Scientific Research and Essays Vol. 5(24), pp. 3914-3925, 18 December, 2010
Available online at http://www.academicjournals.org/SRE
ISSN 1992-2248 ©2010 Academic Journals

Full Length Research Paper

A video-rate color image segmentation using adaptive and statistical membership function

Omid Sojodishijani*, Vahid Rostami and Abd Rahman Ramli

Intelligent System and Robotic Laboratory, Institute of Advanced Technology (ITMA), Universiti Putra Malaysia (UPM), 43400 Serdang, Selangor, Malaysia.

Accepted 24 November, 2010

The color image segmentation is a critical task in many computer vision applications. The function of segmentation is to identify homogeneous regions in an image, based on properties such as intensity, color and texture. Typically, the image segmentation algorithms in video processing system require very high computation power, so it is desirable to develop algorithms for implementation as a real-time system. This paper proposes a novel image segmentation algorithm and its real-time hardware architecture which is capable of dealing with regions color information. In this algorithm, statistical information of regions is used to create fuzzy membership functions in color model components. These membership functions characterize each segment in an image, which are updated dynamically when the image is being scanned. The histogram of color components are estimated by non-symmetric Gaussian function (*NSGF*). To overcome the video-rate limitation, the image is scanned in the raster fashion. Moreover, the hardware architecture of algorithm on FPGA is reported in this paper. Finally, the results of algorithm are analyzed by quantitative performance analyzers.

Key words: Embedded system, adaptive membership function, video-rate image segmentation.

INTRODUCTION

Designing dedicated devices in embedded systems is one of the controversial challenges in the development of smart technology. In embedded systems, some specific tasks and low level functions are implemented on the hardware instead of high level software. This transition is helpful for reducing transactions of main processor, communication rate, cost, and size. To achieve this, we need to develop real-time algorithms suitable for consumer electronic devices.

Data reduction in video image retrieval (Han and Ma, 2002), anomalies detection in industrial inspection systems (Miteran et al., 2003), object recognition and tracking in visual surveillance (Perumal et al., 2008), surgical instruments inside the abdominal cavity (Doignon et al., 2005), tumors detection from medical images (Guvenc et al., 2010), etc, usually require segmenting the natural images in video-rate speed. Due to mounting space and communication speed limitations, it is mostly favourable to perform the process on a single chip.

Obviously, most kind of real-time image processing algorithms need to process extremely pixels without I/O usage in a limited time. For example in video processing to establish 25 images (640x480 pixels) per second, each pixel must be processed in less than 130 ns. In these systems because a large amount of pixels are involved in processing with high computation speed, the software implementation is not suitable and hardware implementation becomes mandatory. FPGAs and ASICs are two classes of hardware devices that fall in terms of performance for real-time applications. Due to high cost for constructing the algorithms onto ASICs, it seems that the FPGAs offer an excellent platform to be used in real-time image processing (Shirazi et al., 1995; Meribout et al., 1999; Draper et al., 2003; Trieu and Maruyama, 2007).

Image segmentation is a process which partitions an image into disjoint homogeneous regions. Due to changing conditions of brightness, shadows and color gradients in natural images, there are many segmentation techniques that provide a fuzzy framework

*Corresponding author. E-mail: o_sojoodi@ieee.org.

to model imprecise regions in these kind of images (Gillet et al., 2001; Liew et al., 2005; Prados-Suarez et al., 2007).

In Prados-Suarez et al. (2007), a supervised method by fuzzy connectivity property between pixels to extract the segments in path based approach was proposed. The membership degrees were computed from resemblance of pixels in the surrounding of seed points. Although this approach provided a suitable homogeneity measure, it is needed to determine the seed points appropriately. Meanwhile, it requires a large amount of memory with imposed limitation for use in embedded systems.

In Liew et al. (2005), an adaptive fuzzy c-means (*FCM*) clustering was presented for segmenting natural images. In this method, the clustering algorithm was capable of utilizing local contextual information to impose local spatial continuity. Furthermore, a dissimilarity index that is insensitive to intensity variation in an objective function of FCM was introduced.

In Trieu and Maruyama (2007), the watershed transformation was used to segment the gray scale image in parallel and pipeline techniques. The hardware was designed on single FPGA chip to establish a 512x512 gray level image with a plausible speed to overcome constraint of real-time systems. In this approach, the entire image is stored in the external memory, and is scanned twice, top-left to bottom-right and bottom-right to top-left. Also, an internal memory buffer was used to keep 5 lines of image for pipeline processing.

In Miteran et al. (2003), a Support Vector Machine (*SVM*) decision method is employed for quality control of industrial products. In this approach, the hyper-rectangles model is developed for features dimension in the classification method. Also, to achieve a process of 10 gray scale images per second and simultaneously using the learned classifier parameters, a VHDL code is implemented on the FPGA automatically. In this paper, we propose a novel color image segmentation algorithm based on fuzzy approach which is capable to implement in the FPGA and can be used in the embedded systems. In our method, the parameters of the fuzzy framework are created and updated by exploiting the statistical information from a color pixels stream. Furthermore, the hardware architecture of proposed algorithm is sketched in this paper. The major contribution of the proposed video-rate image segmentation method is the aggregation of the following advantages in one algorithm:

- This method robustly segments both synthetic and complex natural images, and maintains this performance in a single phase scanning of an image;
- It does not need *a priori* assumptions/knowledge (such as seed points) and predefined parameters;
- It is capable of dealing with various cameras' output color spaces (RGB, HSI, YUV, etc.);
- It extracts more accurately the homogeneous regions by using non-symmetric Gaussian function (*NSGF*) which is derived from Gaussian Mixture Model (*GMM*) to model the histogram of regions;
- The computational costs are abated by employing a parallel and pipeline algorithm for reaching up to 25 frames per second (fps) in frames of 640 x 480 pixels.

Reviewing the literature shows that the real-time image segmentation algorithms usually have only some of the above advantages (Miteran et al., 2003; Meribout et al., 1999; Liew et al., 2005; Prados-Suarez et al., 2007; Trieu and Maruyama, 2007).

Finally, we present the accuracy of proposed method by a comprehensive set of experimental results for natural images with two well known image segmentation approaches as well as proposed method and three unsupervised quantitative evaluators.

The remainder of the paper is organized as follows: Section 2 defines the segment and homogeneity measure in a fuzzy framework. Section 3 provides details of proposed algorithm. Hardware implementation of algorithm in a parallel situation is described in Section 4. In Section 5, algorithm is analyzed by quantitative and unsupervised analyzers. Finally, Section 6 shows the result of applying proposed algorithm to some natural images.

