

# A Comparison of Texture Models for Automatic Liver Segmentation

Mailan Pham<sup>a</sup>, Ruchaneewan Susomboon<sup>b</sup>, Tim Disney<sup>c</sup>, Daniela Raicu<sup>b</sup>, Jacob Furst<sup>b</sup>

<sup>a</sup>Mount Holyoke College, South Hadley, MA 01075, USA

<sup>b</sup>Intelligent Multimedia Processing Laboratory, DePaul University, Chicago, IL 60604, USA

<sup>c</sup>Seattle Pacific University, Seattle WA 98119, USA

This material is based upon work supported by the National Science Foundation under Grant No. 0453456.

## ABSTRACT

Automatic liver segmentation from abdominal computed tomography (CT) images based on gray levels or shape alone is difficult because of the overlap in gray-level ranges and the variation in position and shape of the soft tissues. To address these issues, we propose an automatic liver segmentation method that utilizes low-level features based on texture information; this texture information is expected to be homogenous and consistent across multiple slices for the same organ. Our proposed approach consists of the following steps: first, we perform pixel-level texture extraction; second, we generate liver probability images using a binary classification approach; third, we apply a split-and-merge algorithm to detect the seed set with the highest probability area; and fourth, we apply to the seed set a region growing algorithm iteratively to refine the liver's boundary and get the final segmentation results. Furthermore, we compare the segmentation results from three different texture extraction methods (Co-occurrence Matrices, Gabor filters, and Markov Random Fields (MRF)) to find the texture method that generates the best liver segmentation. From our experimental results, we found that the co-occurrence model led to the best segmentation, while the Gabor model led to the worst liver segmentation. Moreover, co-occurrence texture features alone produced approximately the same segmentation results as those produced when all the texture features from the combined co-occurrence, Gabor, and MRF models were used. Therefore, in addition to providing an automatic model for liver segmentation, we also conclude that Haralick co-occurrence texture features are the most significant texture characteristics in distinguishing the liver tissue in CT scans.

**Keywords:** texture, segmentation, Computed Tomography

## 1. INTRODUCTION

Liver cancer is the fourth most common malignancy in the world. In particular, the hepatocellular carcinoma, the predominant liver cancer, accounts for approximately 6 percent of all cancer cases<sup>7</sup>. During the surgical preparation process, it is important to analyze the spatial information of the relative volume of the lesion compared to the overall liver. Thus, overall liver tissue segmentation is an important first step for computer-aided diagnosis.

In general, the approaches applied for segmentation can be divided into eight categories<sup>1</sup>: 1) threshold approaches, 2) region growing approaches, 3) clustering approaches, 4) Markov Random Field models, 5) artificial neural networks, 6) classifiers, 7) deformable models and 8) atlas-guided approaches. Most of these approaches perform segmentation based on gray-level intensities; however, the gray-levels alone are not sufficient for liver segmentation because many soft tissues in CT have overlapping gray-level ranges. Therefore, the use of higher-order properties of the corresponding anatomical structures is necessary to perform accurate medical image segmentation<sup>2</sup>.

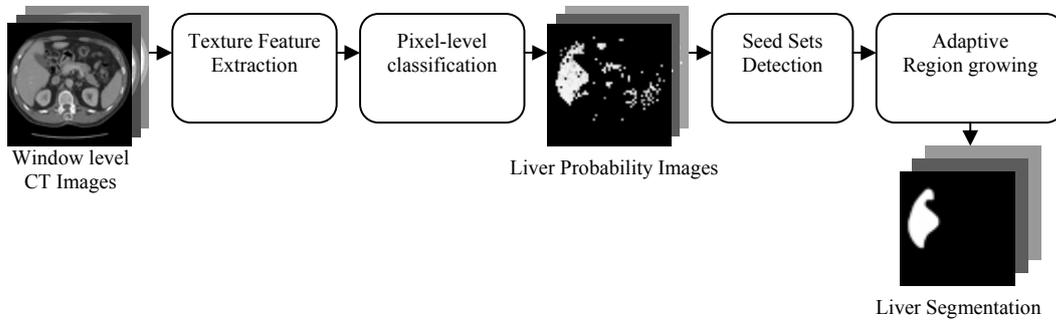
Since the shape of the same organ might also be different across a sequence of 2-D axial slices or, even more, across different patients, texture-based features have been proposed for segmentation as a way to quantify the homogeneity and consistency of soft tissues across multiple slices of a single organ. There are a large number of texture-based segmentation algorithms in the literature; among the most commonly used segmentation algorithms based on texture features are clustering

techniques, region growing, and split-and-merge techniques. Segmentation using these traditional techniques requires considerable amounts of expert interactive guidance. Additionally, these techniques do not incorporate any spatial modeling which can produce poor segmentation results.

In this paper, we propose an automatic approach for segmentation of the liver and compare three different texture models in order to determine the best model or combination of texture features that will produce good segmentation of the liver in CT studies.

