# Assignment A4: Frequency Domain Filtering

***CS 4640***
***Fall 2021***

**Assigned:** 28 September 2021

**Due:** 21 October 2021

For this problem, handin a report (PDF file) as required below as well as the Matlab .m files for the functions described by the headers below, and any help functions you write.

None of the functions should read or write files, write to the interpreter, draw, plot, etc. unless explicitly required by the header.

1. Develop a texture feature analysis tool based on the 2D FFT power spectrum. For every 5x5 region in the image, compute the 2D FFT, compute the power spectrum, and return that as a 25-element feature vector. Develop the function *CS4640_FFT_texture* described below. This function should do the one thing the header says! Discuss any issues arising from this, and show results on a few selected dataset images mixing defect types as follows:

- Compute the FFT texture vectors for the image. I.e., this results in a matrix T of dimension M*Nx25.
- Run the Matlab kmeans function on T to cluster the vectors into 4 or 5 sets: e.g., [cidx,ctr] = kmeans(T,4);
- Reshape the cluster indexes back into an MxN array; i.e., im = reshape(cidx,M,N).
- Evaluate whether this results data in helpful for inspection.

```
function T = CS4640_FFT_texture(im)
% CS4640_FFT_texture - compute FFT texture parameters
```

```
% On input:
%     im (MxN array): input gray level image
% On output:
%     T (M*Nx25 array): texture parameters
%        each texture parameter is a column vector in T
% Call:
%     T = CS4640_FFT_texture(im);
% Author:
%     <Your name>
%     UU
%     Fall 2021
%
```

2. Use CS4640_FFT_texture to develop the following functions:

- CS4640_FFT_cap: determine if the middle bottle has a cap.
- CS4640_FFT_underfilled: determine if the middle bottle is underfilled.
- CS4640_FFT_overfilled: determine if the middle bottle is overfilled.
- CS4640_FFT_white_label: determine if the middle bottle has a white label.

Explain your development of these and discuss any issues that arose. Show results on a few images with a mix of defects,

```
function b = CS4640_FFT_cap(im)
% CS4640_FFT_cap - determine if middle bottle has a cap
% On input:
%     im (MxNx3 array): RGB image
% On output:
%     b (Boolean): 1 if cap exists, else 0
% Call:
%     b = CS4640_FFT_cap(im);
% Author:
%     <Your Name>
%     UU
%     Fall 2021
%
```

```
function b = CS4640_FFT_underfilled(im)
% CS4640_FFT_cap - determine if middle bottle is underfilled
% On input:
%     im (MxNx3 array): RGB image
% On output:
%     b (Boolean): 1 if underfilled, else 0
% Call:
%     b = CS4640_FFT_underfilled(im);
% Author:
%     <Your Name>
%     UU
%     Fall 2021
%


function b = CS4640_FFT_overfilled(im)
% CS4640_FFT_cap - determine if middle bottle is overfilled
% On input:
%     im (MxNx3 array): RGB image
% On output:
%     b (Boolean): 1 if overfilled, else 0
% Call:
%     b = CS4640_FFT_overfilled(im);
% Author:
%     <Your Name>
%     UU
%     Fall 2021
%


function b = CS4640_FFT_white_label(im)
% CS4640_FFT_cap - determine if middle bottle has white label
% On input:
%     im (MxNx3 array): RGB image
% On output:
%     b (Boolean): 1 if white label, else 0
% Call:
%     b = CS4640_FFT_white_label(im);
% Author:
%     <Your Name>
%     UU
```

```
%      Fall 2021
%
```

3. Incorporate the 4 FFT-based defect detection functions into CS4640_inspect_Coke and run it on the 141 dataset. Using the provided defects array (in defects.mat in the A4 files in Canvas), report for each type of defect the number of true positives, false negatives, false positives and false neagtives. If the results are not perfect, discuss what went wrong and what should be investigated to rememdy it.