

تمرین سری سه

درس تصویرپردازی رقمی

فرهاد دلیرانی

۹۶۱۳۱۱۲۵

dalirani@aut.ac.ir

dalirani.1373@gmail.com

فهرست

۱	ابزارهای استفاده شده
۲	تمرین ۱
۲	۲ قسمت a
۳	۳ قسمت b
۳	۳ d قسمت c و
۵	۵ تمرین ۲
۵	۵ قسمت a
۶	۶ قسمت b, c
۷	۷ قسمت d
۹	۹ تمرین ۳
۹	۹ قسمت (a)
۹	۹ زیرقسمت (a1)
۹	۹ زیرقسمت (a2)
۱۰	۱۰ زیرقسمت (a3)
۱۰	۱۰ زیرقسمت (a4)
۱۱	۱۱ زیرقسمت (a5)
۱۱	۱۱ زیرقسمت (a6)
۱۲	۱۲ زیرقسمت (a7)
۱۳	۱۳ زیرقسمت (a8)
۱۳	۱۳ زیرقسمت (a9)
۱۴	۱۴ زیرقسمت (a10)
۲۱	۲۱ قسمت (C)
۲۴	۲۴ قسمت (D)
۲۸	۲۸ تمرین ۴
۲۸	۲۸ قسمت (A)

٣١.....	قسمت (B)
٣٩.....	تمرين ٥)
٣٩.....	قسمت (a)
٣٩.....	زير قسمت (a1)
٤١.....	زير قسمت (a2)
٤٤.....	زير قسمت (a3)
٤٨.....	زير قسمت (a4)
٥١.....	زير قسمت (a5)
٥٤.....	قسمت (b)
٥٤.....	زير قسمت (b1)
٥٦.....	زير قسمت (b2)
٦١.....	زير قسمت (b3)
٦٦.....	تمرين ٦
٦٦.....	قسمت (a)
٧٠.....	قسمت (b)
٧٤.....	قسمت (c)
٧٥.....	قسمت (d)
٨٣.....	قسمت (e)
٩١.....	تمرين ٧)
٩١.....	قسمت (a)
٩٥.....	قسمت (b)
٩٩.....	قسمت (c)
١٠٣.....	تمرين ٨
١٠٣.....	قسمت (a)
١٠٤.....	قسمت (b)
١٠٦.....	قسمت (c)
١٠٧.....	قسمت (d)
١٠٨.....	قسمت (e)

ابزارهای استفاده شده

زبان برنامه نویسی: متلب ۲۰۱۶

محیط توسعه: متلب

سیستم عامل: ویندوز ۱۰

تمرين ١

قسمت a

1)

a)

a 1) $g(t) = \sin(t-2)$

$$f(t-t_0) \longleftrightarrow F(\omega) e^{-j\omega t_0}$$

$$\mathcal{I}\{\sin(t-2)\} = \mathcal{I}\{\sin(t)\} e^{-j\omega \times 2}$$

$$= [-j\pi (\delta(\omega-1) - \delta(\omega+1))] e^{-j\omega \times 2}$$

a 2) $g(t) = \frac{2t^2+1}{t^2+1}$

$$\mathcal{I}\left\{\frac{2t^2+1}{t^2+1}\right\} = 4\pi \delta(\omega) - \pi e^{-| \omega |}$$

a 3) $g(t) = \delta(t-1) + \delta(t+2)$

$$\mathcal{I}\{\delta(t+1) + \delta(t+2)\} = e^{-j\omega} + e^{2j\omega}$$

a 4) $\mathcal{J}(t) = F^{-1}\left\{\frac{1}{f^2+1}\right\} \rightarrow \mathcal{I}\left\{F^{-1}\left\{\frac{1}{f^2+1}\right\}\right\} = \frac{1}{f^2+1}$

a 5) $\Rightarrow g(t) = \cos(t) e^{-|t|}$

$$\mathcal{I}\{\cos(t) e^{-|t|}\} = \frac{2\omega^2+2}{\omega^2+4}$$

a 6) $g(t) = \cos(4\pi t) \cos(6\pi t) \Rightarrow$

$$\mathcal{I}\{\cos(4\pi t) \cos(6\pi t)\} = \frac{1}{2} \pi [8(\omega-10\pi) + \delta(\omega-2\pi) + \delta(\omega+2\pi) + \delta(\omega+10\pi)]$$

