

# Scalable multiresolution color image segmentation

Fardin Akhlaghian Tab<sup>a,\*</sup>, Golshah Naghdy<sup>a</sup>, Alfred Mertins<sup>b</sup>

<sup>a</sup>*School of Electrical, Computer and Telecommunications Engineering, University of Wollongong, Wollongong, NSW 2522, Australia*

<sup>b</sup>*Signal Processing Group, Institute of Physics, University of Oldenburg, Germany*

Received 16 May 2005; received in revised form 20 September 2005

Available online 25 October 2005

---

## Abstract

This paper presents a novel multiresolution image segmentation method based on the discrete wavelet transform and Markov Random Field (MRF) modeling. A major contribution of this work is to add spatial scalability to the segmentation algorithm producing the same segmentation pattern at different resolutions. This property makes it suitable for scalable object-based wavelet coding. To optimize segmentation at all resolutions of the wavelet pyramid, with scalability constraint, a multiresolution analysis is incorporated into the objective function of the MRF segmentation algorithm. Examining the corresponding pixels at different resolutions simultaneously enables the algorithm to directly segment the images in the YUV or similar color spaces where luminance is in full resolution and chrominance components are at half resolution. Allowing for smoothness terms in the objective function at different resolutions improves border smoothness and creates visually more pleasing objects/regions, particularly at lower resolutions where down-sampling distortions are more visible. In addition to spatial scalability, the proposed algorithm outperforms the standard single and multiresolution segmentation algorithms, in both objective and subjective tests, yielding an effective segmentation that particularly supports scalable object-based wavelet coding.

© 2005 Elsevier B.V. All rights reserved.

*Keywords:* Multiresolution image segmentation; Markov random field; Scalability; Color image; Smoothness

---

## 1. Introduction

Image segmentation is the process of dividing an image into homogenous regions, which is an essential step toward higher level image processing such as image analysis, pattern recognition and computer vision. In particular, effective segmentation is crucial for the emerging object-based image/

video standards such as object-based coding standards defined by MPEG-4 [1] and content-based shape descriptor used in MPEG-7 [2].

In scalable object-based coding, a single codestream can be sent to different users with different processing capabilities and network bandwidths by selectively transmitting and decoding the related parts of the codestream [3]. Some of the desirable scalable functionalities are signal to noise ratio (SNR) scalability, spatial scalability and temporal scalability [3]. A scalable bitstream includes embedded parts that offer increasingly better SNR, greater spatial resolution or higher frame rates. Therefore considering the spatial scalability, it is

---

\*Corresponding author. Tel.: +98 2 4221 4398;  
fax: +98 2 4221 3244.

*E-mail addresses:* fat98@uow.edu.au (F.A. Tab),  
golshah@uow.edu.au (G. Naghdy),  
alfred.mertins@uni-oldenburg.de (A. Mertins).

necessary to extract and present objects' shape at different resolutions for the scalable object-based encoder/decoder systems. For an effective scalable object-based coding algorithm, it is essential that the shapes of the extracted objects at different resolutions be similar or equivalently, the pattern of segmented regions should be similar at different resolutions. We call the segmentation algorithm with the similar patterns at different resolutions scalable segmentation.

Multiresolution image segmentation algorithms analyze the image at different resolutions resulting in some advantageous over the single-resolution segmentation such as

- less computational complexity,
- improvement in the convergence rate,
- reduction in over-segmentation cases,
- less sensitivity to noise,
- ability to capture the image structures at different resolutions,
- less dependence on initial segmentation.

These algorithms consider the inter-scales image data correlation in the segmentation procedure. In the most straightforward case, these algorithms consider inter-scale correlation by projecting the lower resolution segmentation result to the next higher resolution as an initial segmentation estimation. The segmentation is further refined at the current higher resolution by a single resolution segmentation. This procedure continues progressively until highest resolution is segmented [4–7]. One of the challenges arising from this approach is that higher resolution segmentations are more rigorous than the lower resolution segmentations and segmentation maps at higher and lower resolutions are not quite identical. For example, some objects are not detected or are partly detected at low resolutions while they are perfectly detected at higher resolutions. This makes the higher resolution segmentation more reliable for object extraction applications. Therefore the semantic segmentation and object extraction algorithms extract objects from the highest resolution segmentation.

In [4], Pappas presents an adaptive clustering algorithm based on maximum a posteriori (MAP) estimation which allows for slow space-varying mean statistics for each region. Local intensity variations are considered in an iterative procedure which estimates the Markov Random Field para-

eters over a sliding window whose size decreases as the algorithm progresses. In the hierarchical mode, segmentation starts at the coarsest resolution. Once the image at this resolution is segmented, the result is used as an initial segmentation for the next finer resolution until the finest resolution is segmented. In [5] the Pappas algorithm is updated by considering the edge information to increase the accuracy of the segmentation and reduce the under-segmentation produced by the Pappas algorithm. In [6] noisy images are decomposed by redundant discrete wavelet transform (RDWT). In RDWT decomposition, first the low pass sub band image is filtered  $L$  times and then down sampling is done again  $L$  times. Similarly, the segmentation starts from lower resolution and the result is propagated to the higher resolution. One of the disadvantages of this algorithm is under-segmentation. In this algorithm low resolution segmentation could be significantly different from highest resolution segmentation and scalability is not provided. Munoz et al. [7] propose a multiresolution image segmentation which integrates both region and boundary information. The image is decomposed into several resolutions. First, at the coarsest resolution, the most relevant edges are detected. Then seeds are placed far from the edges and the region growing algorithms obtain the regions. Using a global energy function and a greedy optimization algorithm, all the pixels are classified. The segmentation is subsequently projected to the next higher resolution where non-boundary pixels are used to model the regions in the higher resolution. The greedy optimization algorithm again obtains the regions. The algorithm cannot detect small objects/regions if they are not detected in the lowest resolutions. Considering the inter-scale correlation in the pixel classification procedure or extension to the scalable mode is not easily possible.

In the second group of segmentation algorithms, the inter-scale correlations is considered in the statistical models and decision at each pixel/block is based on the information of the different resolutions [8–12]. However, often only the causal inter-scale correlation with the latest lower resolution [8,9,12,11] or the next higher resolution is considered [10]. Considering the other resolutions, results in a very complex model and increases the computational complexity. Bouman et al. [8] introduce hierarchical MRF variables defined as a coarse to fine Markov chain of levels. The associated interaction structure is a quadtree which

correlates the current node with the parent at the last resolution. It does not include the spatial correlation at the same resolution and also it is not shift invariant since two pixels that would be adjacent in terms of spatial lattice may be actually apart in the graph structure. In [9] Comer et al. consider the inter-scale correlation as well as intra-scale correlation. They propose a multiresolution texture segmentation which fits an auto regressive model to the pyramid representation of the image. The MAP optimization criterion is replaced with the multiresolution maximization of the posteriori marginal (MPPM) estimation which facilitates the use of EM algorithm to estimate the parameters such as auto regressive model coefficients and prediction variances of different textures. The coarsest resolution is segmented in a single resolution mode and the segmentation is propagated down to the other levels of the pyramid. In this approach, correlation is only between adjacent resolutions and low resolution segmentation error will propagate to higher resolutions. Although some parameters are estimated, many parameters such as spatial interaction or number of segmentation classes are determined experimentally. Wilson et al. [12] update the statistical models used by Bouman et al. [8] with the view that each scale or resolution data is conditioned not only on its immediate predecessor but also directly dependent on its neighbors at its own scale. Segmentation starts from the coarse resolution and is projected to the next resolution as an initial estimation. At each resolution, the image is divided into blocks and every block is classified by a MRF-based segmentation. Because the classification is block based, after the highest resolution optimization, a line processing refines the regions' borders to the actual borders. This method suffers from a number of shortcomings. Detecting new regions in higher resolutions specially small regions is not possible. Interaction between different resolutions are causal from low to highest resolution. It needs a region boundary refinement which increases the computational complexity and also its procedure does not interact with region labeling. The method cannot be extended to scalable mode easily. Kato et al. [10] introduce a novel hierarchical segmentation which includes a three dimensional neighborhood system. In addition to spatial correlation, the interaction with both higher and lower resolutions are considered. The algorithm alternates between parameter estimation and segmentation algorithm.

Unfortunately, the resulting parameter estimation and segmentation procedures require considerable computing time. In addition considering both higher and lower resolutions produces a complex model and increases the computational complexity.

None of these works and similar ones in the literature, consider inter-scale correlation between all pyramid resolutions. In addition their extension to scalable segmentation, producing the same segmentation patterns at different resolution, is nearly impossible or results in an algorithm with large computational complexity. In order to produce similar objects/regions at different resolutions, we present a novel MRF-based multiresolution grey/color image segmentation algorithm which extends the statistical model to consider the correlation between all the resolutions without overly increasing the computational complexity. It produces the same segmentation patterns at different resolutions and it is applicable to object-based wavelet coding algorithms. Bearing in mind the down-sampling of pixels to lower resolution in wavelet decomposition, corresponding pixels at different resolutions are considered as vector of pixels which are supposed to have the same segmentation label. A vector or multiresolution/multidimensional analysis is incorporated in the objective function of the MRF-based segmentation algorithm which aligns the segmentation algorithm with the object-based wavelet coding spatial scalability. This vector analysis keeps wavelet spatial scalability as a constraint which fits multiresolution MRF segmentation to wavelet-based scalable object coding.

