [4] is created with the intention to provide computationally efficient approach which can be used in wide range of computer vision tasks, which is consistent with our motivation. The algorithm uses adaptive thresholding criterion based on the degree of variability in the neighboring regions of the image, and single controlling parameter κ defining the scale of observation. The algorithm runs in time nearly linear in the number of graph edges. Source code of the algorithm is available at `http://people.cs.uchicago.edu/~pff/segment/`.

6.2.1 Methodology

The evaluation database is the BSD dataset [32] which contains 300 images of complex natural scenes. For each image several ground truth (GT) segmentations are available. The BSD collection is divided into training and testing data sets. As non of the evaluated algorithms has learning strategy implemented, algorithms are evaluated on all images. For the assessment of the quality of provided segmentations we adopt the evaluation methodology proposed in the work of Pantofaru and Herbert [33]. The performance measure is the Normalized Probabilistic Rand index (NPR) [35], quantifying the degree of agreement of an algorithmically generated segmentation with a set of manually created GT segmentations. The NPR index is computed from the Probabilistic Rand index (PR) [36] by normalization with respect to its expected value. The expected values are modeled using all of the ground truth data, not just the data for the particular image, with no assumptions on the number and the size of regions in the segmentation. The normalization step facilitates meaningful comparison of scores between segmentations of different granularity of the same image, as well as across segmentations of different images.

6.2.2 Results

The evaluation results are summarized in Fig. 4 and Fig. 5. Each algorithm is evaluated for a reasonable set of parameters settled in [33]. In all plots, the label 'MSE' refers to the mean shift based algorithm as implemented in the EDISON system, with parameters values taken from $h_s \in [4, 16]$ and $h_r \in [4, 16]$. The label 'EGB' refers to the efficient graph-based segmentation algorithm with $\kappa \in [25, 300]$. The fixed bandwidth $h_0 \in [4, 16]$ for the 'FHS' algorithm is taken from the same set of values as the range bandwidth of the MSE algorithm.

The quality of the segmentations provided by each algorithm is presented in Fig. 4. Plot (a) shows the maximal NPR index of each image for each algorithm. Indices are plotted in increasing order for a particular algorithm, thus the same index may not represent the same image across the algorithms. Plot (b) represents maximal NPR index distribution. Best results are achieved by the MSE algorithm, followed by the FHS algorithm. Results achieved by the MSE and the FHS are roughly the same in the mean NPR index achieved for all parameter combinations, presented in plot (c). Better stability with respect to the parameter choice is confirmed in the distribution of the standard deviation shown in plot (d), with more left-biased standard deviation of the FHS when compared to other two algorithms. Stability with respect to the parameter choice is achieved by implementing variable bandwidth. Moreover, the MSE algorithm has two controlling parameters thus it is reasonable to expect higher reception to mutual influences of parameters. For the majority of images all algorithms have the ability to produce segmentations of the acceptable quality with above-zero maximal NPR index. The EGB algorithm demonstrates the lowest performance, with below-zero maximal NPR index for several images.

In order to estimate efficiency of the evaluated algorithms, series of experiments are conducted measuring algorithm running time for different parameters and image resolutions. All tests are carried out on the same architecture with equal run-time conditions ensured. Results are presented in Fig. 5. Plots (a), (b) and (c) show run-time of all three algorithms in the logarithmic scale. Average running time on all images in the BSD collection is shown, with axes on all three plots kept constant to facilitate comparison. Run-time of the FHS algorithm decreases for higher h_0 as this parameter determines discretization of the feature space. For $h_0 > 4$ the FHS algorithm runs faster than the EGB algorithm, which is not sensitive to the parameter κ. Complexity of the MSE algorithm results in longer running times than the other two algorithms.

Running times for images of different resolutions are shown in plot (d). Algorithms are applied on ten images of each resolution and average running times are presented in log-log scale. Controlling parameters for

each algorithm are taken to be values for which the particular algorithm scores the best performance on the BSD collection. Running times of all algorithms are nearly linear with the size of the input data. FHS is the fastest algorithm which is consistent with efficiency achieved on the BSD collection and theoretical complexity analysis.

7 Conclusion

The main contribution of this paper is the novel image segmentation method based on discretization of the color space domain and the adaptive kernel density estimation. For the image segmentation task, spatial constraints of coherent image objects are recovered using the region growing technique where the growing criterion is the affiliation of subsequent pixels to same range domain cluster. Important request for robustness and stability of the segmentation algorithm operating in the erratically changing environment is fulfilled by implementing the variable bandwidth kernel. Using two-step density estimation approach, the kernel bandwidth is made locally adaptive to the data without considerable increase in the computational complexity. Adaptivity to the noise in the input scene is attained using relative noise level. Experimental evaluation of the proposed algorithm has shown that the quality of the provided segmentation is comparable to those of the Mean Shift algorithm, widely adopted in the vision community, while running times of the FHS algorithm are several times faster.

