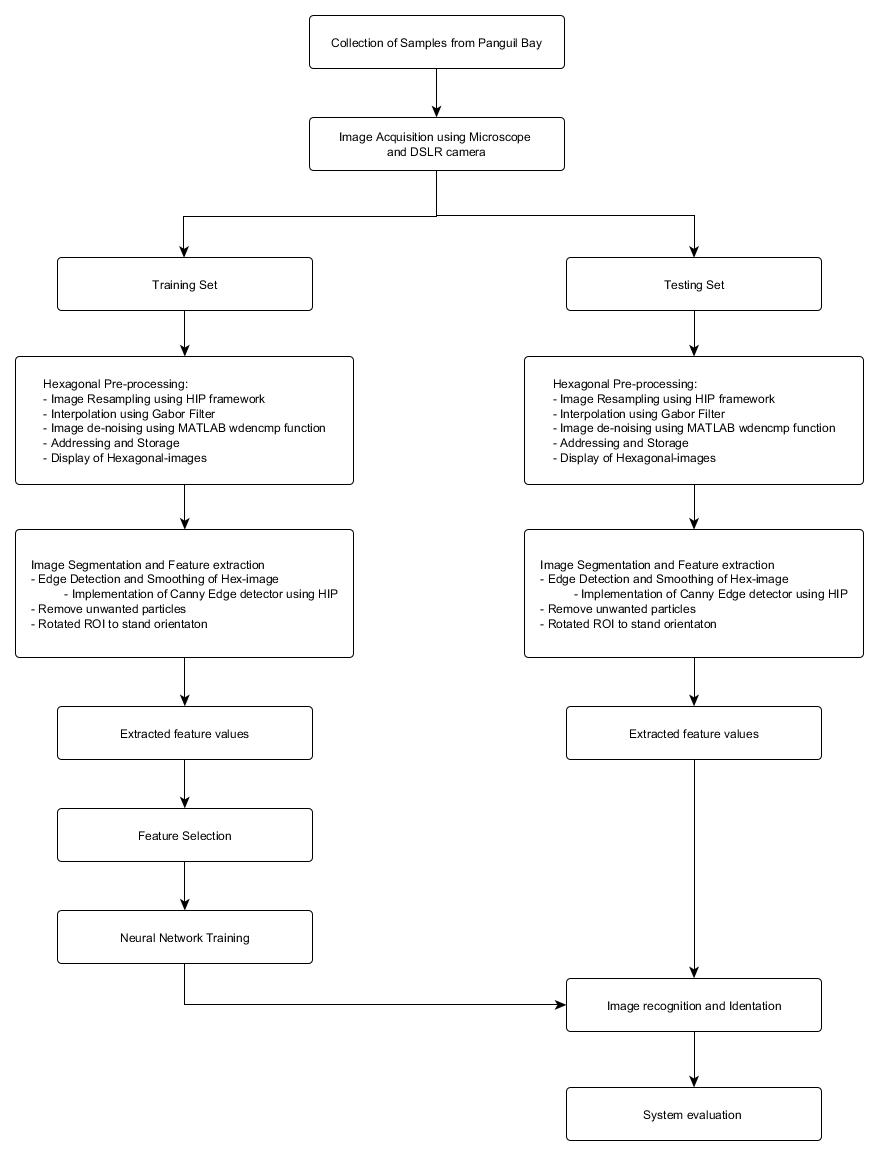
**Chapter 4: Methodology**



**Figure 4.1** Process Flowchart

**4.1 Sample Collection and Image Acquisition**

Five genera of marine copepods commonly encountered in mangrove waters will be examined: Acartia (A. spinicauda), Bestiolina (B. similis), Oithona (O. aruensis, O. dissimilis and O. simplex), Parvocalanus (P. crassirostris) and Tortanus (T. barbatus and T. forcipatus). Copepods will be sampled from four stations from the upper estuary in the Panguil bay to near shore waters of Marandin Lala Lanao Del Norte. Horizontal plankton tows (0.5-1 m depth) using paired 45 cm-diameter bongo nets (180 μm) will be made and collected plankton will be preserved in buffered 10% formaldehyde. In the laboratory, collected copepods will then be sieved through stacked Endecott sieves of 1,000 μm, 500 μm, 250 μm and 125 μm mesh sizes, and the sieved fractions will be preserved in 80% alcohol in individual vials for a long-term preservation.