The multiscale analysis uses different-resolution information concurrently, which produces better results than regular single and multiresolution Bayesian segmentation algorithms. It combines and processes both coarse global information of low resolution with fine local information obtained from higher resolutions of the wavelet decomposition pyramid. Therefore, it combines good features of both single and multiresolution segmentations. While it is noise resistance, it detects objects/regions better than regular multiresolution segmentation and also results in a lower number of regions than single level segmentation. To optimize the objective function of the segmentation algorithm, the Iterated Condition Mode (ICM) algorithm according to [4], matched to the scalable multiscale analysis, is used.

In this work, a smoothness criterion is incorporated in the objective function of the segmentation

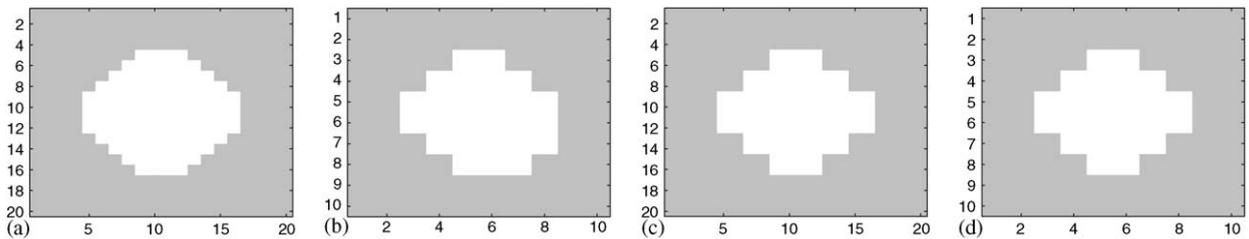


Fig. 1. Circles in different resolutions: (a) closer approximation of a digital circle at high resolution; (b) down-sampling to low resolution; (c) worse approximation of a digital circle at high resolution; (d) down-sampling of (c) to low resolution.

algorithm which results in a more normal or visually pleasing objects/regions. Many natural objects exhibit smooth borders/edges, and distortions such as down sampling often result in rough borders/edges. Hence, to some extent there is correlation between visually pleasing objects and smoothness. In order to accentuate smoothness at lower resolutions, bigger smoothness coefficients are chosen for lower resolutions. By considering different coefficients for the smoothness term of different resolutions, the distortion effects of down-sampling is reduced.

Color images have more information than grey-level images which results in more reliable separation of foreground regions from background in color image segmentation algorithms. It has been recognized that selection of an appropriate color space produces more perceptually effective segmentation results [13,14]. In particular, segmentation in YUV or LUV spaces produces more favorable results than the RGB space [13–15]. Many of the images and image sequences in the databases are in YUV format where Y is in full resolution while U and V components are in half resolution. The fact that the Y, U, and V channels are presented at different resolutions is not considered in any of the existing regular single or multiresolution color image segmentation algorithms. However, this fact calls for a fitted multiresolution algorithm to perform the segmentation task effectively. The proposed algorithm has enough flexibility to directly segment color images. In the vector analysis, only available components of color data at different resolutions are used to classify the vector to one of the segmentation labels. For this reason, the proposed algorithm can segment grey-level images as well.

In order to produce similar objects/regions at different resolutions, an alternative method is single-resolution segmentation followed by down-sampling. In single-resolution region-based image/

video segmentation algorithms, features such as intensity/color, texture, motion, etc. are considered at the highest resolution. In this method, good features of multiresolution segmentation such as reduced noise sensitivity are lost, and producing optimized and visually pleasing objects/regions at different resolutions, as a criterion, is not considered. Furthermore, down-sampling distorts shapes and cannot preserve their topology at lower resolutions for all possible shapes [16]. In other words, achieving visually pleasing objects/regions at higher resolution does not necessarily ensure similar quality at lower resolutions. For example in Fig. 1, down-sampling of two digital circles is compared. It can be seen that down sampling of the better approximation of a digital circle at high resolution can result in worse shape at lower resolution.

This paper is organized as follows. Section 2 refers to the scalability in object-based wavelet coding. In Section 3, the proposed scalable multiresolution segmentation algorithm, which includes a statistical image modeling and optimization processes, is explained. Some experimental results and discussion are presented in Section 4, and finally, conclusions are drawn in Section 5.

## 2. Object-based wavelet coding scalability

Scalability means the capability of decoding a compressed sequence at different data rates. It is useful for image/video communication over heterogeneous networks which require a high degree of flexibility from the coding system. Some of the desirable scalable functionalities are SNR scalability, spatial scalability and temporal scalability [3]. In particular spatial scalability means that, depending on the end user's capabilities (bandwidth, display resolution, etc.), a resolution is selected and all the shape and texture information is sent and decoded at the appropriate resolution. Scalable image/video coding has also different applications such as web

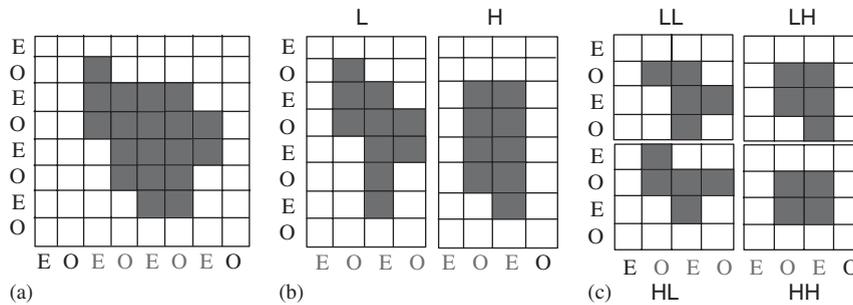


Fig. 2. Decomposition of a non rectangular object with odd-length filters: (a) the object, shown in dark grey; (b) the decomposed object after horizontal filtering; (c) decomposed object after vertical filtering. The letters “E” and “O” indicate the position (even or odd) of a pixel in the horizontal and vertical dimensions.

browsing, image/video database systems, video telephony, etc.

In wavelet-based spatial scalability applications, due to the self similarity feature of the wavelet transform, the shape in lower scale is the shape in the lowpass (LL) subband. The exact relationship between the full-resolution shape and its low-resolution versions depends on the kind of wavelet transform used for the decomposition. In this paper we use an odd length filter (e.g. 9/7), where all shape points with even indices<sup>1</sup> are downsampled for the lowpass band [17]. Fig. 2 further illustrates the wavelet decomposition of arbitrarily shaped objects when using an odd-length filter. The final four-band decomposition is depicted in Fig. 2(c). As a result, every shape pixel with even indices has a corresponding pixel on the lower resolution and every shape pixel on the lower level has a corresponding pixel on the next higher level. By considering the self similarity of the wavelet transform, it is straightforward to suppose that the pixels of a shape with even indices have the same segmentation classifications as the corresponding pixels on the lower level.

The wavelet self similarity extends to all low pass subband shapes of different levels. Therefore the discussed relationship between corresponding pixels is extended to shapes at different scales. Corresponding pixels at different resolutions have the same segmentation class.

### 3. Spatial segmentation algorithm

Markov Random Field statistical modeling is used in many image processing applications. In order to solve an image processing problem by the MRF technique, a statistical image model has to be

fitted to the application which captures the intrinsic character of the image in a few parameters. Image/Video processing problems, including all uncertainties and constraints, can therefore be converted to a mathematical parameter optimization problem [18].

#### 3.1. Statistical color image model

The main challenge in multiresolution image segmentation for scalable object-based wavelet coding is to keep the same relation between extracted objects/regions at different resolutions as it exists between the decomposed objects/regions at different resolutions in a shape adaptive wavelet transform. The other constraint is border smoothness particularly in lower resolutions. Different smoothness coefficients defined at different resolutions give some degree of freedom to put more emphasis on the low-resolution smoothness. To meet these challenges, Markov random field modeling is selected as it includes low level processing at pixel level and has enough flexibility in defining objective functions matched with the problem at hand [18]. We first explain the statistical model of single resolution grey/color image segmentation and then extend it to the scalable multiresolution segmentation mode. In a regular single-level MRF-based image segmentation the problem is formulated using a criterion such as the maximum a posteriori (MAP) probability. The desired segmentation is denoted by  $X$ , and  $Y$  is the observed color image with three channels shown by a three dimensional vector  $Y = [Y_1, Y_2, Y_3]$ . Then according to the Bayes rule, the a posteriori probability density of the segmentation variables can be written as

$$P(X|Y) \propto P(Y|X)P(X), \quad (1)$$

<sup>1</sup>Suppose indices start from zero or an even number.

where  $P(X|Y)$  represents the conditional probability of the segmentation label, given the observation  $Y$ . By assuming the conditional independence of the channels given the segmentation field [13], we have  $P(Y|X) = P(Y_1|X)P(Y_2|X)P(Y_3|X)$ , and the conditional probability in (1) becomes

$$P(X|Y) \propto P(Y_1|X)P(Y_2|X)P(Y_3|X)P(X). \quad (2)$$

The label field  $X$  is normally modeled by a MRF stochastic variable. Spatial continuity is easily incorporated into the segmentation, because it is inherent to MRFs [19]. Using a four or eight pixel neighborhood system considering only pairwise cliques,  $P(X)$  is then a Gibbs distribution [4] and is defined by its energy function  $U(X)$  such that

$$P(X) = \frac{1}{Z} \exp\left(-\frac{1}{T} U(X)\right), \quad U(X) = \sum_{c \in C} V_c(X), \quad (3)$$

where  $C$  is the set of all cliques, and  $V_c$  is the clique potential function. A clique is a set of neighboring pixels. A clique function depends only on the pixels that belong to the clique. In single-resolution segmentation, usually one or two pixel cliques are used as shown in Fig. 3(a), and for one pixel cliques we assume that the one pixel clique potentials are zero, which means that all region types are equally likely [4]. Spatial connectivity of the segmentation is imposed by assigning the following clique function:

$$V_c(s, r) = \begin{cases} -\beta & \text{if } X(s) = X(r), \\ +\beta & \text{if } X(s) \neq X(r), \end{cases} \quad (s, r) \in C. \quad (4)$$

Herein  $\beta$  is a positive number and  $s$  and  $r$  are a pair of neighboring pixels. Note that a low potential or energy corresponds to a higher probability for pixel pairs with identical labels and lower probability for pairs with different labels, which automatically encourages spatially connected regions.