MATERIALS AND METHODS

Fuzzy region implication

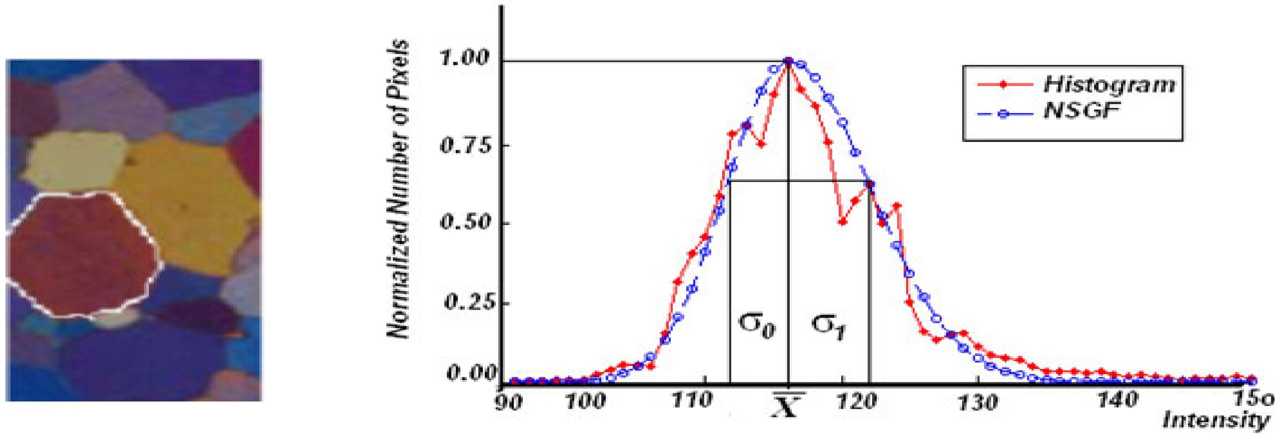
As we know, human intuitive perception of a segment is not precise, that is each pixel of an image belongs to all adjacent segments by different resemblance values. A region consists of a set of connected pixels which are resembled by its features vector. Therefore, we need to define an imprecise relationship function for this resemblance measure. Most researchers have used fuzzy framework by predetermined relationship function to characterize the regions (Gillet et al., 2001; Liew et al., 2005; Prados-Suarez et al., 2007). On the other hand, the synthesized and natural images could be partitioned to uniform and non-uniform (composed by multiple objects) regions (Protiere and Sapiro, 2007). The histogram of each channel in color space can be modeled by a simple Gaussian for uniform regions and a Gaussian Mixture Model (*GMM*) for non-uniform regions (Protiere and Sapiro, 2007; Rother et al., 2004).

Our method is based on an adaptive fuzzy approach and region growing. In this method, a GMM derivation which is called Non-Symmetric Gaussian Function (*NSGF*) is proposed to model the histogram of regions that can be used as membership functions. The combination of the membership functions in each color component defines fuzzy homogeneity measure of a region.

Fuzzy region definition

In this research, the image segmentation algorithm is carried out to partition an image into disjoint sets $S_1, S_2, \dots S_n$ of connected pixels according to the following criteria:

$$I = \bigcup^{i=1..n} S_i$$



(a) (b)

Figure 1. (a) Marked a region in natural image (b) Histogram (dashed line) of region and approximated (Dotted line) by *NSGF*.

And

$$Max\left(\sum_{S\in\{S_1,S_2,\ldots,S_n\}}\sum_{x\in S} H(F_S,x)\right) \quad (1)$$

where $H(F_s,x)$ is a fuzzy homogeneity measure between pixel x and segment S that will be described in the next subsection.

On the other hand, the Probability Density Function (*PDF*) of color features, has been used as a descriptor of regions by some researchers (Gillet et al., 2001; Liew et al., 2005).

In our method, the histograms of color components in each segment are modeled by a Non-Symmetric Gaussian Function (*NSGF*) which is derived from Gaussian Mixture Model (*GMM*) as follows:

$$NSGF^{c}(x) = \sum_{i=0,1} w_i(x)\, g(x,\bar{x},\sigma_i) \qquad where \quad c\in\{r,g,b\}$$

$$w_i(x) = \begin{cases} i & , \; x < \bar{x} \\ 1-i & , \; x \geq \bar{x} \end{cases} \quad (2)$$

Where $\bar{x}$ is the mean value, σ_i represents the variances, w_i is the mixing proportions (which are positive and sum to one) and g is a normalized Gaussian with specified mean and variance. In Equation 2, *NSGF* is approximation of histogram of color component *c*, which is used as membership function of fuzzy homogeneity measure $H(F_S, x)$.

For instance as shown in Figure 1, the histogram of red channel for marked segment is approximated by NSGF in which the left side is a semi Gaussian function with σ_1 variance and right side is the same function with σ_2 variance. However, the algorithm could be established in various color models such as RGB, HIS, HSV and YUV. The RGB has been chosen in our research, because this color space is the most popular output in many imaging devices.

Regarding the approximation of the histogram as mentioned above, the region *S* is specified by F_s which is made by *NSGFs* for color components in a given color model e.g. RGB as follows:

$$F_s = [NSGF_s^{\,c}] \quad where \quad c\in\{r,g,b\} \quad (3)$$

Homogeneity measure

The natural images can be represented as several partitions according to homogeneity measure and contour of regions. In our research, for computing the resemblance degree of each pixel to each region, the fuzzy homogeneity measure is employed. The Dubois-Prade T-norm is utilized (Franke et al., 2000) for determining the homogeneity degree between pixel $X=(x_r, x_g, x_b)$ and region F_s:

$$H(F_s, X) = \frac{\prod_{i\in\{r,g,b\}} NSGF_s^{\,i}(x)}{\underset{i\in\{r,g,b\}}{Max}(c, NSGF_s^{\,i}(x))} \quad (4)$$

Where c is a non zero constant to avoidance of division by zero and the value of *H* is in [0, 1]. The low values of *H* means a large difference between pixel *X* and region *S* while; its high value indicates their high similarity.

