CS6640 8 Oct 2020

- Summary of Fourier Transform

for $u = 0 : M-1$ and $v = 0 : N-1$

$$F(u,v) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y)\, e^{-i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

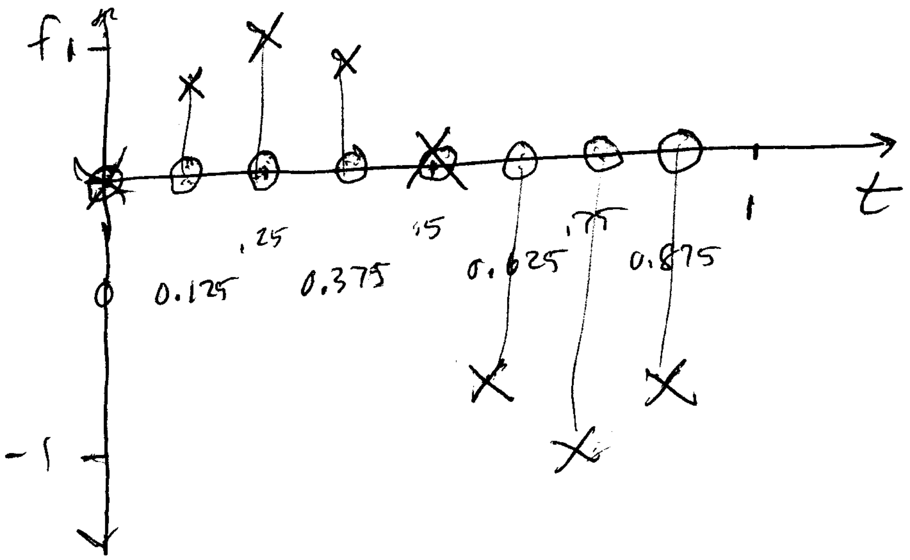
$$f(x,y) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v)\, e^{i2\pi\left(\frac{ux}{M} + \frac{vy}{N}\right)}$$

Consider a simple function in 1D:

$$f(x) = \sin(2\pi x)$$

(see cs6640_week6.m)

Sample at 8 points across $[0, 2\pi]$




f= [0, 0.7071, 1, 0.7071, 0, -0.7071, -1, -0.7071]

8 Oct 2012

Then:

$$F(u) = \begin{cases} \sum_{x=0}^{7} f(x)\, e^{-i2\pi ux/8} & \leftarrow \text{Matlab} \\ \frac{1}{2.8284} * \sum_{x=0}^{7} f(x)\, e^{-i2\pi ux/8} \end{cases}$$

depending on implementation

$\frac{1}{2.8284} \leftarrow \frac{1}{\sqrt{M}}$

Let $\bar{B}_u = [e^{-i2\pi u\cdot 0/8}, e^{-i2\pi u\cdot 1/8}, \ldots, e^{-i2\pi u\cdot 7/8}]^T$

then $F(u) = \frac{1}{2.8284} \bar{f} \cdot \bar{B}_u$

This coefficient is the projection of $\bar{f}$ onto $\bar{B}_u$

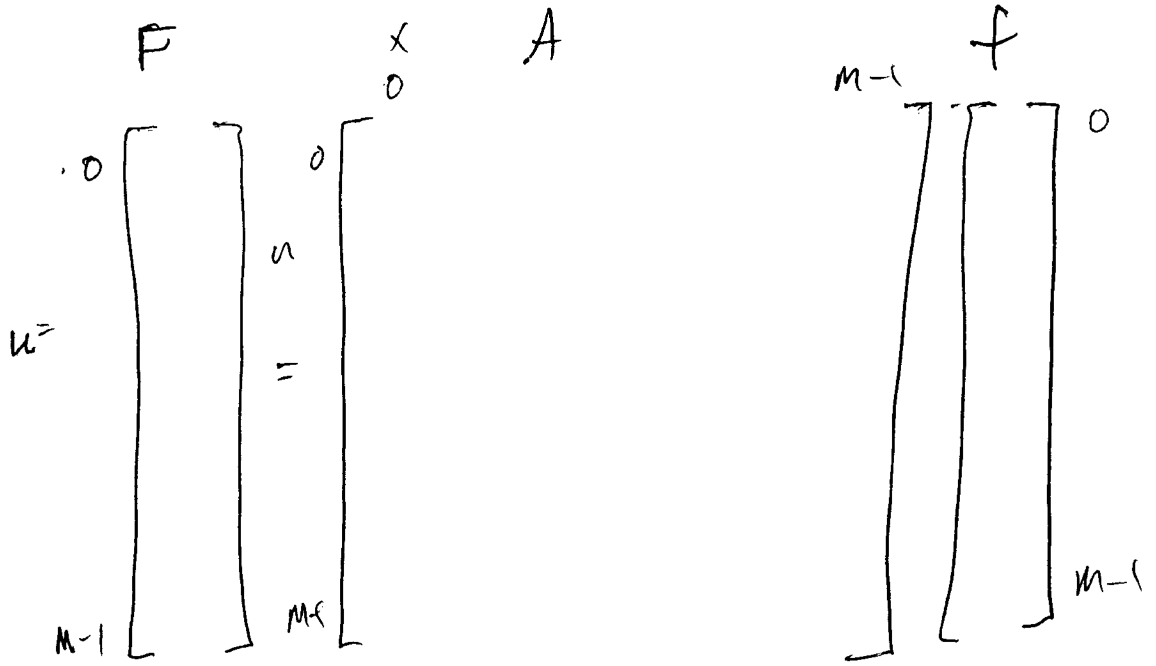
e.g.) $\bar{B}_0 = [1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]^T$

$$\bar{f} \cdot \bar{B}_0 \doteq 0 \quad = \sum f = \sum \begin{cases} 0, 0 \\ 0.7071, 0.7071 \\ 1 \\ -0.7071, -0.7071 \\ -1 \end{cases}$$

In general, the basis vectors are complex

8 Oct 2013

Matrix form:




$$F(0)\quad \begin{bmatrix} 0 \\ \\ \\ \\ \\ \\ \\ \\ \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \\ & & & & & & & \end{bmatrix} \begin{bmatrix} 0 \\ 0.7071 \\ 1 \\ 0.7071 \\ 0 \\ -0.7071 \\ -1 \\ -0.7071 \end{bmatrix}$$

8 Oct 2014

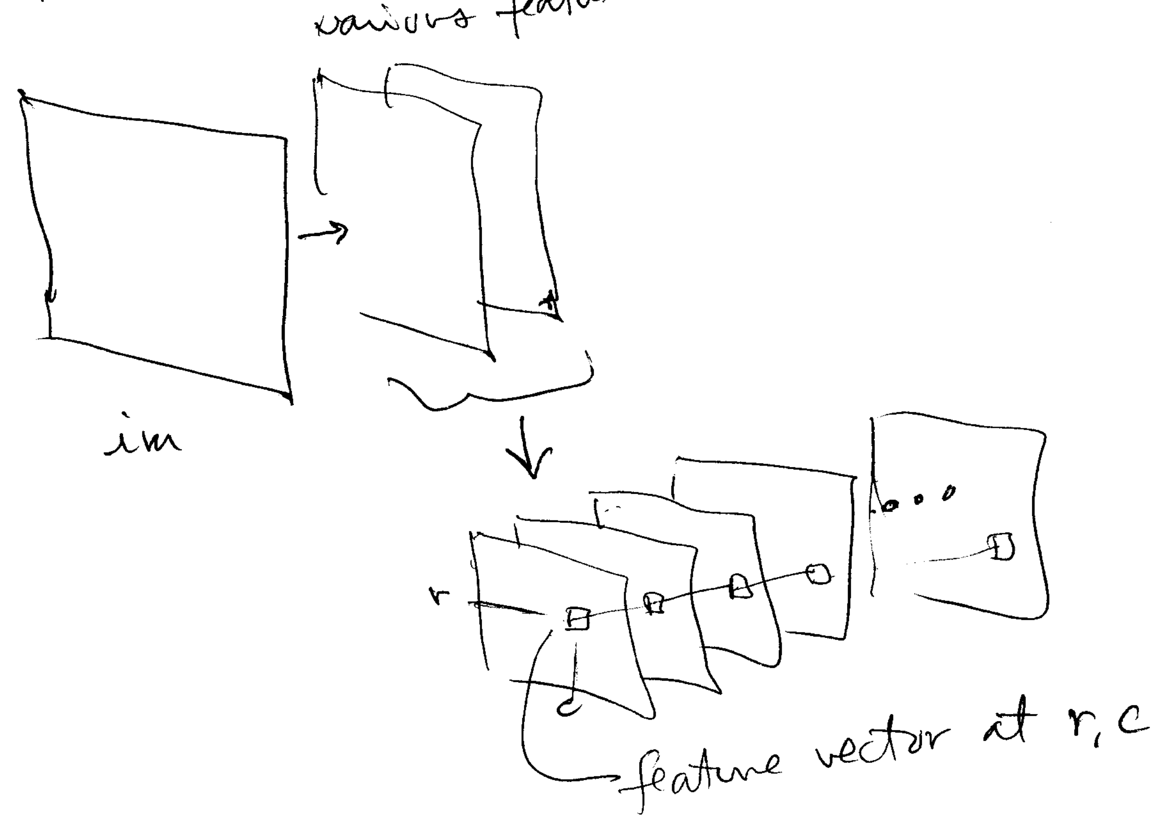
P2

Maps find edges + orientations & use properties

```
[dx,dy] = gradient(double(maplg));
mag = sqrt(dx.^2 + dy.^2);
ori = posori(atan2(dy,dx));
```

(to be clustered by Kmeans)

- feature vectors
  * use mag histogram of subwindow around each pixel
  * " ori "
  * find representative vectors & use Euclidean distance to classify
  * Laws vectors
  * FFT vectors



8 Oct 2015

bottle inspection

e.g., cap detection:

* use window around pixel in r, g & b channels