The conditional probability density  $P(Y_i|X)$ ,  $i = 1, 2, 3$ , is modeled as a white Gaussian process,

with mean  $\mu_{X(s)}^i(s)$  and variance  $\sigma_i^2$  for channel  $i$ . Each region is characterized by a mean vector  $[\mu_{X(s)}^1(s), \mu_{X(s)}^2(s), \mu_{X(s)}^3(s)]$  which is a slowly varying function of  $s$ . Therefore  $P(Y|X)$  can be described by the following equation:

$$P(Y|X) \propto \exp\left\{-\sum_s \left(\sum_{i=1}^3 \frac{1}{2\sigma_i^2} (Y_i(s) - \mu_{X(s)}^i(s))^2\right)\right\}. \quad (5)$$

Considering Eqs. (1), (3) and (5), the conditional probability density of the segmentation variable becomes

$$P(X|Y) \propto \exp\left\{-\sum_s \left(\sum_{i=1}^3 \frac{1}{2\sigma_i^2} (Y_i(s) - \mu_{X(s)}^i(s))^2 + \frac{1}{T} \sum_{r \in \partial_s} V_c(s, r)\right)\right\}. \quad (6)$$

It is easy to see that the parameters  $\sigma_i$ ,  $i = 1, 2, 3$ ,  $T$  and  $\beta$  are interdependent. Therefore, to simplify the expression, the parameters  $2\sigma_i^2$ ,  $i = 1, 2, 3$ , and  $T$  are set to one, and the segmentation result is controlled by the value of  $\beta$  in the  $V_c$  function. The probability density function has two components. One forces the region intensity to be close to the data, and the other imposes spatial continuity. Considering the MAP criterion, we maximize the probability  $P(X|Y)$ , which is equivalent to minimizing the negative value of argument of the exponential function in Eq. (6). This results in the following cost or objective function which has to be minimized with respect to  $X(s)$ :

$$E(X) = \sum_s \left(\sum_{i=1}^3 (Y_i(s) - \mu_{X(s)}^i(s))^2 + \frac{1}{T} \sum_{r \in \partial_s} V_c(s, r)\right). \quad (7)$$

In grey-scale images only the intensity channel exists, and the terms  $Y_1$  and  $Y_2$  in Eq. (7) have to be deleted.

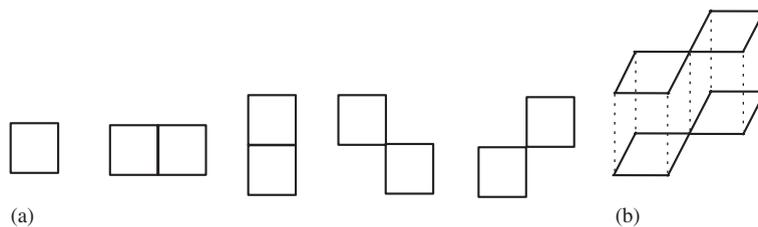


Fig. 3. (a) Normal one and two pixels cliques sets. (b) A clique of two vectors with the vectors' dimension equal to two.

To obtain the final segmentation, this objective function is minimized by one of the several MRF objective minimization methods [18]. To tailor this objective function to scalable multiresolution color image segmentation, initially, the wavelet transform is applied to the original image and a pyramid of decomposed images at various resolutions is created. Let  $Y = [Y_1, Y_2, Y_3]$ , where  $Y_i, i = 1, 2, 3$ , is the intensity of channel  $i$  of the pyramid's pixels. The segmentation of the image into regions at different resolutions will be denoted by  $X$ .

As mentioned earlier, considering scalability, a pixel and its corresponding pixels at all other pyramid levels have the same segmentation label. Therefore they change together during the segmentation process. To change the segmentation label of a pixel, the pixel and all its corresponding pixels at all other levels have to be analyzed together. As a result, an analysis of a set of pixels in a multidimensional space instead of a single resolution analysis needs to be used. The term "vector" is used to refer to multidimensional space. A vector includes corresponding pixels at different resolutions of the pyramid. A symbol  $\{s\}$  shows a vector which includes pixel  $s$ . The dimension of a vector is equal to the number of its pixels which are located at different resolutions. If the corresponding pixels are determined according to the wavelet transform down-sampling, the vector dimension depends on the index of pixels, and it can be 1, 2 or more. Using these primary definitions, the clique concept is extended to vector space. The extended cliques act on two vectors instead of two pixels. Fig. 3(a) shows regular one and two pixels clique sets. In Fig. 3(b), the extension of one of these cliques to the array mode in two dimensional space can be seen.

The extension of clique functions is achieved through the following steps: Eq. (4) is used for cliques of length two at a resolution where pixels  $s$  and  $r$  are two neighboring pixels at the same resolution level. Eq. (8) is defined for multiple levels, where  $\{s\}$  and  $\{r\}$  are vectors corresponding to two neighboring pixels  $s$  and  $r$ . The neighboring pixels of the two vectors  $\{s\}$  and  $\{r\}$  at level  $k$  are denoted as  $s_k$  and  $r_k$ . The lowest resolution which include a pixel of vector  $\{s\}$  is denoted as  $M$  and  $N$  is the dimension of vectors  $\{s\}$  and  $\{r\}$ . A positive value is assigned to the parameter  $\beta$ , so that adjacent pixels, of two neighboring vectors, are more likely to belong to a same region than to different regions. Increasing the value of  $\beta$  decreases

the sensitivity to intensity changes [4].

$$V_c(\{s\}, \{r\}) = \left( \frac{\sum_{k=M}^{M+N-1} L_k}{N} \right) \sum_{k=M}^{M+N-1} (-1)^{L_k} \cdot \beta,$$

$$L_k = \begin{cases} 1 & \text{if } X(s_k) = X(r_k), \\ 0 & \text{if } X(s_k) \neq X(r_k), \end{cases}$$

$$s_k \in \{s\}, r_k \in \{r\}, r_k \in \partial s_k. \quad (8)$$

It is notable that Eq. (8) extends the clique definition to multiresolution mode. In vector  $\{S\}$  corresponding pixels at different resolutions are determined according to the down-sampling relationship of arbitrary shape wavelet transform. In other words the pixels of the lower resolution occur at down-sampled positions of the higher-resolution pixels in the pyramid. However, pixels of  $\{r\}$  are neighbors of  $\{s\}$  at different resolutions. As a result of the clique extension to multiresolution space, segmentation processing will continue in the vector space, therefore, intensity average and segmentation label functions are also extended to vector space. The intensity of pixels at different channels in set  $\{s\}$  form a vector  $Y(\{s\}) = [Y_1(\{s\}), Y_2(\{s\}), Y_3(\{s\})]$ , and similarly,  $\mu(\{s\}) = [\mu_1(\{s\}), \mu_2(\{s\}), \mu_3(\{s\})]$  is the mean vector. Therefore, the objective function is extended to vector space as follows:

$$E(X) = \sum_{\{S\}} \left\{ \sum_{i=1}^3 \|Y_i(\{s\}) - \mu_{X(\{s\})}^i(\{s\})\|^2 + \sum_{\{r\} \in \partial \{s\}} V_c(\{s\}, \{r\}) \right\}. \quad (9)$$

The outer summation is over vectors, while the first inner summation is related to the distances of the pixel's intensities from the estimated average for each channel of color images. The second inner summation is over all neighborhood vectors of vector  $\{s\}$ . The two vectors  $\{s\}$  and  $\{r\}$  are neighbors if pixels of  $\{s\}$  and  $\{r\}$  located at the same resolution are also neighbors. The approach used in this section, expressed by Eqs. (7)–(9), is a generalization of regular single-resolution segmentation to scalable multiresolution segmentation.