PROPOSED METHOD

In this section, the pixel stream oriented algorithm for partitioning a color image to disjoint and homogeneous regions is introduced. Due to video-rate constraints, the accessible information to decide about each pixel in stream is limited to local predecessor pixels. This algorithm is designed to operate an input image in a raster scan fashion from left to right and top to bottom. Therefore, the current pixel *X* has 4 predecessor adjacent pixels named N_1, N_2, N_3, N_4 as depicted in Figure 2. Consequently, only *n+1* structures of information are needed, where *n* is the row size of image. One data structure for each column is retained except the column before the current pixel that has two data structures, N_1 and N_2. Data structures include the segment information that ended in each column as shown in Table 1. This set of data structures can be implemented by a circular linked list as shown in Figure 3.

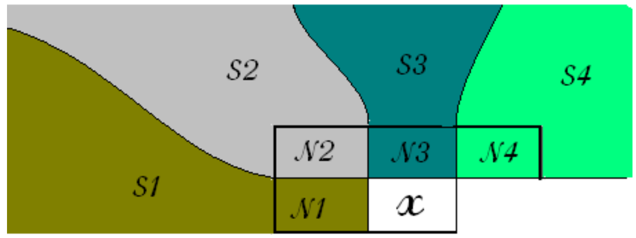


Figure 2. Processing pixel *x* can be connected to one of the four neighbor regions S_1, …, S_4 by four predecessor adjacent pixels N_1,…, N_4.

Table 1. List of attributes per each segment in circular linked list of segments information.

[1]	Property name	[2]	Type	[3]	Meaning
[4]	rSigmaL	[5]	Float	[6]	Variance of left side of red histogram
[7]	rSigmaR	[8]	Float	[9]	Variance of right side of red histogram
[10]	rMu	[11]	Float	[12]	Mean of red histogram
[13]	gSigmaL	[14]	Float	[15]	Variance of left side of green histogram
[16]	gSigmaR	[17]	Float	[18]	Variance of right side of green histogram
[19]	gMu	[20]	Float	[21]	Mean of green histogram
[22]	bSigmaL	[23]	Float	[24]	Variance of left side of blue histogram
[25]	bSigmaR	[26]	Float	[27]	Variance of right side of blue histogram
[28]	bMu	[29]	Float	[30]	Mean of blue histogram
[31]	rPixelCountL	[32]	Float	[33]	Number of pixel in left side of red histogram
[34]	rPixelCountR	[35]	Float	[36]	Number of pixel in side right of red histogram
[37]	gPixelCountL	[38]	Float	[39]	Number of pixel in left side of green histogram
[40]	gPixelCountR	[41]	Float	[42]	Number of pixel in side right of green histogram
[43]	bPixelCountL	[44]	Float	[45]	Number of pixel in left side of blue histogram
[46]	bPixelCountR	[47]	Float	[48]	Number of pixel in side right of blue histogram
[49]	lastX	[50]	Integer	[51]	Column coordination of last pixel that has been added to segment
[52]	lastY	[53]	Integer	[54]	Row coordination of last pixel that has been added to segment

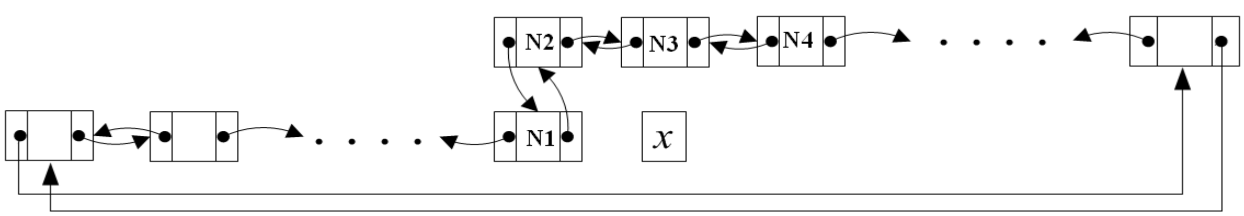


Figure 3. Circular linked list of segments information structure.

However, three phases are simultaneously performed on current pixel in raster scan processing fashion. The first phase that is called homogeneity testing finds out the segment that the current pixel is added to. In the second phase, to avoid over segmentation of the output labeled image by neighboring segments with the same characteristics, a merging process is applied.

In the last phase, noise segments that appear as small segments are detected and eliminated. As shown in Figure 4. The three phases of algorithm, noise suppression (left block), merging process (middle block) and homogeneity testing (right block) are simultaneously applied to the current pixel.

To test the uniformity, the fuzzy homogeneity finds out the best similar segment, *I*, for adding the current pixel *X* to it, as follows:

$$h_s = H(F_s, X) \qquad where \quad s = 0..n \tag{5}$$

and

$$I = ArgMax(h_0.h_1,...,h_n) \tag{6}$$

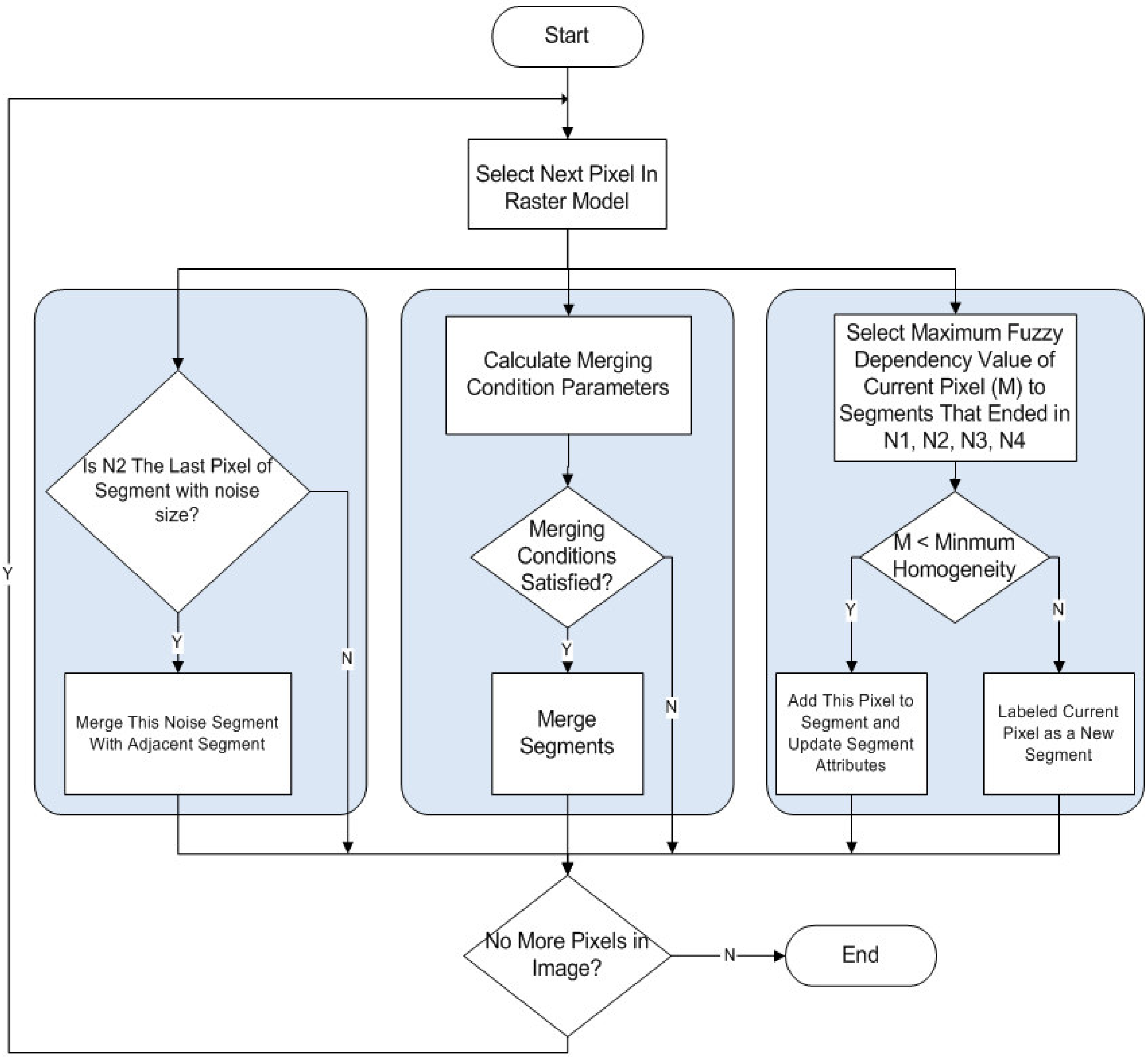


Figure 4. Overall schema of parallel phases of algorithm from left to right: noise suppression, merging process and homogeneity testing.

In Equations (5) and (6), *l* Є *{0, 1... n}*, *n* is the number of neighbors and h_s is homogeneity degree to the neighboring segments. For $l > 0$, $h_0 = \varepsilon$ is minimum value of homogeneity. The value of *l* has two different cases:

Create a New Region: when l=0 implies that current pixel does not satisfy the similarity criteria, therefore; new region will be started at this pixel and feature parameters are initiated.

Append to an existing Region: The current pixel will be added to most similar adjacent segment when $l > 0$. In this case the feature parameters $\bar{x}$, σ_l and σ_r for each component of color model will be updated as follows:

$$\bar{x} \leftarrow \frac{(n-1)\times\bar{x}+x_n}{n} \qquad (7)$$

$$\begin{cases} \sigma_l^2 \leftarrow \dfrac{(n-1)\times\sigma_l^2+(\bar{x}-x_n)^2}{n} & \text{if}(x_n \le \bar{x}) \\ \sigma_r^2 \leftarrow \dfrac{(n-1)\times\sigma_r^2+(\bar{x}-x_n)^2}{n} & \text{if}(x_n > \bar{x}) \end{cases} \qquad (8)$$

where *n* is the total number of pixels in the region as well as current pixel. In the following pseudo code, the homogeneity measures of current pixel and neighbor regions are simultaneously computed. If the minimum value of homogeneity is not satisfied, the current pixel is labeled as a new segment and otherwise it is added to the region that gave the maximum homogeneity value.

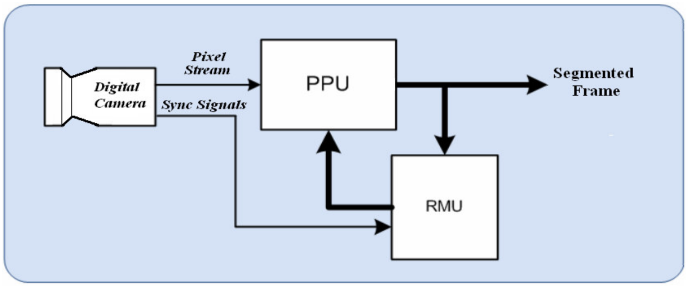


Figure 5. Interconnection between PPU (Parallel Processing Unit) and RMU (Regions Memory Unit).

For (i : i ∈ {1..4 }) Do Parallel
{
$p_i = 1$;
For (c : c ∈ { RED , GREEN , BLUE }) Do Parallel
{
$m_i = NSGF_i^c(x)$;
$s_i = Max\{ m_i \}$;
$p_i = p_i * m_i$;
}
$J_i = p_i / s_i$;
}
$M = Max\{ J_i \}$;
If (M <Minimum homogeneity value)
The current pixel is labeled as a new segment;
Else
The current pixel is labeled with segment number that gave maximum j_i ;

Merging phase is done in proposed algorithm by computing a homogeneity measures in adjacent regions. To this purpose, the homogeneity measure h_{ij} between the feature vectors (F_i and F_j) of adjacent regions is obtained as follows:

$$h_{ij} = Max\ (H\ (F_i, \bar{x}_j), H\ (F_j, \bar{x}_i)) \quad (9)$$

when h_{ij} lies upper than minimum value of homogeneity, that is satisfy the similarity criteria and the two regions are merged into a single region. To reduce the noise segments which appeared as small regions, they are appended to an adjacent segment with higher value of homogeneity measure h_{ij}.

PROPOSED HARDWARE DESIGN

High speed signal processing e.g. video-rate image processing requires about 40 ms processing time per image frame (assuming an image with 640x480 resolutions at 25 fps). Although computers keep getting as fast, there are always new image processing applications that need more processing than is available. To meet the demands of real-time applications, we need to develop new techniques for accelerating image-based applications on commercial hardware such as ASIC, FPGA or FPOA (Miteran et al., 2003; Draper et al., 2003; Trieu and Maruyama, 2007). On the other hand, these devices can be perfectly suitable for execution in parallel and piped elementary digital processing tasks. In addition, the reconfigurability capabilities established for FPGAs and FPOAs constitute a competitive alternative for high performance signal and image processing.

Architecture

The hardware architecture follows aforementioned fuzzy inference and parallel algorithm. There are two main concerns for designing the hardware architecture: the first is how to catch the range of the function with only floating point arithmetic based on IEEE 754 standard (IEEE Standards Board, 1995). The second concern is guaranteeing the low propagation delay.

