CHAPTER 4

IMAGE RETRIEVAL OF BRAIN MR IMAGES USING MOD_LTP

4.1 INTRODUCTION

LBP’s have been shown to have high discriminating power for texture classification (Ojala et al. 2002) and are invariant to monotonic gray-level transformations. LBP itself is sometimes used as a normalization step to reduce lighting problems for other methods (Heusch et al., 2006). However, under large illumination variations the reliability of LBP decreases significantly. In addition to this, they are likely to be sensitive to noise, especially in near-uniform image regions, because the binary pattern is formed by thresholding with the central pixel. As an extension to traditional LBP, Tan & Triggs (2007) proposed Local Ternary Pattern (LTP) based on a 3-valued coding that includes a threshold around zero instead of binary encoding which is less sensitive to noise. LTP inherits most of the other key advantages of LBP such as computational efficiency, simplicity, etc. In a given neighborhood, gray levels above a threshold, $\sigma$ from central pixel are quantized to +1 and the ones below it to -1, and within $\pm \sigma$ are quantized to zero which is shown in Figure 4.1. When LTP is used for content matching, $3^n$ different codes are generated, but the number of codes can be reduced using uniform patterns. For simplicity, each ternary pattern is split into its positive and negative parts as shown in Figure 4.2 and separate histograms and similarity metrics are computed for each part.
The idea of a 3-valued encoding is also proposed by Ahenon & Pietikainen (2007), Loris et al (2010), where a fuzzy thresholding function is
used to make the LBP operator more robust to noise. In all these methods, the threshold of 3-valued encoding is calculated globally, which is sensitive to global illumination changes.

In the previous chapter, as an extension to LBP, MOD_LBP has been used to retrieve similar anatomical structures from Brain MR images in which a binary encoding is applied to form a P-bit binary pattern. This binary pattern formed in a given neighborhood is multiplied by the squared difference between intensity of each pixel and the mean intensity of the neighborhood pixels. The MOD_LBP appears to be a better texture descriptor as the amount of difference between each pixel from the mean value is incorporated. In order to make it robust to noise, the binary encoding is replaced by ternary encoding based on a global as well as local threshold.

The MOD_LBP with ternary encoding based on global or local threshold is termed as Modified Local Ternary Pattern (MOD_LTP) and is given in equation 4.1.

\[
MOD\_LTP(P,R) = \frac{1}{P} \sum_{i=0}^{P-1} s(g_i - g_c)(g_i - \mu)^2, 
\]

where \( s(g_i - g_c) = \begin{cases} 1, & g_i - g_c > \sigma \\ 0, & |g_i - g_c| \leq \sigma \\ -1, & g_i - g_c < -\sigma \end{cases} \), \( P \) is the number of neighboring pixels in the circular neighborhood with radius \( R \) and \( \sigma \) - is the threshold. If the gray value of a pixel is more than the mean gray value by \( \sigma \), it is assigned a value 1. If it is less than the mean gray value of the neighboring pixels by \( \sigma \), -1 is assigned to it, otherwise 0. The threshold can be a user defined one or can be calculated automatically based on the variance of the given neighborhood. The user defined threshold is sensitive to global illumination changes, but the local threshold chosen to automatically make the texture descriptor more robust to the global illumination changes. The ternary pattern so obtained using a
3-value encoding is split into two binary patterns in which first binary pattern is obtained by considering positive components and making negative component as zero. The second binary pattern is obtained by considering the negative components and making positive components as zero. The resultant binary pattern is obtained by concatenating the two binary patterns obtained. The MOD_LTP image is formed by assigning the real number equivalent of the weighted binary number to the central pixel based on the squared gray level difference from the mean value.

The MOD_LTP with a proper choice of threshold is shown to more discriminate texture descriptor compared to MOD_LBP. As MRI is locally smooth, this local measure which gives importance to the local variance act as a better texture. The MOD_LTP with a local threshold, based on the variance of the given neighborhood makes the texture descriptor invariant to monotonic gray level change and robust to intra-image illumination as long as absolute gray scale values are not much affected. As every pixel in the local neighborhood is involved in the MOD_LTP computation, the method is invariant to some basic geometric transformations and intensity variations with respect to that neighborhood.

4.2 METHODOLOGY

Similar to MOD_LBP computation, MOD_LTP is computed using a circular window of size \( w \). The normalized histogram of the resultant image is taken as feature vectors for similarity computation. The images in the database are ranked based on the Bhattacharya coefficient between query and database images. The average rank and accuracy for a set of query images to locate similar images is calculated. The moment features of MOD_LTP are computed spatially over an angularly partitioned area and performance of the retrieval system is evaluated based on the distance function of moment features. In order to achieve an optimum performance, the moment features of
MOD_LTP are reweighted based on the relevance of individual features in the retrieval process. The overview of the procedure is shown in Figure 4.3.