### 3.2. Smoothness criterion

Traditionally, in region-based image/video segmentation algorithms, the image features such as pixels' grey level or color have been considered. In

most of these approaches, emphasis is put on the accuracy of segmentation. However objects/regions shape delineation, and producing a well-pleasing objects’/regions’ shape has not attracted enough attention due to the ill-posed problem nature of the segmentation task [20,21]. In contour/edge-based segmentation algorithms, another important criterion, related to the appearance of the extracted objects/regions, has been considered. In these algorithms, the extracted objects/regions borders are smoothed [22,23]. Ideally, borders are edges in the image, which are one of the most important properties for visual perception. Because most natural objects exhibit smooth edges and distortions such as down-sampling often creates rough edges, there is a correlation between border smoothness and visually pleasing objects. Therefore border smoothness terms corresponding to different resolutions have been added to the objective function to contribute in our MRF-based segmentation approach.

The proposed smoothness definition is based on the border’s curvature, which is the rate of the angle change between a curve and the tangent line to the curve. In a digital environment an estimation of curvature can be used. The estimation is explained in Fig. 4. Minimizing the proposed estimation of smoothness prevents unwanted fluctuations in the border pixels.

A large number of pixels ensures border smoothness at high resolutions, however, at lower resolutions the visual quality can suffer due to down-sampling distortion and lack of sufficient information. To reduce this effect, the smoothness could be enhanced at lower resolutions more rigorously than higher resolutions. The priority is realized by bigger coefficients for lower resolution smoothness. Therefore the objective function is

updated according to the following equation:

$$E(X) = \sum_{\{S\}} \left\{ \sum_{i=1}^3 \|Y_i(\{s\}) - \mu_{X(\{s\})}^i(\{s\})\|^2 + \sum_{\{r\} \in \partial\{s\}} V_c(\{s\}, \{r\}) + \sum_{q \in \{s\}} I_{res(q)} * v(q) \right\}, \tag{10}$$

where  $v(q)$  shows the curvature estimation of pixel  $q$  a pixel of vector  $\{s\}$ , and  $I_{res(q)}$  is a coefficient which decreases when resolution increases. In grey-level images, only the grey-intensity channels are available, therefore it is enough to only consider  $Y_1$  in the second summation in Eq. (10) which gives the following objective function for grey-level images:

$$E(X) = \sum_{\{S\}} \left\{ [Y(\{s\}) - \mu_{X(\{s\})}(\{s\})]^2 + \sum_{\{r\} \in \partial\{s\}} V_c(\{s\}, \{r\}) + \sum_{q \in \{s\}} I_{res(q)} * v(q) \right\}, \tag{11}$$

where  $Y$  is the grey-intensity function and  $\mu$  is the grey-intensity average function.

The proposed smooth object extraction is different from a simple objects’ border smoothness as has been done in [24] which is a filtering of the extracted video object shape to remove the small elongation introduced during the segmentation process, in the following areas. (1) Our smoothing process takes part in the segmentation algorithm and changes the segmentation outcome. (2) With sufficient contrast, the proposed algorithm produces borders that are more faithful to the regions shape. (3) On some occasions, some background pixels are added to the foreground regions to produce better looking shapes especially at different resolution. (4) The smoothness factor could be adjusted for different

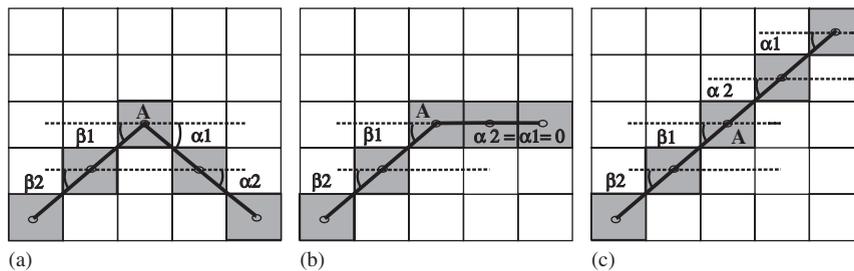


Fig. 4. Curvature estimation;  $k(A) = 05.abs(\alpha_2 + \alpha_1) - (\beta_2 + \beta_1)$ : (a) corner point,  $\beta_1 = \beta_2 = 45, \alpha_1 = \alpha_2 = -45, k = 90$ ; (b) change direction point,  $\beta_1 = \beta_2 = 45, \alpha_1 = \alpha_2 = 0, k = 45$ ; (c)  $\beta_1 = \beta_2 = 45, \alpha_1 = \alpha_2 = 45, k = 0$  same direction.

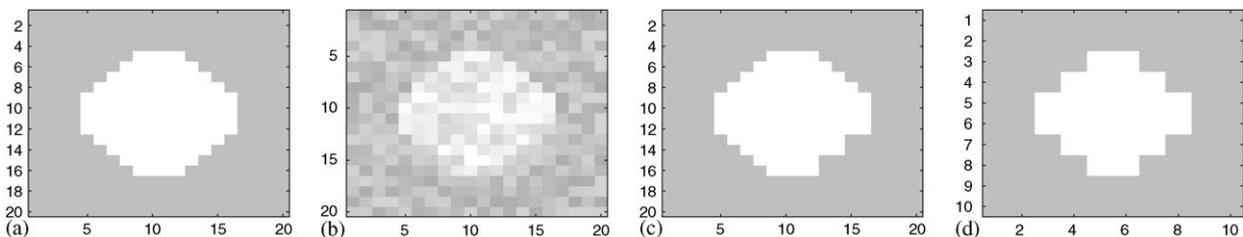


Fig. 5. Scalable segmentation of a digital circle with emphasis on low level smoothness: (a) original image; (b) noisy image; (c) segmentation at  $20 \times 20$  resolution; (d) segmentation at  $10 \times 10$  resolution.

resolutions to produce visually pleasing shapes at different resolutions with scalability as a constraint.

As an example of smoothness effect in spatial segmentation, consider the circle in Fig. 5(a). It has two grey levels, 100 in the background area and 200 in the foreground area. We consider a uniform noise in the range  $(0, 50)$  added to the background and subtracted from the object intensity. This noise changes the image from binary to grey level and reduces the pixels intensity variation of the foreground to the background pixels. The image is segmented by the proposed algorithm at two resolutions  $20 \times 20$  and  $10 \times 10$ . We augment the lower resolution smoothness by decreasing the smoothness coefficients to zero for the highest level and increasing the smoothness coefficient for lower resolution. The results are shown in Fig. 5(c) and (d). In this example, the smoothness criterion has deleted some pixels of the shapes at different resolution. The results could be compared with Fig. 1(a), (b) at low and high resolutions considered as regular segmentation. The proposed segmentation method extracts a more pleasing shape at lower resolutions, albeit sometimes adding some distortion at higher resolution. However, large number of pixels at higher resolutions ensures more smoothness and visually pleasing objects.

### 3.3. MAP estimation

The ICM optimization method [25] is used to minimize the objective function in Eq. (10). The segmentation is initialized with the  $k$ -means clustering algorithm for each channel separately. Then neighboring pixels with equal labels at all three channels form a region. The segmentation estimation is improved using ICM optimization [25]. In single-resolution image segmentation, ICM optimizes the objective function pixel by pixel in a raster-scan order until convergence is achieved. At each pixel, the segmentation of the processed pixel is

updated given the current  $X$  at all other pixels. Therefore only the terms in the objective function related to the current pixel need to be minimized:

$$E(X(s)) = \sum_{i=1}^3 (Y_i(s) - \mu_{X(s)}^i(s))^2 - \sum_{r \in \partial_s} V_c(s, r). \quad (12)$$

ICM, as used in the single-level segmentation algorithm for grey-level images by Pappas [4], and extended to color images by Chang et al. [13], is modified to adapt to the scalable multiresolution segmentation algorithm. Similarly, the objective function term corresponding to the current vector is optimized given the segmentation at all other vectors of the pyramid. The resulting objective function terms related to the current vector are

$$E(X\{s\}) = \sum_{i=1}^3 \|Y_i(\{s\}) - \mu_{X(\{s\})}^i(\{s\})\|^2 + \sum_{\{r\} \in \partial\{s\}} V_c(\{s\}, \{r\}) + \sum_{q \in \{s\}} l_q v(q). \quad (13)$$

For grey-level images there is only the intensity channel and the objective function is simplified to

$$E(X\{s\}) = (Y(\{s\}) - \mu_{X(\{s\})}(\{s\}))^2 + \sum_{\{r\} \in \partial\{s\}} V_c(\{s\}, \{r\}) + \sum_{q \in \{s\}} l_q v(q). \quad (14)$$

During the optimization process for each pixels  $s$  of a vector  $\{s\}$ , the terms  $\mu^i(s)$ ,  $i = 1, 2, 3$ , are estimated by averaging the channel intensities of all pixels that belong to the region  $i$  and are inside a window with width  $w$  centered at pixels  $s$ . The window size  $w$  is doubled when we move to the next higher resolution. The average of any pixel  $s$  and its correspondences at all other levels in  $\{s\}$  are used to classify the pixels of  $\{s\}$  to a label which minimizes Eq. (13). To reduce computational complexity, it is enough to consider only labels of  $\{s\}$  and its neighboring vectors to select the best label by the energy minimization through Eq. (13). Therefore for the pixels inside a region there is no computation

Table 1  
Number of vectors with different lengths

Length of vectors	1	2	3	4	5
Number of vectors	$3/4 M \times N$	$3/16 M \times N$	$3/64 M \times N$	$3/256 M \times N$	$1/256 M \times N$

and the regions' border are gradually refined. Furthermore, this border processing prevents isolated noise pixels from becoming a new cluster, resulting in fewer wrongly detected boundaries [26].

