Med-Auxiliator

Med-Auxiliator aims to process the medical images to predict the ailments. Processing of the medical image databases does not aim to replace the physician by predicting the problem of a particular case but to assist him/her in diagnosis. The visual characteristics of a disease carry diagnostic information and oftentimes visually similar images correspond to the same disease category.

The Purpose of the project is to design a system which enhances the quality of medical scans and has some diagnostic capabilities in order to identify the defect. The proposed system should also retrieve visually similar images for reference.

Objectives

* To extract the features such as colour, texture and shape.
* To transform a discrete image which is defined at one set of coordinate locations to a new set of coordinate points i.e., converting from rectangular to hexagonal grid.
* To enhance the scans of medical images for better visibility.
* To diagnose the ailment.
* To provide visual indication of the affected area.
* To provide a brief description of the problem detected.
* To retrieve images of similar cases for reference.

Methodology

Algorithm

1. Remove noise in the query image using a median filter.

2. The image is resampled to hexagonal grid from the rectangular grid.

The approach followed for resampling is:

for each odd line

find the mid-point between two adjacent pixels by simple linear interpolation.

3. Feature extraction:

Color: Get maximum and minimum pixel value.

Calculate standard deviation, variance and mean.

Texture: Calculate texture parameters using Gray Scale co-occurrence matrix (GLCM)

For calculating GLCM measures for each pixel:

1. Read the input image.

2. Convert the data type to double and Zero pad the image

3. Extract a 3×3 window image from the input image and compute the co- occurrence texture measure

4. Estimate the texture parameters for the obtained texture image

5. Repeat the step3 and step4 by moving the window till the end of the image

6. Display various texture parameters by normalizing them

4. Detection of Edges:

Apply Gabor filter bank for hexagonal sampled grid choosing 3 orientations 00, 600 and 1200.

5. Shape: the shapes using moment invariants.

The feature Vector is stored in the database in the following form:

|  |  |  |
| --- | --- | --- |
| Color  Std., variance, mean, max, min | Texture  4X4 = 16 values | Shape  7X4 = 28 values |

6. Retrieval:

For K-means clustering:

6.1. Initialize cluster centers randomly in texture image

6.2. For all the pixels in the image do the following

a) Compute the Euclidean distance of the feature vector from the cluster for every other cluster.

b) Assign the pixel to that cluster whose center yields the minimum distance from the feature vector

6.3. Update the cluster centers by computing the mean of the feature vectors of the pixels belonging to that cluster

6.4. Between two consecutive updates, if the changes in the cluster centers are less than a specified value, then stop

Else go to step 6.2

7. Similarity comparison: use Euclidean distance for similarity comparison.

8. Defect detection: Use color and texture parameters to identify the defect.

Give Visual indication of the affected area.

Generate text report of the defect.