**Figure 4.3** Block diagram of image retrieval scheme using MOD_LTP

### 4.2.1 Feature extraction

The MOD_LTP is computed using Equation (4.1) and the pixel value of MOD_LTP is brought into the range of integer values [0,Z]. Normalized histogram of MOD_LTP is taken as feature vectors and Bhattacharya coefficient is used for defining similarity between the query image and database images. The moment features of MOD_LTP over an angularly partitioned grid are also extracted with respect to a reference line (line with respect to which 2D brain image exhibits maximum symmetry). The
central moments of orders 1 to 25, Rotational Scale and Translational (RST) invariant features (Schalkoff 1989) obtained from the central moments, and the eccentricity is calculated for each partitioned grid and Euclidean distance is used to calculate similarity between query and database images. The retrieval performance is evaluated using individual features as well as the features together. Since the presence of the some features degrade the performance of the entire system, individual features are reweighted as explained in chapter 3 section 3.2.3.2 based on its role in the retrieval process.

4.3 RESULTS

The experiment is performed on T2 weighted clinical data set as well as BrainWeb simulated database [BrainWeb].

The slices (T2 weighted) used in this work, were acquired on a 1.5 Tesla, General Electric (GE) – Signal Hex MR Scanner from Pushpagiri Medical College Hospital, Tiruvalla, Kerala, India. Axial, 2D, 5mm thick slice images, with a slice gap of 1.5mm were acquired with the Field-of-view (FOV) of range 220mm to 250mm. The T2 (TR / TE(eff.) of 3500-4500/ 85-105 (eff.) ms) images were collected using Fast Spin Echo (FSE) sequences with a matrix size of 320 × 224 (Frequency × Phase) and a NEX (Averages) of 2.

The unregistered brain MR images (460 numbers) of different persons are categorized into 4 levels as shown in Figure 4.4. In order to test the performance of the retrieval system, 10 images are randomly chosen from each level and average rank accuracy is calculated for retrieving first 10 relevant images. The similarity computation is performed based on histogram features and moment features of MOD_LTP. In order to attain optimum performance, moment features are reweighted 5 times based on the role of individual features in the retrieval performance. A comparison has been made with rotational invariant LBP ($LBP^{(a)}_{8,1}$), rotational invariant uniform LBP
and MOD_LBP in retrieving 10 relevant images in each level and is shown in the Table 4.1.

Figure 4.4 Different levels of T2 weighted axial MR brain slices.

The performance of the retrieval system is evaluated using various user defined thresholds. Figure 4.5 shows the average rank of different level images based on various thresholds. In an ideal situation, the average rank is expected to be 1, but practically it is not possible due to various challenges. The average rank can be made least by selecting proper threshold. It shows a better average rank corresponds to a threshold 0.1 in L_1, 0.02 in L_2, 0.03 for L_3 and L_4.
<table>
<thead>
<tr>
<th>Class</th>
<th>Method</th>
<th>No of relevant images retrieved</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>L1</td>
<td>a)</td>
<td>2.47</td>
</tr>
<tr>
<td></td>
<td>b)</td>
<td>2.53</td>
</tr>
<tr>
<td></td>
<td>c)</td>
<td>2.73</td>
</tr>
<tr>
<td></td>
<td>d)</td>
<td>1.48</td>
</tr>
<tr>
<td></td>
<td>e)</td>
<td>1.17</td>
</tr>
<tr>
<td></td>
<td>f)</td>
<td>1.43</td>
</tr>
<tr>
<td>L2</td>
<td>a)</td>
<td>1.56</td>
</tr>
<tr>
<td></td>
<td>b)</td>
<td>1.81</td>
</tr>
<tr>
<td></td>
<td>c)</td>
<td>1.69</td>
</tr>
<tr>
<td></td>
<td>d)</td>
<td>1.63</td>
</tr>
<tr>
<td></td>
<td>e)</td>
<td>1.87</td>
</tr>
<tr>
<td></td>
<td>f)</td>
<td>1.68</td>
</tr>
<tr>
<td>L3</td>
<td>a)</td>
<td>5.8</td>
</tr>
<tr>
<td></td>
<td>b)</td>
<td>7.26</td>
</tr>
<tr>
<td></td>
<td>c)</td>
<td>3.73</td>
</tr>
<tr>
<td></td>
<td>d)</td>
<td>1.52</td>
</tr>
<tr>
<td></td>
<td>e)</td>
<td>1.87</td>
</tr>
<tr>
<td></td>
<td>f)</td>
<td>1.83</td>
</tr>
<tr>
<td>L4</td>
<td>a)</td>
<td>1.64</td>
</tr>
<tr>
<td></td>
<td>b)</td>
<td>1.59</td>
</tr>
<tr>
<td></td>
<td>c)</td>
<td>1.32</td>
</tr>
<tr>
<td></td>
<td>d)</td>
<td>1.06</td>
</tr>
<tr>
<td></td>
<td>e)</td>
<td>1.00</td>
</tr>
<tr>
<td></td>
<td>f)</td>
<td>1.48</td>
</tr>
</tbody>
</table>
Figure 4.5 The Average Rank in retrieving 10 relevant images using various user defined threshold