Let us consider the overall optimization algorithm now. As mentioned above, the initial segmentation of the pyramid is obtained by the  $k$ -means clustering algorithm [27,28]. The pyramid's pixels are processed progressively from low to high resolutions. At each resolution, pixels are visited in a raster scan order. The intensity average  $\mu^i(s)$ ,  $i = 1, 2, 3$ , at each pixel  $s$  and its corresponding pixels at the other resolutions for all possible classes are estimated with a pre-determined window size  $w$  used for estimation. We then update the estimate of  $X\{s\}$  using the ICM approach with a multi-level analysis using Eq. (14). By updating the segmentation labels of pixels at the current resolution, the corresponding pixels at the other levels are also updated. After convergence at the current resolution, the algorithm moves to the next higher resolution and updates the estimates of  $\mu$  and  $X$  and so on, until all resolutions are processed. The stopping criterion at each resolution is the number of  $X$  update which should be below a pre-defined threshold. To reduce the number of iterations, other convergence criteria can also be used. The whole procedure is repeated with a smaller window size. The algorithm stops when the minimum window size for the lowest level is reached. We have considered the minimum window size being eight for the lowest level.

Scalable and multi-dimensional analysis ties pixels in high and low resolutions together, so that high resolution refinement influences low resolution refinement, too. On the other hand, optimization includes several stages of refinements from low to high resolution with decreasing window size. Therefore, the proposed segmentation algorithm with its objective function and the optimization routine performs repeated low to high resolution segmentation refinement and feedback from high to low resolution segmentation until convergence is reached. The combination of the proposed objective function and the optimization method provide

effective low to high resolution and high to low resolution interactions, therefore providing a reliable and scalable segmentation algorithm.

To quantify the computational complexity of the proposed vector-based processing, the vector size as well as the number of iterations required for the optimization process have to be considered. Because the down-sampling from a given resolution to the next lower one reduces the number of pixels to the  $\frac{1}{4}$ th, the average length of vectors is not much larger than one. To give an example, we consider the segmentation of an image of size  $M \times N$ , using five levels of down-sampling. This means that vectors exist with lengths from one to five. Table 1 shows the number of vectors of each length. The average vector length is easily calculated to be equal to 1.33. Since the number of iterations needed for convergence is image dependent, it is difficult to assign a value to it. However, from the average vector size it can be confidently asserted that vector-based processing increases the computational complexity only moderately. Experimental results confirm that vector processing along with the proposed optimization process increases the computational complexity by less than a factor of 2.5. The major part of this increase is related to the proposed optimization process, which includes several times low to high resolution processing, and not to the vector length itself. On the other hand, as experiments have shown, imposing the smoothness criterion to the segmentation algorithm contributes more towards increasing the computational complexity than the vector-based processing. Smoothness estimation is repeated at each pixel increasing the computational complexity considerably. In different examples, it has been seen that the smoothness criterion increases the complexity of the scalable segmentation by more than a factor of 6.

#### 4. Experimental results and discussion

In this section, experimental results obtained using the algorithm introduced in Section 3 are presented. The results are compared with the regular single-level and multiresolution segmenta-

tion algorithms [4,13]. In the first step, the image is decomposed into three resolutions, using the  $(\frac{2}{7})$  wavelet filter. Then in each level of the decomposition, the image is segmented while scalability between regions in different resolutions, as required for the arbitrary shape wavelet transform, is achieved with the proposed algorithm. Some parameters such as the number of the labels, the continuity  $\beta$  and smoothness coefficients are entered into the algorithm. Automatic determination of these parameters is beyond the scope of this work and could be the subject of further research.

As a first example, the proposed algorithm is tested using frame 5 of the SIF sequence Table Tennis. For grey-level images only the intensity of the grey channel is considered in Eq. (14). Fig. 6 represent the results achieved by the proposed multiresolution scalable, regular single and multiresolution segmentation algorithms. The result of the proposed scalable multiresolution segmentation algorithm is presented only at the finest resolution, because the lower resolutions results have the same patterns and figures.

In regular multiresolution segmentation algorithms, brief, coarse and filtered versions of the image are processed at the lower resolutions, therefore, some small size or low contrast regions are not detected. This drawback is called under-segmentation. In contrast in the proposed algorithm the effects of high resolutions on low-resolution results in the detection of details which is not possible using regular multiresolution segmentation. In other words, the sensitivity to grey-level changes is increased, resulting in a better detection of small or low-contrast objects especially in low resolutions. Table 2 shows the number of detected regions of the Table Tennis image in three spatial resolutions for different segmentation algorithms. The proposed scalable segmentation detects more relevant regions than the regular multiresolution algorithm. For example, consider the segmentation of the textured wall and detection of the ball in the Table Tennis image as presented in Fig. 6 by the proposed

multiresolution scalable, the regular single-resolution, and the multiresolution segmentation algorithm. The single-level segmentation detects the ball, but it also detects a number of spurious regions due to the textured background as the number of regions in Table 2 shows. This drawback is called over-segmentation. The regular multiresolution algorithm misses the ball at different resolutions altogether. The proposed algorithm, however, detects the ball as well as avoiding unsightly segmentation of the textured background.

It is significant to note that while our algorithm has improved sensitivity to grey-level variation it still maintains noise tolerance. To test the scalable segmentation algorithm on noisy images, first a uniform noise in the range  $(-30, +30)$  is added to the Table Tennis images, and then different segmentation algorithms were performed. The number of misclassified pixels for the Table Tennis object including arm, racket and ball (11 033 pixels) as well as the entire image pixels are counted. The results in Table 3 confirm that the proposed algorithm can deal with noisy images as effectively as multiresolution segmentation and much better than single-level segmentation. This result confirms that the introduced multilevel segmentation algorithm maintains most advantages of multiresolution segmentations over single-level segmentations such as better segmentation of noisy images.

The proposed segmentation can be used in general segmentation applications. However, it is especially suited to scalable wavelet-based image object coding which allows us only the pixels belonging to an arbitrarily shaped object to be coded [29,30]. To facilitate “object-of-interest” extraction, in a semi-automatic procedure the user determines the rough boundary of the “object-of-interest” through a graphical user interface (GUI). Subsequently, all the regions with a predetermined percentage of their area inside this closed contour are selected as the regions belonging to the “object-of-interest”. Combining all the selected regions creates the final object. As an example, a

Table 2  
Number of regions in Table Tennis segmentation

Seg. algorithm	60 × 120	120 × 176	240 × 352
Multiresolution	19	55	164
Scalable	42	83	184
Single level	—	—	314

Table 3  
Misclassified pixels in noisy image

Algorithm	Multiresolution	Scalable	Single resolution
Object.	%4.13	%3.04	%8.85
Image	%7.96	%6.20	%18.74

user has roughly determined the objects of interest in Fig. 8. Subsequently the exact borders of the object at different resolutions are determined. The extracted objects by both the scalable segmentation and regular single-resolution segmentation algorithms at three different resolutions are shown in Fig. 7. A comparison of the extracted objects confirms the superiority of the scalable segmentation algorithm in a subjective test. As a complementary step, using a quantitative test, the border smoothness is measured at different resolutions. Table 4 shows the average smoothness for all border pixels in scalable segmentation and single-resolution

segmentation, down-sampled to lower levels at three different resolutions. The quantitative test results, similar to subjective test, show that the proposed algorithm ensures smoother edges compared to the down-sampled versions of single-resolution segmentation. Finally, the extracted image object can be coded by scalable object-based coding algorithms [30].

In the second example, frame 34 of the Mother and Daughter sequence is segmented. The image is in qcif format and is given in the YUV color space. Similar to many other sequences in databases, U and V, the chrominance components, are in half

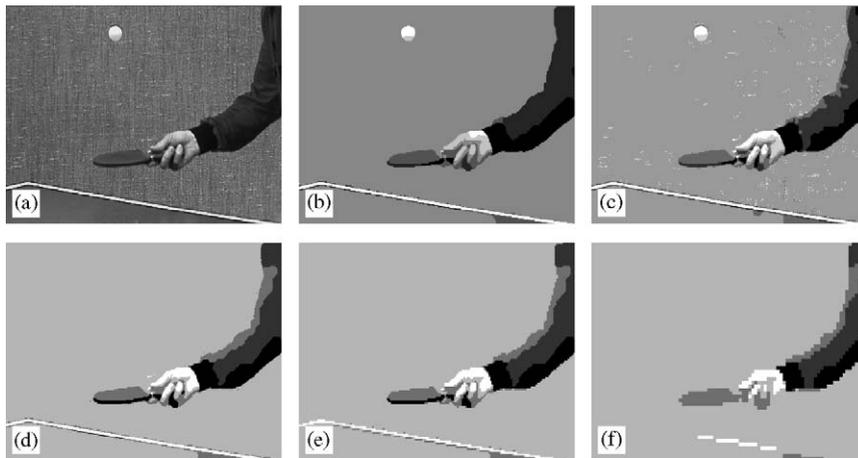


Fig. 6. Table Tennis image segmentation with  $k = 6$  clusters and  $\beta = 100$ : (a) the main image; (b) segmentation by the proposed scalable algorithm at  $240 \times 352$ ; (c) regular single level segmentation; (d) regular multiresolution segmentation  $240 \times 352$ ; (e)  $120 \times 176$  segmentation; (f)  $60 \times 88$  segmentation.

