We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS BOOK CITATION INDEX INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

26

Pixon-Based Image Segmentation

Hamid Hassanpour[1], Hadi Yousefian[2] and Amin Zehtabian[3]
[1]*Shahrood University of Technology*
[2]*Iran University of Science & Technology (IUST)*
[3]*Noshirvani University of Technology (NIT)*
Iran

1. Introduction

The pixon concept was introduced by Pina and Puetter in 1993. The pixon they introduced was a set of disjoint regions with constant shapes and variable sizes. Their pixon definition scheme was a local convolution between a kernel function and a pseudo image. The drawback of this scheme was that after selecting the kernel function, the shape of the pixons could not vary. Yang and Jiang presented a new pixon definition scheme, whose shape and size can vary simultaneously. They also used the anisotropic diffusion equation to form the pixons and finally they have combined the pixon concept and MRF for segmentation of the images. Recently, another well-behaved pixon-based image representation is proposed [Lei Lin et al., 2008]. In their presented scheme the pixons combined with their attributes and adjacencies construct a graph, which represents the observed image. They used a Fast QuadTree Combination (FQTC) algorithm to extract the good pixon-representation. These techniques integrated into MRF model. The main disadvantage of MRF-based methods is that in these algorithms the minimization problem of objective function is very time consuming. The most novel method which uses pixon concept to segment the images is introduced by Hassanpour et al. In this method, first a pre-processing step is performed which applies the wavelet thresholding technique. This step is suitable for image smoothing due to the noise reduction property of wavelet thresholding. To avoid over-smoothed problem, the value of the threshold must be assigned properly. Then, the pixon-based algorithm is used to form and extract the pixons. Finally, the Fuzzy C-Means (FCM) algorithm is applied to segment the image. The advantage of using pixons is that after forming the pixons the decision level changes from pixels to pixons and this decreases the computational time, because of the fewer number of pixons compared to number of pixels. This is the key aspect of pixon-based algorithms in image segmentation.

2. Pixon-based methods

2.1 Traditional Pixon-Based method (TPB)

The TPB method is known as one of the simplest pixon-based approaches applied for image segmentation. The method is mainly composed of two following steps: (1) form the pixons, and (2) segment the image.

2.1.1 Description of pixon model

In any pixon definition scheme, the ability to control the number of degrees of freedom used to model the image is the key aspect. In other word, the pixon definition scheme should yield an optimum scale description of the observed image. The pixon definition scheme which is used in this method can be described as follows:

$$IP = \bigcup_{j=1}^{m} P_j \tag{1}$$

where IP is the pixon-based image model; m is the number of pixons; P_j is a given pixon, which is made up of a set of connected pixels, a single pixel or even a sub-pixel. The mean value of the connected pixels making up of the pixon is defined as the pixon intensity. Both the shape and size of each pixon vary according to the observed image. After the pixon-based image model is defined, the image segmentation problem is transformed into a problem of labeling pixons. The procedure to determine the set of pixons, i. e. their shape and size, can be divided into three steps: 1) obtain a pseudo image, which has at least the same resolution as the observed image; 2) use an anisotropic diffusion filter to form the pixons; and 3) use a simple hierarchical clustering algorithm to extract the pixons.

Obtaining the Pseudo Image; The pseudo image is a basic image to form the pixons and to obtain a segmented image, which is derived from the observed image. Suppose the dimension of the observed image is $D_M \times D_N$, then the dimension of the pseudo image can be $lD_M \times lD_N$, where $l = 2^n$. When n = 0, the pseudo image is the observed image itself. When $n \geq 1$, the pseudo image can be obtained by the following iterative process

$$i_{2^n}(i,j) = \begin{cases} I_{2^{n-1}}\left(\frac{i}{2},\frac{j}{2}\right) & i,j \text{ is even} \\ \frac{1}{2}\left[I_{2^{n-1}}\left(\frac{i}{2},\frac{j-1}{2}\right) + I_{2^{n-1}}\left(\frac{i}{2},\frac{j+1}{2}\right)\right] & i \text{ is even}, j \text{ is odd} \\ \frac{1}{2}\left[I_{2^{n-1}}\left(\frac{i-1}{2},\frac{j}{2}\right) + I_{2^{n-1}}\left(\frac{i+1}{2},\frac{j}{2}\right)\right] & i \text{ is odd}, j \text{ is even} \\ \frac{1}{4}\left[\left(\frac{i-1}{2},\frac{j}{2}\right) + I_{2^{n-1}}\left(\frac{i-1}{2},\frac{j}{2}\right) + I_{2^{n-1}}\left(\frac{i-1}{2I_{2^{n-1}}},\frac{j}{2}\right) + I_{2^{n-1}}\left(\frac{i-1}{2},\frac{j}{2}\right)\right] & i,j \text{ are odd} \end{cases} \tag{2}$$

Where $i = 0, 1, \ldots, 2^M \times D_M - 1$ and $j = 0, 1, \ldots, 2^N \times D_N - 1$.

In the above iterative process, I_1 corresponds to the observed image. The essence of the process is increasing the resolution through interpolation to describe the image parts, which have a lot of details.

Parameter n is of great importance. If $n \geq 1$, then the resolution of the pseudo image is larger than the original image and the finally pixons formed are probable to be a sub-pixel. So, it determines the smallest size of the pixons. In the image parts, where the intensities of nearing pixels are similar, which means having little information, the intensity of newly inserted pixels will be similar with the intensities of the pixels in the observed image, from which the new pixels are obtained through interpolation.

So there is a little difference whether the pixons are derived from the original observed image or interpolated pseudo image. However, in the image parts, where have a lot of

details, it will be better to derive the pixons from the interpolated pseudo image than from the original observed image. So, it is probable that a pixon is a sub-pixel to fully model the corresponding image parts. Therefore, if the image has many details, it should be large, otherwise it should be small. In current implementation, we let $n = 0$.

Formulation of Pixon; To form the pixons based on the pseudo image, let us consider the following anisotropic diffusion equation [Perona & Malik, 1990]:

$$\frac{\partial I(x,y,t)}{\partial t} = C(x,y,t)\left(\frac{\partial^2 I(x,y,t)}{\partial x^2} + \frac{\partial^2 I(x,y,t)}{\partial y^2}\right) + \frac{\partial I(x,y,t)}{\partial x} \cdot \frac{\partial C(x,y,t)}{\partial x} + \frac{\partial I(x,y,t)}{\partial y} \cdot \frac{\partial C(x,y,t)}{\partial y} \tag{3}$$

where C(x,y,t)is the diffusion coefficient, which controls the diffusion strength.

The partial differential equation is used to model the heat diffusion process. In regions with a large diffusion coefficient, the temperature tends to be uniform. While temperature differences will be retained in regions with small diffusion coefficients. We can view the pseudo image intensity as the temperature of the temperature field and the transformation of the gradient as the diffusion coefficient. The transformation function is

$$c(x,y,t) = \frac{1}{\left(1 + \frac{|\nabla I|^2}{k^2}\right)} \tag{4}$$

where K is a constant.

To be convenient, the solution of the diffusion equation is called solution image. In the solution image, the intensity of the regions having less information (having fewer edges) will tend to be uniform and vice versa. So, the "regions" having similar intensity in the solution image can be regarded as the pixons in our image model. The diffusion equation can be approximately solved by the following discrete formulation:

$$I(x,y,t+\Delta t) = I(x,y,t) + \Delta t(d_n c_n + d_s c_s + d_e c_e + d_w c_w) \tag{5}$$

where

$$\begin{aligned}
d_n &= I(x,y-1,t) - I(x,y,t) \quad c_n = \frac{1}{1+\left(\frac{d_n}{k}\right)^2} \\
d_s &= I(x,y+1,t) - I(x,y,t) \quad c_s = \frac{1}{1+\left(\frac{d_s}{k}\right)^2} \\
d_e &= I(x+1,y,t) - I(x,y,t) \quad c_e = \frac{1}{1+\left(\frac{d_e}{k}\right)^2} \\
d_w &= I(x-1,y,t) - I(x,y,t) \quad c_w = \frac{1}{1+\left(\frac{d_w}{k}\right)^2}
\end{aligned} \tag{5}$$

To ensure the convergence of the above iteration process, the parameter Δt should not be too large (here, $\Delta t = 0.25$). Larger values of K increase the pixon size. To describe the details of the image, K could not be too large (here $K = 5$)

Extraction of the Pixons; After forming the pixons according to the pseudo image, a segmentation method is applied based on hierarchical clustering to extract them. For this purpose, initially each pixel represents a cluster. Then the clusters are merged according to their intensities and made greater pixons. The mean value of the connected pixels making up of the pixon is defined as the pixon intensity. Both the shape and size of each pixon can vary according to the observed image.

To stop the algorithm, a threshold value, T, is assigned and the mergence process iterates until the difference between intensities of two adjacent pixons would be smaller than the threshold value (here, T = 10).

