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

6,900
Open access books available

186,000
International authors and editors

200M
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

A Design for Stochastic Texture Classification Methods in Mammography Calcification Detection

Hong Choon Ong and Hee Kooi Khoo
Universiti Sains Malaysia,
Malaysia

1. Introduction

A texture is a pattern represented on a surface or a structure of an object. Classifying a texture involves pattern recognition. Integration of texture classification is a one step ahead for machine vision application. For the past few decades until now, many researchers from various fields have been trying to develop algorithms to do texture discrimination. Alfréd Haar created the first discrete wavelet transform (DWT) in Haar (1911) which led to the development of the fast Fourier transform (FFT) and the other forms of DWT to detect periodic signals. Haralick had developed grey level co-occurrence probabilities (GLCP) with statistical features to describe textures (Haralick et al., 1973). The hidden Markov model (HMM) is another statistical method used for pattern recognition (Rabiner, 1989).

The classification process starts with image acquisition to retrieve the partial information carried by the test image. The general framework of texture classification is divided into two important stages, which are feature extraction and feature selection. Feature extraction is a process of transforming a texture into a feature domain based on the intensity distribution on a digital image. On the other hand, feature selection is a process of integrating a set of conditional statements and routines, so that the computing system can logically decide which pixels belong to which texture.

The texture classification methods are the tools which have been used to assist in medical imaging. For example, to classify the breast cancer tumour in Karahaliou et al. (2007) based on mammography imagery, to detect abnormalities in patients using Magnetic Resonance Imaging (MRI) imagery (Zhang et al., 2008), and precisely segmenting human brain images on Computed Tomography (CT) imagery for visualization during surgery (Tong et al., 2008).

In food science, texture classification techniques aid on improving the quality of food and decrease the rate of food poisoning cases. For instance, a non-supervised method of texture classification is proposed to estimate the content of Intramuscular Fat (IMF) in beef on Du et al. (2008) and thus improving the meat quality.

In remote sensing, texture classification is used to analyze satellite Synthetic Aperture Radar (SAR) imagery for multiple purposes. The analyzed satellite imagery might be used to monitor flood situations (Seiler et al., 2008), large scale construction planning, archeological research, weather forecasting, geological study etc. Texture segmentation is also applied in

the study of global warming by detecting and estimating the size of the icebergs at the North Pole from time to time (Clausi, 2002a).

Another breakthrough in the advancement of security is in biometric authentication. One of the biometric measures is in using the human eye as an identity of a person. The human iris contains texture patterns which are unique in nature and thus texture classification techniques can be applied on this iris imagery (Bachoo & Tapamo, 2005).

The first section has briefly introduced the texture classification and its applications. We highlighted the importance of texture classification in science. A literature review on the current existing methods is discussed in Section 2 and 3. The main challenge of texture classification is pointed out. Section 4 explains the theories behind our research methodologies. The non-parametric statistics are developed and used to generate statistical features from a given textured image. In the feature extraction section, we will provide a general formulation for our proposed method called cluster coding and explain briefly the parameters being used for texture classification purposes. Furthermore, the solution to solve the misclassification problem is also provided. A practical analysis on the digital mammogram using our proposed method is shown in Section 5. We will conclude with a summary of our contributions in Section 6.

There are four goals in this chapter which are as follows: -

a.   To show how statistical texture descriptions can be used for image segmentation.
b.   To develop algorithms for feature extraction based on texture images.
c.   To develop algorithms for texture segmentation based on texture features analyzed.
d.   To illustrate the above objectives using an application on a mammogram image.

2. Feature extraction

There are several mathematical models which are used to extract texture features from an image. This section briefly discusses the five main feature extraction methods, namely autocorrelation function (ACF), Gabor filter, DWT, HMM, and GLCP. Besides, some combination of methods using different models are also developed to increase the robustness of texture segmentation system. We will briefly discuss two hybrid methods; these are Gabor wavelet and wavelet-domain HMM.

Gabor filter is one of the texture descriptors based on the Fourier transform (FT). A discrete FT is first taken from the test image to generate a two dimensional (2D) sinusoidal signal. For texture recognition, the Gabor filter bank contains a list of sub-bands of different signals generated by different textures. After the sub-bands are determined, the 2D signal of test image will multiply one of the chosen sub-bands and yield only the frequencies that match the sub-band. The product is then transformed back by taking the inverse FT and this leaves only the location of the texture feature which matches the signal. The process continues with each possible sub-band and produces the locations where the same signals occur (Petrou & Sevilla, 2006).

Gabor filter is useful in adapting sinusoidal signals whereby it can be decomposed into a weighted sum of sinusoidal signals. Thus Gabor filter is suitable to decompose textural information. Experiments done by Clausi & Deng (2005) stated that the Gabor filter can well recognize low and medium frequencies, but it produces inconsistent measurements for high frequencies due to the noise in the signal. The feature domain generated by Gabor filters are

not distinctive enough for high frequencies and thus could affect the segmentation result (Hammouda & Jernigan, 2000).

Texture pattern can also be modeled as a transitional system using HMM. A Markov model assumes a texture pattern have a finite number of states and times. Each probability of a state is determined by the previous probability of the state. Three issues can arise from HMM observations; these are evaluation, decoding, and learning. HMM evaluation is to compare the probabilities of different models which best describe the texture feature. HMM decoding is to decompose and provide an estimated basis of texture patterns based on the HMM observations. The HMM learning searches for which model parameters best describe the texture pattern (Sonka et al., 2007).

The HMM can generate a consistent measurement for texture patterns based on the best probabilities. However, one or more observations which produce undesired probabilities could generate disorderly sequences due to the noisy pattern in the test image (Sonka et al., 2007).

ACF is another method to describe texture patterns. The function helps to search for repeated patterns in a periodic signal. The function also identifies the missing basis of texture patterns hidden under the noisy patterns. In using the ACF, the mean of each image is adjusted before applying the general formula. Thus we are actually computing the normalized auto-covariance function. One can characterize a texture pattern by inferring the periodicity of the pattern (Petrou & Sevilla, 2006).

The ACF feature is well demonstrated and distinctive between textures on a three dimensional (3D) graph. In feature selection, inferring the periodicity of a texture feature is done by observing several threshold points of the auto-covariance function and then counting the number of peaks for each threshold in a fixed variation. This may result in a random fluctuation and texture segmentation may fail because of two issues; there is not enough information by taking only one dimensional (1D) threshold to compare and an appropriate set of standard deviation of the distances between peaks are needed to know when the periodicity end. For example, the lagged product estimator and time series estimator are proposed to select ACF feature in Broerson (2005). But to appropriately characterize the texture pattern by its periodicity is still an active area of research.

