CSE 490 GZ
Assignment 6
Due February 20, 2004

1. In this problem you will investigate how the GLA algorithm works in 2-dimensions. Consider 2-dimensional vectors with each coordinate having 15 values, 0 to 15. Let our training set be $X = \{(0,0),(1,1),(2,2),...,(15,15)\}$. (Note: let's assume that the rounding function rounds down on .5, for example round of 6.5 is 6.)

   (a) Starting with vectors $c(0) = (0,7)$ and $c(1) = (15,7)$ run the GLA algorithm until there is no decrease in distortion.

   (b) What happens to the GLA if the starting vectors are $c(0) = (0,15)$ and $c(1) = (14,1)$?

   (c) Run the GLA algorithm with the splitting strategy. When splitting a codeword c, create a new codeword $c' = c + (1,1)$.

2. In this problem you will explore Orchard's algorithm, an alternative to the k-d tree for fast full search. The algorithm is explained in the paper on fast full search in the reading on the web. Suppose there are n codewords $c[i]$, $0 \le i \le n-1$. We construct a $n \times n$ array A where $A[i,j]$ where $A[i,0] = i$, $A[i,1]$ is the index of the closest codeword to codeword i, $A[i,2]$ is the index of the second closest codeword to codeword i, and so on. Generally, $A[i,j]$ is the index of the j-th closest codeword to codeword i. In a simple version of Orchard's algorithm, suppose we are searching for the closest codeword to vector c. Every search starts with codeword indexed $k = 1$, $c[1]$. Compute $r = \|c - c[1]\|$. Generally, r is the distance from c to $c[k]$, where k is index of the closest codeword to c found so far. In each iteration of the algorithm we search the subarray $A[k,j]$, $j = 1,2,\dots n$, until either (i) we find the first k' such that $\|c - c[k']\| < r$ or (ii) we find the first k' such that $\|c[k] - c[k']\| \ge 2r$ or (iii) the subarray is searched and neither (i) nor (ii) occurs. In case (i) we set k to k' and recompute $r = \|c - c[k]\|$ and do another iteration. In cases (ii) and (iii) we are done and k is the index of the closest codeword to c.

   (a) Consider the codewords $(4,4),(10,4),(7,7),(4,10),(10,10)$ indexed 0 to 4, respectively. Construct the array A needed to run Orchard's algorithm.

   (b) Run Orchard's algorithm on the input $(9,7)$.

   (c) Generally, Orchard's algorithm will be run for many inputs in the encoding process.

i. Explain in detail how taking square roots can be avoided.

ii. Explain how an additional data structure (array) can be constructed during the sorting phase so that no distance computation has to be made in the test used in (ii).

iii. Explain how an additional data structure (array) can be used during the algorithm to avoid computing the distance twice between c and a codeword in the test used in (i).