The proposed segmentation technique presents the efficient and versatile low-level tool, which can be easily integrated with other image processing techniques. The FHS method is implemented in the intelligent Forest Fire (iForestFire) video surveillance system [37], within the module for automatic forest fire detection. Output is taken from several stages of the FHS algorithm and integrated with other low-level techniques like color-based pixel classification, edge-preserve filtering and shape analysis.

General grid-based clustering technique proposed in section 4 does not scale well with the dimension of the feature space, due to the time and space complexity of multidimensional mesh processing and the curse of dimensionality problem inherent to all density-based approaches. However, this technique should be applicable to the processing of other low-level image features, as well as to other problems when efficient processing of large number of low-dimensional data samples is required.

References

[1] Comaniciu, D., and Meer, P.: *"Mean Shift: A Robust Approach Toward Feature Space Analysis"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, Vol. 24(5), pp. 603-619

[2] Shi, J., and Malik, J.: *"Normalized Cuts and Image Segmentation"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, Vol. 22(8), pp. 888-905

[3] Meer, P.: *"Robust techniques for computer vision"*, in G. Medioni and S. B. Kang (Eds.) "Emerging Topics in Computer Vision", 2004, Prentice Hall, pp. 107-190

[4] Felzenszwalb, P. F., and Huttenlocher, D. P.: *"Efficient Graph-Based Image Segmentation"*, International Journal of Computer Vision, 2004, Vol. 59(2), pp. 167-181

[5] Scott, D. W., and Sain, S. R.: *"Multi-Dimensional Density Estimation"*, in C. R. Rao and E. J. Wegman (Eds.) "Data Mining and Computational Statistics", Elsevier, Vol. 23, 2004.

[6] Cheng, H. D., Jiang, X. H., Sun, Y., Wang, J.: *"Color image segmentation: advances and prospects"*, Pattern Recognition, 2001., Vol 34(12), pp. 2259-2281

[7] Mushrif, M. M., and Ray, A. K.: *"Color image segmentation: Rough set theoretic approach"*, Pattern Recognition Letters, 2008, Vol 29(4), pp. 483-493

[8] Chen, T. Q., and Lu, Y.: *"Color image segmentation: an innovative approach"*, Pattern Recognition, 2002, Vol. 35(2), pp. 395-405

[9] Wang, X. Z., Wang, Y. D., Wang, L. J.: *"Improving fuzzy c-means clustering based on feature-weight learning"*, Pattern Recognition Letters, 2004, Vol. 25(2), pp. 1123-1132

[10] Yu, Z., and Wong, H.: *"A Rule Based Technique for Extraction of Visual Attention Regions Based on Real-Time Clustering"*, IEEE Transactions on multimedia, 2007, Vol. 9(4), pp. 766-784

[11] Kurugollu, F., Sankur, B., Harmanci, A. E.: *"Color image segmentation using histogram multithresholding and fusion"*, Image and Vision Computing, 2001, Vol. 19(13), pp. 915-928

[12] Baradez, M. O., McGuckin, C. P., Forraz, N., Pettengell, R., Hoppe, A.: *"Robust and automated unimodal histogram thresholding and potential applications"*, Pattern Recognition, 2004, Vol. 37(6), pp. 1131-1148

[13] Fan, J., Zeng, G., Body, M., Hacid, M.: *"Seeded region growing: an extensive and comparative study"*, Pattern Recognition Letters, 2005, Vol. 26(8), pp. 1139-1156

[14] Espindola, G. M., Camara, G., Reis, I. A., Bins, L. S., Monteiro, A. M.: *"Parameter selection for region-growing image segmentation algorithms using spatial autocorrelation"*, International Journal of Remote Sensing, 2006, Vol. 27(14), pp. 3035-3040

[15] Zhang, Z. H., Chen, C. B., Sun, J., Chan, K. L.: *"EM algorithms for Gaussian mixtures with split-and-merge operation"*, Pattern Recognition, 2003, Vol. 36(9), pp. 1973-1983

[16] Rotem, O., Greenspan, H., Goldberger, J: *"Combining Region and Edge Cues for Image Segmentation in a Probabilistic Gaussian Mixture Framework"*, IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17-22 June 2007.