قسمت b

1)

b)

b1) $\operatorname{Re}\{X(j\omega)\} = 0$ این مخصوصیت را نشان کنید. $X(j\omega) = \frac{1}{j\omega + 2}$ مولار 2 می باشد. $\operatorname{Re}(X(j\omega)) = \operatorname{Re}\left(\frac{1}{j\omega + 2}\right) = \frac{1}{(j\omega)^2 + 4} < 0$

b2) $\operatorname{Im}\{X(j\omega)\} = 0$ این مخصوصیت را نشان کنید. $X(j\omega) = \frac{6}{j\omega + 5}$ مولار 5 می باشد. $\operatorname{Im}(X(j\omega)) = \operatorname{Im}\left(\frac{6}{j\omega + 5}\right) = \frac{6}{(j\omega)^2 + 25} < 0$

b3) $\int_{-\infty}^{\infty} X(j\omega) d\omega = 0$ این مخصوصیت را نشان کنید. $X(j\omega) = \frac{6}{j\omega + 5} + \frac{1}{j\omega - 1} + \frac{2}{j\omega - 3} + \frac{1}{j\omega - 6}$

b4) $\int_{-\infty}^{\infty} \omega X(j\omega) d\omega = 0$ این مخصوصیت را نشان کنید. $X(j\omega) = \frac{6}{j\omega + 5} + \frac{1}{j\omega - 1} + \frac{2}{j\omega - 3} + \frac{1}{j\omega - 6}$ $\operatorname{Re}(X(j\omega)) = 0$

b5) $X(j\omega)$ is periodic این مخصوصیت را نشان کنید.

b6) there is a real α such that $e^{j\omega\alpha}$ is real این مخصوصیت را نشان کنید. $X(j\omega) = \frac{6}{j\omega + 5} + \frac{1}{j\omega - 1} + \frac{2}{j\omega - 3} + \frac{1}{j\omega - 6}$ $e^{j\omega\alpha} = \cos(\omega\alpha)$

قسمت c و d

1)

c)

$$c_1) \quad x(t) = t e^{-2t} u(t) \quad h(t) = e^{-4t} u(t)$$

$$X(\omega) = \frac{1}{(2+j\omega)^2} \quad H(\omega) = \frac{1}{4+j\omega}$$

$$Y(\omega) = X(\omega)H(\omega) = \frac{1}{(2+j\omega)^2} * \frac{1}{(4+j\omega)}$$

$$\text{Convolution} = \mathcal{I}^{-1} \left\{ \frac{1}{(2+j\omega)^2} + \frac{1}{(4+j\omega)} \right\}$$

$$c_2) \quad x(t) = t e^{-2t} u(t) \quad h(t) = t e^{-4t} u(t)$$

1.50 L, 1.50 ↑

$$c_3) \quad x(t) = e^{-t} u(t) \quad h(t) = e^t u(-t)$$

$$X(\omega) = \frac{1}{1+j\omega} \quad H(\omega) = \frac{1}{1-j\omega}$$

$$Y(\omega) = X(\omega)H(\omega) = \left[\frac{1}{1+j\omega} \right] \left[\frac{1}{1-j\omega} \right]$$

$$\text{Convolution} = \mathcal{I}^{-1} \left\{ \left[\frac{1}{1+j\omega} \right] \left[\frac{1}{1-j\omega} \right] \right\}$$

2)

$$u_{\max} = 1_0 \pi$$

$$\frac{1}{T} > 2(1_0 \pi) \rightarrow \boxed{T < \frac{1}{2_0 \pi}}$$

تمرين ٢

قسمت a

2)

a)

2D Continuous Fourier Transform

$$\mathcal{F}\{\sin(\omega_0 x)\} = \frac{\pi}{J} [\delta(\omega - \omega_0) - \delta(\omega + \omega_0)]$$

$$\mathcal{F}\{\cos(\omega_0 x)\} = \pi [\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$$

$$\mathcal{F}\{\sin 8\pi x\} = \frac{\pi}{J} [\delta(\omega - 8\pi) - \delta(\omega + 8\pi)]$$

$$\mathcal{F}\{\cos 2\pi y\} = \pi [\delta(\omega - 2\pi) + \delta(\omega + 2\pi)]$$

$$\mathcal{F}\{f(x, y)\} = \mathcal{F}\{\sin(8\pi x) + \cos(2\pi y)\} = \mathcal{F}\{\sin(8\pi x)\} + \mathcal{F}\{\cos(2\pi y)\}$$

$$= \frac{\pi}{J} [\delta(\omega - 8\pi) - \delta(\omega + 8\pi)] + \pi [\delta(\omega - 2\pi) + \delta(\omega + 2\pi)]$$