The pixon-based image model is represented by a graph structure G = (Q,E) , where Q is the finite set of vertices of the graph and E is the set of edges of the graph (Figure 1).

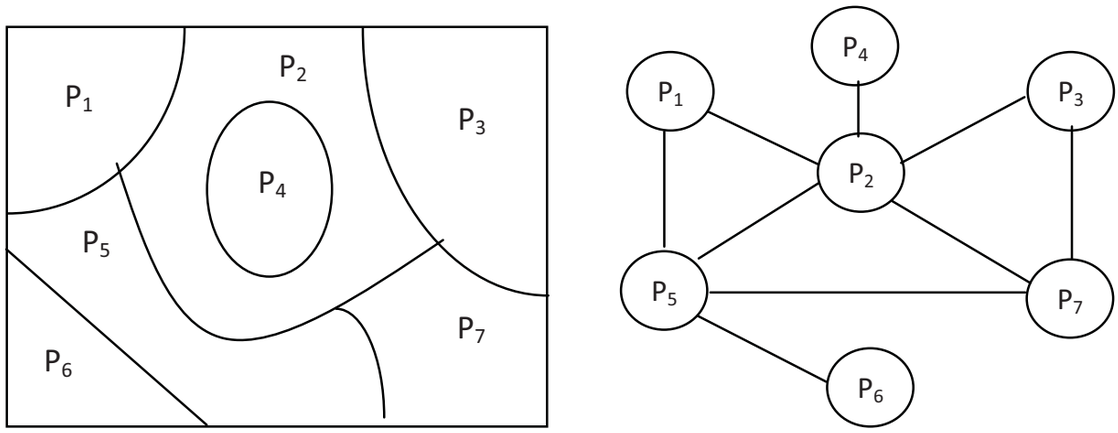


Fig. 1. (a) Pixon model of image, and (b) the corresponding graph structure

After the pixon-based image model is defined, the image segmentation problem is transformed into a problem of labeling pixons.

While the pixons are extracted, the image is divided into a set of disjoint regions. The extraction of pixons can be considered as a primary segmentation. In TPB method, to obtain the final segmented image, the combination of pixons is continued until the end condition of process occurs. This condition is the number of segments in final segmentation purpose.

2.2 MPB method

Pixon-based image segmentation using Markov Random Field (MRF) model is presented by (Lei Lin, et al 2008). In this method, first an image is expressed as a pixon-based model. As we said before, pixons are a set of disjoint regions with variable shape and size. These pixons are combined with their attributes and adjacencies construct a graph which represents the observed images. Then using this pixon-representation, a Markov Random Field (MRF) model is presented to segment the images.

In current procedure, a set of significant attributes of pixons and edges are introduced into the pixon-representation. These attributes are integrated into the MRF model and the

Bayesian framework to obtain a weighted pixon-based algorithm. Also, a Fast Quad Tree Combination (FQTC) algorithm is used to extract the good pixon representation.

2.2.1 Definition of pixon representation

Definition 1. Let $X = \{X_i\}_{i=1}^{M}$ be the set of all the image pixels. A subset of X is a pixon if and only if all the pixels in it are connected. A pixon is then denoted by $P_i = \{X_{ij}\}_{j=1}^{n_i}$.

An attribute vector of the pixon is extracted from the observed image

$$\bar{P}_i = (n_i, b_i, max_i, min_i, \mu_i, \sigma_i^2) \tag{6}$$

where n_i is the number of pixels in P_i, b_i is the perimeter of P_i, namely the length of the boundary between P_i and the other part of the observed image, max_i, min_i, μ_i and σ_i^2 are the maximum, minimum, mean and variance of the observed image intensities in P_i, respectively. Let $I(x_{ij})$ denotes the image intensity on the pixel x_{ij}. The attributes of the pixon intensity can be obtained by

$$\begin{aligned} max_i &= \max\left(I(X_{ij}) | X_{ij} \epsilon P_i\right) \\ min_i &= \min\left(I(X_{ij}) | X_{ij} \epsilon P_i\right) \\ \mu_i &= \sum_{j=1}^{n_i} I(X_{ij}) / n_i \\ \sigma_i &= \sum_{j=1}^{n_i} (I(X_{ij}))^2 / n_i - \mu_i^2 \end{aligned} \tag{7}$$

Definition 2. A set of pixons, $= \{P_i\}_{i=1}^{N}$, is a pixon-representation if and only if

$$\begin{aligned} &P_i \neq \varnothing \\ &P_i \bigcap P_j = \varnothing \text{ , if } i \neq j \\ &\bigcup_{i=1}^{N} P_i = X \end{aligned} \tag{8}$$

The above definition shows that the pixon-representation segments the image into a set of disjoint regions. A set of edges, E, can be acquired from these regions,

$$E = \left\{E_{ij} \mid P_i, P_j \in P \text{ and } P_i, P_j \text{ are adjacent}\right\} \tag{9}$$

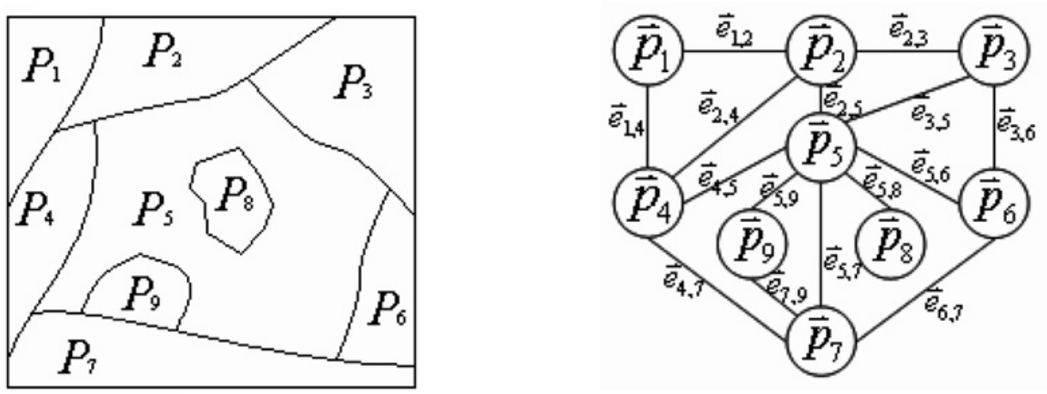
where P_i and P_j are adjacent if $\exists X_{ik} \in P_i$ and $X_{ji} \in P_j$, which are neighboring pixels to each other in the image.

The strength of an edge can be defined as the length of the boundary between the two adjacent pixons, which is denoted by b_{ij}, so $b_i = \sum_j b_{ij}$. An attribute vector, $\bar{e}_{ij}$, is used to denote all the attributes of an edge.

The pixons and edges, combined with their attribute vectors, construct a graph, $G=\{P, E\}$, which represents the observed image, as shown in Fig. 2.

2.2.2 Shortest pixon-representation with respect to a discriminant

There are two trivial pixon-representations, $P_0=\{X\}$ and $P_1=\{\{x_i\}\,|\,x_i\in X\}$. The former takes all the image pixels as one pixon; the latter takes each pixel as a pixon, which is a lossless representation. In order to represent the image using as few pixons as possible while limiting the representation error, the shortest pixon-representation with respect to a discriminant is defined.



(a) (b)

Fig. 2. An example of Pixon-Representation. (a) The Pixon map, in which the boundaries between adjacent Pixons are shown; and (b) The corresponding graph, which combines the attribute vectors of Pixons and edges to represent the observed image.

Definition 3. A function $f(p)\geq 0$ of pixons is a pixon error function if and only if

$$\begin{aligned} &f(p)=0 \text{ ,if } P=\{x_i\}\text{ ,}\\ &f(P_i)\geq f(P_j)\text{ ,if } P_i\supseteq P_j \end{aligned} \tag{10}$$

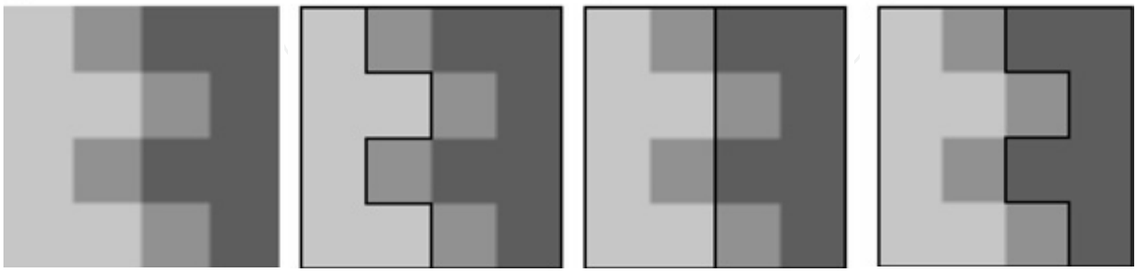
Definition 4. For a given pixon error function, $f(.)$, and a non-negative constant, T, the inequality, $f(.)\leq T$, defines a pixon discriminant.