We employ basic arithmetic floating point operations that are fully parallel and cascade-able with the other components. The hardware architecture consists of two major components: Parallel Processing Unit *PPU* and Regions Memory Unit *RMU* (Figure 5). The major computation of algorithm is performed in *PPU* by basic floating point operations e.g. adder/subtractor/multiplier that is called Process Elements *PE*. The *PPU* computes the homogeneity value and label of pixels by evaluating the largest similarity of input vector to feature vectors. Simultaneously, it recalculates the feature vectors with new pixel and sends updated feature vectors to the Regions Memory Unit.

Parallel processing unit

As shown in Figure 6, h_j is computed in Fuzzy Homogeneity Component – *FHC* – using the input vector *X* and its neighboring feature vectors F_i. The results then will be latched for synchronization. At last, the floating point comparator encoder computes *ArgMax* of h_j which has been done by *FHC*. The *output enable* of new region processing block will be activated by *l*=0. This block initiates parameters and increases the last region number. The N_i blocks compute the feature vectors of regions with new input vector parameters. The floating point multiplexer selects the region number and updated region vector that indicates the largest similarity to the input vector.

Fuzzy homogeneity component

This component executes Equation (5) in parallel manner as shown in Figure 7. Each *FHC* has one sub-tractor, one Look Up Table – *LUT*- and five multiplier operations in the propagation of output line. The adder/subtractor floating point functions use Adaptive Look Up Tables –*ALUT*- and two floating point multipliers from the FPGA fabric (Shirazi et al,. 1995). So, the outputs of multipliers will be valid after 5 pulses and the adder/subtractor will be ready after 11 pulses (Altra, Inc., 2006). Therefore, the maximum latency of h_j FHC's output is 37 pulses.

Implementation

The hardware architecture is designed by using Altera Quartus II software that is capable of implementing on the Stratix II EP1S30 device (Altra, 2006). The Altera devices are high performance and high speed which are major consumption of image processing. Each component of ALTRA devices has *ready* and *done* signals to be easily assembled into large scale and massed component designs (Altra, 2006). Table 2 shows the resources usage of

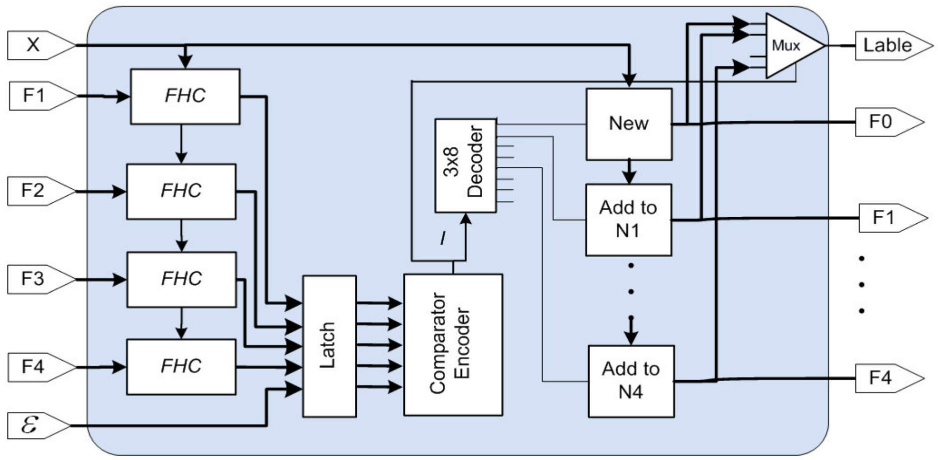


Figure 6. The Parallel Processing Unit finds out the most similar F_i to X and computes the new value of F_i.

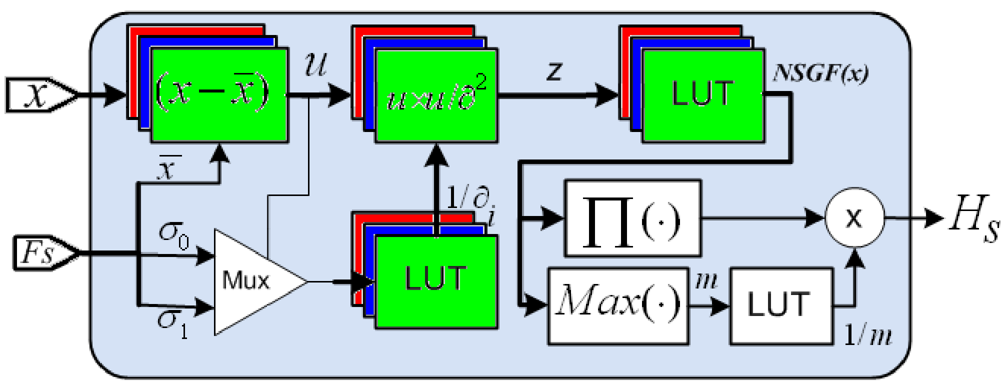


Figure 7. The FHC computes homogeneity measure of pixel X to feature vector F in three channels.

Table 2. Available and required elements.

Elements	EP1S30	Required
ALUT	35580	1530
ALM	71,160	3060
Embedded memory (bits)	3,317,184	1,689,600
18 x 18-bits multipliers	48	44
Clock Speed	300 MHZ	284 MHZ

designed system and availablity of EP1S30. To the peak performance calculation, the Altera device family consumes approximately 3% of the total utilization. Thus, by global clock in 300 MHz the maximum speed for processing a frame is 11.64 MHz. Therefore this architecture is capable of performing a frame size of 640x480 pixels less than 40ms.

QUANTITATIVE EXPERIMENTAL RESULTS

We provide a comparative environment to present the accuracy and effectiveness of proposed algorithm, with three unsupervised quantitative evaluators (Liu and Yang, 1994; Borsotti et al., 1998; Zhang et al., 2004) and also two well known segmentation algorithms (Comaniciu and Meer, 2002; Felzenszwalb and Huttenlocher, 2004). In the next subsections, the three objective evaluators are briefly described.

Liu and Yang's evaluation functions

Liu and Yang (Liu and Yang, 1994) proposed an objective quantitative evaluation function F for image segmentation as follow:

$$F(I) = \sqrt{N}\sum_{i=1}^{N}\frac{e_i^{\,2}}{\sqrt{A_i}} \tag{10}$$

Where N is the number of extracted regions, A_i is area of region i and *squared color error* of region i is defined as:

$$e_i^2 = \sum_{x \in \{r,g,b\}} \sum_{p \in R_i} (C_x(p) - \overline{C}_x(R_i))^2 \quad (11)$$

And

$$\overline{C}_x(R_i) = \left(\sum_{p \in R_i} C_x(p)\right) \Big/ A_i \quad (12)$$

Borsotti et al. (1998) evaluation functions

The developed function of F has been proposed by *Borsotti* and his colleagues (Borsotti et al., 1998). This function is named Q and is defined as follows:

$$Q(I) = \frac{1}{1000 \quad A_I} \sqrt{N} \sum_{i=1}^{N} \left[\frac{e_i^2}{1 + \log A_i} + (\frac{N(A_i)}{A_i})^2 \right] \quad (13)$$

Where $N(A_i)$ is the number of regions in the segmented image having an area of exactly A_i and A_I as the image size.