2)

b), c) $\{1, 2, 3, 4, 5\}$

b) $f(0) = 1 \quad f(1) = 2 \quad f(2) = 3 \quad f(3) = 4 \quad f(4) = 5$

$$F(u) = \sum_{x=0}^{m-1} f(x) e^{-j2\pi ux/m}$$

$$F(0) = \frac{1}{5} \left[1 \times e^{-j2\pi(0)(0)} + 2 \times e^{-j2\pi(1)(0)} + 3 \times e^{-j2\pi(2)(0)} + 4 \times e^{-j2\pi(3)(0)} + 5 \times e^{-j2\pi(4)(0)} \right]$$

$$= \frac{1}{5} [1+2+3+4+5] = \frac{15}{5} = 3$$

$$F(1) = \frac{1}{5} \left[1 \times e^{-j2\pi(1)(1)} + 2 \times e^{-j2\pi(1)(1)} + 3 \times e^{-j2\pi(1)(1)} + 4 \times e^{-j2\pi(1)(1)} + 5 \times e^{-j2\pi(1)(1)} \right]$$

$$= \frac{1}{5} [1+2e^{-j\frac{2\pi}{5}} + 3e^{-j\frac{4\pi}{5}} + 4e^{-j\frac{6\pi}{5}} + 5e^{-j\frac{8\pi}{5}}]$$

$$F(2) = \frac{1}{5} \left[1 \times e^{-j2\pi(2)(0)} + 2 \times e^{-j2\pi(2)(0)} + 3 \times e^{-j2\pi(2)(0)} + 4 \times e^{-j2\pi(2)(0)} + 5 \times e^{-j2\pi(2)(0)} \right]$$

$$= \frac{1}{5} [1+2e^{-j\frac{4\pi}{5}} + 3e^{-j\frac{8\pi}{5}} + 4e^{-j\frac{12\pi}{5}} + 5e^{-j\frac{16\pi}{5}}]$$

$$F(3) = \frac{1}{5} \left[1 \times e^{-j2\pi(3)(0)} + 2 \times e^{-j2\pi(3)(0)} + 3 \times e^{-j2\pi(3)(0)} + 4 \times e^{-j2\pi(3)(0)} + 5 \times e^{-j2\pi(3)(0)} \right]$$

$$= \frac{1}{5} [1+2e^{-j\frac{6\pi}{5}} + 3e^{-j\frac{12\pi}{5}} + 4e^{-j\frac{18\pi}{5}} + 5e^{-j\frac{24\pi}{5}}]$$

$$F(4) = \frac{1}{5} \left[1 \times e^{-j2\pi(4)(0)} + 2 \times e^{-j2\pi(4)(0)} + 3 \times e^{-j2\pi(4)(0)} + 4 \times e^{-j2\pi(4)(0)} + 5 \times e^{-j2\pi(4)(0)} \right]$$

$$= \frac{1}{5} [1+2e^{-j\frac{8\pi}{5}} + 3e^{-j\frac{16\pi}{5}} + 4e^{-j\frac{24\pi}{5}} + 5e^{-j\frac{32\pi}{5}}]$$

c) $F(x) = \sum_{u=0}^{m-1} F(u) e^{-j2\pi ux/m}$
برای محاسبی تبدیل مکافون فوریه ای کمتر از محدود است
استفاده ای کمتر کرد صورت مبارکری مقدارها و نتایج را درست نماییم

$f(0) = 1 \quad f(1) = 2 \quad f(2) = 3 \quad f(3) = 4 \quad f(4) = 5$

2)

\mathbf{f})

$$c_1 = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \quad -j2\pi \left(\frac{ux}{M} + \frac{vy}{N} \right)$$

$$F(u, v) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi \left(\frac{ux}{3} + \frac{vy}{3} \right)}$$

$$F(u, v) = \frac{1}{9} \sum_{x=0}^2 \sum_{y=0}^2 \frac{1}{9} e^{-j2\pi \left(\frac{ux}{3} + \frac{vy}{3} \right)}$$

$$= \frac{1}{81} \sum_{x=0}^2 \sum_{y=0}^2 e^{-j2\pi \left(\frac{ux}{3} + \frac{vy}{3} \right)}$$

~~$\times 2V \times j2\pi \left(\frac{ux}{3} + \frac{vy}{3} \right)$~~

$$F(u, v) = \frac{1}{81} \left[e^{-j2\pi \left(-\frac{u}{3} + \frac{v}{3} \right)} + e^{-j2\pi \left(-\frac{u}{3} + \frac{-1}{3} \right)} + e^{-j2\pi \left(-\frac{u}{3} + \frac{1}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{v}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{-1}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{1}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{v}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{-1}{3} \right)} + e^{-j2\pi \left(\frac{u}{3} + \frac{1}{3} \right)} \right]$$

$v = -1, 0, 1 \quad , \quad u = -1, 0, 1$

$\rightarrow F(u, v) = \sum_{u=-1}^1 \sum_{v=-1}^1 f(u, v)$

2)

d)

~~2~~

$$C_2 : \begin{bmatrix} -1 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\begin{aligned}
 F(u, v) &= \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) e^{-j2\pi \left(\frac{ux}{M} + \frac{vy}{N} \right)} \\
 &= \frac{1}{3} \sum_{x=0}^2 \sum_{y=0}^2 f(x, y) e^{-j2\pi \left(\frac{ux}{3} + \frac{vy}{3} \right)} \\
 &= \frac{1}{3} \left[(-1) e^{-j2\pi \left(\frac{(0)u}{3} + \frac{(0)v}{3} \right)} + (-1) e^{-j2\pi \left(\frac{(1)u}{3} + \frac{(0)v}{3} \right)} + (-1) e^{-j2\pi \left(\frac{(0)u}{3} + \frac{(1)v}{3} \right)} \right. \\
 &\quad \left. + (-1) e^{-j2\pi \left(\frac{(1)u}{3} + \frac{(1)v}{3} \right)} + (-1) e^{-j2\pi \left(\frac{(2)u}{3} + \frac{(0)v}{3} \right)} + (-1) e^{-j2\pi \left(\frac{(1)u}{3} + \frac{(2)v}{3} \right)} \right]
 \end{aligned}$$

میانی $F(u, v)$ را می‌دانیم $v = -1, 0, 1$ و $u = -1, 0, 1$ باشد ω میانی (u, v) را داشته باشیم.

$\omega = (u, v)$

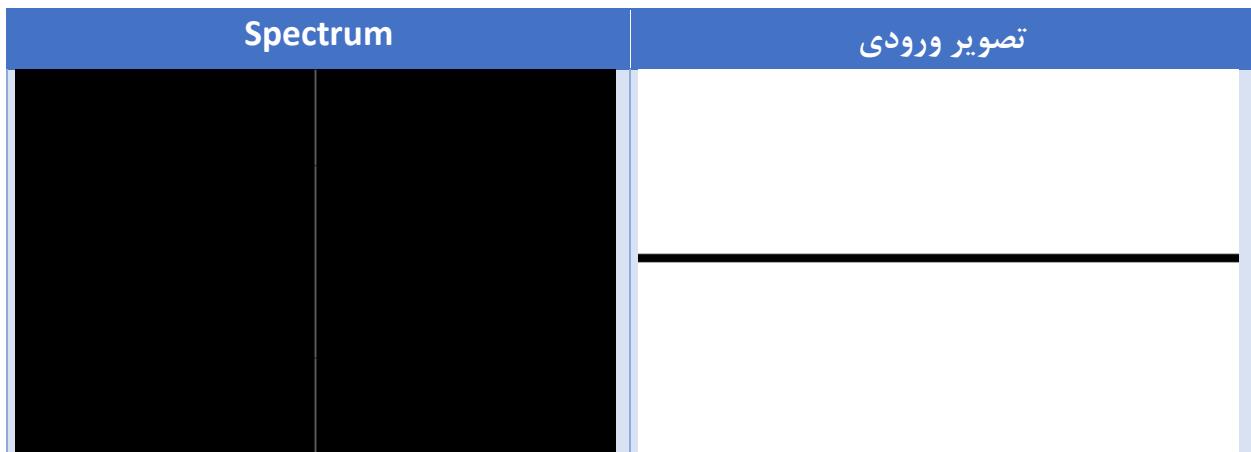
تمرین ۳

کدهای این قسمت در پوشه‌ی p3 قرار دارد.