Instead of using the Gabor filter for feature extraction, wavelet is well known today as a flexible tool to analyze texture. Wavelet is a function whereby the basic function, namely the mother wavelet is being scaled and translated in order to span the spatial frequency domain. The DWT is done by the product of a corresponding signal generated by a pattern and the complex conjugate of the wavelet, and then integrating over all the distance points conveyed by the signal. In texture analysis, a tree data structure namely a packet wavelet expansion is used to split the signal into smaller packets and expand a chosen band at each level of resolution (Petrou & Sevilla, 2006).

The DWT can become rotation invariant in texture analysis by taking the logarithm of the frequency sub-band. However this will damage the frequency components of the corresponding signal generated by a texture pattern and thus may obtain inaccurate result (Khouzani & Zadeh, 2005).

GLCP method is a non-parametric solution whereby the textures are described in a discrete domain (Petrou & Sevilla, 2006). GLCP statistics are used to preserve the spatial characteristics of a texture. The selection of certain texture is possible as it is based on the

statistical features. The best statistical features that are used for analysis are entropy, contrast, and correlation (Clausi, 2002a). However, further analysis in Jobanputra & Clausi (2006) shows that correlation is not suitable for texture segmentation. GLCP statistics can also be used to discriminate between two different textures. Boundaries can be created from the shift on statistical feature while moving from one texture to another (Jobanputra & Clausi, 2006). Clausi & Zhao (2002) also proposed grey level co-occurrence linked-list (GLCLL) structure and grey level co-occurrence hybrid histogram (GLCHH) structure in Clausi & Zhao (2003) to this non-parametric solution for storing purpose in order to speed up the computational time for GLCP feature extraction.

Wavelet-domain HMM can be described as a finite state machine in the wavelet domain. The hidden Markov tree (HMT) can identify the characteristics of the joint probabilities of DWT by capturing the scale dependencies of wavelet co-efficient via Markov chains (Fan & Xia, 2003). Since this method is based on the wavelet domain, the disorder sequence of signal generated by the noise in the texture patterns may weaken the cross-correlation between DWT sub-bands (Fan & Xia, 2003). Ming et al. (2008) proposed the wavelet hidden-class-label Markov random field to suppress the specks of noise, but there are some small blobs of noises still appearing in the results of the segmented imagery.

3. Feature selection

The texture feature selection stage is the most important part of the texture segmentation process because it determines which pixels belong to which texture of an image. The use of parameters of a chosen method is crucial to the output of the result.

K-means algorithm has become a popular clustering method which is used for pattern recognition. Given the number of clusters, K, the algorithm will start at a random K number of centres. Then, each of the centres will group the features using the closest distances or Euclidean distance measures. The locations of the features with the same cluster will determine the new centre for each cluster. The process will then repeat until the centre of each texture class remains the same.

K-means algorithm assumes that all clusters are in spherical shape, but it may return inappropriate result for non-spherical clusters (Clausi, 2002b). In the real life 2D feature set is not always in spherical shape and normally the number of classes is unknown.

Support vector machines (SVM) algorithm is a slow but highly accurate clustering method. The SVM training algorithm was introduced by Boser et al. (1992). The purpose of SVM is to map feature vectors into a higher dimensional feature space, and then creating a separating hyperplane with maximum margin to group the features. Support vectors (SVs) contain highlighted pixels that help to create the margins or boundaries in an image. The higher dimensional space is defined by a kernel function. Some of the popular kernels are shown in Schölkopf & Smola (2002). A combined GLCP and SVM technique has been proposed for two class segmentation with significant result in Khoo et al. (2008).

Expectation-maximization (EM) algorithm is a statistical estimation algorithm used for finding maximum likelihood and estimates of parameters in probabilistic models. The parameters involve means, variances, and weights (Tong et al., 2008). The EM algorithm starts by initializing the parameters to compute the joint probability for each cluster. The algorithm then iterates to re-estimate the parameters and maximize the probability for each cluster until the set of convergence values of probabilities are obtained (Bishop, 2006). The convergence

probabilities are only dependent upon the statistical parameters, so we must carefully choose these parameters, especially the parameters for texture patterns (Diplaros et al., 2007).

The self-organizing map (SOM) is an unsupervised single layer artificial neural network (ANN). In the SOM training environment, a digital image is mapped as a grid. A set of neurons will be placed at random grid points where each neuron is stored as a cluster centre (Chen & Wang, 2005). SOM clusters regions which have similar pattern and separates the dissimilar patterns based on a general distance function (Martens et al., 2008). The SOM learning process is similar to the K-means clustering where it iterates until each of the cluster centre converges to the centre of the possible texture patterns.

The advantage of SOM is that the more number of neurons are placed in the grid, the higher classification result will be obtained. However, if the numbers of neurons are too large, SOM may end up with over classification (Martens et al., 2008). On the other hand, the numbers of neurons required is unknown. Furthermore, the classification using SOM may fail at the local minimum in training (Abe, 2005).

4. Methodology

A texure pattern can be identified by

a. Counting the number of repetition of its primitive,
b. Average brightness of the primitive, and
c. The structural distribution of its primitive.

In a stochastic texture, it is not possible to determine (a), (b), and (c). To classify this texture, a measure of its density has to be computed. Therefore, the statistical modelling of GLCP is used to extract textural features. A general classification approach, namely the cluster coding is proposed. This approach preserves the textural information of a given textured image.

4.1 Enhanced two-dimensional statistical features

Given a textured image, the grey level intensities of the pixel set can be retrieved. A grid with equal sized cells is fitted onto the image. For each cell, the grey level intensities of the local neighbourhood pixels are taken into consideration when computing the joint probabilities density function. The Equation 1 shows the statistical feature for a cell, $v(i,j)$ based on the joint probability density function. i and j are the positions of a cell on the image.

$$v(i,j) = \sum_{s,t=0}^{G-1} \left(\frac{F(s,t)}{\sum_{s,t=0}^{G-1} F(s,t)} \right) \tag{1}$$

where $F(s,t)$ represents the frequency of occurrence between grey levels, s and t. G is the quantized grey levels which forms a $G{\times}G$ co-occurrence matrix.

