```
%Code to apply Shannon Fano coding to a grayscale image
clc;
clear all;
close all;
I=imread("anu.jpg");
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=-Σ(p*log2(p)) for binary coding.

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
--------------------------------------
255         0.019146     0000000
254         0.013537     0000001
252         0.007802     000001
253         0.007502     0000100
```

```
242         0.007443     0000101
243         0.007247     0000110
247         0.006914     0000111
244         0.006860     00010000
241         0.006541     00010001
249         0.006499     0001001
251         0.006354     0001010
239         0.006239     0001011
248         0.006208     00011000
246         0.005894     00011001
238         0.005437     0001101
237         0.004931     0001110
236         0.004881     0001111
127         0.004843     00100000
240         0.004789     00100001
132         0.004779     00100010
--------------------------------------
Entropy(H)=7.8871 bits/pixel
AverageCodeLength(Lavg)=7.9456 bits/pixel
CodingEfficiency=99.26 %
```

Published with MATLAB® R2021a