# Computational Photography Assignment 3

*Single Michael*
*08-917-445*

## Task 1

**Def(Linear)** Given two measurable function $f$ and $g$ and $\alpha$, $\beta \in \mathbb{C}$. 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 $S_\delta$ 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) $\mathcal{F}\left\{f\right\}(x) = \sum_{k=x-1}^{x+2} f(k)$ is *linear* and is *shift invariant.*

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

*Proof.*

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

□

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

*Proof.*

$$\begin{aligned}
\mathcal{F}\left\{S_\delta(f(x))\right\} &= \mathcal{F}\left\{f(x+\delta)\right\} \\
&= \sum_{k=x-1+\delta}^{x+2+\delta} f(k) \\
&= S_\delta(\sum_{k=x-1}^{x+2} f(k)) \\
&= S_\delta(\mathcal{F}\left\{f\right\}(x)
\end{aligned}$$

□

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

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

*Proof.*

$$a = a$$

□

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

*Proof.*

$$a = a$$

□

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

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

*Proof.*

$$a = a$$

□

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

*Proof.*

$$a = a$$

□

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

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

*Proof.*

$$a = a$$

□

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

*Proof.*

$$a = a$$

□

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

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

*Proof.*

$$a = a$$

□

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

*Proof.*

$$a = a$$

□

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

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

*Proof.*

$$a = a$$

□

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

*Proof.*

$$a = a$$

□

## 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 1 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 1.
  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**

## Task 3

## Task 4

## Task 5

## Task 6