## 2. METHODOLOGY

Our approach for evaluating texture models for automatic liver segmentation consists of the following stages as shown in Figure 1: first, texture features are calculated at the pixel level using one of the three texture models; then, a probability image of the organ of interest is obtained by applying a binary classification model (liver/non-liver) obtained using pixel-based texture features. Since the classifier model does not incorporate any spatial information, an adaptive split-and-merge segmentation algorithm is applied on the organ probability image to overcome this drawback and remove the noise introduced by misclassified pixels; finally, the segmented organ boundaries determined at the end of the previous stage are iteratively refined using a region growing algorithm<sup>18,19</sup>.



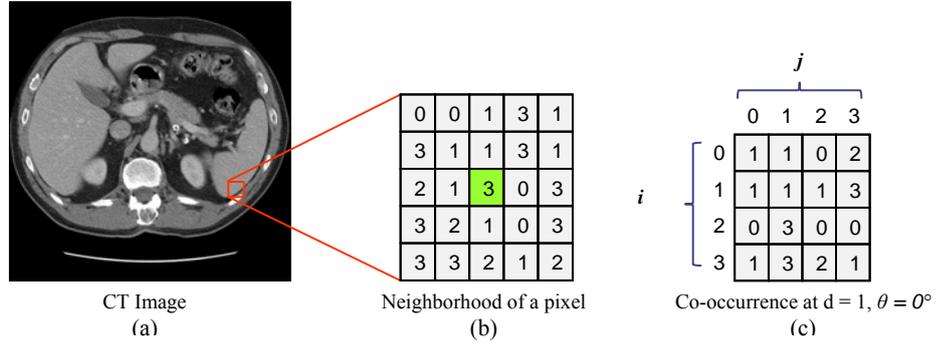
**Figure 1.** Diagram of the proposed approach

### 2.1. Texture Feature Extraction

Pixel-level texture extraction is the process of analyzing and quantifying the texture patterns within a small neighborhood of size  $N$  by  $N$  around a pixel of interest. In the following subsections we present the three texture models along with the texture features generated for each one of them.

#### 2.1.1. Co-occurrence Matrices

As a statistical method for texture extraction, co-occurrence matrices focus on the distribution and the relationships among the gray levels in an image<sup>3</sup>. The general idea of a co-occurrence matrix is to represent an image's texture characteristics by counting pixel intensity pairs, using a matrix that keeps track of all the pixel-pair counts as shown in Figure 2. At the pixel level, the normalized co-occurrence matrix,  $P_{ij}(d, \theta)$ , counts all the pixel-pairs within the  $N$  by  $N$  neighborhood (where  $d=1, \dots, N-1$  is the displacement vector,  $\theta$  is the angle ( $0^\circ, 45^\circ, 90^\circ, 135^\circ$ ),  $i$  represents the gray-level along the vertical direction (row) and  $j$  represents the gray-level along the horizontal direction (column).



**Figure 2.** (a) original 2-D CT image with the pixel of interest highlighted; (b) the 5 by 5 neighborhood around the corresponding pixel; (c) the co-occurrence matrix calculated for  $d = 1, \theta = 0^\circ$

Once the co-occurrence matrix is calculated for each pixel, nine Haralick texture descriptors (entropy, energy, contrast, sum average, variance, correlation, maximum probability, inverse different moment, and cluster tendency) are calculated<sup>17</sup>.

Entropy: Measure the randomness of gray-level distribution.

$$-\sum_i \sum_j P_{ij} \log P_{ij} \quad (1)$$

Energy: Measure the occurrence of repeated pairs within an image.

$$\sum_i \sum_j P_{ij}^2 \quad (2)$$

Contrast: Measure the local contrast in an image.

$$\sum_i \sum_j (i-j)^2 P_{ij} \quad (3)$$

Sum Average: Measure the average of the gray-level within an image.

$$\frac{1}{2} \sum_i \sum_j (iP_{ij} + jP_{ij}) \quad (4)$$

Variance: Measure the variation of gray level distribution.

$$\frac{1}{2} \sum_i \sum_j ((i-\mu_r)^2 P_{ij} + (j-\mu_c)^2 P_{ij}) \quad (5)$$

Correlation: Measure a correlation of pixel pairs on gray-levels.

$$\sum_i \sum_j \frac{(i-\mu_r)(j-\mu_c)P_{ij}}{\sqrt{\sigma_r^2 \times \sigma_c^2}} \quad (6)$$

Maximum Probability (MP): Determine the most predominant pixel pair in an image.