Definition 5. A pixon-representation is called the shortest pixon representation with respect to a given discriminant, $f(.)\leq T$, if its number of pixons is least among all the pixon-representation satisfying $\forall P_i\in P,\ f(P_i)\leq T$.

In general, using the pixon attribute vector to describe the region of the observed image will loss some information, so a pixon error function is used to denote the error between the pixon and the region of the observed image. In this method error function is defined as $f(P_i)=\max_i-\min_i$. With a given discriminant $f(.)\leq T$ the shortest pixon-representation use the least number of pixons to represent the image, so we consider it the best pixon-representation whose pixons' errors do not exceed the threshold, T.

2.2.3 Extraction of pixon-representation

The shortest pixon-representation with respect to a discriminant is not unique, as shown in Fig. 3. And it is hard to extract the shortest one from a large and complex image. In this section, an approach to extract a GOOD pixon-representation is presented, which combines the adjacent pixons of the lossless pixon-representation, $P_1 = \{\{x_i\} \mid x_i \in X\}$, iteratively, until no pixons can be combined considering the given discriminant. The obtained good pixon-representation is dependent on the order of combination besides the discriminant.



(a) (b) (c) (d)

Fig. 3. The non-uniqueness of the Shortest Pixon-Representation. (a) Observed image, whose pixel intensities are among 100, 150, and 200; (b), (c) and (d) are three of its shortest Pixon-Representations when $f(P_i) = \max_i - \min_i \le 50$ is given as a discriminant. The black lines overlapping on the image are the boundaries of Pixons.

2.2.3.1 Combination of adjacent pixons

The adjacent pixons in a pixon-representation, $G = \{P, E\}$, can be combined to form a new pixon, denoted by $P_{new} = P_i \oplus P_j$, whose attribute vector, $\bar{P}_{new}$, can be obtained from $\bar{P}_i$ and $\bar{P}_j$,

$$\begin{aligned} n_{new} &= n_i + n_j \\ b_{new} &= b_i + b_j - 2b_{ij} \\ max_{new} &= \max(max_i, max_j) \\ min_{new} &= \max(min_i, min_j) \\ \mu_{new} &= (n_i\mu_i + n_j\mu_j)/n_{new} \\ \sigma^2{}_{new} &= \left[n_i(\sigma_i^2 + \mu_i^2) + n_j(\sigma_j^2 + \mu_j^2)/n_{new} - \mu_{new}^2 \right] \end{aligned} \tag{11}$$

where b_{ij} is the edge strength, i.e. the length of the boundary between P_i and P_j .
It can be proved that $P - \{P_i, P_j\} + \{P_{new}\}$ is still a pixon-representation. And the edge set of the new pixon-representation can be obtained from E by combining the edges connecting the same two pixons after the pixon combination.

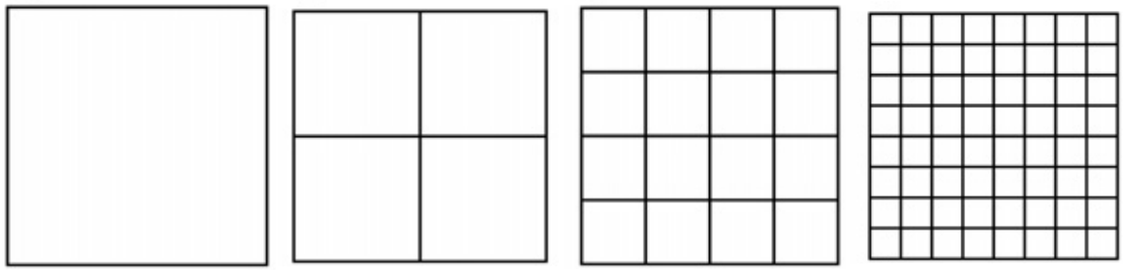
2.2.3.2 Combination-based extraction of pixon-representation

Given a discriminant, $f(.) \le T$, the edge error function is defined as $f_E(E_{ij}) = f(P_i \oplus P_j)$. Since $P_1 = \{\{x_i\} \mid x_i \in X\}$ satisfies all the discriminants, the shortest pixon-representation

with respect to $f(.) \le T$ can be extracted by combining the pixons of P_1, the lossless representation, until all error function values of the edges are larger than T.
In fact, the pixon-representation obtained by combination scheme may not always be the shortest, which is dependent on the order of combinations. However, it is a substitute to the shortest, for the number of pixons has been sharply cut down.

2.2.3.3 Fast Quad Tree Combination algorithm

A fast Quad Tree combination algorithm is used to extract the shortest pixon-representation here. Firstly, a QuadTree-based multi-resolution pixon-representation is constructed, as shown in Fig. 4.



(a) (b) (c) (d)

Fig. 4. The QuadTree-based multi-resolution Pixon-Representation. (a) Coarsest scale Pixon-Representation which uses the whole image as one Pixon; (b), (c), (d) is the followed scale pixon-Representation, which are obtained by subdividing each square of the coarser scale into four equal squares. The square in the finest scale only includes one pixel.

Then a initial pixon-representation with respect to $f(.) \le T_{qt}$, $T_{qt} \in [0, T]$, is extracted by coarse-to-fine selecting a set of disjoint squares from the multi-resolution pixon-representation, which satisfy $f(.) \le T_{qt}$. Finally, the pixons connected by the edge with the minimal edge error are combined iteratively, until the minimal edge error is larger than T.
If the image region is not a square whose edge length is the power of 2, the multi-resolution pixon-representation can be constructed as follows. Firstly, the image is put into a large enough square like (a) in Fig. 4. For each scale, the pixon is then defined as the set of pixels falling into a square of this scale; the squares including no pixel are ignored. An example using the fast Quad Tree combination algorithm is given in Fig. 5, where the error function is defined as

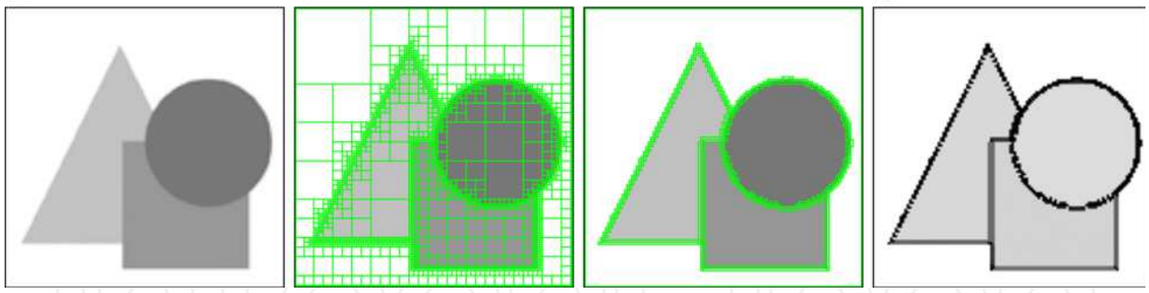
$$f(P_i) = max_i - min_i \tag{12}$$

2.2.4 Image segmentation based on pixon-representation

In this method, a Markov random field model-based image segmentation approach under Bayesian framework is used based on pixon-representation. The noise model of the Bayesian framework in this approach is based on the pixel intensity.

2.2.4.1 Bayesian framework

Let I be the observed image and S be the segmented image. In the Bayesian segmentation framework, the segmented image is obtained by maximizing the posterior probability,



(a) (b) (c) (d)

Fig. 5. The fast QuadTree Combination Algorithm. (a) Observed image (13689 Pixels); (b) Initial Pixon-Representation (2115 Pixons). (c) Final Pixon-Representation (493 Pixons) after iterative Pixon combination; (d) The Pixon size map of the final Pixon-Representation, where the image intensity denotes the local Pixon size. The green lines in (b) and (c) are the boundaries between adjacent Pixons

$$S^* = \arg\max_{s} P(S|I) \tag{13}$$

where

$$P(S|I) \propto (PI|S)P(S). \tag{14}$$

We assume $I = S + N$, where N is independent Gaussian white noise. Then the conditional probability is

$$P(I|S) = \prod_{k=1}^{K} \prod_{x_i \in \Gamma_K} \frac{1}{\sqrt{2\pi}\sigma_K} \exp\left(-\frac{(I(x_i) - u_k)^2}{2\sigma_K{}^2}\right) \tag{15}$$

where K is the number of classes, Γ_K is the set of pixels segmented into the Kth class, and u_k is the intensity mean of pixels in Γ_K. Let $G = \{P, E\}$ be a pixon-representation of I. Since the characteristics of pixels in each pixon are similar, we assume that the pixels in one pixon will be segmented into the same class. So using (7) and (15), we get

$$\begin{aligned} P(I|S) &= \prod_{k=1}^{K} \prod_{P_i \subset \Gamma_K} \prod_{x_{ij} \in P_i} \frac{1}{\sqrt{2\pi}\sigma_K} \exp\left(-\frac{\left(I(x_{ij}) - u_k\right)^2}{2\sigma_K{}^2}\right) \\ &= \prod_{k=1}^{K} \prod_{P_i \subset \Gamma_K} (2\pi)^{-n_i/2} \sigma_K{}^{-n_i} \exp\left(-\frac{n_i[(\mu_i - u_K)^2 + \sigma_i{}^2]}{2\sigma_K{}^2}\right) \end{aligned} \tag{16}$$

The computation of $P(I|S)$ is simplified since the number of pixons is far less than that of pixels. $P(S)$ is the prior probability. In this method, the MRF model based on the pixon-representation is adopted to define the prior probability distribution as follows.