Histogram equalization is applied to achieve higher separability between to different textures in the image. The feature set, v is distributed in a frequency histogram in order to calculate the cumulative distribution function, cdf_k using Equation 2,

$$cdf_k = \begin{cases} f_k & , \quad k = 0 \\ cdf_{k-1} + f_k, & \quad k \in \{1,2,...,G-1\} \end{cases} \tag{2}$$

The general definition for histogram equalization (Gonzalez & Woods, 2006), H_k is

$$H_k = \frac{cdf_k - cdf_{\min}}{t - cdf_{\min}} \times (G-1), \ k \in \{1,2,...,G-1\} \tag{3}$$

where cdf_k, a cumulative distribution function is the total number of feature sets and min is the minimum for the set of cumulative distribution function. The design process is shown in Figure 1.
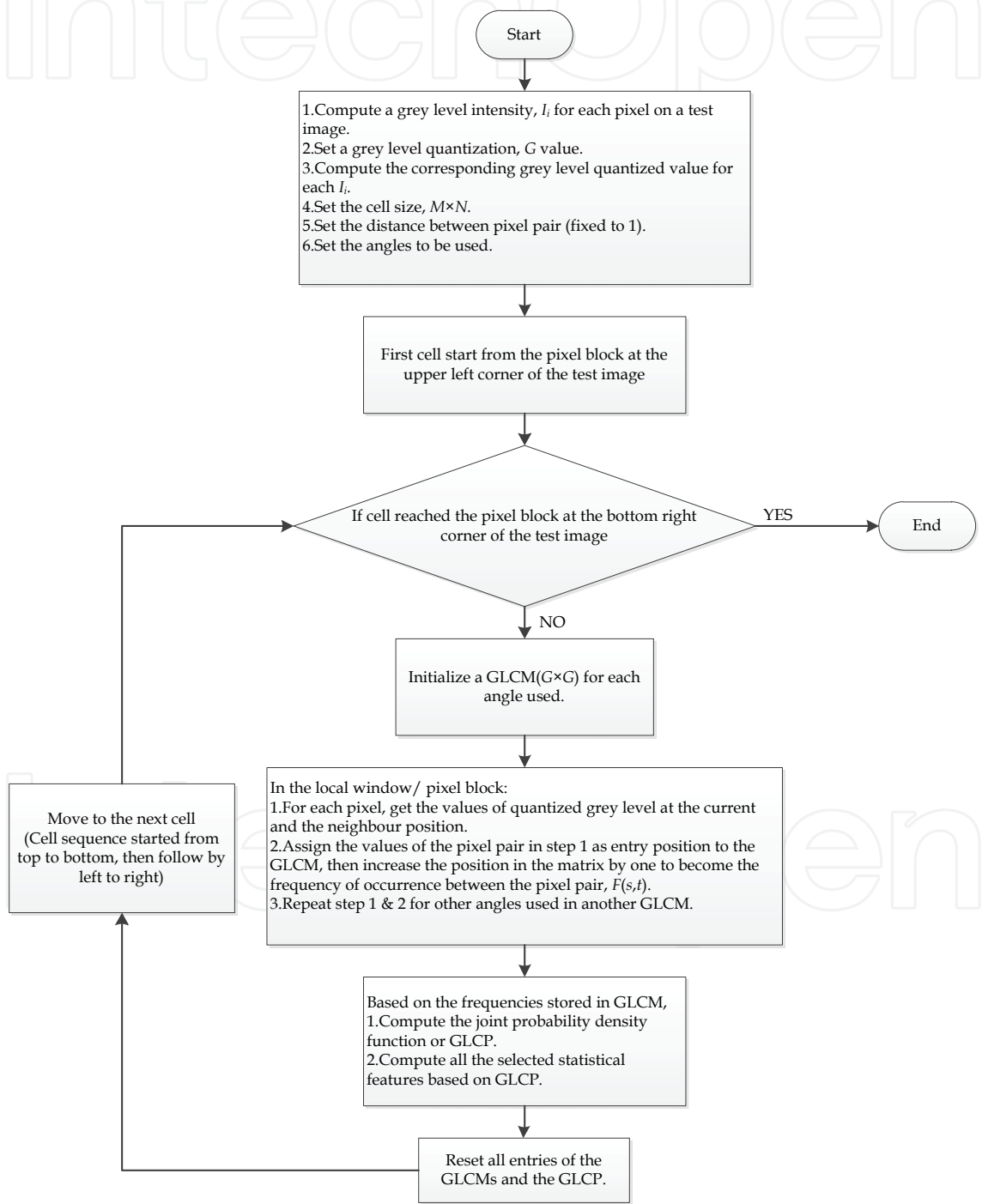
```
                              ┌─────────┐
                              │  Start  │
                              └─────────┘
                                   │
        ┌──────────────────────────────────────────────────────┐
        │ 1.Compute a grey level intensity, Iᵢ for each pixel on a test │
        │ image.                                                        │
        │ 2.Set a grey level quantization, G value.                     │
        │ 3.Compute the corresponding grey level quantized value for    │
        │ each Iᵢ.                                                       │
        │ 4.Set the cell size, M×N.                                     │
        │ 5.Set the distance between pixel pair (fixed to 1).           │
        │ 6.Set the angles to be used.                                  │
        └──────────────────────────────────────────────────────┘
                                   │
        ┌──────────────────────────────────────────────────────┐
        │    First cell start from the pixel block at the              │
        │       upper left corner of the test image                    │
        └──────────────────────────────────────────────────────┘
```

Fig. 1. The flowchart of the GLCP algorithm for feature extraction.

Flowchart boxes:

Start

1. Compute a grey level intensity, I_i for each pixel on a test image.
2. Set a grey level quantization, G value.
3. Compute the corresponding grey level quantized value for each I_i.
4. Set the cell size, $M \times N$.
5. Set the distance between pixel pair (fixed to 1).
6. Set the angles to be used.

First cell start from the pixel block at the upper left corner of the test image

If cell reached the pixel block at the bottom right corner of the test image — YES → End

NO

Initialize a GLCM($G \times G$) for each angle used.

In the local window/ pixel block:
1. For each pixel, get the values of quantized grey level at the current and the neighbour position.
2. Assign the values of the pixel pair in step 1 as entry position to the GLCM, then increase the position in the matrix by one to become the frequency of occurrence between the pixel pair, $F(s,t)$.
3. Repeat step 1 & 2 for other angles used in another GLCM.