Zhang et al. (2004) entropy-based objective evaluation method

The objective segmentation evaluation method based on information theory has been proposed by Zhang (Zhang et al., 2004). In this method the entropy *E* as the basis for measuring the uniformity of pixel characteristics within a segmentation region is used.

$$E = H_l(I) + H_r(I) \quad (14)$$

Where

$$H_l(I) = -\sum_{i=1}^{N} \frac{A_i}{A_I} \log \frac{A_i}{A_I} \quad (15)$$

And the expected region entropy of segmentation *I* is:

$$H_r(I) = \sum_{i=1}^{N} (\frac{A_i}{A_I}) H(R_i) \quad (16)$$

And the entropy for region *i* is defined as:

$$H(R_i) = -\sum_{m \in v_i^{(v)}} \frac{L_i(m)}{A_i} \log \frac{L_i(m)}{A_i} \quad (17)$$

Where $L_i(m)$ denotes the number of pixels in region *i* that has a value of m for feature in the image.

The above objective evaluators obtain a measurable criteria of segmented image which can be used to judge between algorithms.

Comaniciu and Meer (2002) developed the mean shift-based segmentation algorithm. Their method is an appropriate development of discontinuity preserving smoothing algorithm. Each pixel in this algorithm has a significant mode of the joint domain density located in its vicinity. To produce segmented images with mean shift-base concept for comparison, the *EDISON* application developed by Robust Image Understanding Laboratory at Rutgers University is used. In order to generate the segmented images with diverse number of regions, we selected the *Spatial Bandwidth* parameter between 4 and 14 and also *Range Band width* parameter in the range of 4 and 17. To compare the results, another algorithm proposed by Felzenszwalb (2004) is used. In this algorithm, the prediction of evidence value was selected to determine a proper boundary in adjacent regions using a graph-based representation of the input image. A set of testing images with diverse number of regions is prepared by selecting *K* in [0.4, 1] and *sigma* from 100 to 1000. The mentioned algorithms as well as the proposed algorithm were applied to *House* and *Stone Wall* images. The results of the proposed algorithm on these images with varied number of extracted regions are shown in Figures 10 and 11.

The efficiency of algorithms (which are measured by aforementioned evaluators) on *House* and *Stone Wall* images with diverse number of segmented regions are shown in Figures 8 and 9, respectively. In all these diagrams, the smaller values of evaluators are more favorable considered. Although, the proposed approach emphasizes on designing a segmentation algorithm suitable for hardware implementation, the results show that the accuracy and quality of outcome of proposed method is plausible close to well known algorithms.

QUALITATIVE DISCUSSION

To illustrate the quality of result, we applied the proposed algorithm to natural scenes with diverse complexity (Figures 12, 13 and 14). These results show that we can perform this algorithm in wide range of application domain such as robotic systems, automatic home surveillance, military aims, industrial quality inspection, and weather forecasting and so on. Also, hardware module can be easily embedded in home automation systems explained in Perumal et al. (2008).

Conclusion

In this paper, a new method for extracting segments from images in a parallel fashion for video-rate application is presented. In this method, segments are extracted while images are being scanned in a raster fashion. To verify the uniformity in segments, the homogeneity measure