Local binary Pattern is sensitive to noise due to blind thresholding. But by the choice of proper threshold, sensitivity to noise in uniform region or near uniform can be reduced. This has been tested with different noise levels (1%, 3%, 5% and 7%). The input SNR is calculated using the equation,

\[
ISNR = 10 \log_{10} \left[ \frac{\sum I_{org}^2}{\sum (I_{noisy} - I_{org})^2} \right]
\]

(4.2)

where \(I_{org}\) is the original image and \(I_{noisy}\) is the image with various noise levels. The output SNR is calculated using the equation...
\[ FSNR = 10 \log_{10} \left( \frac{\sum I_{\text{Filt}}^2}{\sum (I_{\text{noisy}} - I_{\text{Filt}})^2} \right) \] (4.3)

where \( I_{\text{Filt}} \) is the filtered image obtained using either MOD_LBP or MOD_LTP. The SNR of MOD_LTP and MOD_LBP over various noise levels is shown in Figure 4.6.

![Figure 4.6 The SNR of MOD_LTP and MOD_LBP over various noise level](image)

The SNR of MOD_LTP (20db) is more than that of MOD_LBP (5.6db) in the presence of 1% noise. The SNR of the MOD_LTP becomes closer to that of MOD_LBP when the noise level present in the image is above 9%. The SNR of MOD_LTP over different user specified threshold is given in Figure 4.7. It is seen that the SNR is high when the user specified threshold is in the range of 0.24 to 0.36 for all levels.
Figure 4.7 SNR of MOD_LTP over different user specified threshold

The Table 4.2 shows the time for LBP computation of different LBP variants when implemented in an unoptimized Matlab with specifications, Windows 7, Intel Core 2 duo at 2.10GHz 3 GB of DDR3 RAM, Matlab version 7.11(R2010b). The results show that $LBP_{P,R}^{ui}$ takes much lesser time than that of $LBP_{P,R}^{ri}$. The results also reveal that as the window size increases the time computation also increases. The rate of change of increase of time with respect to window size is very less in the case of MOD_LBP and MOD_LTP compared to other variants. The time taken for LBP computation is 5 times more (0.207s for $LBP_{8,1}^{ui}$ and 1.19s for $LBP_{16,2}^{ri}$) when window size changes from window size 3 to window size 5. But in the case of MOD_LTP,
the time computation is only 3 times greater as the window size increases from 3 to 5. Since the feature extraction of the database is done offline, the system response time may not be deviated much compared to MOD-LBP, but less than that of MOD-LBP. The average response time of the system is less than 10 seconds for the given database.

**Table 4.2 Time taken for different variants of LBP computation**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Time(S)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$LBP_{8,1}^{ri}$</td>
<td>0.218 s</td>
</tr>
<tr>
<td>$LBP_{8,1}^{rsu}$</td>
<td>0.207 s</td>
</tr>
<tr>
<td>$LBP_{16,2}^{ri}$</td>
<td>6.14 s</td>
</tr>
<tr>
<td>$LBP_{16,2}^{rsu}$</td>
<td>1.19 s</td>
</tr>
<tr>
<td>MOD_LBP(8,1)</td>
<td>0.706 s</td>
</tr>
<tr>
<td>MOD_LBP(24,2)</td>
<td>2.09 s</td>
</tr>
<tr>
<td>MOD_LTP(8,1)</td>
<td>0.68 s</td>
</tr>
<tr>
<td>MOD_LTP(24,2)</td>
<td>2.04 s</td>
</tr>
</tbody>
</table>

**DISCUSSION AND CONCLUSION**

This chapter illustrates how MOD_LTP is used to locate relevant slices from the MR image database. It shows that the MOD_LTP with a proper choice of threshold is more discriminate and less sensitive to noise in uniform or near uniform regions. The system performance has been evaluated based on user defined global threshold as well as variance based local threshold. The automatic choice of threshold based on local variance makes the system more robust to global illumination changes.