[17] Pappas, T. N.: *"An adaptive clustering algorithm for image segmentation"*, IEEE Transactions On Signal Processing, 1992, Vol. 40(4), pp. 901-914

[18] Chang, M. M., Sezan, I., Tekalp, M.: *"Adaptive Bayesian Segmentation of Color Images"*, Journal of Electronic Imaging, 1994, Vol. 3(4), pp. 404-414

[19] Zaher Al Aghbari, and Ruba Al-Haj, *"Hill-manipulation: An effective algorithm for color image segmentation"*, Image and Vision Computing, 2006., Vol 24(8), pp. 894-903

[20] Deng, Y., and Manjunath, B. S.: *"Unsupervised segmentation of color-texture regions in images and video"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001., Vol. 23(8), pp. 800-810

[21] Wangenheim, A. V., Bertoldi, R. F., Abdala, D. D., Richter, M. M.: *"Color image segmentation guided by a color gradient network"*, Pattern Recognition Letters, 2007., Vol 28(13), pp. 1795-1803

[22] Hinneburg, A., and Keim, D. A.: *"A General Approach to Clustering in Large Databases with Noise"*, An International Journal of Knowledge and Information Systems, 2003., Vol. 5(4), pp. 387-415

[23] Fukunaga, K., and Hostetler, L.: *"The estimation of the gradient of a density function, with applications in pattern recognition"*, IEEE Transactions on Information Theory, 1975, Vol. 21(1), pp. 32-40

[24] Cheng, Y.: *"Mean shift, mode seeking, and clustering"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, Vol. 17(8), pp. 790-799

[25] Fukunaga, K.: *"Statistical Pattern Recognition"*, in C. H. Chen, L. F. Pau, P. S. P. Wang (Eds.) "Handbook of Pattern Recognition and Computer Vision", World Scientific Publishing Co. Pte. Ltd., Singapore, pp. 33-60, 1993.

[26] Comaniciu, D., Ramesh, V., Meer, P.: *"The Variable Bandwidth Mean Shift and Data-Driven Scale Selection"*, Proc. of the 8th IEEE International Conference on Computer Vision, Vancouver, Canada, 2001, Vol. 1, pp. 438-445

[27] Abramson, I. S.: *"On Bandwidth Variation in Kernel Estimates-A Square Root Law"*, The Annals of Statistics, 1982, Vol. 10(4), pp. 1217-1223

[28] Mitra, S., and Acharya, T.: *"Data Mining: Multimedia, Soft Computing, and Bioinformatics"*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2003.

[29] Liao, W., Liu, Y., Choudhary, A.: *"A Grid-based Clustering Algorithm using Adaptive Mesh Refinement"*, 7th Workshop on Mining Scientific and Engineering Datasets, Lake Buena Vista, Florida, USA, 2004.

[30] Park N. H., and Lee, W. S.: *"Statistical grid-based clustering over data streams"*, ACM SIGMOD Record, 2004, Vol. 33(1), pp. 32-37

[31] Guliato, D., Rangayyan, R. M., Carnielli, W. A., Zuffo, J. A., Desautels, J. E. L.: *"Segmentation of Breast Tumors in Mammograms by Fuzzy Region Growing"*, Proc. of the 20th Annual International Conference of the IEEE Engineering in Medical and Biology Society, Hong Kong, China, 1998, Vol. 2, pp. 1002-1005

[32] Martin, D., Fowlkes, C., Tal, D., Malik, J.: *"A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics"*, Proc. of the 8th IEEE International Conference on Computer Vision, Vancouver, Canada, July 2001, Vol. 2, pp. 416-423

[33] Pantofaru, C., and Hebert, M.: *"Comparison of Image Segmentation Algorithms"*, tech. report CMU-RI-TR-05-40, Robotics Institute, Carnegie Mellon University, 2005.

[34] Christoudias, C. M., Georgescu, B., Meer, P.: *"Synergism in low level vision"*, Proc. of the 16th International Conference on Pattern Recognition, Quebec, Canada, 2002, Vol. 4, pp. 150-155

[35] Unnikrishnan, R., Pantofaru, C., Hebert, M.: *"A Measure for Objective Evaluation of Image Segmentation Algorithms"*, Proc. of the 2005 IEEE Conference on Computer Vision and Pattern Recognition, Workshop on Empirical Evaluation Methods in Computer Vision, San Diego, CA, USA, 2005, Vol. 3, pp. 34-41

[36] Unnikrishnan R., and Hebert, M.: *"Measures of Similarity"*, Proc. of 7th IEEE Workshop on Applications of Computer Vision, Breckenridge, CO, USA, 2005, pp. 394-400

[37] Stipaničev, D., Vuko, T., Krstinić, D., Štula, M., Bodrožić, Lj.: *"Forest Fire Protection by Advanced Video Detection System - Croatian Experiences"*, Third TIEMS Workshop - Improvement of Disaster Management System, Trogir, Croatia, 2006.