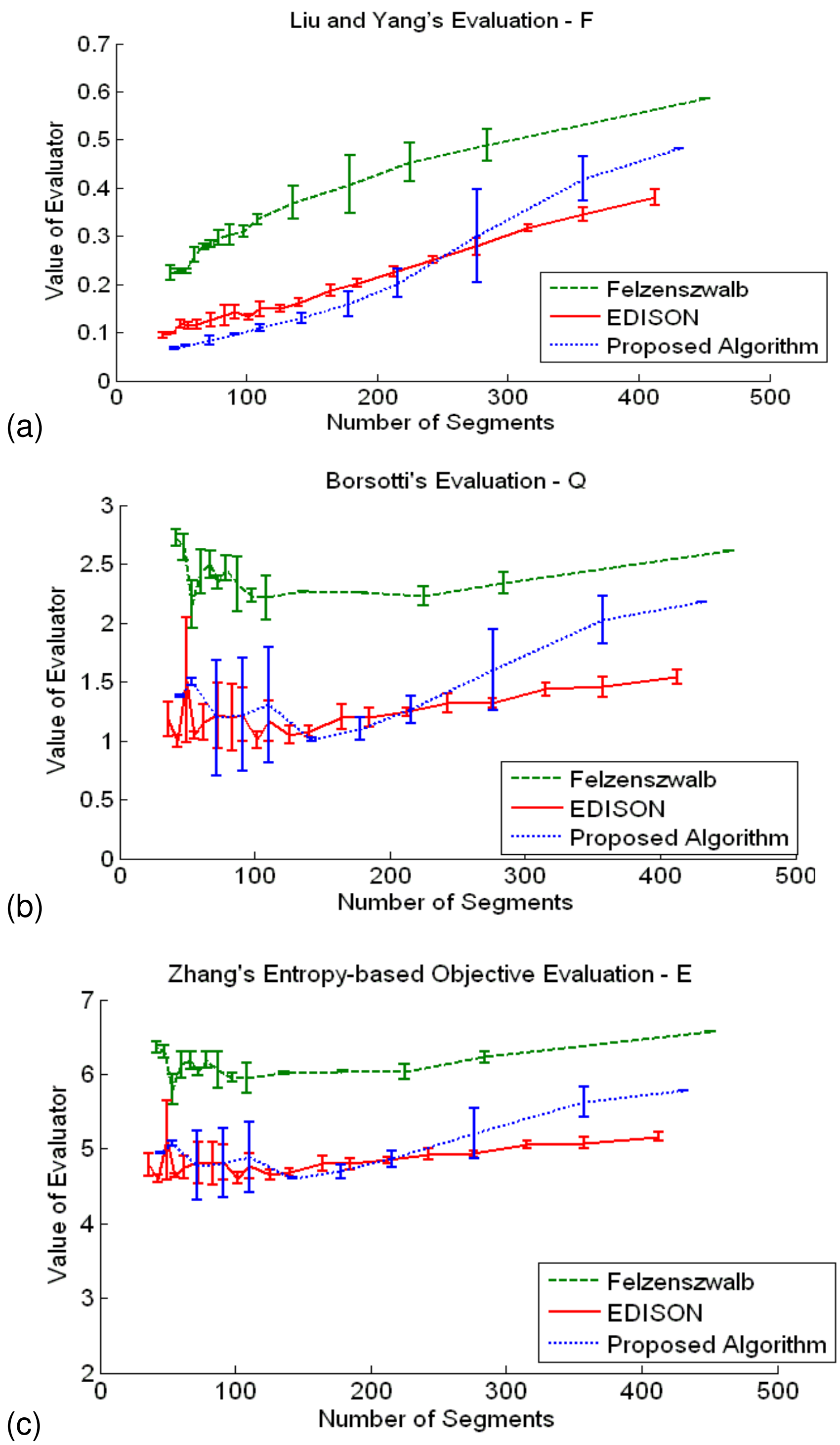


Figure 8. Result of applying (a) Liu and Yang evaluator (b) Borsotti et al evaluator and (c) Zhang et al entropy-based evaluator to the *House* image with diverse number of extracted segments.

based on fuzzy approach is used. For obtaining this measure, histograms of color components in arbitrary color model are estimated with non-symmetric Gaussian function to create fuzzy membership function. The membership functions are adapted automatically while the image is being scanned. This algorithm does not need multiple scanning and it is independent of color model. Moreover, it does not need *a priori* knowledge

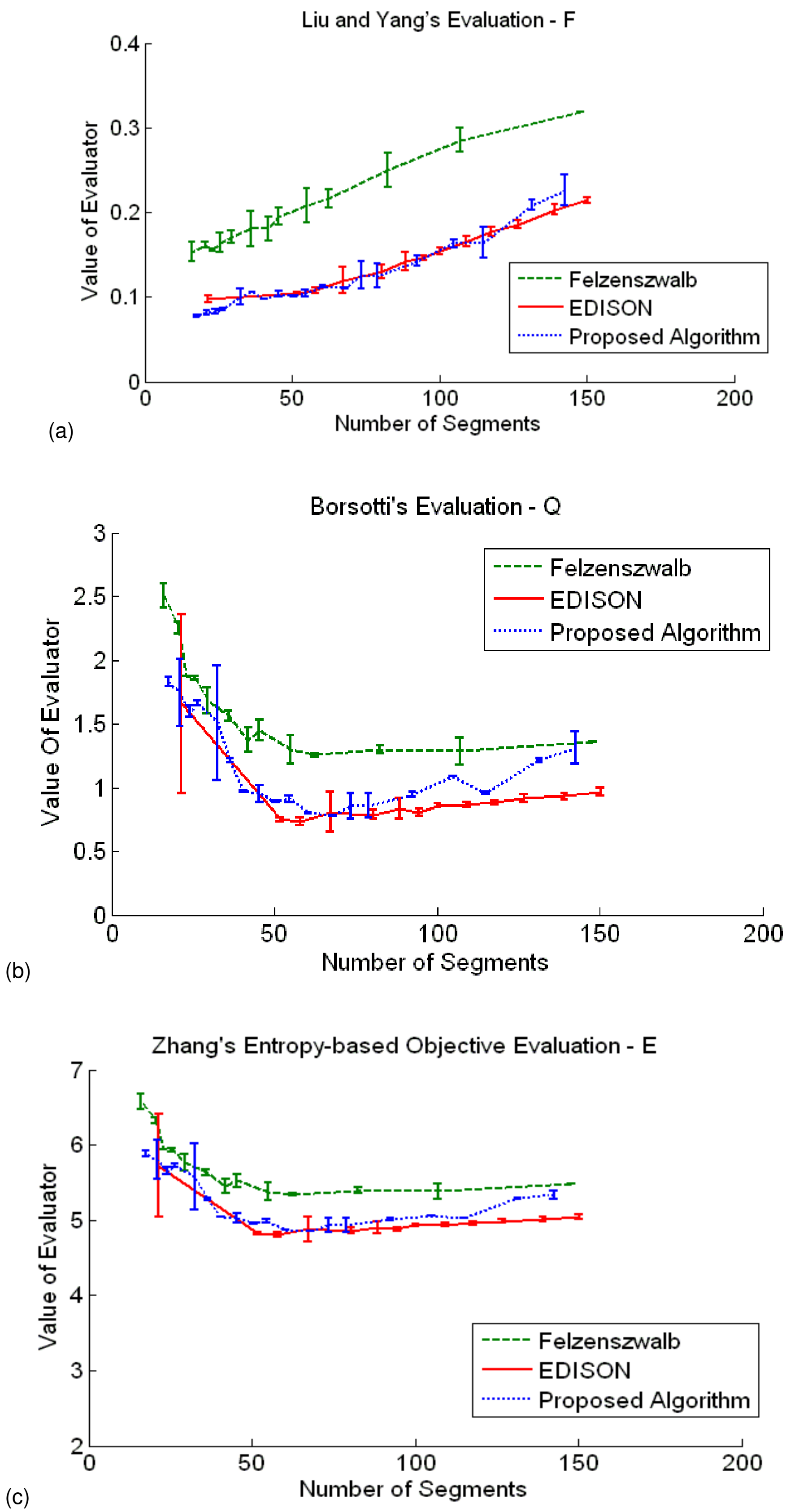
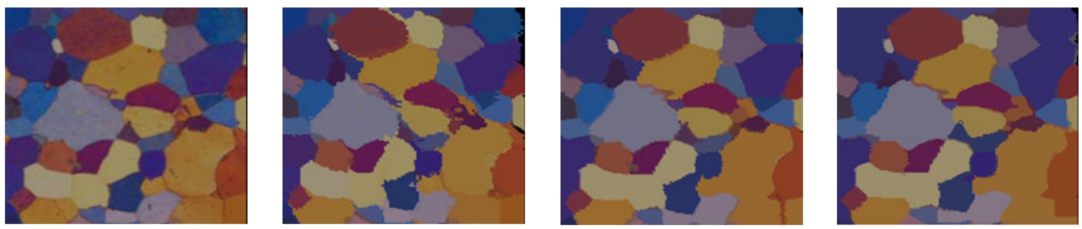


Figure 9. Result of applying (a) Liu and Yang evaluator (b) Borsotti et al evaluator and (c) Zhang et al entropy-based evaluator to the *Stone Wall* image with diverse number of extracted segments.