2.2.4.2 MRF model based on pixon-representation

A neighborhood system of the graph, $G = \{P, E\}$, is defined as

$$N(P) = \{N(P_i) \mid P_i \in P\} \tag{17}$$

where

$$N(P_i) = \{P_j \mid \exists e_{ij} \in E\}, 1 \le i \le N \tag{18}$$

is the neighborhood of each pixon.

Let $\Lambda = \{\lambda_1, \ldots, \lambda_K\}$ be the set of possible labels denoting the classes in the segmented image and $L = \{l_1, \ldots, l_K\}$ be a family of random variables where $l_1 \in \Lambda$ denotes the label of ith pixon and N is the number of pixons. The segmented image S can then be described by the event, $L = \omega$, since we assume that the pixels in one pixon will be segmented into the same class.

Let Ω be the set of all possible configurations, $\Omega = \{\omega = (\omega_1, \ldots, \omega_N) \mid \omega_i \in \Lambda\}$. L is a MRF with respect to the neighborhood, $N(P)$, if

$$P(L = \omega) > 0, \forall \omega \in \Omega \tag{19}$$

$$P(l_i = \omega_i | l_j = \omega_j, P_i \ne P_j) = P(l_i = \omega_i | l_j = \omega_j, P_j \in N(P_i)), \quad \forall P_i \in P \text{ and } \omega \in \Omega \tag{20}$$

where $P(.)$ and $P(.|.)$ are the joint and conditional probability density functions, respectively.

The configurations of MRF obey a Gibbs distribution [Hammersley & Clifford, 1971]

$$P(\omega) = 1 / Z.\exp(-U(\omega) / T) \tag{21}$$

where Z is a normalizing constant and T is a constant called temperature. $U(\omega)$ is the energy function, which is a sum of clique potentials $V_c(\omega)$ on all possible cliques, i.e.

$$U(\omega) = \sum_{c \in C} V_c(\omega) \tag{22}$$

In this method, the set of cliques is defined as

$$C = \{c_i \mid c_i = \{P_i\} \cup N(P_i), P_i \in P\} \tag{23}$$

where each pixon in $G = \{P, E\}$ defines one clique. And the clique potential is defined by

$$V_{c_i}(\omega) = w_c(\bar{P}_i) \sum_{P_{ij} \in N(P_i)} w_e(b_{ij}, b_i, b_j) w_p(\bar{P}_i, \bar{P}_j) \eta_{ij} \tag{24}$$

where η_{ij} is a binary variable which has the value 1 if P_i and P_j have the same label and the value 0 otherwise; $w_c(\bar{P}_i) = n_i$ is the clique weight; $w_e(b_{ij}, b_i, b_j) = b_{ij} / b_i$ is the normalized edge weight; and $w_p(\bar{P}_i, \bar{P}_j) = 1/|\mu_i - \mu_j|$ is the pixon distance weight that denotes the difference of image characteristics between two pixons.

In all, the prior probability is defined as

$$P(S) = P(\omega) = \frac{1}{Z}\exp(-\frac{1}{T}\sum_{P_i \in P} n_i \sum_{P_{ij} \in N(P_i)} \frac{b_{ij}}{b_i}\frac{\eta_{ij}}{|\mu_i - \mu_j|}) \tag{25}$$

2.2.4.3 Optimization

From (13) and (14), the optimal segmented image can be written as

$$S^* = \arg\min_s(-\ln P(I|S) - \ln P(S)). \tag{26}$$

Using (16) and (25), the objective function is then obtained,

$$F(S) = F(\omega) = \sum_{k=1}^{K}\sum_{P_i \subseteq \Gamma_K} [n_i(\frac{[(\mu_i - u_K)^2 + \sigma_i^2]}{2\sigma_K^2} + \ln\sigma_k) + \alpha n_i \sum_{P_{ij} \in N(P_i)} \frac{b_{ij}}{b_i}\frac{\eta_{ij}}{|\mu_i - \mu_j|}] \tag{27}$$

where $\alpha = 1/T$ is a weight of MRF model, which denotes the tradeoff between the fidelity to the observed image and the smoothness of the segmented image. The constant term has been removed from the objective function.

The class number K and the weight α are given before optimization. The initial segmented image is obtained using Fuzzy C-Means (FCM) clustering, and the initial parameters of each class are estimated from the initial segmented image, i.e. the means u_k and variances σ_k. Then the threshold T is computed, the value of T should not be too large, otherwise the pixon will contain many pixels which actually belong to two different classes. So we using follow empirical function:

$$T = \min_{0<i,k\le K, i\ne j}(|u_i - u_j| - \sigma_i - \sigma_j) \tag{28}$$

Finally, the segmented image and the parameters are optimized, simultaneously.

Let $F(\omega, I_{i,new})$ denote the objective function value when the ith label of ω is changed into $I_{i,new}$ and $\Delta F(\omega, I_{i,new})$ denote $F(\omega, I_{i,new}) - F(\omega)$. The optimization is described as follows

1. Initialize the number of classes K; the total number of iteration NUM; $u_1, \ldots, u_k$ and $\sigma_1, \ldots, \sigma_k$ according to an initial segmentation, which is obtained using FCM method; compute the threshold T; and the iteration index $j = 0$;
2. Extraction of pixon-representation, then initialize the pixon-based image model: assign a label λ_k to each pixon P, which minimizes the expression $|\mu_P - u_{k_l}|$
3. Find the best label for each pixon, $l_{i,best}, 1 \le i \le N$, which minimizes $\Delta F(\omega, I_{i,new})$.
4. Find the $\Delta F(\omega, I_{min,best})$, satisfying $\Delta F(\omega, I_{min,best}) \le \Delta F(\omega, I_{i,best}), 1 \le i \le N$
5. If $\Delta F(\omega, I_{min,best}) < 0$ and $j < NUM$, go to step 4, otherwise stop iteration.
6. Update the best label of each pixon and re-estimate u_k, σ_k using new ω.
7. $j = j + 1$, Go to step 3.

In fact, $\Delta F(\omega, I_{i,best})$ can be calculated using the correlative terms with the ith label in $F(x)$, i.e.

$$F_i(\omega, I_{i,new}) = n_i \left(\frac{(\mu_i - u_{l_{i,new}})^2 + \sigma_i^2}{2\sigma_{l_{i,new}}^2} + \ln \sigma_{l_{i,new}} \right) + \alpha \sum_{P_{ij} \in N(P_i)} \left(\frac{n_i b_{ij}}{b_i} + \frac{n_j b_{ij}}{b_j} \right) \frac{\eta_{ij,new}}{|\mu_i - \mu_j|} \quad (29)$$

2.3 WPB method

Pixon-based approach using wavelet thresholding is a recently developed image segmentation method [Hassanpour et al, 2009]. In this method, a wavelet thresholding technique is initially applied on the image to reduce noise and to slightly smooth the image. This technique causes an image not to be oversegmented when the pixon-based method is used. Indeed, the wavelet thresholding, as a pre-processing step, eliminates the unnecessary details of the image and results in a fewer pixon number, faster performance and more robustness against unwanted environmental noises. The image is then considered as a pixonal model with a new structure. The obtained image is segmented using the hierarchical clustering method (Fuzzy C-Means algorithm).

2.3.1 Pre-Processing step

As mentioned above, the wavelet thresholding technique is used as a pre-processing step in order to smooth the image. For this purpose, by choosing an optimal wavelet level and an appropriate mother wavelet, the image is decomposed into different channels, namely low-low, low-high, high-low and high-high (LL, LH, HL, HH respectively) channels and their coefficients are extracted in each level. The decomposition process can be recursively applied to the low frequency channel (LL) to generate decomposition at the next level. The suitable threshold is achieved using one of the different thresholding methods and then details coefficients cut with this threshold. Then, inverse wavelet transform is performed and smoothed image is reconstructed.

2.3.1.1 Wavelet thresholding technique

Thresholding is a simple non-linear technique which operates on the wavelet coefficients. In this technique, each coefficient is cut by comparing to a value as the threshold. The coefficients which are smaller than the threshold are set to zero and the others are kept or modified by considering the thresholding method. Whereas the wavelet transform is good for energy compaction, the small coefficients are considered as noise and large coefficients indicate important signal features [Gupta & kaur, 2002]. Therefore, these small coefficients can be cut with no effect on the significant features of the image.