**4.2 Image acquisition**

Specimens of copepod will be randomly pipetted onto a microscope slide from the preserved samples and each identified to species level under a compound microscope (Olympus BH2). To enable the dorsal aspect of the identified copepod to be imaged, often the copepod body had to be rotated. Body rotation could be easily achieved by first placing two short nylon fishing lines (0.36 mm diameter) on either side of the specimen and gently moving a cover slip placed over them by using the tip of the index finger. The desired view of the copepod body will be acquired by an Olympus digital camera (DP26) connected to a computer installed with an imaging software (Olympus cellSens Standard ver. 1.12) for real-time viewing, capturing and storing of the images. The built-in function in cellSens called Extended Focus Imaging (EFI) will be used to create a single plane image with sharp, in-focus details and high contrast. The EFI function recorded the image data as the sample was gradually focused through from top to bottom to obtain single dorsal image of the copepod with all body parts. Besides, the contrast and brightness of the images were set to the best before they were captured using cellSens software. The resolution of the captured images was standardised (2448 × 1920 pixels) and all the images were saved in uncompressed Tagged Image File Format (TIFF) by renaming them according to the date when the images were captured.

**4.3 Hexagonal Pre-processing**

**4.3.1 Image Resampling using HIP framework**

The regular image will undergo resampling to obtain a hexagonal image using the framework proposed by Middleton in 2005. It used the Python Imaging Library (PIL) to handle images. The proposed code can handle both grayscale images and coloured images. The requirement for the code to run would be as follows:

1. The image should be a valid PIL image
2. The layers in the image should be determined.
3. The spacing between point in the hex lattice should be determined
4. A kernel should be decided



**4.3.2 Interpolation using Gabor Filter**

The interpolation is done by using Gabor filter using following manner. Hexagonal sampled grid has 3 directional symmetry in 0o, 60o and 120o orientations. Due to these three axes of symmetry of hexagonal grid, we select three different orientation of Gabor filter along in 0o, 60o and 120o and the filtering is done in these three orientations. The three filtered images will be superimposed to get the interpolated image. The interpolated image is then used for wavelet based de-noising (Jeevan and Krishnakumar, 2016).

**4.3.3 Image de-noising scheme**

a. Perform wavelet decomposition of image obtained after interpolation.

b. Perform de-noising using the MATLAB function ‘wdencmp’**.**

**4.4.4 Addressing and Storage**

This process will be made using the HIP framework code where the latter provides a class that implements HIP arithmetic via new data type known as a Hexint. The method to be used is somewhat a modified form of the GBT system by Gibson in 1982.

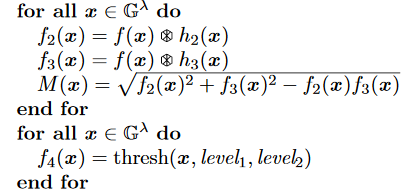
After the Addressing process, the images will be stored in the local machine for further processing.

**4.4.5 Display of Hexagonal-images**

For the visualization of the hexagonal images the HIP framework will be used where the code requires that the Python OpenGL extension be installed as part of Python installation. The provided code is a simple viewer that allows rotation and scaling of the resulting images. This will handle both spatial and frequency domain HIP images.

**4.5 Canny edge detector using HIP**

The Canny edge detector will be used for segmentation and feature extraction due to the reasons discussed in the previous chapter. The algorithm for this operation is as follow.To compute two masks *h2* and *h3,* oriented as in Prewitt edge detector the Gaussian smoothing and derivative operations are combined. The HIP address is converted to Cartesian coordinates before computing the oriented masks. The specific mask weights are found using the directional derivatives of the Gaussian function at the Cartesian coordinates.



The smoothing process is a built-in function of the Canny edge detector and can be utilized to denoise the image from salt-and-pepper noise from the water.

**4.6 Image resampling from Hex to Square image**

The image will be converted back to square lattice in order to perform the remaining steps. The HIP will be used and will require the following:

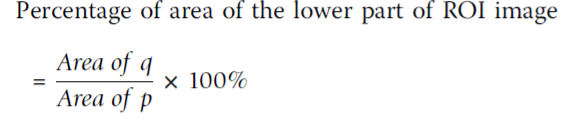
The requirement for the code to run would be as follows:

1. The HIP image
2. Radius to resample
3. The spacing between point in the hex lattice should be determined
4. A kernel should be decided

**4.6 Image segmentation and Feature extraction**

The following general steps will be used to perform segmentation and to ready the image for segmentation. We will introduce Matlab software used by Leow in 2015 in his Novel study “Automated identification of copepods using digital image processing and artificial neural network”