Based on the frequencies stored in GLCM,
1. Compute the joint probability density function or GLCP.
2. Compute all the selected statistical features based on GLCP.

Reset all entries of the GLCMs and the GLCP.

Move to the next cell (Cell sequence started from top to bottom, then follow by left to right)

Fig. 1. The flowchart of the GLCP algorithm for feature extraction.

4.2 Cluster coding classification

Figure 2 demonstrates the basic concept of the cluster coding algorithm. The classification process is run on a split-and-merge operation. The splitting part is shown in Figure 2(b) and 2(c), while the merging part is shown in Figure 2(d).
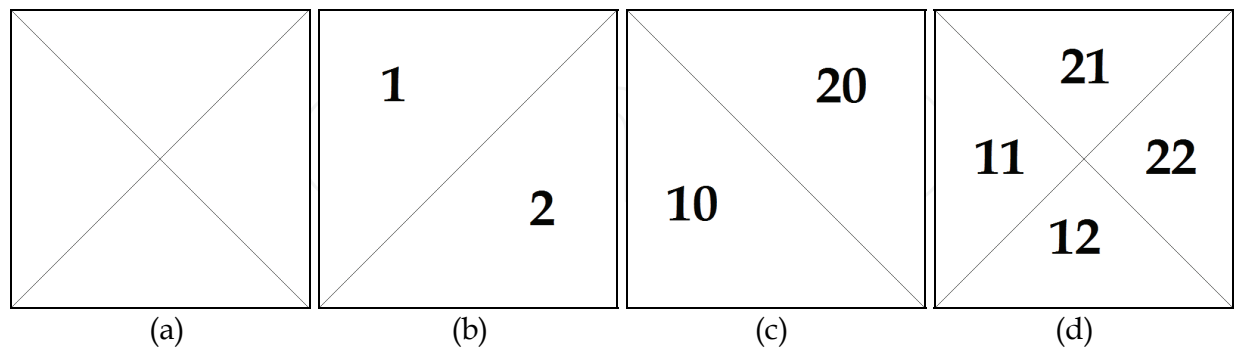

(a) (b) (c) (d)

Fig. 2. Cluster coding conceptual diagram. (a) Desired segments. (b) Feature A coded in 2 clusters. (c) Feature B coded in 2 clusters. (c) Summation of codes from (b) and (c).

Each feature set is split up into coded clusters using the mean shift estimator (MSE). Mean shift estimator locate the local maxima of a feature set. The MSE is defined as,

$$MSE(v(i,j)) = \frac{K\left[|v(i,j) - v(i+\delta_1, j+\delta_2|)\right]v(i,j)}{K\left[|v(i,j) - v(i+\delta_1, j+\delta_2)|\right]} \tag{4}$$

where K is the smoothing kernel function. δ_1 and δ_2 are the shifting parameters of $v(i,j)$. The mode seeking of one-dimensional version of feature set is illustrated in Figure 3. In the figure, the dotted line is the splitting point at the position 3.5. The flowchart for cluster coding is shown in Figure 4.
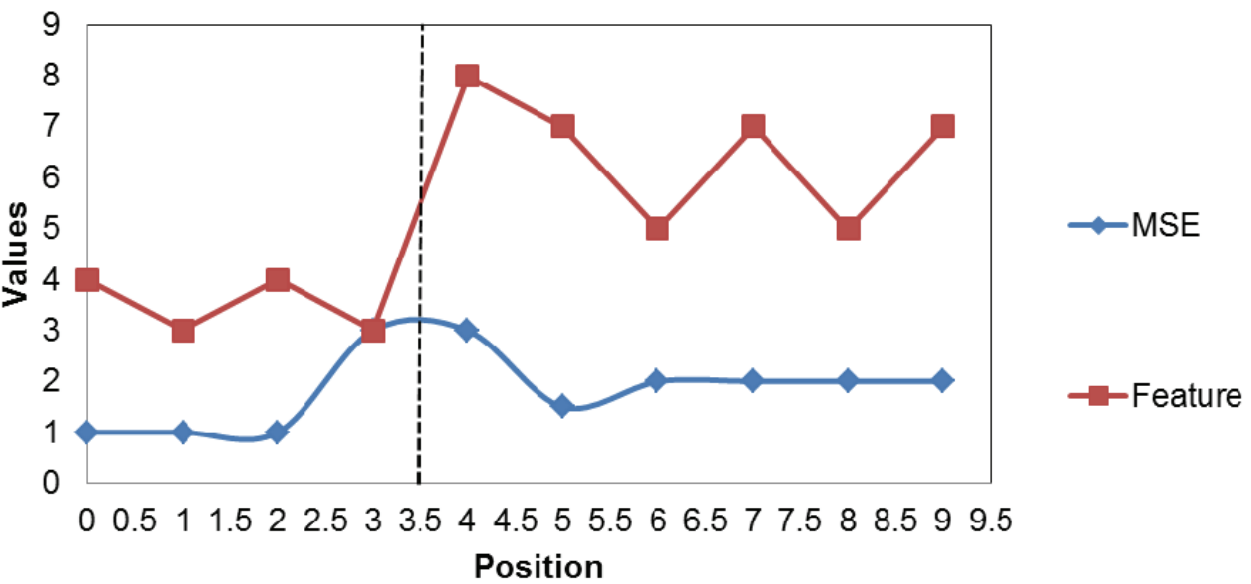
Fig. 3. Line graph of the feature values and the corrresponding MSE.

Start

1. Determine feature to be used
2. Configure a suitable threshold using MSE for each domain.
3. Split each feature into 2 clusters.

1. Each regions assigned with a unique code.
2. Sum up all codes from each region in every feature.

The initial block start at the top left of the image.

Check if the current cell reached the bottom right of the image — YES

NO

Move to the next cell (Cell sequence started from top to bottom, then follow by left to right)

Check for the first code — NO → Check if codes exist — YES

YES   NO

1. Create a new cluster.
2. Plus 1 count for the population of the new cluster.

Plus 1 count for the population of the existing cluster.

Sort each cluster in descending order (Start from the biggest population of the cluster to the smallest population of the cluster)

Check if there exist clusters with small population. — YES → Assign a special label for these clusters as undetermine cluster.

NO

End

Fig. 4. The flowchart of the cluster coding algorithm.

4.3 Correction for misclassification

Each classified region on the image is identified. Two rules are set, in order to examine the region as to whether it is a misclassified blob or a texture class on the image. If the region is the texture class, then

Rule I: The located region has achieved the minimum population, and
Rule II: There is no dominated population encircled by it's neighbourhood region.

The region reclassifies if there exist a dominated population surrounding its neighbourhood region. Figure 5 provides the detail flowchart for this approach.
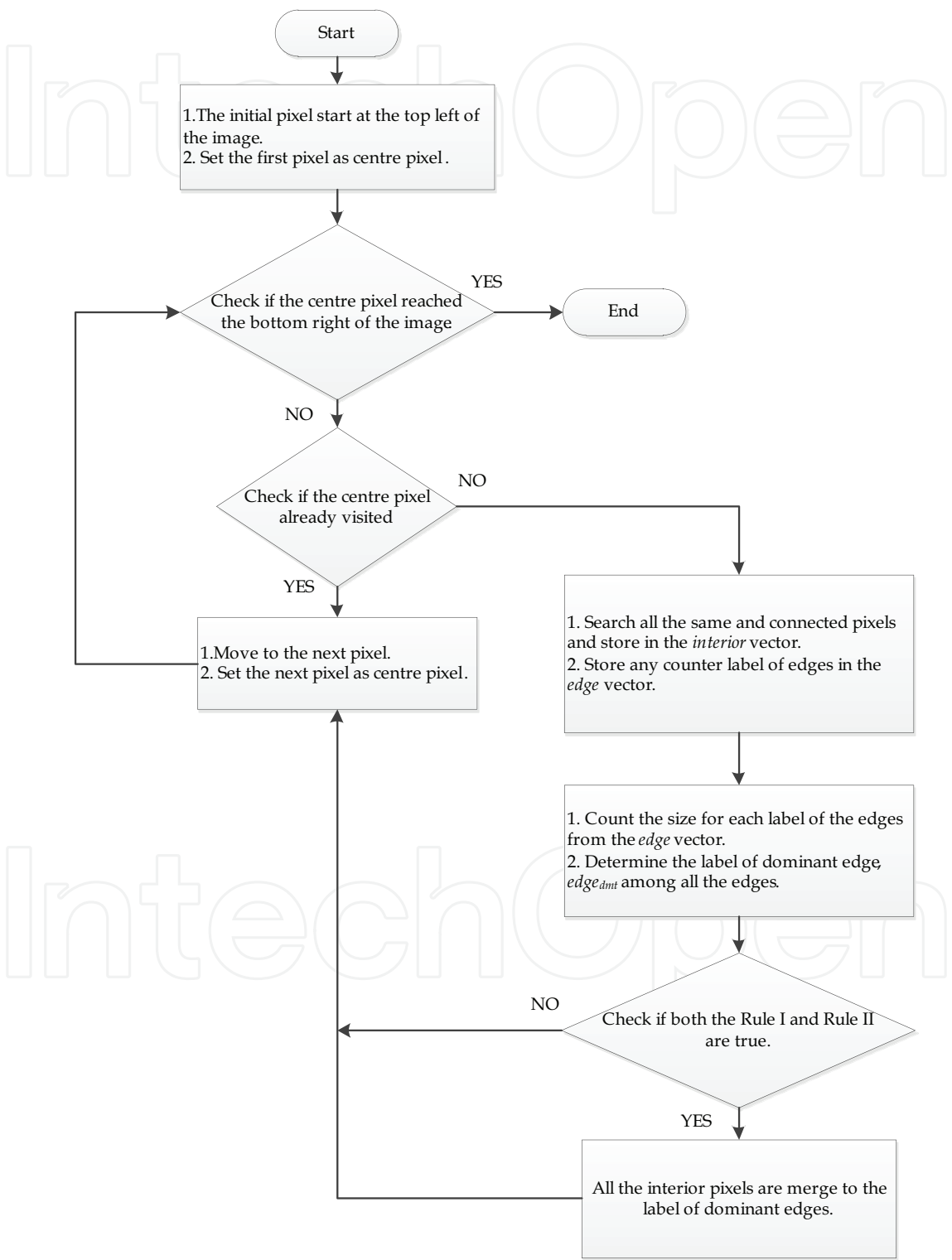
Fig. 5. The flowchart of the modified flood fill algorithm for cluster merged.

5. Results and applications

The design in Section 4 is developed using C++ programming language and implemented on a computer with the specification of 1.66GHz Intel Core 2 Dual processor, 3.37GB DDR2 RAM memory, and 256MB nVidia Quadro FX 1500M GPU for image display. This section is divided into three subsections; the initial tests using artificially created texture images followed by the application background and the application results.