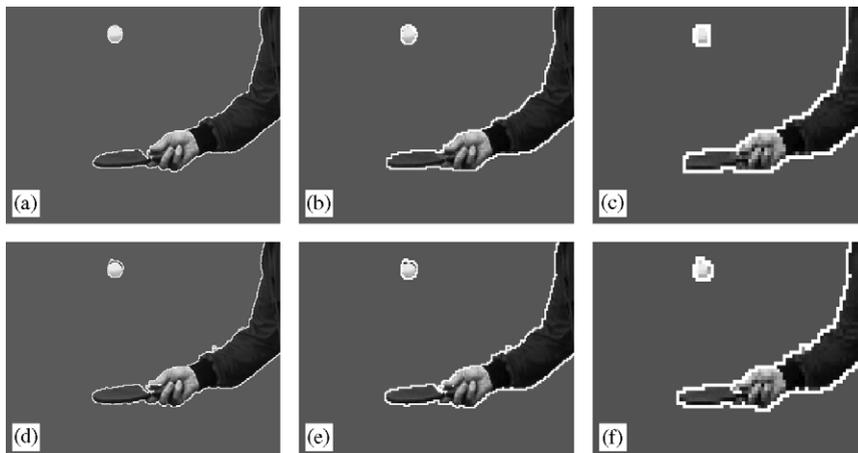


Fig. 7. Table Tennis object extraction, objects are extracted from scalable segmentation in the first row and from single-resolution segmentation in the second row: (a)  $240 \times 352$ ; (b)  $120 \times 176$ ; (c)  $60 \times 88$ ; (d)  $240 \times 352$ ; (e)  $120 \times 176$ ; (f)  $60 \times 88$ .

resolution. Regular color-image segmentation needs the information in the same resolution. Therefore, in the first solution, the image is segmented in grey-

Table 4  
Means of segmentation curvature estimation

Algorithm	60 × 88	120 × 176	240 × 352
Downsample	25.58	22.68	21.33
Scalable	17	17.3	15.17
Improvement	%33.2	%23.65	%28.9

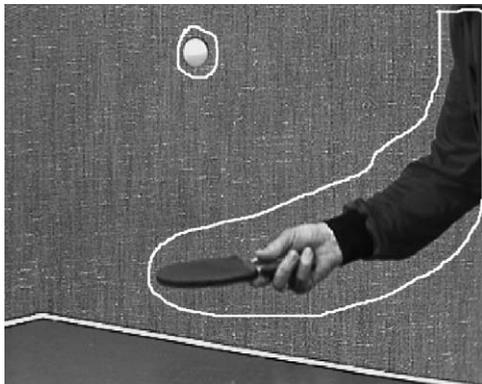


Fig. 8. Table Tennis object selection by user.

level space by a single-resolution statistical image segmentation algorithm [4]. The result is shown in Fig. 9(b). The left area of the daughter's face has not been well separated from the background because there is not enough grey-level contrast between face and background. The same shortcoming happens for the other grey-level segmentation algorithms except when there is over-segmentation with a large number of detected regions, which is not desired for segmentation applications. To successfully separate object's regions from background, color segmentation is performed as an alternative solution. The proposed scalable segmentation algorithm can perform color segmentation using half-resolution chrominance components. The result of segmentation by the scalable color image segmentation is shown in Fig. 9(d). The number of regions in grey-level segmentation is 273 while in color segmentation it is 112, which shows a reasonable color image segmentation algorithm.

In the third example, the color image of Lena in YUV space, where Y is in  $256 \times 256$  resolution and the chrominance components U and V are in half resolution, is segmented. The original image is shown in Fig. 10(a). In the first experiment, U and V are projected to higher resolution by a 1:4 pixel

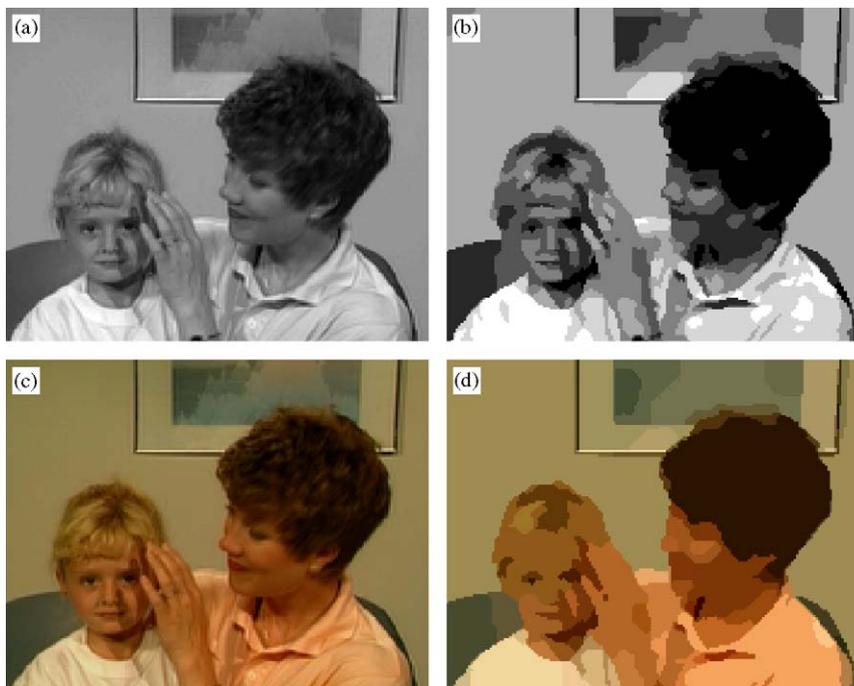


Fig. 9. Frame 34 of qcif size Mother and Daughter image sequence segmentation with  $k = 7, 2, 2$  clusters and  $\beta = 40$ : (a) original grey-level image; (b) regular grey-level single-resolution segmentation; (c) color image of Mother and Daughter where U and V are in half resolution; (d) proposed scalable segmentation.

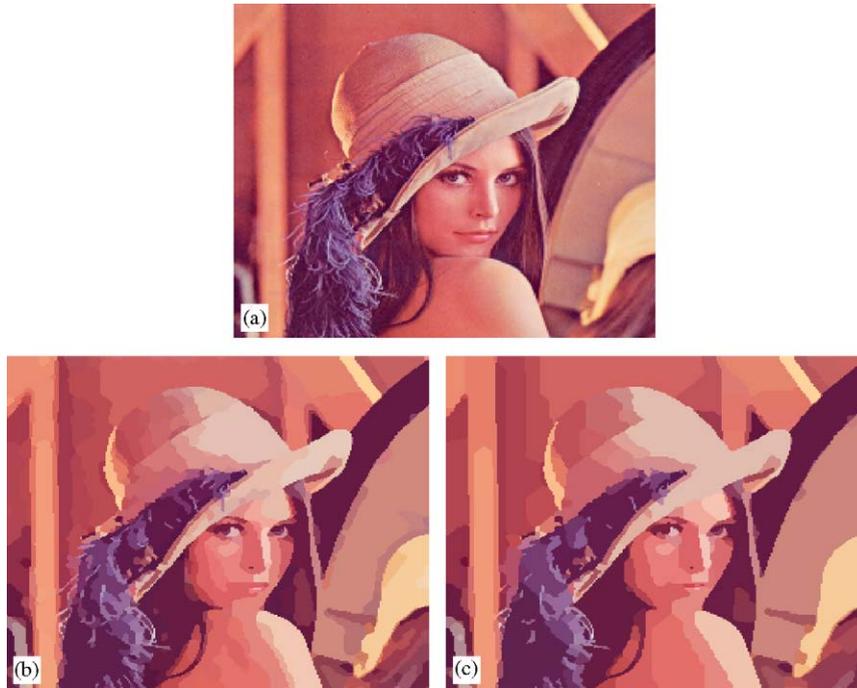


Fig. 10. Lena image segmentation at  $256 \times 256$  with  $k = 6, 4, 4$  clusters and  $\beta = 100$ : (a) original image; (b) regular single-resolution segmentation where U and V are projected to higher resolution; (c) proposed scalable segmentation where U and V are in half resolution.

transform and then single-resolution segmentation is performed [13]. The result is shown in Fig. 10(b). It can be seen that the top part of the hat is not well separated from the background. Finally, result for the proposed multiresolution scalable segmentation algorithm, which uses half resolution U and V, is shown in Fig. 10(c). This algorithm can separate all the foreground (Lena) regions from the background successfully. It is interesting to note that the single-resolution method divides the image into 578 regions, while the proposed scalable segmentation separates the image into 427 regions, which is a %26 reduction in the number of regions. This confirms that the proposed algorithm reduces the number of regions or over-segmentation compared to single-level segmentations while still separating the objects' regions from the background. Similarly, single-resolution segmentation in RGB space cannot separate the hat from the background and divides the image into 779 regions with similar parameters. Considering the over-segmentation and under-segmentation of respectively of single resolution segmentation and multiresolution segmentation algorithms, the failure to separate object's regions in single resolution will for sure lead to even bigger failure in multiresolution segmentation algorithm.

