```
% Gamma Distribution Maximum Likelihood Estimation
% Please run 'Init_Read' before this script

% Fitting training data
[Ks, Thetas] = gammaMLFitting(TrainSet);

% Set the priors
% Since we do not have prior knowledge of the data, the priors are
 uniform
Priors = ones(MAX_CLASS, 1)./MAX_CLASS;

% For each test sample (coming from different classes), test against
% all the classes
Likelihoods = cell(1, 1);
Denominators = cell(1, 1);
Posteriors = cell(1, 1);
for i = 1 : MAX_CLASS % For each test set
    SampleLikelihoods = zeros(size(TestSet{i}, 1), MAX_CLASS);
    for r = 1 : size(TestSet{i}, 1) % in each test sample
        for p = 1 : MAX_CLASS % against each trained sample
   % The likelihood of a sample is the product of likelihood of every
 pixels
            SampleLikelihoods(r, p) = prod(gampdf(TestSet{i}(r, :),
 Ks{p}, Thetas{p}));
        end
    end

 % Store the likelihoods of each sample in each test set
    Likelihoods{i} = SampleLikelihoods;
end

% Calculate Posterior using Bayes' rule.
% This snippet is one of the examples given by the textbook
for i = 1 : MAX_CLASS
    Denominator{i} = 1 ./ (Likelihoods{i} * Priors);
    Posteriors{i} = (Likelihoods{i} * diag(Priors));
    Posteriors{i} = (Posteriors{i}.' * diag(Denominator{i}));
end

% Check the amount of samples that are correctly labeled.
% For each test sample (coming from different classes), check if the
% maximum posterior is the correct class
CorrectCount = zeros(1, MAX_CLASS);
for i = 1 : MAX_CLASS
    [M,I] = max(Posteriors{i});
    CorrectCount(i) = nnz(I==i);
end

% Print stuff
CorrectPercentages = CorrectCount./TestCount*100;
TotalPercentage = sum(CorrectCount)/sum(TestCount);
disp(sprintf('Correct/Total: %.2f%%', TotalPercentage*100));
```

```
Correct/Total: 50.77%
```

Published with MATLAB® R2016a