Let $X = \{X_{i,j},\ i, j = 1, 2 \ldots M\}$ denotes the $M \times M$ matrix of the original image. The two dimensional orthogonal Discrete Wavelet Transform (DWT) matrix and its inverse are implied by W and W^{-1}, respectively. After applying the wavelet transform to the image matrix X, this matrix is subdivided into four sub-bands namely LL, HL, LH and HH [Burrus et al., 1998].

Whereas the LL channel possesses the main information of the image signal, we apply the hard or soft thresholding technique to the other three sub-bands which contain the details coefficients. The outcome matrix which is produced after utilizing the thresholding level is denoted as $\hat{L}$ matrix. Finally, the smoothed image matrix can be obtained as follows:

$$\hat{X} = W^{-1}\hat{L} \tag{30}$$

The brief description of the hard thresholding is as follows:

$$\gamma(Y) = \begin{cases} Y & if\ |Y| > T \\ 0 & otherrwise \end{cases} \tag{31}$$

where Y is an arbitrary input matrix, $\gamma(Y)$ is the hard thresholding function which is applied on Y, and T indicates the threshold value. Using this function, all coefficients less than the threshold are replaced with zero and other coefficients are kept unchanged.

The soft thresholding acts similar to the hard one, except that in this method the values above the threshold are reduced by the amount of the threshold. The following equation implies the soft thresholding function:

$$\eta(Y) = \begin{cases} sign(Y)(|Y| - T) & if\ |Y| > T \\ 0 & otherrwise \end{cases} \tag{32}$$

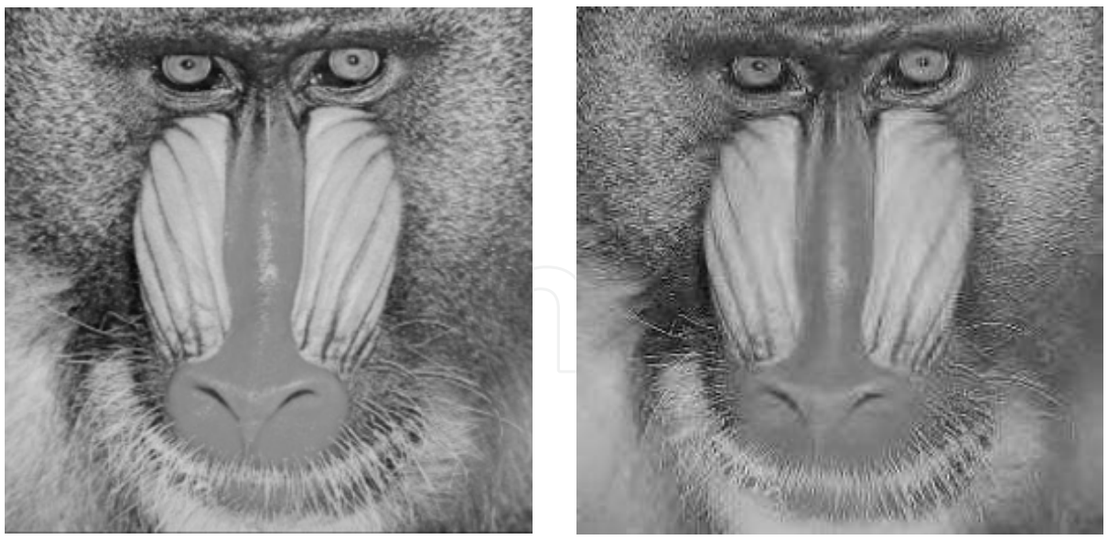
where Y is the arbitrary input matrix, $\eta(Y)$ is the soft thresholding function and T indicates the threshold value. The researchs indicates that the soft thresholding method is more desirable in comparison with the hard one because of its better visual performance. The hard thresholding method may cause some discontinuous points in the image and this event may be a discouraging factor for the performance of our segmentation.

Three methods are presented to calculate the threshold value, namely Visushrink, Bayesshrink and Sureshrink. The method Visushrink is based on applying the universal threshold [Donoho & Johnstone, 1994]. This thresholding is given by $\sigma\sqrt{2\log M}$ where σ is standard deviation of noise and M is the number of pixels in the image. This threshold does not adapt well with discontinuities in the image. Sureshrink is also a practical wavelet procedure, but it uses a local threshold estimated adaptively for each level [Jansen, 2001]. The Bayesshrink rule uses a Bayesian mathematical framework for images to derive subband-dependent thresholds. These thresholds are nearly optimal for soft thresholding, because the wavelet coefficients in each subband of a natural image can be summarized adequately by a Generalized Gaussian Distribution (GGD) [Chang et al., 2000].

2.3.1.2 Algorithm and results

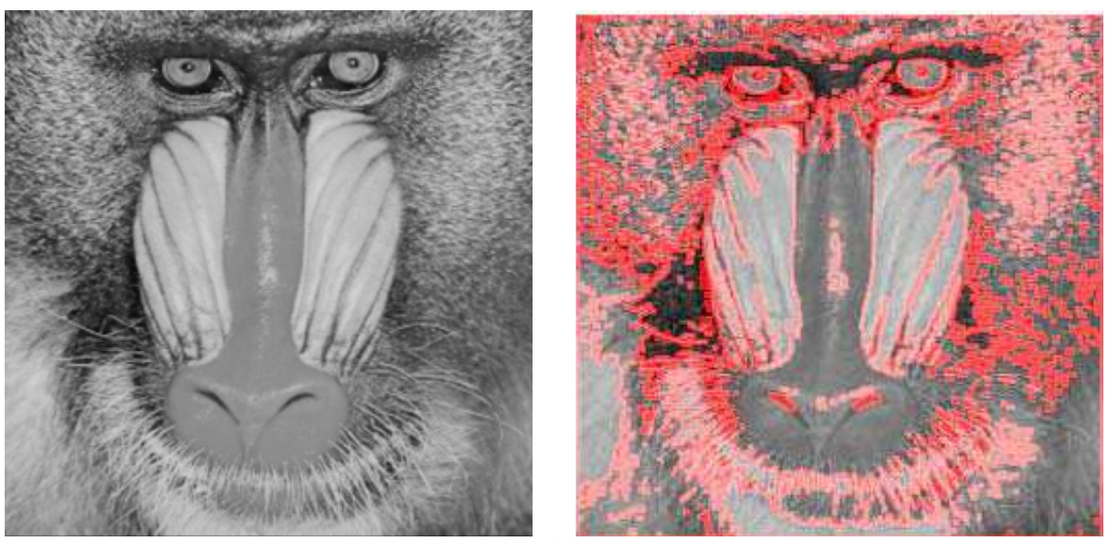
Our implementations on several different types of images show that "*Daubechies*" is one of the most suitable wavelet filters for this purpose. An image is decomposed, in our case, up to 2 levels using *8-tap Daubechies* wavelet filter. The amount of the threshold is assigned by the *Bayesshrink* rule and this value may be different for each image. This algorithm can be expressed as follows. First image is decomposed into four different channels, namely LL, LH, HL and HH. Then the soft thresholding function is applied on these channels, except on LL. Finally the smoothed image is reconstructed by inverse wavelet transform. Figure 6 shows the result of applying wavelet thresholding on the Baboon image. It can be inferred from this figure that the resulted image has fewer discontinuities than the original image and its smoothing degree increased and will be resulted in a fewer number of pixons.

In order to obtain a better view about pixonal image, we indicate the effect of pixon forming stage on an arbitrary image. As illustrated in Fig. 7, the boundaries between the adjacent pixons are sketched so that the image segments are more proper.



(a) (b)

Fig. 6. Result of applying wavelet thresholding technique on Baboon image: (a) Original image, and (b) smoothed image



(a) (b)

Fig. 7. The effect of applying the pixon forming algorithm to the baboon image: (a) The original image, (b) the output image with boundaries between pixons

2.3.2 Image Segmentation using pixon method

In this approach the wavelet thresholding technique is used as a pre-processing step to make the image smoothed. This technique is applied on the wavelet transform coefficients of image using the soft thresholding function. The output of pre-processing step is then used in the pixon formulation stage. In TPB algorithm, after obtaining the pseudo image, the

anisotropic diffusion equation was used to form the pixons. In WPB algorithm, utilizing the wavelet thresholding method as a pre-processing stage eliminates the necessity of using the diffusion equations. After forming and extracting the pixons, the Fuzzy C-Means (FCM) algorithm is used to segment the image. The FCM algorithm is an iterative procedure described in the following [Fauzi & Lewis, 2003].