List of Figures

(a) Original image

(b) Original image

(c) $h_0 = 8$

(d) $h_0 = 8$

(e) $h_0 = 16$

(f) $h_0 = 16$

(g) $h_0 = 16$, min. size = 50pix

(h) $h_0 = 16$, min. size = 50pix

Figure 2: Segmentation for different h_0. In images (g) and (h) segments smaller than 50 pixels are connected to the neighbor with the most similar color. Region boundaries are superimposed to emphasize details.

(a) Original image

(b) MS EDISON

(c) FHS, $h_0 = 4$

(d) EGB

(e) FHS, $h_0 = 12$

(f) JSEG

(g) FHS, $h_0 = 20$

(h) GNM

Figure 3: Comparison of the FHS with other methods. The output differs for different algorithms. The EGB uses random coloring scheme. JSEG algorithm superimposes region boundaries on the original image.

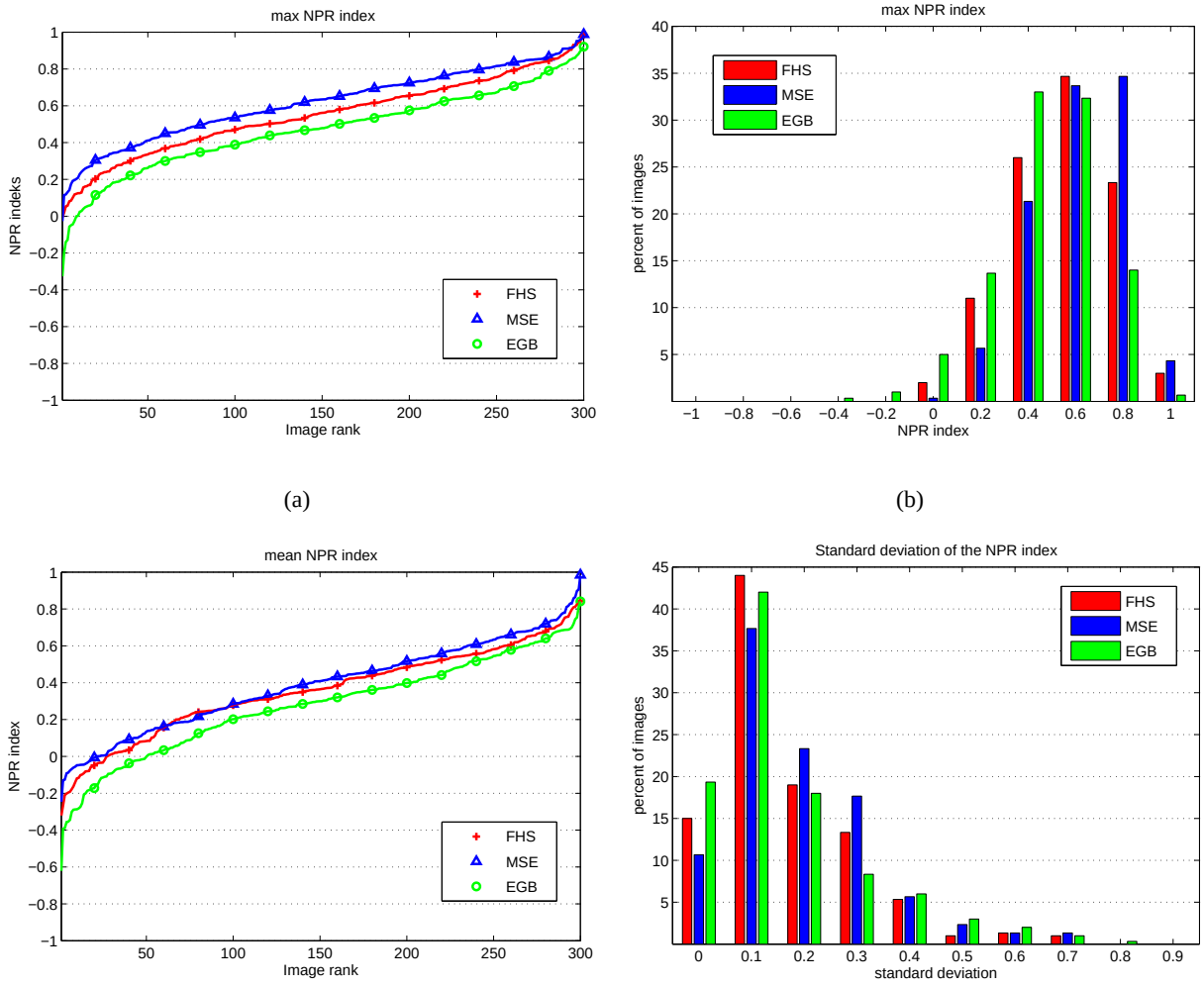
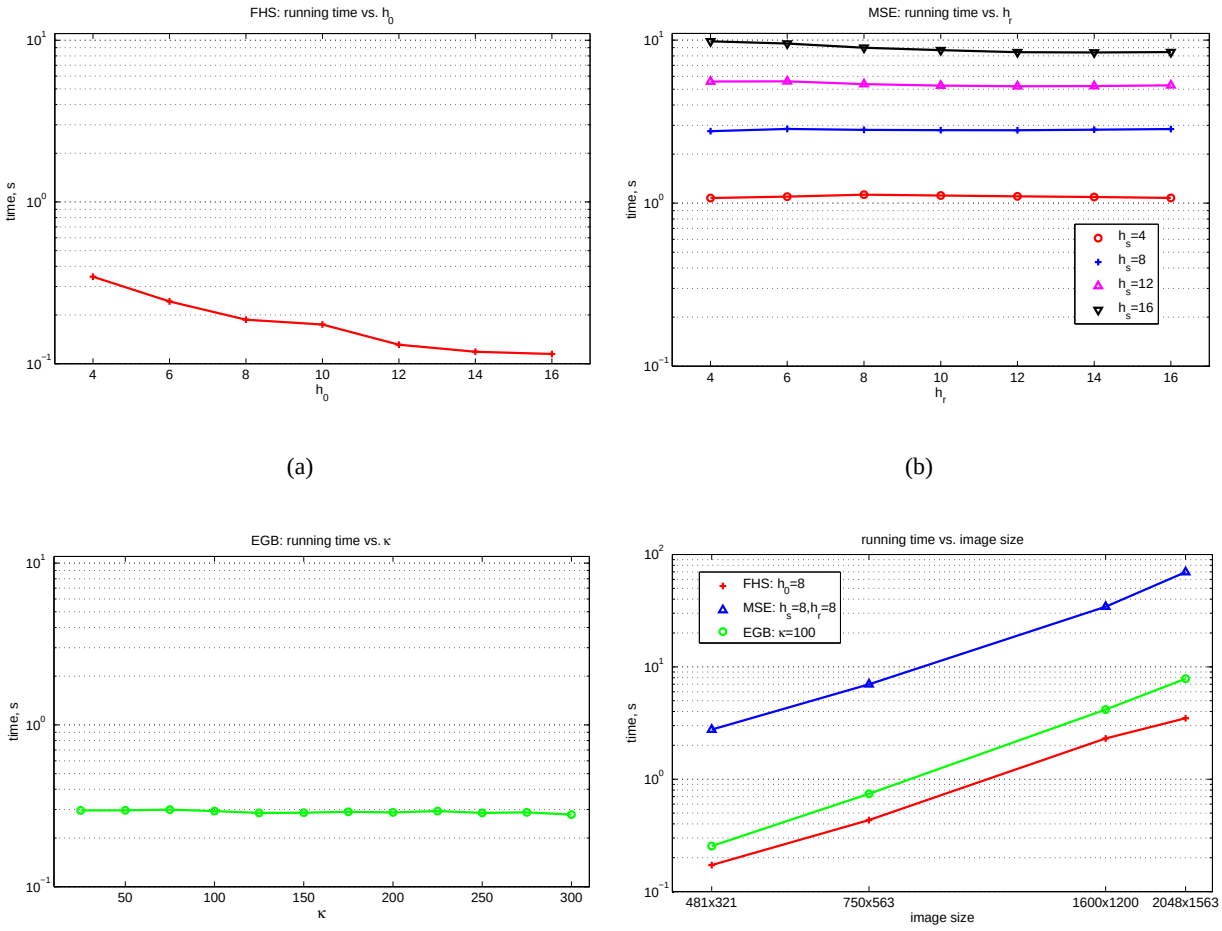


Figure 4: NPR index varying algorithm parameters: (a) maximal NPR index, (b) distribution of the maximal NPR index, (c) mean NPR index, (d) standard deviation of the NPR index



(a) (b)

(c) (d)

Figure 5: Algorithms running time: Plots (a), (b) and (c) show running times of evaluated algorithms vs. controlling parameters. Plot (d) represents running time vs. image resolution.