$$\underset{i,j}{\text{Max}} P_{ij} \quad (7)$$

Inverse Difference Moment (IDM): Measure the smoothness of an image

$$\sum_i \sum_j \frac{P_{ij}}{1+|i-j|^2} \quad (8)$$

Cluster Tendency (CT): Measure the grouping of pixels that have similar gray-level values

$$\sum_i \sum_j (i-\mu_r + j-\mu_c)^2 P_{ij} \quad (9)$$

where  $\mu_r, \mu_c, \sigma_r^2, \sigma_c^2$  are the means and variance of row and column defined as follow:

$$\mu_r = \sum_i \sum_j i P_{ij}, \quad \mu_c = \sum_i \sum_j j P_{ij}$$

$$\sigma_r^2 = \sum_i \sum_j (i-\mu_r)^2 P_{ij}, \quad \sigma_c^2 = \sum_i \sum_j (j-\mu_c)^2 P_{ij}$$

Since the nine descriptors will be calculated for each of the (N-1)\*4 co-occurrence matrices, there will be (N-1)\*4\*9 texture descriptors which are not necessary independent from each other. In order to reduce the feature set and select the most important features, an information-gain criterion<sup>21</sup> is applied on the feature set. At the end of this stage, each pixel will be characterized by a  $p$ -dimensional feature vector, where  $p$  indicates the number of most discriminative features as determined by the information gain criterion; the feature vector is further normalized by a min-max normalization technique applied with respect to each individual feature.

For computational efficiency purposes, in the co-occurrence matrix implementation we represent only the gray-levels that appear within the pixel neighborhood under consideration; this implementation is possible because the texture features are not affected by the co-occurrence probabilities equal to zero which were produced by the intensities which do not appear in the corresponding pixel neighborhood. Therefore, rather than having a 256 by 256 co-occurrence matrix for each pixel neighborhood, the co-occurrence matrix will be, in the worst case scenario (when all possible values within the pixel neighborhood are different), of size  $(2*N)^2$ .

### 2.1.2. Gabor Filters

In contrast to the statistical based co-occurrence matrix method, Gabor filtering is a transform based method for extracting texture information. The use of Gabor filters is motivated by Gabor filtering being *strongly correlated with the human visual system*<sup>4</sup>. Gabor filters have also been successfully used in a number of other projects with the goal of extracting texture information in order to perform texture-based segmentation<sup>5,6</sup>.

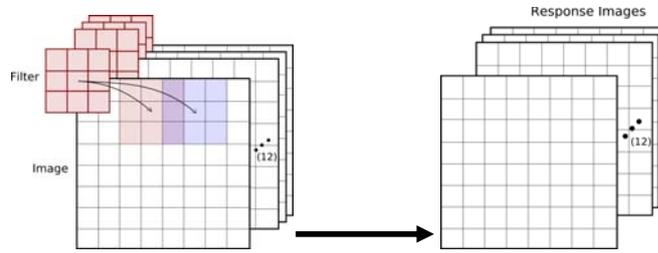
Gabor filtering is a way of extracting feature information from an image in the form of a response image. Several filters with varying parameters are applied to an image to acquire the response. A Gabor filter is a sinusoid function modulated by a Gaussian and is defined by the following equation<sup>5</sup>:

$$G(x, y) = \exp\left(\frac{-x_{\theta}^2 - \gamma^2 \frac{y_{\theta}^2}{\sigma^2} + \frac{2\pi x_{\theta} i}{\lambda}}{\sigma^2}\right) \quad (10)$$

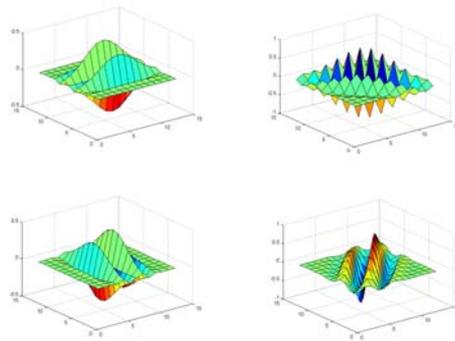
where  $x_{\theta} = x \cdot \cos(\theta) + y \cdot \sin(\theta)$ ,  $y_{\theta} = -x \cdot \sin(\theta) + y \cdot \cos(\theta)$

$\sigma$  is the standard deviation of the Gaussian function,  $\lambda$  is the wavelength of the harmonic function,  $\theta$  is the orientation, and  $\gamma$  is the spatial aspect ratio which is left constant at 1/2. The spatial frequency bandwidth is the ratio  $\sigma / \lambda$  and is held constant and equal to 0.56.

We convolve the image with 12 different Gabor filters tuned to four orientations ( $\theta$ ) and three frequencies ( $1/\lambda$ ) as shown in Figure 3; each filter generates two texture features: the mean and standard deviation with respect to its local neighborhood. Figure 4 shows the Gabor filters when the orientation varied from 0 to  $3\pi/4$  and frequency varied from 0.3 to 0.5 (stepping by 0.1).