Given M input data $\{x_m; m = 1,...,M\}$, the number of clusters $C(2 \le C < M)$, and the fuzzy weighting exponent $w, 1 < w < \infty$, initialize the fuzzy membership functions $u^{(0)}_{c,m}$ with $c = 1,...,C$ and $m = 1,...,M$ which are the entry of a $C \times M$ matrix $U^{(0)}$. The following procedure is performed for iteration $l = 1,2,...$:

1. Calculate the fuzzy cluster centers v_c^l with $v_c = \sum_{m=1}^{M} (u_{c,m})^w x_m \; / \; \sum_{m=1}^{M} (u_{c,m})^w$
2. Update $U^{(l)}$ with $u_{c,m} = 1 / \sum_{i=1}^{C} (\frac{d_{c,m}}{d_{i,m}})^{\frac{2}{w-1}}$ where $(d_{i,m})^2 = x_m - v_i^2$ and $.$ is any inner product induced norm.
3. Compare $U^{(l)}$ with $U^{(l+1)}$ in a convenient matrix norm. If $U^{(l+1)} - U^{(l)} \le \varepsilon$ stop; otherwise return to step 1.

The value of the weighting exponent, w determines the fuzziness of the clustering decision. A smaller value of w, i.e. w is close to unity, will give the zero/one hard decision membership function, and a larger w corresponds to a fuzzier output. Our experimental results suggest that w = 2 is a good choice.

Figure 8 illustrates this method block diagram.

3. Evaluation of the pixon-based methods

In this section the pixon-based image segmentation methods are applied on several standard images and the results of these implementations are extracted. For this purpose, commonly used images such as baboon, pepper and cortex are selected and the performance of applying the mentioned methods on them is compared. In order to evaluate these methods numerically, several experiments have been carried out on different standard images and some criteria such as number of the pixons in image, pixon to pixel ratio, normalized variance and computational time are used which are introduced in following.

3.1 Measurements

Computational time; In most applications, the time which is consumed to perform algorithms is an important parameter to evaluate them. So, researchers always seek to decrease the computational time.

Number of pixons and pixon to pixel ratio; As expressed previously, after forming the pixons, the image segmentation problem transformed to labeling the pixons. So, decrement in the number of pixons and related pixon to pixel ratio results in a decrement in computational time. Certainly it should be noted that the details of the image do not eliminate in this way.

Variance and Normalized Variance; One of the most important parameters used to evaluate the performance of image segmentation methods is the variance of each segment. The smaller value of this parameter implies the more homogeneity of the region and consequently the better segmentation results. Assume that after the segmentation process,

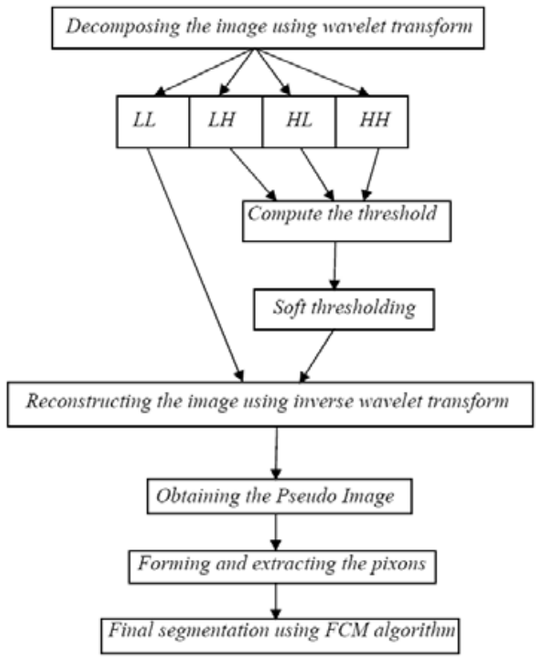


Fig. 8. The block diagram of the proposed method

the images are divided into K segments with different average values which we have called these segments as "Classes". In addition to the typical variance, the normalized variance of each image can be calculated. If N_k and $V(k)$ denotes the number of the pixels and the variance of each class respectively, the normalized variance of each image can be determined as below:

$$V^* = \frac{V_1}{V_2} \tag{33}$$

where

$$V_1 = \sum_{k=1}^{K} \frac{N_k V(k)}{N} \tag{34}$$

and

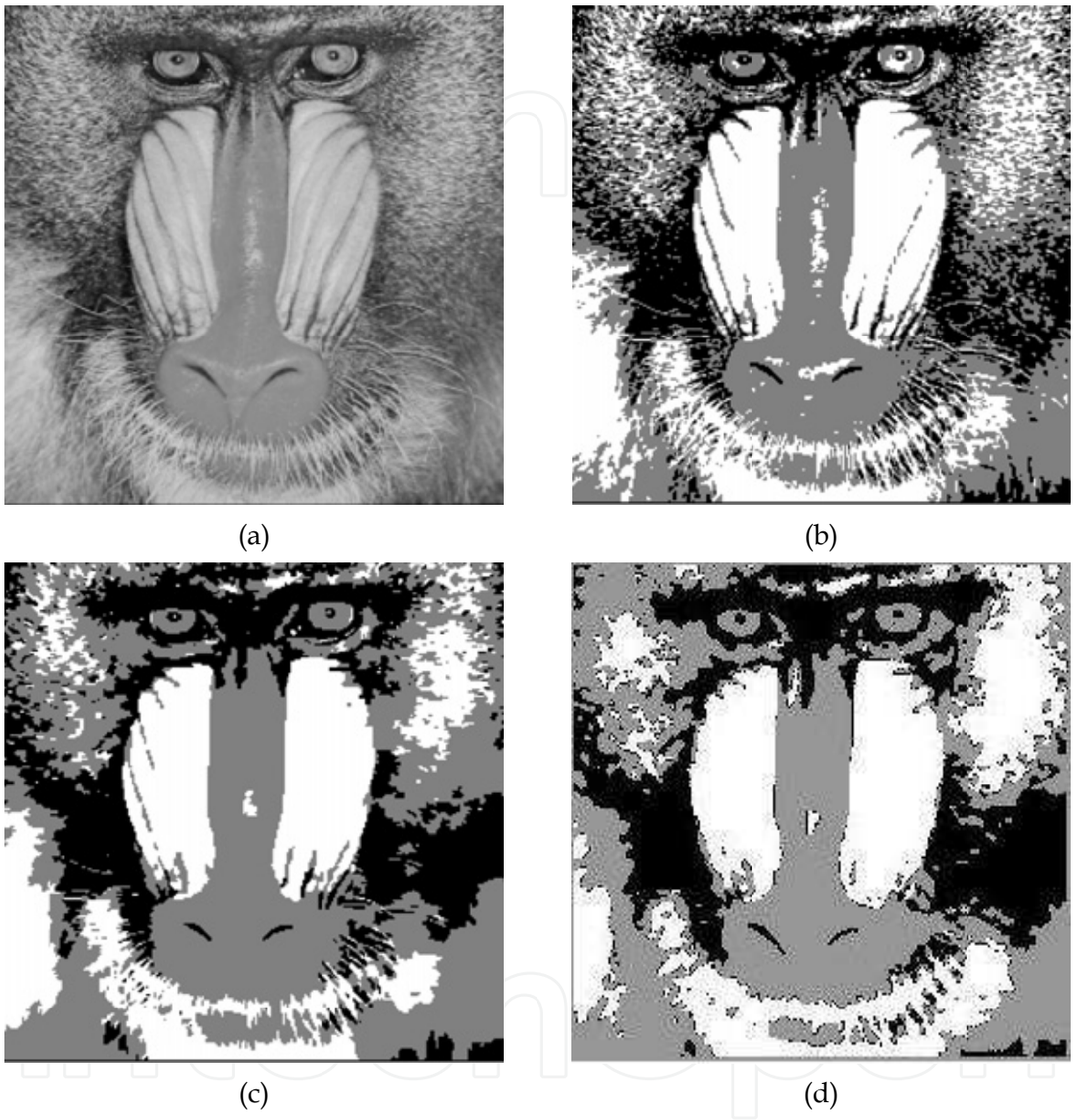
$$V_2 = \sum_{k=1}^{K} \frac{(I(x,y) - M)^2}{N} \tag{35}$$

In the above equations, k denotes the number of classes, $I(x,y)$ is the gray level intensity, M and N are the averaged value and the number of pixels in each image respectively.

3.2 Experimental results

In this section, results of applying the TPB, MPB and WPB methods on several standard images are considered. Figs. 9(a), 10(a) and 11(a) are the Baboon, Pepper and Cortex images used in this experiment. Figs. 9(b), 10(b), 11(b) and 9(c), 10(c), 11(c) show the segmentation

results of TPB and PMB methods, respectively. The segmentation results of WPB method are illustrated in Figs. 9(d), 10(d) and 11(d). As shown in these figures, the homogeneity of regions and the discontinuity between adjacent regions, which are two main criteria in image segmentation, are enhanced in WPB method.