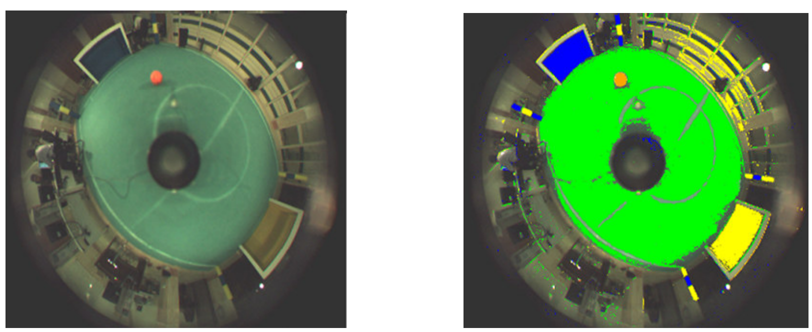
(a) (b) (c) (d)

Figure 10. (a) Original image of *"House"*, segmented images with varied number of extracted regions (b) 448 regions (c) 234 regions (d) 85 regions.



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

Figure 11. Original image of *"Stone Wall"*, segmented images with varied number of extracted regions (b) 144 regions (c) 81 regions (d) 62 regions.

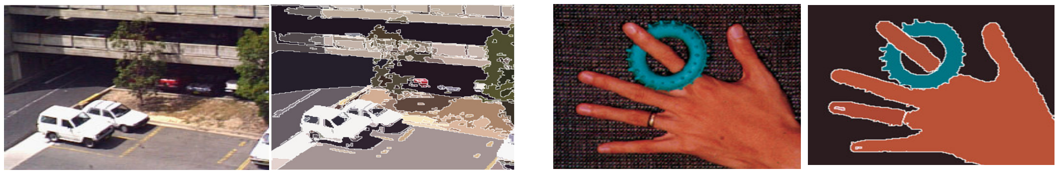


(a) (b)

Figure 12. (a) Original omni directional image of soccer robot field and (b) the image with Blue, Green, Yellow and Orange highlighted segments.

(a) (b)

Figure 13. (a) Original Image (Comaniciu and Meer, 2002), (b) Segmented image by proposed algorithm.



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

Figure 14. Original images (a), (c) (Comaniciu and Meer, 2002) and region boundaries delineated over the original images (b), (d).

such as seed points. Different experimental results show the effectiveness and robustness of this algorithm on various applications with complex images. Also, the designed hardware of this algorithm on FPGA shows that real- time limitation is fully covered.

ACKNOWLEDGMENT

This research was partially supported by Universiti Putra Malaysia under RUGS research grant No. 91148.

REFERENCES

Han J, Ma KK (2002). Fuzzy color histogram and its use in color image retrieval. IEEE Trans. Image Process., 11(8): 944-952.
Miteran J, Bouillant S, Bourennane E (2003). SVM approximation for real-time image segmentation by using an improved hyper rectangles-based method. Real-Time Imaging, 9(3): 179-188.
Perumal T, Ramli AR, Chui YL (2008). Design and implementation of SOAP-based residential management for smart home systems. IEEE Trans. Consumer Electronics, 54(2): 453-459.
Doignon C, Graebling P, Mathelin M (2005). Real-time segmentation of surgical instruments inside the abdominal cavity using joint hue saturation color feature. Real-Time Imaging, 11(5-6): 429-442.
Guvenc U, Demirci R, Karagul T (2010). Light refraction based medical image segmentation. Sci. Res. Essays, 5(10): 1127-1132.
Shirazi N, Walters A, Athanas P (1995). Quantitative analysis of floating point arithmetic on FPGA based custom computing machines. IEEE Symposium on FPGAs for Custom Computing Machines. California.
Meribout M, Nakanishi M, Ogura T (1999). A Real-Time image segmentation on a massively parallel architecture. Real-Time Imaging, 5(4): 279-291.
Draper BA, Beveridge JR, Böhm APW, Ross C, Chawathe M (2003). Accelerated image processing on FPGAs. IEEE Trans. Image Process., 12(12): 1543-1551.
Gillet A, Macaire L, Botte-Lococq C, Postaire J (2001). Color image segmentation by fuzzy morphological transformation of the 3d color histogram. 10th IEEE International Conference on Fuzzy Systems, 2: 824-824.
Liew AWC, Yan H, Law NF (2005). Image segmentation based on adaptive cluster prototype estimation. IEEE Trans. Fuzzy Syst., 13(4): 444-453.
Prados-Suarez B, Chamorro-Martinez J, Sanchez D, Abad J (2007). Region-based fit of color homogeneity measures for fuzzy image segmentation. Fuzzy Sets and Systems, 158(3): 215-229.
Trieu DBK, Maruyama T (2007). Real-Time image segmentation based on parallel and pipelined watershed algorithm. J. Real-Time Image Proc., 2(4): 319-329.
IEEE Standards Board (1995). IEEE standard for binary floating-point arithmetic. Technical Report ANSI/IEEE Std. 754.
Altra, Inc. (2006). Stratix Device Handbook. <www.altera.com/literature/hb/stx/stratix_section_1_vol_1.pdf >
Liu J, Yang YH (1994). Multi-resolution color image segmentation. IEEE Trans. Pattern Anal. Mach. Intelligence, 16(7): 689-700.
Borsotti M, Campadelli P, Schettini R (1998). Quantitative evaluation of color image segmentation results. Pattern Recogn. Lett., 19(8): 741-748.
Zhang H, Fritts J, Goldman SA (2004). An entropy-based objective evaluation method for image segmentation. In: Proceeding of SPIE-Storage and Retrieval Methods and Applications for Multimedia, pp. 38-49.
Comaniciu D, Meer P (2002) Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intelligence, 24(5): 603-619.
Felzenszwalb PF, Huttenlocher DP (2004). Efficient graph-based image segmentation. Int. J. Comput. Vision, 59(2): 167-181.
Protiere A, Sapiro G (2007). Interactive image segmentation via adaptive weighted distances. IEEE Trans. Image Process., 16(4): 1046-1057.
Rother C, Kolmogorov V, Blake A (2004). GrabCut-Interactive foreground extraction using iterated graph cuts. ACM Trans. Graphics, 23(3): 309-314.
Franke K, Koppen M, Nickolay B (2000). Fuzzy image processing by using dubois and prade fuzzy norms. in Proceeding of International Conference on Pattern Recognition.