**Figure 3.** Visualization of convolving a Gabor filter with an image to produce 12 response images



**Figure 4.** Visualization of Gabor filters varying by  $\theta$  parameter. Top left  $\theta = 0$ , top right  $\theta = \pi/4$ , bottom left  $\theta = \pi/2$ , bottom right  $\theta = 3\pi/4$

### 2.1.3. Markov Random Fields

Falling into the model-based category, Markov Random Fields (MRF) capture the local contextual information of an image<sup>8</sup>. The application of MRF to extract textual information was first done by Jain and Cross in 1983<sup>3</sup>. Since then, Markov

Random Fields have been gaining increasing popularity because of their ability to create an image model that can be successfully used for image classification, segmentation, and texture synthesis<sup>9</sup>.

In a MRF model, the image is represented by a two-dimensional lattice. The value at each pixel in the lattice is a random variable; for example, for gray scale images with 256 gray levels, each random variable can take a value in the set  $\{0, 1, 2, \dots, 255\}$ <sup>9,10</sup>.

A random variable  $X$  on the lattice  $S$  with neighborhood system  $\delta_s$  is said to be a Markov random field if for all  $s \in S$

$$p(X_s | X_r \text{ for } r \neq s) = p(X_s | X_{\delta_s}) \quad (\text{Markovianity})$$

In order for a site to be a MRF, it must exhibit this attribute of Markovianity that is, the value of the random variable is dependent only on its neighbors<sup>8</sup>. In a Gaussian Markov Random Field (GMRF), the image is represented on a local conditional probability distribution that is assumed to be Gaussian<sup>11</sup>. We extracted four features from the GMRF. These four features correspond to the four orientations between a neighboring pixel pair<sup>16</sup>. To obtain our final feature vectors in our liver segmentation process, we used an algorithm based on the work proposed by Cesmeli: 1) estimate the four GMRF parameters, 2) derive four new feature vectors from the estimated parameters, and 3) filter the feature vectors with noise suppression<sup>11</sup>.

The least-square approach, Equation (11), was used to estimate the set of four parameters for a second order GMRF model:  $\hat{\Theta} = [\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3, \hat{\theta}_4]^T$ , where  $\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3, \hat{\theta}_4$  correspond to the  $0^\circ, 90^\circ, 45^\circ,$  and  $135^\circ$  directions, respectively.

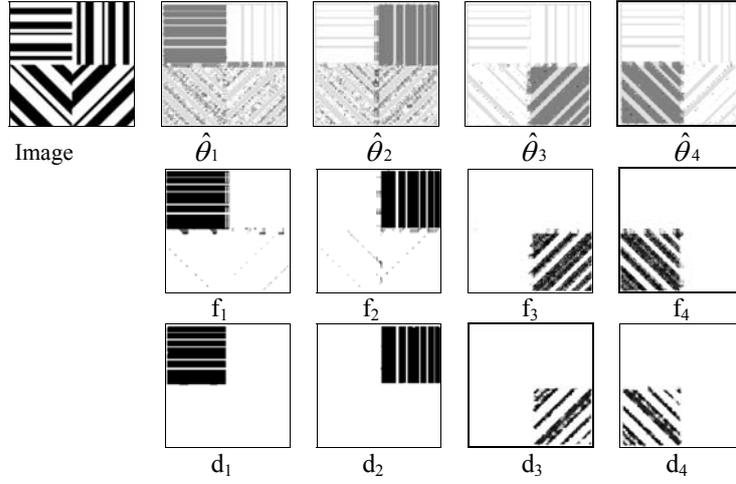
$$\hat{\Theta} = \left[ \sum_{r, r \pm \tau; j \in R(s)} Q(r) Q(r)^T \right]^{-1} \left[ \sum_{r, r \pm \tau; j \in R(s)} Q(r) y_r \right] \quad (11)$$

where  $Q(r) = [(y_{r+\tau_1} + y_{r-\tau_1}), \dots, (y_{r+\tau_4} + y_{r-\tau_4})]$ ,  $\tau$  stands for the orientation  $0^\circ, 90^\circ, 45^\circ,$  and  $135^\circ$ , respectively,  $r$  is the pixel location in the image, and  $R(s)$  is the estimation window. For example,  $y_{r+\tau_1} + y_{r-\tau_1}$  corresponds to the  $0^\circ$  orientation and gives the sum of the two intensity values of the neighbor pixels that are to the left and the right of the pixel at location  $r$ .