(a) (b)

(c) (d)

Fig. 9. Segmentation results of the Baboon image: (a) Original image, (b) TPB's method, (c) WPB's method, and (d) WPB's method

In addition, several experiments have been carried out on the different images and the average results are drawn in several tables. In Table 1, the number of pixons and the ratio of Pixon-Pixel in the three methods are shown. As can be seen from this table we can find that these parameters are decreased significantly in WPB method in comparison with two other methods which resulted from applying wavelet thresholding technique before forming pixons. Table 2 shows the computational time required of the three methods (Intel(R) Core(TM)2 Duo CPU 2.20 GHz processor, with MATLAB 7.4). By using pixon concept with

wavelet thresholding technique in the WPB method, the computational cost is sharply reduced. Since the MRF technique, because of its complicated mathematical equations, is a time consuming process, the MPB method expends much time compared to TPB method.
In this experience, after the segmentation process, the images are divided into three segments or Classes. The variance and average of each class are listed in Tables 3-5, for mentioned images. In most cases, the variance values of the classes of different images in WPB method are smaller in comparison with the other methods. In order to investigate the performance of methods more exact, the normalized variance of each image after applying the

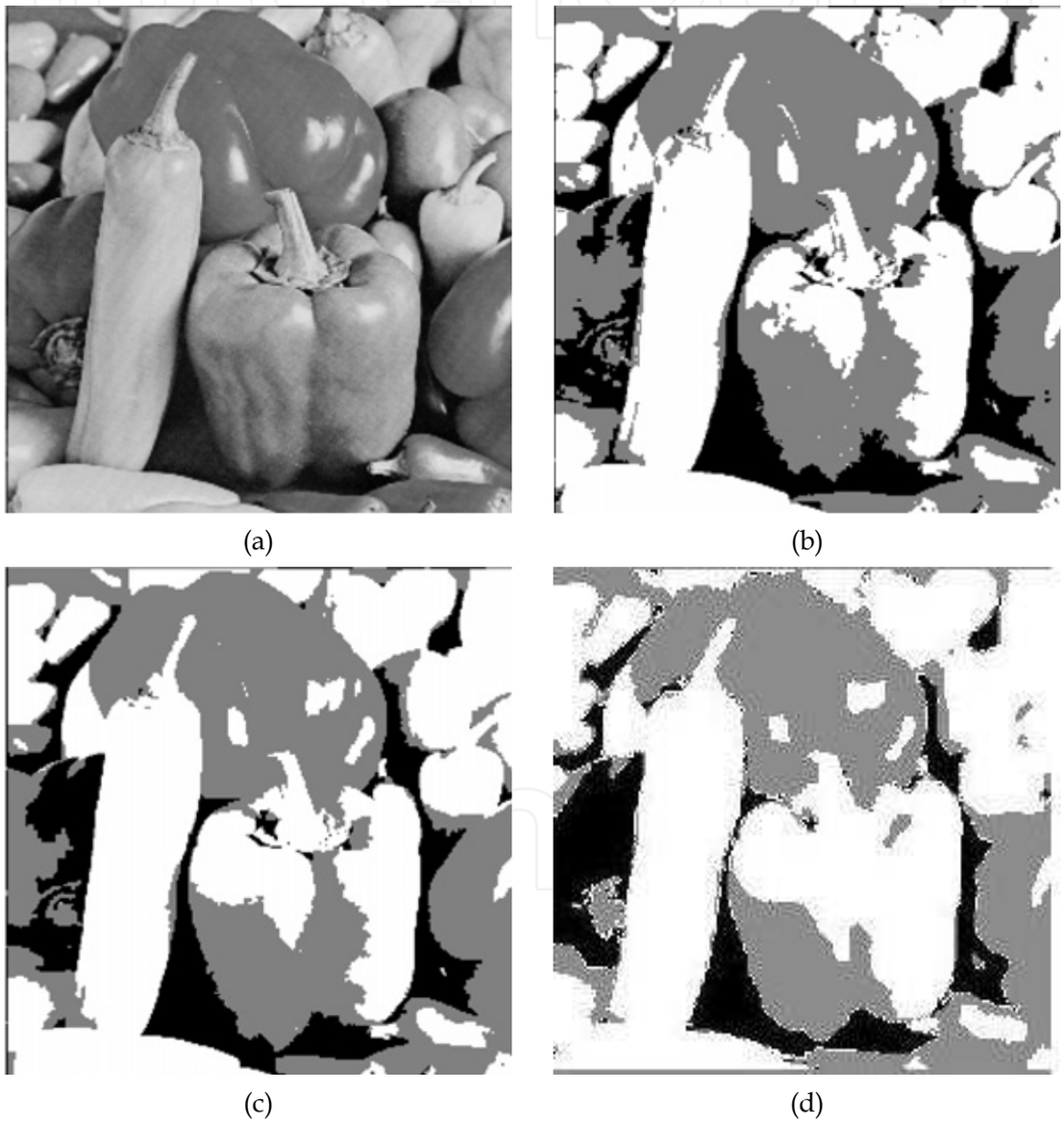


Fig. 10. Segmentation results of the Pepper image: (a) Original image, (b) TPB's method, (c) MPB's method, and (d) WPB's method

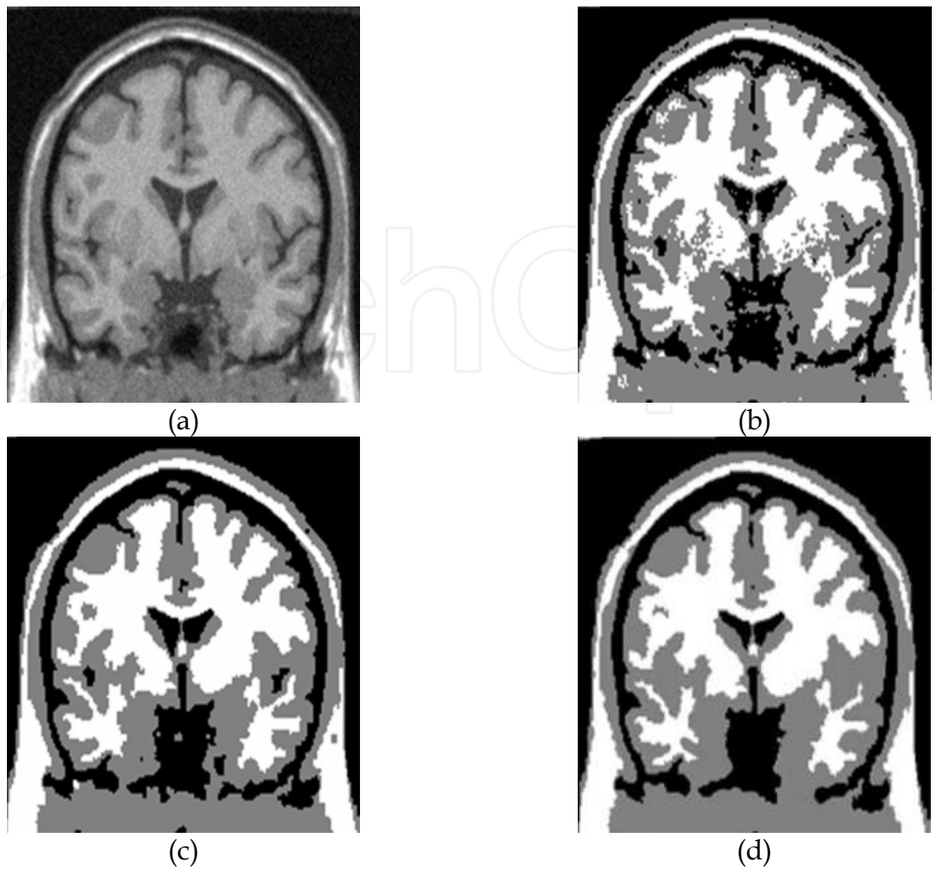


Fig. 11. Segmentation results of the Cortex image: (a) Original image, (b) TPB's method, (c) MPB's method, and (d) WPB's method

three methods are calculated too. The normalized variance results illustrated in the tables demonstrate that in the pixon-based approach which used wavelet (WPB method), the amount of pixels in each cluster is closer to each other and the areas of images are more homogenous.

Images (Size)	The number of pixels	The number of pixons			The ratio between the number of pixons and pixels		
		TPB's method	MPB's method	WPB's method	TPB's method	MPB's method	WPB's method
Baboon (256×256)	262144	83362	61341	25652	31.8 %	23.4 %	9.79 %
Pepper (256×256)	262144	31981	24720	13221	12.2 %	9.43 %	5.04 %
Cortex (128×128)	16384	1819	1687	1523	11.1 %	10.2 %	9.3 %

Table 1. Comparison of the number of pixons and the ratio between the number of pixons and pixels, among the three methods

Images	TPB's method (ms)	MPB's method (ms)	WPB's method(ms)
Baboon	18549	19431	15316
Pepper	15143	17034	13066
Cortex	702	697	633

Table 2. Comparison of the computational time, between the three methods

Method	Parameter	class 1	class 2	class 3
TPB's method	average	168.06	127.28	84.25
	variance	12.18	11.06	17.36
	Normalized Variance	**0.0279**		
MPB's method	average	167.86	126.45	82.18
	variance	12.05	11.55	16.67
	Normalized Variance	**0.0259**		
WPB method	average	170.40	128.36	83.95
	variance	11.34	11.46	16.96
	Normalized Variance	**0.0212**		

Table 3. Comparison of variance values of each class, for the three algorithms (Baboon).