In the next example, frame 30 of the qcif sequence foreman is considered. The original image is in YUV format where Y is in full resolution but U and V are in half resolution. In Fig. 11(a) the original image can be seen. The image is segmented with the proposed scalable multiresolution segmentation. The result is compared with the other segmentation algorithms. To perform other algorithms, U and V of color components are again projected to full resolution by a 1:4 pixel transform and the regular single, multiresolution and the proposed scalable segmentation algorithms are performed. The initial segmentation estimation comes from  $k$ -means clustering for different channels and the number of classes were chosen as  $k = 10, 4, 2$  for the used YUV or RGB color channels, respectively. In the first experiment the components U and V are projected to the next higher resolution and then the proposed scalable and the regular multiresolution segmentation algorithms are performed. The results can be seen in Figs. 11(b) and (c). It is clear that in this example regular multiresolution segmentation cannot separate the foreground (foreman) from background regions. This is more pronounced in separating the left area of the hat from the background. Furthermore some other details such as the

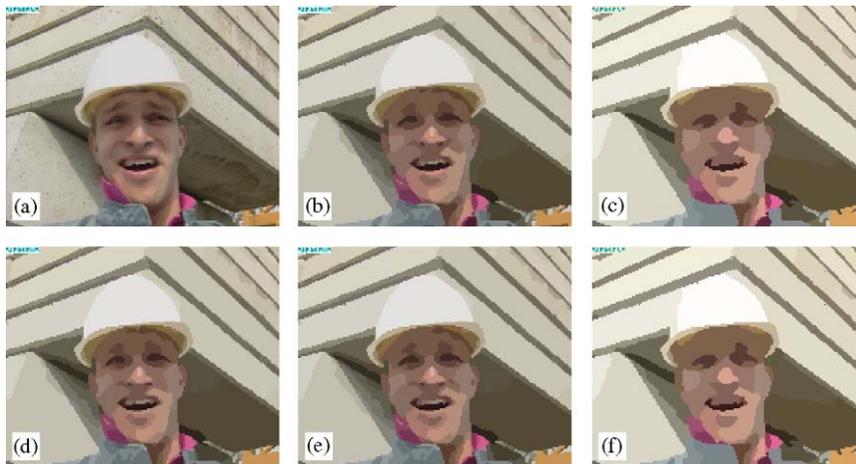


Fig. 11. Segmentation of frame 32 of qcif size foreman sequence with  $K = 10, 4, 2$  clusters at different channels: (a) original image; (b) scalable segmentation where U and V are projected to higher resolution; (c) regular multiresolution segmentation; (d) scalable segmentation where YUV are in full resolution; (e) scalable segmentation in YUV space where U and V are in half resolution; (f) scalable segmentation in RGB space where components are in full resolution.

left eyebrow has not been detected. Similarly, the scalable segmentation using projected U and V to higher resolution could not detect the corner of the hat.

The image with real full resolution size of U and V is segmented and will be considered as a ground truth for comparison in the following. The qcif size U and V components are taken from the available YUV, CIF size image sequence. Segmentation by the scalable segmentation which uses real full resolution qcif size U and V are in Fig. 11(c). Fig. 11(d) shows the segmentation with the proposed algorithm which uses full resolution Y and half resolution U and V components. As can be seen, the proposed algorithm separates the foreground regions from the background successfully, as the scalable algorithm with the full-resolution information does. As a statistical test, scalable segmentation using half resolution U and V (Algorithm A) and scalable segmentation using projected U and V (Algorithm B) are compared with the ground truth. The number of misclassified pixels in the proposed algorithm using half resolution U (Algorithm A) and V is about %30 of the ones of the algorithm which uses the projected U and V components in high resolution (Algorithm B). The numbers of misclassified pixels at different resolutions are shown in Table 5. In the Fig. 11(e) the segmented image in the RGB space using the full resolution information is shown. The right and top area of the hat are not separated well. To remedy the problem, we have to increase the number of

Table 5  
Misclassified pixels in Foreman image segmentation

Resolution	$72 \times 88$	$144 \times 176$	$288 \times 352$
Algorithm A	50	186	786
Algorithm B	123	511	2118
Improvement	%59	%64	%63

classes from 10, 4, 2, to 10, 10, 10 classes to separate the hat resulting in the increase of the number of regions which is over-segmentation. Increasing the number of regions will increase the computational complexity of the segmentation algorithm. The number of regions with  $K = 10, 4, 2$  is 279 for the proposed algorithm in YUV space and 337 for RGB space which increases to 739 regions for  $K = 10, 10, 10$  in RGB space.

As the last segmentation example, the  $256 \times 256$  Guitar image is segmented. The original grey-level image and the single resolution segmentation are shown in Figs. 12(a) and (b). The segmentation result shows that many meaningful regions are not well detected, and border fluctuations occur in some areas of the image. Figs. 12(c)–(e) show the multi-resolution segmentation results. Although multiresolution segmentation has less border fluctuations, the resulting under-segmentation means that more semantic regions are missed than in single resolution segmentation. Table 6 shows the number of regions for single and multiresolution segmentation algorithms at different resolutions.

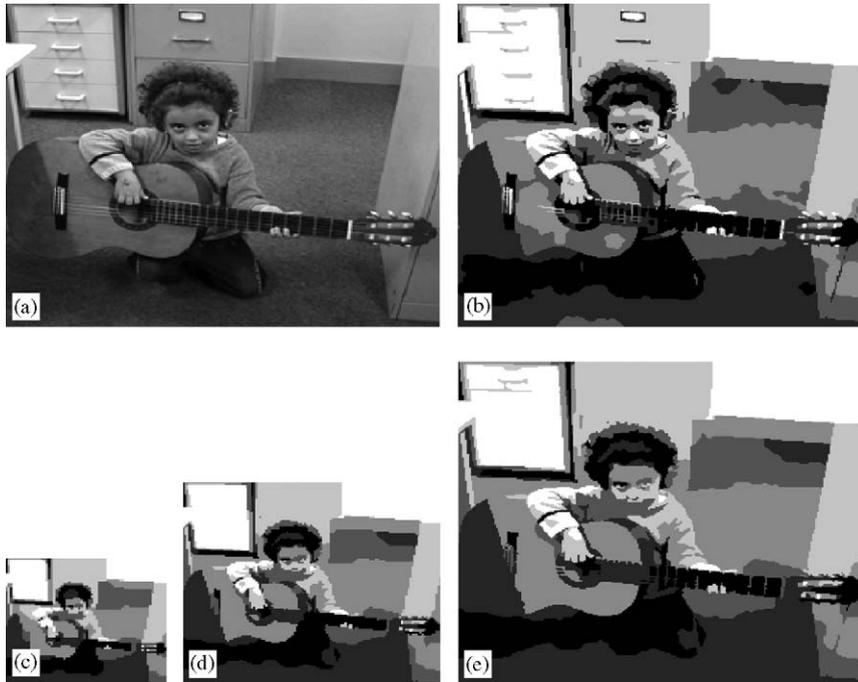


Fig. 12. Guitar segmentation: (a) the  $256 \times 256$  grey image; (b) single-resolution segmentation; (c) multiresolution  $64 \times 64$  segmentation; (d) multiresolution  $128 \times 128$  segmentation; (e) multiresolution  $256 \times 256$  segmentation.

Table 6  
Number of regions for segmentation of grey Guitar

Segmentation	$88 \times 72$	$144 \times 176$	$288 \times 352$
Uni-resolution seg.	82	189	447
Multiresolution seg.	82	98	157

To enhance the segmentation process, in a further experiment, color information has been used in the segmentation algorithm. Fig. 13(a) shows the original  $256 \times 256$  Guitar color image. The single-resolution segmentation of the color image in the YUV color space is shown in Fig. 13(b). 560 regions have been detected with many non-meaningful regions, which indicates over-segmentation. Border roughness decreases for many regions creating a better visual quality. Figs. 13(c)–(e) show the standard multiresolution segmentations of the color image at different resolutions. While border smoothness is increased, many meaningful regions have not been detected. This, therefore, over-corrects the over-segmentation of the single-resolution segmentation, resulting in under-segmentation. Many meaningful regions of the Guitar instrument

and filing cabinet have not been well detected and are mixed irreversibly with the background. The numbers of regions are shown in Table 7. Finally, the proposed scalable multiresolution segmentation with the smoothness constraint at three different resolutions is shown in Figs. 13(f)–(h). It can be seen that the most important and meaningful regions have been extracted and that the segmentation maps at different resolutions are similar. The borders are significantly smoother.