Equation (11), by multiplying the inverse of the summation of the correlation matrices (4x4) with the summation of the vectors (4x1), yields four parameters. Usually, the four MRF parameters are directly used as the feature vectors; however, as Cesmeli stated and could be seen in Figure 5, a new set of feature vectors  $[f_1, f_2, f_3, f_4]$  can be derived from the ones given by equation (11) and they have more discriminatory power in detecting different textures<sup>11</sup>:

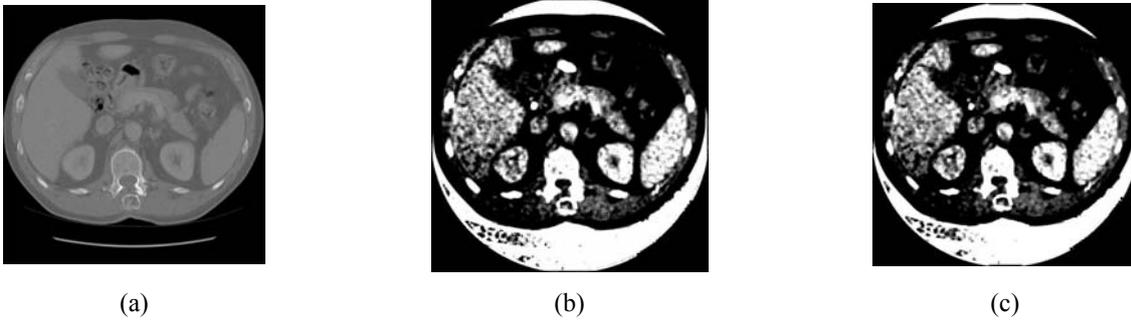
$$f_j = \frac{1}{(u^2)} \sum_{r, r \pm \tau; j \in R(s)} [y(r) - \hat{\theta}_j Q_j(r)]^2, \quad j = 1, \dots, 4 \quad (12)$$

where  $u$  is equal to the size of the estimation window. Since the new features are sensitive to noise, they are filtered with an extended version of an edge preserving noise suppressing quadrant filter (EPNSQF) following the approach from<sup>11</sup> for an estimation window of size 5 by 5.



**Figure 5.** Visualization of an artificial image, its original four parameters:  $\hat{\theta}_1, \hat{\theta}_2, \hat{\theta}_3, \hat{\theta}_4$ , its new feature vectors:  $f_1, f_2, f_3, f_4$ , and its noise-suppressed feature vectors:  $d_1, d_2, d_3, d_4$ .

Each estimation window is further divided into five 3 by 3 sub-windows  $N_{s,j}$  and the mean and variance of each feature for each sub-window are calculated. Next, the total variance is calculated for each parallel sub-windows in  $f_1, f_2, f_3$ , and  $f_4$ . The sub-window with the lowest total variance is the winner and its mean replaces the pixel value. For example, if  $N_5$  is the winner sub-window, the mean value  $N_{5,1}$  would replace the pixel value in  $f_1$ , the mean value of  $N_{5,2}$  would replace the pixel value in  $f_2$ , the mean value of  $N_{5,3}$  would replace the pixel value in  $f_3$ , and the mean value of  $N_{5,4}$  would replace the pixel value in  $f_4$ . Figure 6 shows an example of a texture image for the  $f_3$  feature vector (Fig. 6.b) and for its noise-suppressed feature vector  $d_3$  (Fig. 6.c) calculated for the abdominal CT image from Fig. 6.a.



**Figure 6.** (a) original CT image, (b) its feature vector  $f_3$ , and (c) its final feature vector  $d_3$  after EPNSQF

## 2.2. Pixel-based texture classification

In our approach, a binary Classification and Regression Tree (C&RT) model is applied on the pixel-level data generated in the previous step in order to generate decision rules for the classification of the organ of interest. The C&RT model is chosen because it does not make any assumptions about the distribution of the data, its computational efficiency, easy interpretability of the output data in the form of decision rules, and good classification performance as shown in Varma et. al. 2003<sup>14</sup>. Once the optimal decision tree is built, each terminal node will produce a decision rule for the organ of interest and a probability will be associated with each rule in the form of IF-THEN- PROBABILITY expression; the probability will be

given by the ratio between the number of true organ pixels classified as the organ of interest and total number of pixels found at that terminal node. For instance, the rule,

IF [ $f_1(d, \theta; \gamma) < 0.34, 0.65 < f_5(d, \theta; \gamma) \leq 0.8$ ] THEN Liver with PROBABILITY=.9,

denotes a probability rule obtained from a terminal node in which 90% of the pixels found at that node were indeed liver pixels and the only descriptors used by the rule were the co-occurrence texture descriptors,  $f_1$  and  $f_5$ . At the end of the classification process, the decision rules will be applied to each pixel within the slice to be segmented; the output will be an organ probability image that will be used later for the adaptive split-and-merge segmentation stage; the probability image will also be used for the region growing stage in which pixels with lower probabilities will be added as the algorithm advances from the center of the organ towards the boundary.