Method	Parameter	class 1	class 2	class 3
TPB's method	average	190.59	123.29	35.47
	variance	16.64	21.89	21.79
	Normalized Variance	**0.0263**		
MPB's method	average	191.68	125.27	34.39
	variance	16.28	22.66	22.30
	Normalized Variance	**0.0251**		
WPB's method	average	189.75	122.56	37.17
	variance	15.87	22.86	20.30
	Normalized Variance	**0.0217**		

Table 4. Comparison of variance values of each class, for the three algorithms (Pepper).

Method	Parameter	class 1	class 2	class 3
TPB's method	average	22.44	93.71	197.23
	variance	12.59	11.67	14.11
	Normalized Variance	**0.0131**		
MPB's method	average	21.34	91.65	199.50
	variance	12.75	10.33	13.93
	Normalized Variance	**0.0119**		
WPB's method	average	24.25	92.49	196.72
	variance	11.37	10.51	12.81
	Normalized Variance	**0.0101**		

Table 5. Comparison of variance values of each class, for the three algorithms (Cortex).

4. Conclusion

This chapter provided an introduction to the pixon-based image segmentation methods. The pixon is a set of disjoint regions with variable shapes and sizes. Different algorithms were introduced to form and extract the pixons. Pixon-based methods were divided into three classes: TPB method, which used from the traditional pixon definition to segment the image; MPB method, which combined the pixon concept and MRF to obtain the segmented image; and WPB method, which segmented the image by a pixon-based approach utilizing the wavelet thresholding algorithm. The chapter was concluded with illustration of experimental results of applying these methods on different standard images.

5. References

Andrey P. & Tarroux, P. (1998). Unsupervised segmentation of Markov random field modeled textured images using selectionist relaxation, IEEE Trans. Pattern Anal. Machine Intell., vol. 20, pp. 252–262.

Bonnet, N. ; Cutrona, J. & Herbin, M. (2002). A 'no-threshold' histogram-based image segmentation method, *Pattern Recognition*, Volume 35, Issue 10, pp. 2319-2322.

Burrus, C. S. ; Gopinath, R. A. & Guo, H. (1998). Introduction to Wavelets and Wavelet Transforms, Prentice Hall,New Jersey.

Comaniciu, D. & Meer, P. (2002). Mean shift: a robust approach toward feature space analysis, *IEEE Trans. Pattern Anal.* Mach. Intell. 24 (5), pp. 1–18.

Donoho, D. L. &. Johnstone, I.M. (1994). Ideal spatial adaptation via wavelet shrinkage, *Biometrica*, Vol. 81, pp. 425-455.

Elfadel I. M. & Picard, R. W. (1994). Gibbs random fields, cooccurrences, and texture modeling, *IEEE Trans. Pattern Anal.* Machine Intell., vol. 16, pp. 24–37.

Fauzi M. F. A. & Lewis, P. H. (2003). A Fully Unsupervised Texture Segmentation Algorithm, *British Machine Vision Conference 2003*, Norwich, UK. pp.1201-1206

Francisco de A.T. de Carvalho, (2007). Fuzzy c-means clustering methods for symbolic interval data, *PatternRecognition Letters*, Volume 28, Issue 4, pp. 423-437.

Gonzalez, R. C. & Woods, R.E. (2004). *Digital Image Processing*, Prentice Hall,

Gupta,S. & kaur, L. (2002). Wavelet Based Image Compression using Daubechies Filters, *8th National conference on communications*, I.I.T. Bombay, pp. 88-92.

Hassanpour, H & Yousefian, H. (2010). A Pixon-Based Approach for Image Segmentation Using Wavelet Thresholding Method, *International Journal of Engineering(IJE)*, Vol. 23, pp. 257-268.

Jansen, M. (2001). Noise Reduction by Wavelet Thresholding, Springer Verlag New York Inc., Pages. 875-879.

Chang, S. G. ; Yu, B. & Vetterli, M. (2000). Adaptive Wavelet Thresholding for image Denoising and compression, *IEEE Trans. Image Processing*, Vol.9, pp.1532-1545.

Kato, Z. ; Zerubia, J. & Berthod, M. (1999). Unsupervised parallel image classification using Markovian models, *Pattern Recognit.*, vol. 32, pp. 591–604.

Kim, I.Y. & Yang, H.S. (1996). An integration scheme for image segmentation and labeling based on Markov random field model, *IEEE Trans. Pattern Anal. Mach. intell.* Vol.18 No.1. pp. 69–73.

Lakshmanan, S. & Derin, H. (1989). Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing, *IEEE Trans. Pattern Anal.* Machine Intell., vol. 11, no. 8, pp. 799–813.

Lin, L. ; Zhu, L. & Yang, F. & Jiang, T. (2008). A novel pixon-representation for image segmentation based on

Markov random field, *Image and Vision Computing journal of ELSEVIER*, Vol.26, pp. 1507–1514.

Papamichail, G.P. & Papamichail, D.P. (2007). The k-means range algorithm for personalized data clustering in e-commerce, *European Journal of Operational Research*, Volume 177, Issue 3, pp. 1400-1408.

Piña, R. K. & Pueter, R. C. (1993). Bayesian image reconstruction: The pixon and optimal image modeling, *P. A. S. P.,* vol. 105, pp. 630–637.

Perona P. & Malik, J. (1990). Scale-space filtering and edge detection using anisotropic diffusion, *IEEE Trans. Pattern Anal.* Machine Intell., Vol.12, No. 7, pp. 629–639.

Puetter, R. C. (1995). Pixon-based multiresolution image reconstruction and the quantification of picture information content, *Int. J. Imag. Syst. Technol.*, vol. 6, pp. 314–331.

Shi, J. & Malik, J. (2000). Normalized cuts and image segmentation, *IEEE Trans. Pattern Anal*. Mach. Intell. 22 (8) pp. 888–905.

Yang, F. & Jiang, T. (2003). Pixon-based image segmentation with Markov random fields, *IEEE Trans. Image Process.* 12 (12) , pp. 1552–1559.

Zhu, S.C. & Yuille, A. (1996). Region competition: unifying snakes, region growing, and byes/mdl for multi-band image segmentation, *IEEE Trans. Pattern Anal*. Mach. Intell. 18 (9) ,pp. 884–900.

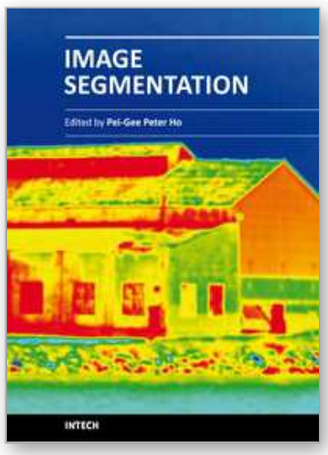


Image Segmentation
Edited by Dr. Pei-Gee Ho

ISBN 978-953-307-228-9
Hard cover, 538 pages
Publisher InTech
Published online 19, April, 2011
Published in print edition April, 2011

It was estimated that 80% of the information received by human is visual. Image processing is evolving fast and continually. During the past 10 years, there has been a significant research increase in image segmentation. To study a specific object in an image, its boundary can be highlighted by an image segmentation procedure. The objective of the image segmentation is to simplify the representation of pictures into meaningful information by partitioning into image regions. Image segmentation is a technique to locate certain objects or boundaries within an image. There are many algorithms and techniques have been developed to solve image segmentation problems, the research topics in this book such as level set, active contour, AR time series image modeling, Support Vector Machines, Pixon based image segmentations, region similarity metric based technique, statistical ANN and JSEG algorithm were written in details. This book brings together many different aspects of the current research on several fields associated to digital image segmentation. Four parts allowed gathering the 27 chapters around the following topics: Survey of Image Segmentation Algorithms, Image Segmentation methods, Image Segmentation Applications and Hardware Implementation. The readers will find the contents in this book enjoyable and get many helpful ideas and overviews on their own study.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Hamid Hassanpour, Hadi Yousefian and Amin Zehtabian (2011). Pixon-Based Image Segmentation, Image Segmentation, Dr. Pei-Gee Ho (Ed.), ISBN: 978-953-307-228-9, InTech, Available from: http://www.intechopen.com/books/image-segmentation/pixon-based-image-segmentation

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen