Computational Photography Assignment 3

Single Michael
08-917-445

Task 1

Def(Linear) Given two measurable function f and g and α, $\beta \in \mathbb{C}$. An operator $\mathcal{F}$ is called *linear* if the following identity holds true:

$$\mathcal{F}\left\{\alpha f + \beta g\right\}(x) = \alpha \mathcal{F}\left\{f\right\}(x) + \beta \mathcal{F}\left\{g\right\}(x) \tag{1}$$

Def(Shift Invariant) Given a measurable function f. An operator $\mathcal{F}$ is called *shift invariant* if the following identity holds true:

$$\mathcal{F}\left\{S_\delta f\right\}(x) = S_\delta(\mathcal{F}\left\{f\right\})(x) \tag{2}$$

where the shift operator S_δ is defined as the following:

$$S_\delta(f)(x) = f(x + \delta) \tag{3}$$

For any given function f and small number $\delta \in \mathbb{R}$.

Note that notational convention the operation notion for any operator F, we write $F(f)(x)$ instead of $F(f(x))$. In words we apply the operator F (i.e. the functional F) to the function f. Thus is would even be possible to omit the argument x. However to make the reasoning clear, I will keep it.

Therefore, directly applying the definition from equation 3 to equation 2 we can also define the property *shift invariant* as the follwoing:

$$\mathcal{F}\left\{S_\delta(f(x))\right\} := \mathcal{F}\left\{f(x+\delta)\right\} = S_\delta(\mathcal{F}\left\{f\right\}(x) \tag{4}$$

Relying on the definitions of the equations before, equation 4 and equation 1 let I solved this task,

(a) $\mathcal{F}\left\{f\right\}(x) = e^{f(x)}$ is *not linear* and is *shift invariant.*

Show: $\mathcal{F}$ is not *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f + \beta g\right\}(x) &= e^{\alpha f(x) + \beta g(x)} \\ &= e^{\alpha f(x)} e^{\beta g(x)} \\ &\neq \alpha e^{f(x)} + \beta e^{g(x)} \\ &= \alpha \mathcal{F}\left\{f\right\}(x) + \beta \mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is *shift invariant*

Proof.

$$\begin{aligned}\mathcal{F}\left\{S_\delta(f(x))\right\} &= \mathcal{F}\left\{f(x+\delta)\right\} \\ &= e^{f(x+\delta)} \\ &= S_\delta(e^{f(x)}) \\ &= S_\delta(\mathcal{F}\left\{f\right\}(x)\end{aligned}$$

□

(b) $\mathcal{F}\left\{f\right\}(x) = f(x)f(x-1)$ is *not linear* and is *shift invariant.*

Show $\mathcal{F}$ is not *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f+\beta g\right\}(x) &= (\alpha f(x)+\beta g(x))(\alpha f(x-1)+\beta g(x-1)) \\ &= \alpha^2 f(x)f(x-1)+\beta^2 g(x)g(x-1)+\alpha\beta(f(x)g(x-1)+f(x-1)g(x)) \\ &\neq \alpha f(x)f(x-1)+\beta g(x)g(x-1) \\ &= \alpha\mathcal{F}\left\{f\right\}(x)+\beta\mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is *shift invariant*

Proof.

$$\begin{aligned}\mathcal{F}\left\{S_\delta(f(x))\right\} &= \mathcal{F}\left\{f(x+\delta)\right\} \\ &= f(x+\delta)f(x-1+\delta) \\ &= S_\delta(f(x)f(x-1)) \\ &= S_\delta(\mathcal{F}\left\{f\right\}(x)\end{aligned}$$

□

(c) $\mathcal{F}\left\{f\right\}(x)=\sum_{k=x-4}^{x+2} f(k)$ is *linear* and is *shift invariant*.

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f+\beta g\right\}(x) &= \sum_{k=x-4}^{x+2}(\alpha f(k)+\beta g(k)) \\ &= \alpha\sum_{k=x-4}^{x+2} f(k)+\beta\sum_{k=x-4}^{x+2} g(k) \\ &= \alpha\mathcal{F}\left\{f\right\}(x)+\beta\mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is *shift invariant*

Proof.

$$\begin{aligned}
\mathcal{F}\{S_\delta(f(x))\} &= \mathcal{F}\{f(x+\delta)\} \\
&= \sum_{k=x-4+\delta}^{x+2+\delta} f(k) \\
&= \sum_{k=x-4}^{x+2} f(k+\delta) \\
&= S_\delta(\sum_{k=x-4}^{x+2} f(k)) \\
&= S_\delta(\mathcal{F}\{f\}(x)
\end{aligned}$$

$\square$

(d) $\mathcal{F}\{f\}(x) = f(2x)$ is *linear* and is *not shift invariant.*

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}
\mathcal{F}\{\alpha f + \beta g\}(x) &= \alpha f(2x) + \beta g(2x) \\
&= \alpha \mathcal{F}\{f\}(x) + \beta \mathcal{F}\{g\}(x)
\end{aligned}$$

$\square$

Show $\mathcal{F}$ is not *shift invariant*

Proof.

$$\begin{aligned}
\mathcal{F}\{S_\delta(f(x))\} &= \mathcal{F}\{f(x+\delta)\} \\
&= f(2(x+\delta)) \\
&= f(2x+2\delta) \\
&\neq f(2x+\delta) \\
&= S_\delta(f(2x)) \\
&= S_\delta(\mathcal{F}\{f\}(x)
\end{aligned}$$

$\square$

(e) $\mathcal{F}\{f\}(x) = \sin(2x)f(x)$ is *linear* and is *not shift invariant.*

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f + \beta g\right\}(x) &= \sin(2x)\left(\alpha f(x) + \beta g(x)\right)\\ &= \sin(2x)\alpha f(x) + \sin(2x)\beta g(x)\\ &= \alpha\sin(2x) f(x) + \beta\sin(2x) g(x)\\ &= \alpha\mathcal{F}\left\{f\right\}(x) + \beta\mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is not *shift invariant*

Proof.

$$\begin{aligned}\mathcal{F}\left\{S_\delta(f(x))\right\} &= \mathcal{F}\left\{f(x+\delta)\right\}\\ &= \sin(2x) f(x+\delta)\\ &\neq \sin(2x+\delta) f(x+\delta)\\ &= S_\delta(\sin(2x) f(x))\\ &= S_\delta(\mathcal{F}\left\{f\right\}(x)\end{aligned}$$

□

(f) $\mathcal{F}\left\{f\right\}(x) = x f(x)$ is *linear* and is *not shift invariant.*

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f + \beta g\right\}(x) &= x(\alpha f(x) + \beta f(x))\\ &= x\alpha f(x) + x\beta f(x)\\ &= \alpha x f(x) + \beta x f(x)\\ &= \alpha\mathcal{F}\left\{f\right\}(x) + \beta\mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is not *shift invariant*

Proof.

$$\begin{aligned}
\mathcal{F}\{S_\delta(f(x))\} &= \mathcal{F}\{f(x+\delta)\} \\
&= xf(x+\delta) \\
&\neq (x+\delta)f(x+\delta) \\
&= S_\delta(xf(x)) \\
&= S_\delta(\mathcal{F}\{f\}(x)
\end{aligned}$$

□

(g) $\mathcal{F}\{f\}(x) = f(x) - f(x-5)$ is *linear* and is *shift invariant.*

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}
\mathcal{F}\{\alpha f + \beta g\}(x) &= (\alpha f(x) + \beta g(x)) - (\alpha f(x-5) + \beta g(x-5)) \\
&= (\alpha f(x) - \alpha f(x-5)) + (\beta g(x) - \beta g(x-5)) \\
&= \alpha(f(x) - f(x-5)) + \beta(g(x) - g(x-5)) \\
&= \alpha\mathcal{F}\{f\}(x) + \beta\mathcal{F}\{g\}(x)
\end{aligned}$$

□

Show $\mathcal{F}$ is *shift invariant*

Proof.

$$\begin{aligned}
\mathcal{F}\{S_\delta(f(x))\} &= \mathcal{F}\{f(x+\delta)\} \\
&= f(x+\delta) - f(x-5+\delta) \\
&= S_\delta(f(x) - f(x-5)) \\
&= S_\delta(\mathcal{F}\{f\}(x)
\end{aligned}$$

□

(h) $\mathcal{F}\{f\}(x) = f(x-3) - 2f(x-12)$ is *linear* and is *shift invariant.*

Show $\mathcal{F}$ is *linear*

Proof.

$$\begin{aligned}\mathcal{F}\left\{\alpha f+\beta g\right\}(x) &= (\alpha f(x-3)+\beta g(x-3))-2(\alpha f(x-12)+\beta g(x-12))\\ &= (\alpha f(x-3)-2\alpha f(x-12))+(\beta g(x-3)-2\beta g(x-12))\\ &= \alpha(f(x-3)-2f(x-12))+\beta(g(x-3)-2g(x-12))\\ &= \alpha\mathcal{F}\left\{f\right\}(x)+\beta\mathcal{F}\left\{g\right\}(x)\end{aligned}$$

□

Show $\mathcal{F}$ is *shift invariant*

Proof.

$$\begin{aligned}\mathcal{F}\left\{S_\delta(f(x))\right\} &= \mathcal{F}\left\{f(x+\delta)\right\}\\ &= f(x-3+\delta)-2f(x-12+\delta)\\ &= S_\delta(f(x-3)-2f(x-12))\\ &= S_\delta(\mathcal{F}\left\{f\right\}(x)\end{aligned}$$

□

Task 2

Given an $m \times n$ monochromatic (i.e. there is only one color-channel) Image I. Give an algorithm how to apply box-filtering on this image. Furthermore analyse the asymptotic complexity of this algorithm.

Algorithm 1 Moving Average box filter

Input: Grayscale *Image* I with resolution $m \times n$
Output: Box filtered Image *Image* $\hat{I}$

Procedures: $getDimensions(Image)$, $zeros(height, width)$

$[h, w] = getDimensions(I)$
$\hat{I} = zeros(h, w)$
$r = \lceil \frac{w-1}{2} \rceil$
Foreach $Pixel\, p \in Image\, I$ **do**
  $contribution = 0$
  Foreach $Pixel\, p_n \in r - Neighborhood\ \mathcal{N}_r(p)$ **do**
    $contribution = contribution + I(p + p_n)$
  end for
  $\hat{I}(p) = \frac{contribution}{m \cdot n}$
end for

Remarks:

- By pixels in the Algorithm we are referring to the coordinates of the pixel in the image. Therefore p corresponds to the x and y coordinates of pixel p in the Image I.
- $I(p)$ denotes accessing the pixel-(color)-values in the images at the position of the pixel p in the image I.
- $\mathcal{N}_r(p)$ denotes the neighborhood with radius r around a given pixel p. In the context of pixel-coordinates, think of it as a box-grid, centred at the pixel coordinates of p. This grid has a radius of r. This means there are r neighbors (pixel-coordinates in the grid) below, on top, on the left and on the right of p.
- Our algorithm can easily be extended for color Images by simply applying the same algorithm to each color-channel separately.
- The assumption of being provided by a m by n can easily be extended for the case when $n \neq m$. This only will affect the computation of the radius r in algorithm .
  Computing $\lceil 0.5 \cdot \left(\lceil \frac{m-1}{2} \rceil + \lceil \frac{n-1}{2} \rceil \right) \rceil$ would be a valid option in order to compute r.
- If w (i.e. n) is odd, then $\lceil \frac{w-1}{2} \rceil$ is equal to $\frac{w-1}{2}$.
- The procedure *getDimensions* returns the width-and height resolution of a provided Image.
- The procedure *zeros* creates a new image with the provided resolutions.

Aysmptotic Complexity Next let us have a closer look into algorithm .

- The most outer foreach loop will iterate over each pixel. This loop corresponds to two for loops, one over iterating over each row in the image, another iterating over each column in the image. Assuming there are h rows and w columns in the image, this most outer foreach loop has an asymptotic complexity of $\mathcal{O}(h \cdot w)$. From the assignment description we are supposed to assume that there are m pixels in the image. Thus, we conclude, the most outer foreach loop has an asymptotic complexity in $\mathcal{O}(m)$.
- The inner foreach loop iterates over each neighorhood pixel around the current pixel in iteration (according the the most outer foreach loop). Assuming we are allowed to ignore boundary issue, we always visit a neighborhood of $\mathcal{O}\left((2r+1)^2\right)$ pixels. Referring to the definition of r at line 3 in algorithm we see that r linearly depends on the width of the image (note that we assume width is equal to height for the given image). Therefore $\mathcal{O}\left((2r+1)^2\right)$ is in the complexity class $\mathcal{O}\left(width^2\right)$.

We can conclude the following: The asymptotic complexity of algorithm is in $\mathcal{O}\left(k^2 m\right)$. Note that k denotes the width w from before and m is the number of pixels.

Task 3

Task 4

Task 5

Task 6