### 2.3. Adaptive Split & Merge segmentation

Pixel-level classification is often an ultimate objective for image segmentation<sup>15</sup>. However, classifiers generally do not perform any spatial modeling and this makes the segmentation process sensitive to the noise produced by the misclassified pixels. To overcome this limitation, the split-and-merge algorithm is applied on the organ probability image. The algorithm involves two main steps: 1) the selection of initial seed regions and 2) the selection of adjacent regions. Once the seed set is determined, the splitting constraints are relaxed. The process looks only at the regions adjacent to the seed set and splits these regions further if the two relaxed splitting criteria (decreased probability threshold and a new minimum sub-region size) are not satisfied. Again, the homogeneity threshold (average of probability within the neighborhood window) is relaxed down to a value that is equal or very close to the overall sensitivity as given by the decision tree:

$$sensitivity = \frac{TP}{TP + FN} \quad (13)$$

$$specificity = \frac{TN}{TN + FP} \quad (14)$$

where TP is the number of true positives (pixels classified as liver when they were actually liver), TN is the number of true negatives (pixels classified as non-liver when they were actually non-liver), FN is the number of false negatives (pixels classified as non-liver when they were actually liver) and FP is the number of false positives (pixels classified as liver when they were actually non-liver).

Setting the threshold to a value close to overall sensitivity may lead to a potential early termination (because of generally high sensitivities) of the segmentation process which may cause the system not to include some of the organ boundary pixels. Therefore, a refinement of the boundary is required as a post-processing stage in order to include the organ boundary pixels in the final segmentation results.

### 2.4. Adaptive region growing

To refine the boundary of the segmented organ, an adaptive region growing approach is deployed as a post-processing step. First, the process attempts to find the boundary-pixels from segmented regions. Around each boundary-pixel, a window of size  $N$  by  $N$  is formed and the average probability within that window is calculated. The boundary-pixel will be merged with the existent segmented region only if it satisfies the region constraint meaning that the pixel's average probability within the corresponding window meets the threshold; the threshold for boundary refinement is more relaxed compared with those used in the process of splitting since, at the organ boundary, the average of the organ's probabilities tends to be lower. For each repetition of the adaptive region growing process, the threshold is lowered by 5%. The adaptive region growing process repeats until either the threshold has reached the value of 60% or no new pixels are added in the result of the region segmentation.

## 3. PRELIMINARY RESULTS

Our preliminary results are based on data extracted from normal CT images obtained from Northwestern Memorial Hospital (NMH) PACS. The data consists of multiple, serial, axial computed tomography images derived from helical, multi-detector CT abdominal and chest acquisitions using a HiSpeed CT/i scanner (GE Medical Systems). The images were transferred via Ethernet to a nearby computer workstation in DICOM format of size 512 by 512 and have 12-bit gray level resolution.

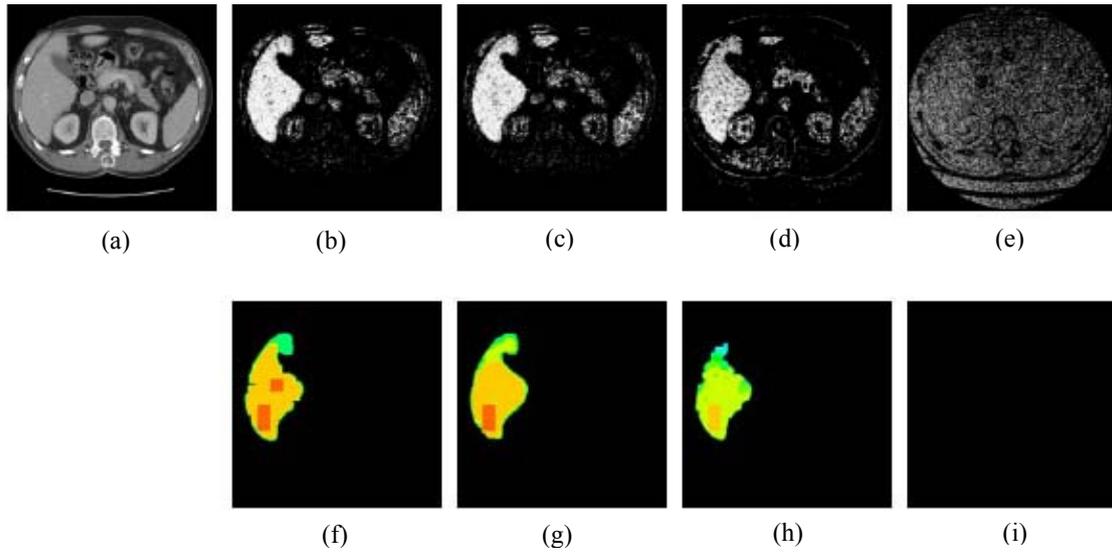
In order to obtain the a priori knowledge for the C&RT and each texture model, we randomly selected pixels from one CT slice in which the organ of interest was present such that 50% of the total numbers of data pixels are from the liver class and the rest are from the “unknown” (anything not liver) class. For each pixel, a 9x9 window is used to extract the pixel-level texture. For Haralick co-occurrence features, all nine descriptors were calculated for each of the direction and distance of all possible combinations in the 9x9 window; therefore, there will be  $(9-1)*4*9$  texture descriptors which are not necessary independent from each other. In order to reduce the feature set and select the most important features, we select 40 features that provide the highest information-gain. The number of co-occurrence features is obtained experimentally as the number of features that provide the optimal classification results. For MRF texture model, the 4 variances of the four texture orientations ( $d_1, d_2, d_3, d_4$ ) will be applied. We obtain 24 features from Gabor texture model from 12 different Gabor filters, in which each filter generates two texture features: the mean and standard deviation with respect to its local neighborhood.