5.1 Initial test

An initial testing procedure is being carried out to verify the design mentioned in Section 4. Three Brodatz's textures are used to fill the regions as can be seen in Figure 6(a). The oriental rattan (D64) is fitted in the Times New Roman font type of the character 'S'. The non-linearity of the 'S' shape in Figure 6(a) is intentionally created to add the complexity of the classification process.
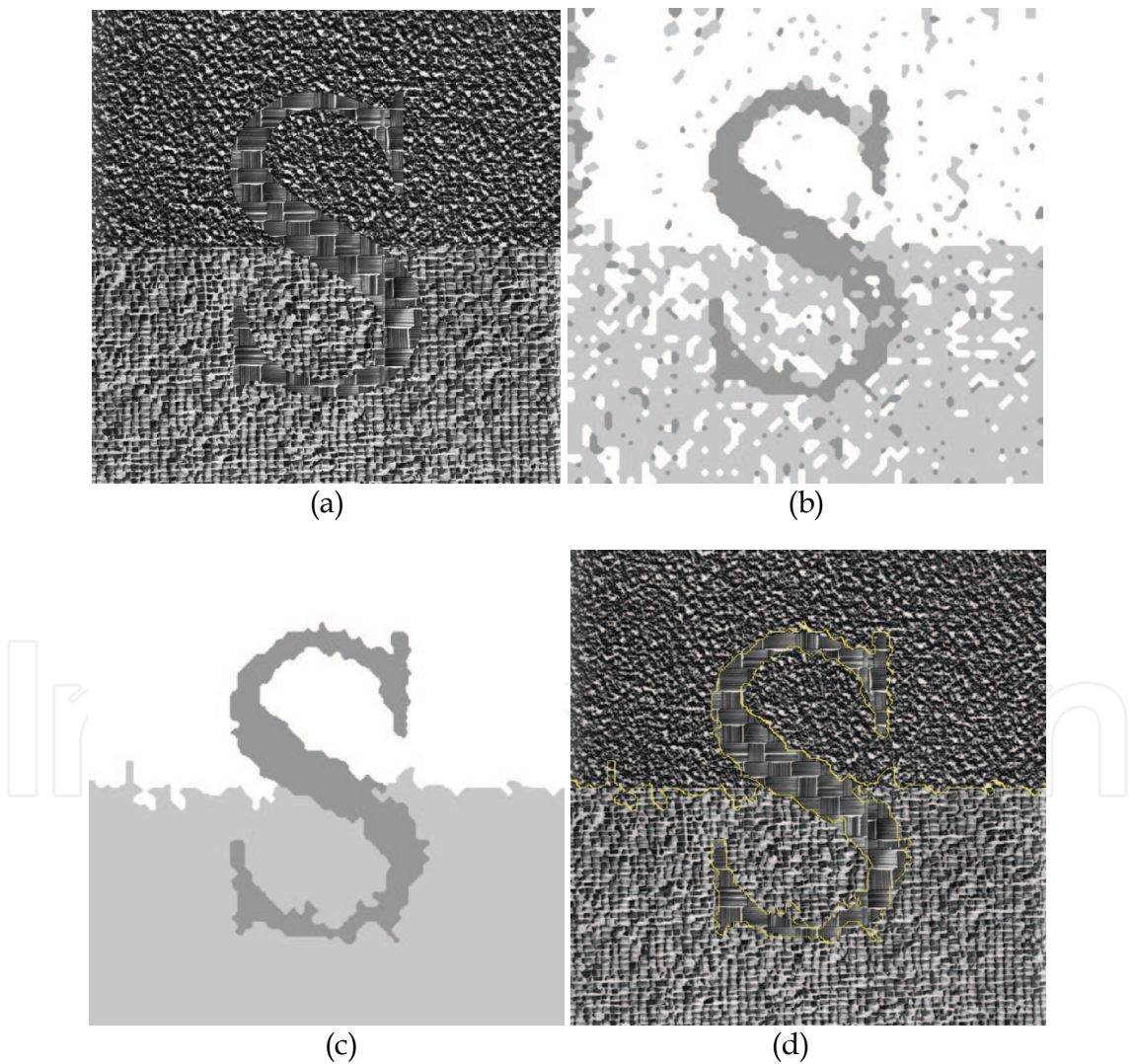
Fig. 6. Brodatz's textures classification test. (a) Texture mosaic contains handmade paper (D57), oriental-rattan (D64), and raffia (D84). (b) Classification with cluster coding. (c) Blobs correction with Rule I and II in Section 4.3. (d) Edge detection based on (c).

Although there are misclassfied blobs in Figure 6(b), the reclassification has effectively corrected the entire classified image as illustrated on Figure 6(c). Figure 6(d) is the final classification result based on the edge detection in Figure 6(c). The edges in Figure 6(d) is captured by detecting the boundaries between the different classes using both the vertical line and horizontal line scans.

Test Data	Accuracy (%)	
	Euclidean Distance K-means Clustering	Proposed Method
Bi-dimensional data sets, Hammouche, et al. (2006)	78.7178	97.2484

Table 1. Comparative performance with K-means Clustering.

We also perform a quantitative comparison with the Euclidean distance based K-means clustering (EDKMC) using the image data sets from Hammouche, et al. (2006) with a summary of the accuracy shown in Table 1. The proposed method has achieve less classification errors with only 2.7516% as compared to EDKMC errors with 21.2822%.

5.2 Mammographic scanner

Mammographic scanner is a specialized device which is used to detect early stage of breast cancer. The device makes use of a low dose X-ray, which is a safer way to make diagnosis as compared to other vision machines which use higher doses of radiation, such as normal X-ray and Positron Emission Tomography (PET).

Besides, the introduction of digital mammography has made it possible to acquire image data from the electrical signals of the device for analysis. The classification algorithms can be integrated to analyse the image data with the computer.

The device still emits ionizing radiation, which could cause mutations of human cells. Therefore an appropriate design of algorithm is crucial to reduce false positive of diagnosis and at the same time reduces the frequencies of scanning.

5.3 Mammogram results and analysis

Three possible masses can be seen in the mammogram, which are the calcification clusters, cysts, and fibroadenomas. Table 2 explains the details of these masses. The masses need to have a multiple viewing observations because the ductal carcinoma or cancer cells might be hidden beneath them. Figure 7 shows the examples of masses appearance in mammograms. Further diagnosis can be found in Breast cancer - PubMed Health (2011).

Masses	Description
Calcification clusters	Calcium salts crystalized in the breast tissue.
Cysts	Fluid-filled masses.
Fibroadenomas	Movable and rounded lumps.

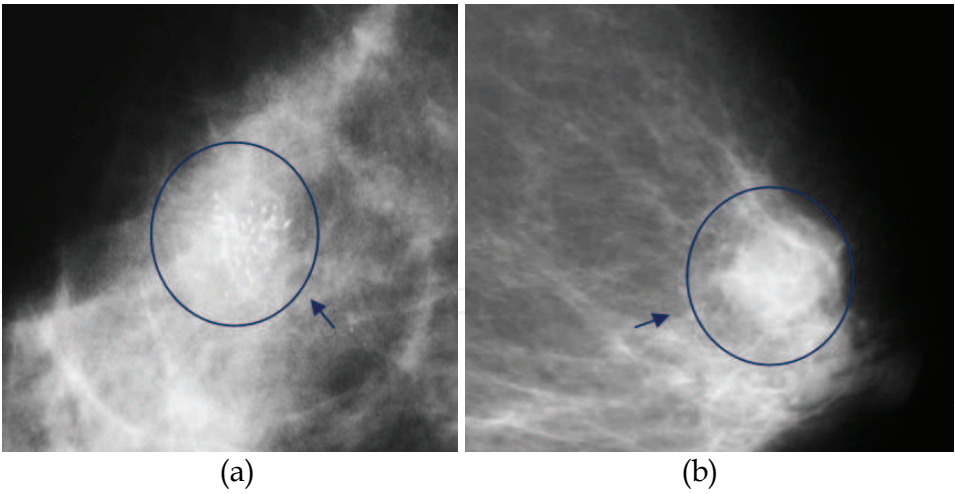Table 2. The common masses appearance in mammogram.

Fig. 7. Masses appearance in mammograms (Ribes, et al., 2008). (a) A sample of microcalcification clusters. (b) A sample of simple cyst.
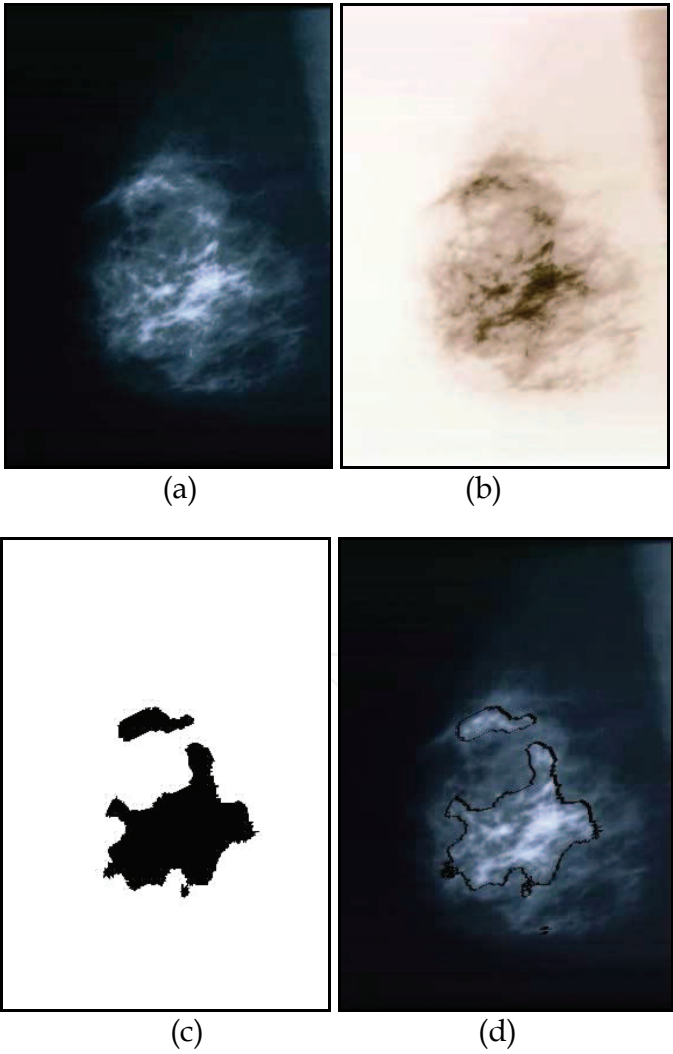

Fig. 8. Mammogram test. (a) A mammogram captured from a woman's left breast (BreastCancer.org, 2011). (b) Image negative. (c) Areas of macrocalcification clusters using the approach in Section 4. (d) Edge detection based on (c).

In Figure 7, doctors may have difficulty to spot cancer cells in these masses, particularly the calcification clusters in a mammogram. This is because the areas contain textural information which are random in nature. Statistical features can be used to measure the correlation between the breast tissues and the masses.

Figure 8(a) shows a women's left breast. Notice that the breast contains calcification clusters scattered around the tissues and there is no sign of cyst or fibroadenomas, as appeared in the mammogram.

Negative image is first obtained to remove the unnecessary dark areas in the mammogram (Gonzalez & Woods, 2006), as shown in Figure 8(b). The $v(i, j)$ of the enhanced-GLCP is then computed to produce 2D feature space for classification. Finally, the classification is carried out using the proposed cluster coding algorithm.

The propose method highlights the areas of calcification which have the potential spread of cancer cells. The end result in Figure 8(d) is optimum since the areas of calcification are in the closed boundaries.

6. Conclusion

The semantic of cluster coding with MSE is easy to understand and program. The enhanced 2D statistical features have obtained separability between different textures using histogram equalization. The design has been successfully applied in the mammogram for calcification clusters detection purpose. This approach can help the doctor or radiologist to focus on the critical areas of the mammogram.

7. Acknowledgment

8. References

Abe, S. (2005). *Support vector machines for pattern classification*, Springer Science+Business Media, LLC, ISBN 978-1849960977, Spring Street, New York

Bachoo, A. K. & Tapamo J. R. (2005). Texture detection for segmentation of iris images, *Proceedings of SAICSIT 2005 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries*. pp.236-243, ISBN 1-59593-258-5, South African Institute for Computer Scientists and Information Technologists, Republic of South Africa, 2005

Bishop, C. M. (2006). *Pattern recognition and machine learning*, Springer Science+Business Media, LLC, ISBN 978-0387310732, Spring Street, New York

Boser, B. E.; Guyon, I. M. & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers, *Proceedings of COLT 1992 5th Annual ACM Workshop on Computational Learning Theory*, pp.144-152, ISBN 0-89791-497-X, New York, United States, 1992

Breast cancer - PubMed Health (July 2011). Available from http://www.ncbi.nlm.nih.gov/pubmedhealth/PMH0001911

BreastCancer.org. (July 2011). Available from
        http://breastcancer.org

Broerson, P.M.T. (2005). The uncertainty of measured autocorrelation functions, *Proceedings of IEEE AMUEM 2005 International Workshop on Advanced Methods for Uncertainty Estimation in Measurement*, pp.90-95, ISBN 0-7803-8979-4, Niagara Falls, Ontario, Canada, May 13, 2005.

Chen, C. H. & Wang, P. S. P. (2005). Handbook of pattern recognition and computer vision, In: Statistical Pattern Recognition, (3rd Ed.), pp.17-19, World Scientific Co., ISBN 981-256-105-6, Danvers, United States

Clausi, D. A. (2002a). An analysis of co-occurrence texture statistics as a function of grey level quantization. *Canadian Journal of Remote Sensing*, Vol.28, No.1, (2002), pp.45-62, ISSN 0703-8992

Clausi, D. A. (2002b). K-means Iterative Fisher (KIF) unsupervised clustering algorithm applied to image texture segmentation. *Pattern Recognition*, Vol.35, (September 2002), pp.1959-1972, ISSN 0031-3203

Clausi, D. A. & Deng, H. (2005). Design-based texture feature fusion using Gabor filters and co-occurrence probabilities. *IEEE Transactions on Image Processing*, Vol.14, No.7, (July 2005), pp.925-936, ISSN 1057-7149

Clausi, D. A. & Zhao, Y.P. (2002). Rapid extraction of image texture by co-occurrence using a hybrid data structure. *Computers & Geosciences*, Vol.28, No.6, (July 2002), pp.763-774, ISSN 0098-3004

Clausi, D. A. & Zhao, Y.P. (2003). Grey Level Co-occurrence Integrated Algorithm (GLCIA): a superior computational method to rapidly determine co-occurrence probability texture features. *Computers & Geosciences*, Vol.29, No.7, (August 2003), pp.837-850, ISSN 0098-3004

Diplaros, A. ; Vlassis, N. & Gevers, T. (2007). A spatially constrained generative model and an EM algorithm for image segmentation. *IEEE Transactions on Neural Networks*, Vol. 18, No.3, (May 2007), pp.798-808, ISSN 1045-9227

Du, C. J.; Sun, D. W.; Jackman, P. & Allen, P. (2008). Development of a hybrid image processing algorithm for automatic evaluation of intramuscular fast content in beef M. longissimus dorsi. *Meat Science*, Vol.80, No.4, (December 2008), pp.1231-1237, ISSN 0309-1740

Fan, G. L. & Xia, X. G. (2003). Wavelet-based texture analysis and synthesis using hidden Markov models. *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, Vol.50, No.1, (January 2003), pp.106-120, ISSN 1057-7122

Gonzalez, R. C. & Woods, R. E. (2006). *Digital Image Processing*, (3rd ed.), Prentice Hall, Inc., ISBN 978-0131687288, New Jersey, United States

Haar, A. (1911). Zur Theorie der orthogonalen Funktionen-Systeme. *Mathematische Annalen*, Vol.71, No.1, (1911), pp.38–53, ISSN 0025-5831

Hammouche, K., Diaf, M. & Postaire J.G. (2006). A clustering method based on multidimensional texture analysis, Vol.39, No.7, (July 2006), pp.1265-1277, ISSN 0031-3203

Hammouda, K. & Jernigan, E. (2000). Texture segmentation using Gabor filters. Department of Systems Design Engineering, University of Waterloo, N2L 3G1

Haralick, R.M.; Shanmugam, K. & Dinstein, I. (1973). Textural Features for Image Classification. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol.3, No.6, (November 1973), pp.610-621, ISSN 0018-9472

Jobanputra, R. & Clausi, D. A. (2006). Preserving boundaries for image texture segmentation using grey level co-occurring probabilities. *Pattern Recognition*, Vol.39, No.2, (February 2006), pp.234-245. ISSN 0031-3203

Karahaliou, A.; Boniatis, I.; Sakellaropoulos, P.; Skiadopoulus, S.; Panayiotakis, G. & Costaridou, L. (2007). Can texture of tissue surrounding microcalcifications in mammography be used for breast cancer diagnosis? *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, Vol.580, No.2, (October 2007), pp.1071-1074, ISSN 0168-9002

Khoo, H. K.; Ong, H. C. & Wong, Y. P. (2008). Image texture classification using combined grey level co-occurrence probabilities and support vector machines, *Proceedings of CGIV 2008 5th International Conference on Computer Graphics, Imaging and Visualization*, pp.180-184, ISBN 978-0-7695-3359-9, Penang, Malaysia, August 26-28, 2008

Khouzani, K. J. & Zadeh, H. S. (2005). Rotation-invariant multiresolution texture analysis using radon and wavelet transforms. *IEEE Transactions on Image Processing*, Vol. 14, No.6, (June 2005), pp.783-795, ISSN 1057-7149

Martens, G.; Poppe, C.; Lambert, P. & de Walle, R. V. (2008). Unsupervised texture segmentation and labeling using biologically inspired features. *Proceedings of IEEE 10th Workshop on Multimedia Signal Processing*, pp.159-164, ISBN 978-1-4244-2294-4, Cairns, Queensland, Australia, October 8-10, 2008

Ming, L.; Wu, Y. & Zhang, Q. (2009). SAR image segmentation based on mixture context and wavelet hidden-class-label Markov random field. *Computers and Mathematics with Applications*, Vol.57, No.6, (March 2009), pp.961-969, ISSN 0898-1221

Petrou, M. & Sevilla, P.G. (2006). *Image processing dealing with texture*, John Wiley & Sons Ltd., ISBN 978-0470026281, West Sussex, England

Ribes, R.; Luna, A. & Ros, P.R. (2008). *Learning diagnosis imaging: 100 essential cases*, Springer-Verlag Berlin Heidelberg, ISBN: 978-3-540-71207-7, Berlin, German

Rabiner, L.R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, Vol.77, No.2, pp.257-286, ISSN: 0018-9219, February 1989

Schölkopf, B. & Smola, A.J. (2002). *Learning with kernels: support vector machines, regularization, optimization, and beyond*, the MIT Press, ISBN 978-0262194754, London, England.

Seiler, R.; Schmidt, J.; Diallo, O. & Csaplovics, E. (2009). Flood monitoring in a semi-arid environment using spatially high resolution radar and optical data. *Journal of Environmental Management*, Vol.90, No.7, (May 2009), pp.2121-2129, ISSN 0301-4797

Sonka, M.; Hlavac, V. & Boyle, R. (2007). *Image processing analysis and Machine Vision*. International Thomson Publishing Inc., ISBN 978-0534953935, Pacific Grove, California

Tong, H. L.; Fauzi, M.F.A. & Komiya, R. (2008). Segmentation of CT brain images using K-means and EM clustering. *Proceedings of CGIV 2008 5th International Conference on Computer Graphics, Imaging and Visualization*, pp.339-344, ISBN 978-0-7695-3359-9, Penang, Malaysia, August 26-28, 2008

Zhang, J.; Tong, L. Z.; Wang, L. & Li, N. (2008). Texture analysis of multiple sclerosis: a comparative study. *Magnetic Resonance Imaging*, Vol.26, No.8, (October 2008), pp.1160-1166, ISSN 0730-725X

Human-Centric Machine Vision

Edited by Dr. Fabio Solari

Recently, the algorithms for the processing of the visual information have greatly evolved, providing efficient and effective solutions to cope with the variability and the complexity of real-world environments. These achievements yield to the development of Machine Vision systems that overcome the typical industrial applications, where the environments are controlled and the tasks are very specific, towards the use of innovative solutions to face with everyday needs of people. The Human-Centric Machine Vision can help to solve the problems raised by the needs of our society, e.g. security and safety, health care, medical imaging, and human machine interface. In such applications it is necessary to handle changing, unpredictable and complex situations, and to take care of the presence of humans.

INTECH
open science | open minds