APPENDIX G

vqglbg.m

```
%
% Vector Quantization
%
function r = vqlbg(d, k)
e   = 0.01;
r   = mean(d, 2);
dpr = 1000;

for i = 1:log2(k)
  r = [r*(1+e), r*(1-e)];
  while (1 == 1)
   z = disteu(d, r);
   [m, ind] = min(z, [], 2);
   t = 0;
   for j = 1:2^i
     r(:, j) = mean(d(:, find(ind == j)), 2);
     x = disteu(d(:, find(ind == j)), r(:, j));
     for q = 1:length(x)
       t = t+x(q);
     end
   end
   if ((dpr -t)/t < e)
     break;
   else
     dpr = t;
   end
 end
end
```

noise,m

```
function d = noise(s, fs)

%%%% Noise

var = 0.001;                                          % noise variance

mean = 0;                                             % noise mean

n = randn(size(s)) * var + mean*ones(size(s));

%%%%%
```

```
signal = n+s;                      % add Gaussian noise to the signal y

Yfft=fft(s);                                  % FFT of original signal

Xfft=fft(signal);                            % FFT of signal with noise
```