## 5. Conclusions

We have presented a multiresolution scalable grey-level/color image segmentation algorithm which extracts objects/regions with similar segmentation pattern at different resolutions, which is useful for object-based wavelet coding applications. In addition to scalability, a new quantitative criterion is added to the segmentation algorithm. This criterion, which is a smoothness function based on the pixel segmentation labels, represents the visual quality of the objects/regions at different resolutions. To reduce the down-sampling distortion in object-based wavelet transforms, different smoothness coefficients are considered for different

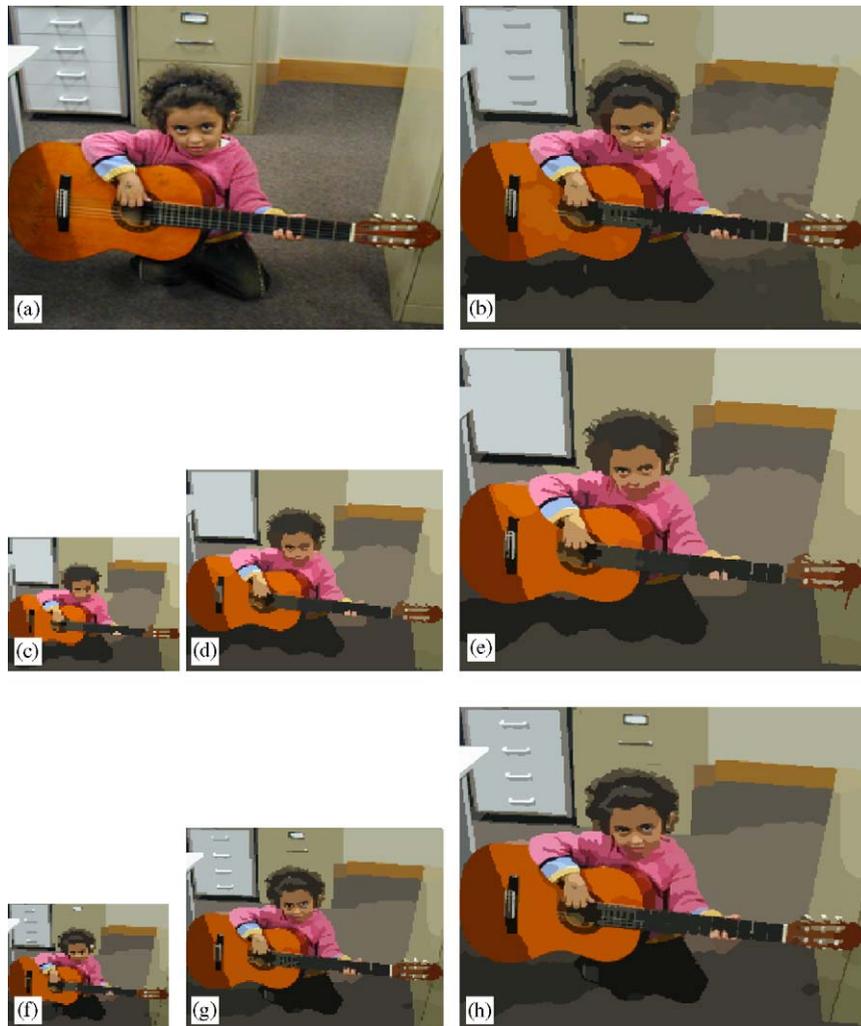


Fig. 13. Guitar segmentation: (a) the  $256 \times 256$  grey image of Guitar; (b) single-resolution segmentation; (c) multiresolution segmentation at  $64 \times 64$ ; (d) multiresolution segmentation at  $128 \times 128$ ; (f) scalable segmentation at  $64 \times 64$ ; (g) scalable segmentation at  $128 \times 128$ ; (h) scalable segmentation at  $256 \times 256$ .

Table 7  
Number of regions for segmentation of color Guitar

Segmentation	$88 \times 72$	$144 \times 176$	$288 \times 352$
Single resolution seg.	139	308	447
Multiresolution seg.	139	143	172
Scalable seg.	173	288	342

resolutions. The proposed multiscale analysis incorporate in the objective function of Bayesian segmentation, improves the sensitivity to grey-level variations while maintaining high performance in

noisy environments. The novel objective function gives flexibility to the proposed algorithm to segment YUV color images where Y is in full resolution but U and V are in half resolution.

## References

- [1] F. Pereira, T. Ebrahimi, The MPEG-4 Book, Prentice-Hall PTR, Englewood Cliffs, NJ, 2003.
- [2] B.S. Manjunath, P. Salembier, T. Sikora, Introduction to MPEG-7 Multimedia Content Description Interface, Wiley, New York, 2002.
- [3] H. Danyali, Highly scalable wavelet image and video coding for transmission over heterogeneous networks, Ph.D. Thesis,

- School of Electrical, Computer and Telecommunication Engineering, University of Wollongong, Wollongong, NSW, Australia, 2003.
- [4] T.N. Pappas, An adaptive clustering algorithm for image segmentation, *IEEE Trans. Image Process.* 40 (4) (April 1992) 901–914.
- [5] Y.A. Tolias, N.A. Kanlis, S.M. Panas, A hierarchical edge-stressing algorithm for adaptive image segmentation, in: *Electronics, Circuits, and Systems, 1996. ICECS '96, Proceedings of the Third IEEE International Conference*, vol. 1, pp. 199–202.
- [6] L. Zheng, J.C. Liu, A.K. Chan, W. Smith, Object-based image segmentation using dwt/rdwt multiresolution markov random field, in: *Acoustics, Speech, and Signal Processing, 1999, ICASSP '99. Proceedings, 1999 IEEE International Conference*, vol. 6, pp. 3485–3488.
- [7] X. Munoz, J. Marti, X. Cufi, J. Freixenet, Unsupervised active regions for multiresolution image segmentation, in: *Pattern Recognition, 2002. Proceedings, 16th International Conference*, vol. 2, pp. 905–908.
- [8] C.A. Bouman, M. Shapiro, A multiscale random field model for bayesian image segmentation, *IEEE Trans. Image Process.* 13 (1994) 162–177.
- [9] M.L. Comer, E.J. Delp, Segmentation of textured images using a multiresolution gaussian autoregressive model, *IEEE Trans. Image Process.* 8 (3) (1999) 408–420.
- [10] Z. Kato, J. Zerubia, M. Berthod, Unsupervised parallel image classification using Markovian models, *Pattern Recognition* 32 (1999).
- [11] C.T. Li, Multiresolution image segmentation integrating gibbs sampler and region merging algorithm, *Signal Process.* 83 (2002).
- [12] R. Wilson, C.T. Li, A class of discrete multiresolution random fields and its application to image segmentation, *IEEE Trans. Pattern Anal. Machine Intell.* 25 (2003).
- [13] M.M. Chang, M.I. Sezan, A.M. Tekalp, Adaptive bayesian segmentation of color images, *J. Electronic Imaging* 3 (4) (1994) 404–414.
- [14] J. Luo, R.T. Gray, H.C. Lee, Towards physics-based segmentation of photographic color images, in: *Image Processing, 1997. Proceedings, International Conference, 1997*, vol. 3, pp. 58–61.
- [15] J. Gao, J. Zhang, M.G. Fleming, A novel multiresolution color image segmentation technique and its application to dermatoscopic image segmentation, in: *Image Processing, 2000. Proceedings, 2000 International Conference, 1997*, vol. 3, pp. 408–411.
- [16] G. Borgefors, G. Ramella, G. Sanniti di Baja, S. Svenson, On the multiscale representation of 2d and 3d shapes, *Graphical Models and Image Processing* 61 (1) (1999) 44–62.
- [17] A. Mertins, S. Singh, Embedded wavelet coding of arbitrary shaped objects, in: *Proceedings of SPIE 4076-VCIP'00, Pert, Australia, June 2000*, pp. 357–367.
- [18] S.Z. Li, *Markov Random Field Modeling in Image Analysis*, second ed., Springer, Tokyo, Japan, 2001.
- [19] A. Murat Tekalp, *Digital Video Processing*, second ed., Prentice-Hall, USA, 1995.
- [20] J. Gao, Object shape delineation during tracking process, in: *Image Processing, 2003. Proceedings, 2003 International Conference, 2003*, vol. 3, pp. III–969–72 vol. 2.
- [21] M. Bertero, T.A. Poggio, V. Torre, Ill-posed problems in early vision, *Proc. IEEE* 76 (8) (1988) 869–889.
- [22] M. Kass, A. Witkin, D. Terzopoulos, Snakes: active contour models, *Int. J. Comput. Vis.* 1 (4) (1987) 321–331.
- [23] J. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Machine Intell.* 8 (6) (1986) 679–998.
- [24] F. Marques, J. Llach, Tracking of generic objects for video object generation, in: *International Conference on Image Processing (ICIP), vol. 3, 1998*, pp. 628–632.
- [25] J. Besag, On the statistical analysis of lattice systems, *J. Roy. Statist. Soc. Ser. B* 48 (3) (1986) 259–279.
- [26] T. Meier, K.N. Ngan, G. Grebbin, A robust markovian segmentation based on highest confidence first (hcf), in: *IEEE International Conference on Image Processing, vol. 1, 1997*, pp. 216–219.
- [27] S. Ray, R. Turi, Determination of number of clusters in  $k$ -means clustering and application in colour image segmentation, in: *Proceedings of the Fourth International Conference on Advances in Pattern Recognition and Digital Techniques, 1999*, pp. 137–143.
- [28] C. Rosenberger, K. Chehdi, Unsupervised clustering method with optimal estimation of the number of clusters: application to image segmentation, in: *Proceedings of 15th International Conference on Pattern Recognition, vol. 1, 2000*, pp. 656–659.
- [29] A. Said, W.A. Pearlman, A new fast and efficient image codec based on set partitioning in hierarchical trees, *IEEE T. Circ. Syst. Vid.* 6 (1996) 243–250.
- [30] H. Danyali, A. Mertins, Highly scalable image compression based on split for network applications, in: *Proceedings of IEEE International Conference on Image Processing, Rochester, NY, USA, September 2002*, pp. 217–220.