* + 1. The images will be converted to binary images with appropriate threshold.
    2. Using the *imclearborder* function from the matlab, borders will be cleared and the holes that occurred during the process of converting the grayscale image into binary image will be filled using the *imfill* function.
    3. Small particles below 50000 pixels will be excluded to ensure only the copepods are segmented for feature.
    4. Orientation represented by the angle between the x-axis and the major axis of the ellipse that has the same second-moments as the region of interest (ROI) will be obtained using region properties function in Matlab. Image rotation will be done using the *imrotate* function so that the ROI has an orientation of 90 degrees.
    5. The ROI of the copepod will be cropped by getting the coordinates of the boundary of copepods.
    6. Features will be extracted from the shape descriptors represented by the binary images of the ROI using region properties function in Matlab. The measurements like area, convex area, eccentricity, major axis length, minor axis length, perimeter, solidity, equivdiameter (sqrt(4\*area/pi)), extent and orientation will be determined.
    7. As seen in the ROI images of copepod, the lower part showed distinct shapes across the eight species. In view of this distinct attribute, a secondary feature will be derived by assigning 60% of the ROI image height measured from the posterior end (end of urosome) to the anterior end (head of copepod) of copepod body as the lower part of ROI image. This ratio will be selected after conducting several tests using a set of ratios (90%, 80%,70%, 60% and 50%). This derived feature will be calculated as:



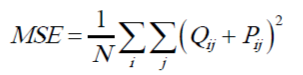
Where p is the total area of ROI image and q is the area of the lower part of ROI image.

**4.7 Feature Selection**

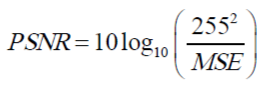
To avoid overfitting in the Neural Network training and to increase performance, not all the 11 extracted features will be used. The extracted features will be evaluated to make sure that only significant features will be selected to classify the copepods into their respective taxa. Forward stepwise discriminant analysis (FSDA) was used to aid the selection of the most useful features (StatSoft Inc.). In order to visualize how well a selected feature clustered the specimens in the training set into the eight classes (species), 2D and 3D scatter plots will be graphed with different combinations of features as the axes.

**4.8 Gabor and Wavelet de-noising Evaluation**

For the performance analysis, Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR), two commonly used measures for quantifying the error between images, will be used. MSE indicate the average difference of the pixel throughout the image. If MSE is higher the difference of the pixel between the original and the processed image is also higher. The MSE between two images P and Q is defined by,

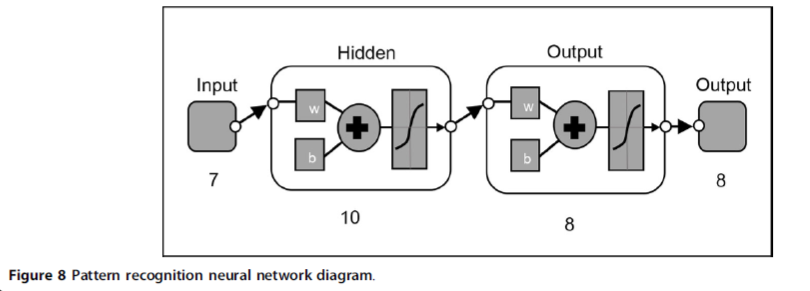


where, the sum over i and j denotes the sum over all pixels in the images, and N is the number of pixels in each image. The PSNR between two images is given by,



**4.9 Artificial Neural Network Training and Performance Evaluation**

An Artificial Neural Network (ANN) will be used as the pattern recognition tool to classify the extracted features values into the eight classes (species). The architecture of the ANN is a two-layer feed-forward network with sigmoid hidden (ten nodes) and output (eight nodes) neurons and the network will be trained with scaled conjugate gradient backpropagation. A total of 240 sample images will be used in the training set with 30 images from each class. The input data for the input nodes of the network will have seven selected features of each specimen from the training set, whereas the target data defined eight desired output classes. The 240 samples will be divided into three sets, the training set (168 samples, or 70% of samples), validation set (36 samples, 15%) and testing set (36 samples, 15%). The data from the training set will be used for network training; the validation set for measuring network generalization and terminating training before overfitting; and the testing set for independent measure of network performance during and after training. The performance of the network training will be evaluated using Mean Square Error (MSE) and confusion matrices. The training stopped when the MSE of the samples in the validation set started to increase indicating that the network generalization stopped improving. The network will be trained several times to get the trained network with best performance. Another 160 independent samples (20 samples for each species) will be used for system performance evaluation. The trained network will be simulated using the testing data as input and the output will be then compared to the predicted data and recorded in a confusion matrix.



**Figure 4.3** Pattern recognition neural network diagram