```
%Code to apply Shannon Fano coding to a grayscale image
clc;
clear all;
close all;
I=imread("img.png");
if size(I,3)==3
    I=rgb2gray(I);
end
figure
imshow(I);
counts=imhist(I);
p=counts/sum(counts);

symbols=find(p>0)-1;

p=p(p>0);
%Finding frequency of each gray level intensity.
%Normalizing histogram counts into
%Extracting only those intensity values that
%Removing all zero probability gray levels.
[p_sorted,idx]=sort(p,'descend');

symbols_sorted=symbols(idx);

%Sorting probabilities from highest to
%Rearranging symbols in the same sorted
codes=strings(1,length(symbols_sorted));
%Creating an empty string array to store Shannon-Fano binary codes.
codes=shannon_fano(symbols_sorted,p_sorted,codes,1,length(p_sorted));
%Calling the recursive function that generates Shannon-Fano codes.
disp("Top 20 Shannon-Fano Codes for Image Symbols:");
disp("GrayLevel   Probability     Code");
disp("--------------------------------------");
for i=1:min(20,length(symbols_sorted))
    fprintf("%3d         %.6f     %s\n", ...
        symbols_sorted(i),p_sorted(i),codes(i));
end
%Displaying only the most frequent gray levels and their corresponding
 codes.
Lavg=0;
for i=1:length(p_sorted)
    Lavg=Lavg+p_sorted(i)*strlength(codes(i));
end
%Computing the average code length using probability weighted sum.
H=0;
for i=1:length(p_sorted)
    H=H-p_sorted(i)*log2(p_sorted(i));
end
%Applying Shannon entropy formula H=-#(p*log2(p)) for binary coding.
1
disp("--------------------------------------");
fprintf("Entropy(H)=%.4f bits/pixel\n",H);
fprintf("AverageCodeLength(Lavg)=%.4f bits/pixel\n",Lavg);
fprintf("CodingEfficiency=%.2f %%\n",(H/Lavg)*100);
%Efficiency indicates how close coding is to the theoretical entropy limit.
function codes=shannon_fano(symbols,p,codes,startIdx,endIdx) %shannon fano
if startIdx>=endIdx
return;
end
%Stopping recursion when only one symbol remains.
    totalProb=sum(p(startIdx:endIdx));
%Calculating total probability of the current symbol group.
    runningSum=0;
    splitIdx=startIdx;
for i=startIdx:endIdx
        runningSum=runningSum+p(i);
%Finding cumulative probability until it reaches half of total.
if runningSum>=totalProb/2
            splitIdx=i;
break;
end
end
for i=startIdx:splitIdx
        codes(i)=codes(i)+"0";
end
%Assigning binary 0 to the first probability subset.
for i=splitIdx+1:endIdx
        codes(i)=codes(i)+"1";
end
%Assigning binary 1 to the second probability subset.
    codes=shannon_fano(symbols,p,codes,startIdx,splitIdx);
    codes=shannon_fano(symbols,p,codes,splitIdx+1,endIdx);
%Recursively repeating the splitting until all symbols get a unique code.
end
```

```
Top 20 Shannon-Fano Codes for Image Symbols:
GrayLevel   Probability     Code
--------------------------------------
 15         0.030728     000000
 12         0.029807     000001
 11         0.027862     00001
 13         0.026006     00010
224         0.023823     00011
 10         0.022920     001000
 16         0.022282     001001
 14         0.021703     00101
221         0.021110     001100
219         0.021011     001101
220         0.020328     00111
223         0.019739     010000
218         0.019169     010001
222         0.018625     010010
 17         0.017646     010011
 18         0.015485     010100
  9         0.015184     010101
217         0.014519     01011
225         0.014371     0110000
 19         0.012219     0110001

ans =

     1

--------------------------------------
Entropy(H)=6.8818 bits/pixel
AverageCodeLength(Lavg)=6.9626 bits/pixel
CodingEfficiency=98.84 %
```

Published with MATLAB® R2021a