In order to generate and test the decision rules, these selected pixels were randomly divided into 66% for training and 34% for testing the C&RT. An optimal C&RT was generated for each texture model and for their combination; the sensitivity and specificity metrics were used to evaluate the performance of the classifier for each of the texture model. As seen in Table 1, co-occurrence has a testing sensitivity of 85% and specificity of 84% which are higher than the values obtained for the Gabor and MRF models alone. The co-occurrence sensitivity is greater than of MRF by 7% and greater than of the Gabor sensitivity results by 40%. Gabor does not extract the necessary texture information in order for it to correctly distinguish different organ textures at the abdominal level. Furthermore, when combining the three texture models, the sensitivity level is almost the same (increased by only 1%) as the one obtained by using occurrence alone. This information leads us to believe that the pixel-pair technique (co-occurrence) gives enough information in order to quantify the texture liver for the purpose of its segmentation.

**Table 1.** Performance metrics for three texture models

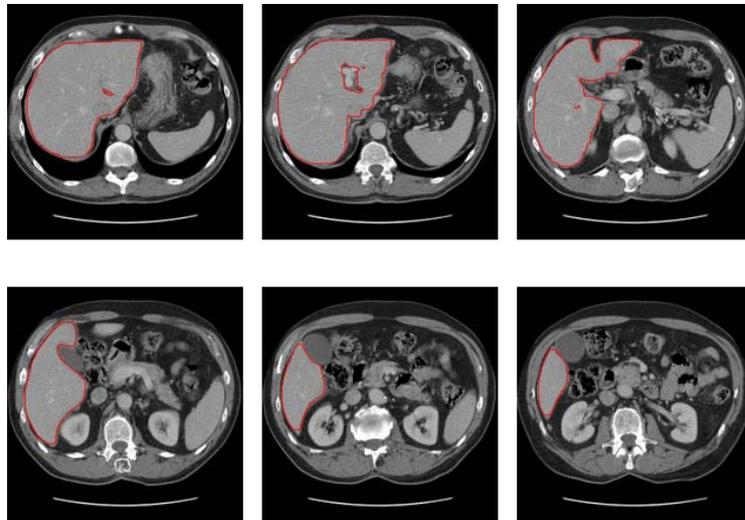
		Co-occurrence, MRF & Gabor	Co-Occurrence	MRF	Gabor
<b>Sensitivity</b>	Training	94.38%	93.89%	86.34%	55.21%
	Testing	85.48%	84.80%	82.92%	52.94%
<b>Specificity</b>	Training	94.27%	93.70%	85.27%	59.59%
	Testing	85.13%	84.08%	81.48%	57.65%

After generating the decision rules, they were applied to generate probability images for each CT image. Figure 7 shows examples of liver segmentation in one particular slice that was obtained using different texture models. We noticed that Gabor provides scattered liver classification (Fig. 7(e)), while the other texture models provide more compact predicted liver regions. Fig. 7.(f-j) show the segmentation done at different probability thresholds, where the different colors represent different probabilities. We found that the use of all texture features gives the highest probability of the initial (seed) region as shown in Fig. 7.(f); however, the model loses its accuracy more when compared with the Co-occurrence matrix as the segmentation gets closer to the boundary. We also noticed that, from all models, the co-occurrence produces the smoothest organ boundary.



**Figure 7.** Example of the liver segmentation: (a) the original image; (b – e) the liver probability image generated by C&RT, where (b) represents the liver probability image generated by the co-occurrence, MRF and Gabor texture features; and (c), (d), (e) show the liver probability images generated by co-occurrence, MRF and Gabor texture models respectively. (f) shows liver segmentation based on the combined co-occurrence, MRF and Gabor texture features; and (g-i) show the segmentation results based on Haralick, MRF, and Gabor, respectively.

Since the co-occurrence texture model performs the best, we show the results of our segmentation approach based on co-occurrence matrices for several 2-D CT slices (Figure 8). The red line represents the segmentation boundary of the liver as detected by our proposed approach. We noticed that co-occurrence texture model was able to capture liver regions without capturing other structures (such as vessels) inside the liver; and the boundary of the segmented liver was smooth.



**Figure 8.** Different examples of liver segmentation showing the correct delineation of the liver in various CT slices based on the Haralick co-occurrence texture model.

#### 4. CONCLUSIONS

In this paper we present an automatic approach for soft tissue segmentation. The approach is presented with respect to liver segmentation, but it can be easily extended to any other soft tissue by setting appropriately the values of the parameters for the splitting and merging algorithm and for the region growing refinement step. Our proposed segmentation approach also emulates the human process of segmenting regions of interest by assigning decreasing probabilities to the pixels as they get closer to the boundaries. For instance, when two or more radiologists will manually segment the same region of interest, due to the variability in the image interpretation, pixels within the core of the region will be found in the segmentation results produced by all radiologists, while pixels close to the boundary will be marked as pixels of that region by only some of the radiologists. Therefore, the pixels within the organ's core will receive higher probabilities while the boundary pixels will receive lower probabilities corresponding to the lower level of confidence in assigning the close to the boundary pixels to the region.

Since no texture model is known to work the best for liver segmentation, we explored three different texture models and found that co-occurrence texture model performs the best. Since Gabor filters performed the worst, as future work, we would like to investigate other values for the Gabor filters in order to see if the segmentation results can be improved. Furthermore, we will investigate the use of the proposed approach to also perform automatic volumetric liver segmentation.

#### REFERENCES

1. D.L. Pham, C. Xu, J.L. Prince, "Current methods in medical image segmentation" *Annual Review of Biomedical Engineering*, Vol. 2, 2000
2. K.I. Chang, K.W. Bowyer, M. Sivagurunath, "Evaluation of texture segmentation algorithms," *IEEE Conf. on Computer Vision and Pattern Recognition*, 1999
3. A. Materka and M. Strzelecki, "Texture analysis methods - a review," *Technical Report.*, Technical University of Lodz, Institute of Electronics, 1998
4. T. Galatard, J. Montagnat, and I. E. Magnin, "Texture based medical image indexing and retrieval: application to cardiac imaging," *Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pp. 135–142, 2004
5. J. Puzicha, T. Hofmann, and J. M. Buhmann, "Non-parametric similarity measures for unsupervised texture segmentation and image retrieval," in *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, 1997.
6. D. A. Clausi and M. E. Jernigan, "Designing Gabor filters for optimal texture separability," *Pattern Recognition*, vol. 33, pp. 1835–1849, 2000.
7. R. Lu, P. Marziliano, and C. H. Thng, "Liver Tumor Volume Estimation by Semi-automatic Segmentation Method", *Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2005
8. C. Bouman, "Markov random fields and stochastic image models," *IEEE International Conference on Image Processing*, 1995.
9. C. Chen, L. Pau, and P. W. (eds.), "The Handbook of Pattern Recognition and Computer Vision (2nd Edition)," *World Scientific Publishing Company*, 1998.
10. M. Goktepe, N. Yalabik, and A. Volkan, "Unsupervised segmentation of gray level Markov model textures: Hierarchical self organizing maps," *Proceedings of the 1996 International Conference on Pattern Recognition*, pp. 90–94, 1996.
11. E. Cesmeli and D. Wang, "Texture segmentation using Gaussian-Markov random fields and neural oscillator networks," *IEEE Transactions on Neural Networks*, vol. 12, pp. 394–404, March 2001
12. S. Krishnamachari and R. Chellappa, "GMRF models and wavelet decomposition for texture segmentation," *Proceedings of the 1995 International Conference on Image Processing*, pp. 568–571, 1995.
13. F. Cohen, Z. Fan, and M. Patel, "Classification of rotated and scaled textured images using Gaussian Markov random field models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 192–202, February 1991.

14. M. Varma, A. Zisserman, "Texture classification: are filter banks necessary?" *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, 2003
15. R. Lerman, D.S. Raicu, J.D. Furst, "Contrast enhancement of soft-tissues in Computed Tomography images," *Proceedings of SPIE Medical Imaging Conference*, 2006
16. P. P. Ohanian and R. C. Dubest, "Performance evaluation for four classes of textural features," *Pattern Recognition* 25(8), p. 819, 1992
17. R.M. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification", *IEEE Trans. on Systems, Man, and Cybernetics*, vol. Smc-3, no.6, pp. 610-621, 1973
18. R. Susomboon, D.S. Raicu, and J.D. Furst, "Automatic Single-Organ Segmentation in Computed Tomography Images", *IEEE International Conference on Data Mining*, December 2006
19. R. Susomboon, D.S. Raicu, J.D. Furst, and D.S. Channin, "A Probabilistic Approach for Automatic Liver Segmentation", *The 92nd Scientific Assembly and Annual Meeting of Radiology Society of North America (RSNA'06)*, Chicago, IL, USA, November 26 - December 1st, 2006
20. R. Susomboon, D.S. Raicu, and J.D. Furst, "Pixel-Based Texture Classification of Tissues in Computed Tomography", *CTI Research Symposium*, April 2006
21. M. Sonka, V. Hlavac, R. Boyle, "Image Processing, Analysis, and Machine Vision", *Pacific Grove: Brooks/Cole Publishing Company*, 1999