(قسمت a)

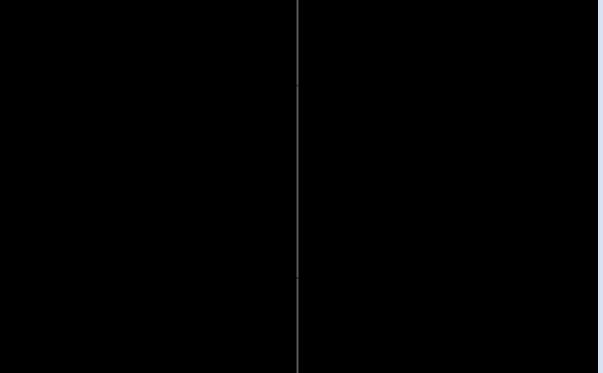
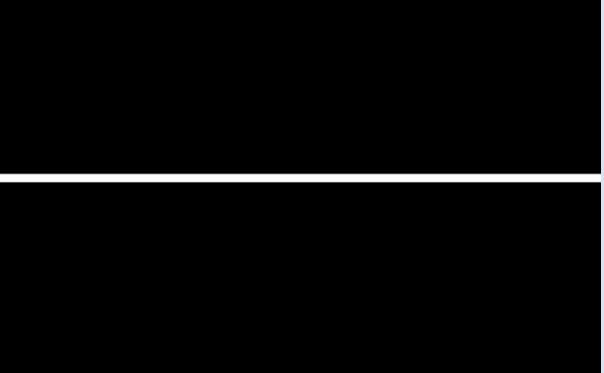
زیرقسمت (a1)

تصویر ورودی یک خط افقی سیاه در پس زمینه‌ی سفید است. با توجه به فرمول تبدیل فوریه و خصیت‌های آن، از آنجایی که خط در راستای افقی است، تبدیل فوریه‌ی آن باید خطی سفید در راستایی عمود بر آن باشد. برای این کار کد p3a_a1.m را نوشته‌ام که یک تصویر خط سیاه رنگ افقی را دریافت می‌کند و آن را Spectrum رسم می‌کند. در خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایید):



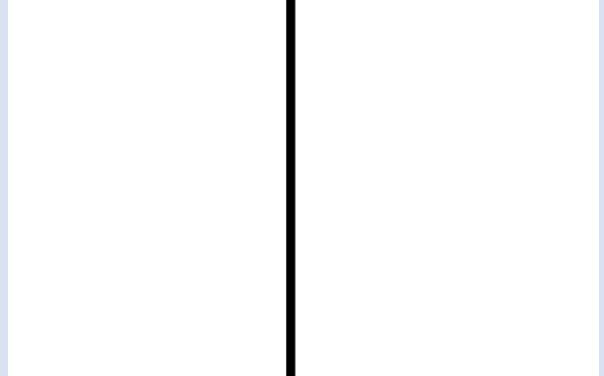
زیرقسمت (a2)

تصویر ورودی یک خط افقی سفید در پس زمینه‌ی سیاه است. با توجه به فرمول تبدیل فوریه و خصیت‌های آن، از آنجایی که خط در راستای افقی است، تبدیل فوریه‌ی آن باید خطی سفید در راستایی عمود بر آن باشد. برای این کار کد p3a_a2.m را نوشته‌ام که یک تصویر خط سفید رنگ افقی را دریافت می‌کند و آن را Spectrum رسم می‌کند. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایید):

Spectrum	تصویر ورودی
	

زیرقسمت (a3)

تصویر ورودی یک خط عمودی سیاه در پس زمینه‌ی سفید است. با توجه به فرمول تبدیل فوریه و خاصیت‌های آن، می‌توان گفت با دوران خط سیاه افقی قسمت a1 به اندازه‌ی نود درجه، خط عمودی را می‌توان ساخت، بنابراین خط عمودی سیاه برابر است با دوران spectrum خط افقی سیاه به اندازه‌ی نود درجه، به این ترتیب Spectrum برابر با یک خط افقی سفید است. برای این کار کد p3a_a3.m را نوشته‌ام که یک تصویر خط سیاه رنگ عمودی را دریافت می‌کند و آن را رسم می‌کند. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوششی p3/a مراجعه فرمایید):

Spectrum	تصویر ورودی
	

زیرقسمت (a4)

تصویر ورودی یک خط قطری سیاه در پس زمینه‌ی سفید است. با توجه به فرمول تبدیل فوریه و خاصیت‌های آن، می‌توان گفت با دوران خط سیاه افقی قسمت a1 به اندازه‌ی ۴۵ درجه، خط قطری را می‌توان ساخت، بنابراین

خط قطري سياه برابر است با دوران spectrum خط افقی سياه به اندازه‌ی ۴۵ درجه، به اين ترتيب Spectrum برابر با يك خط قطري سفيد در قطري مخالف قطر خط است.

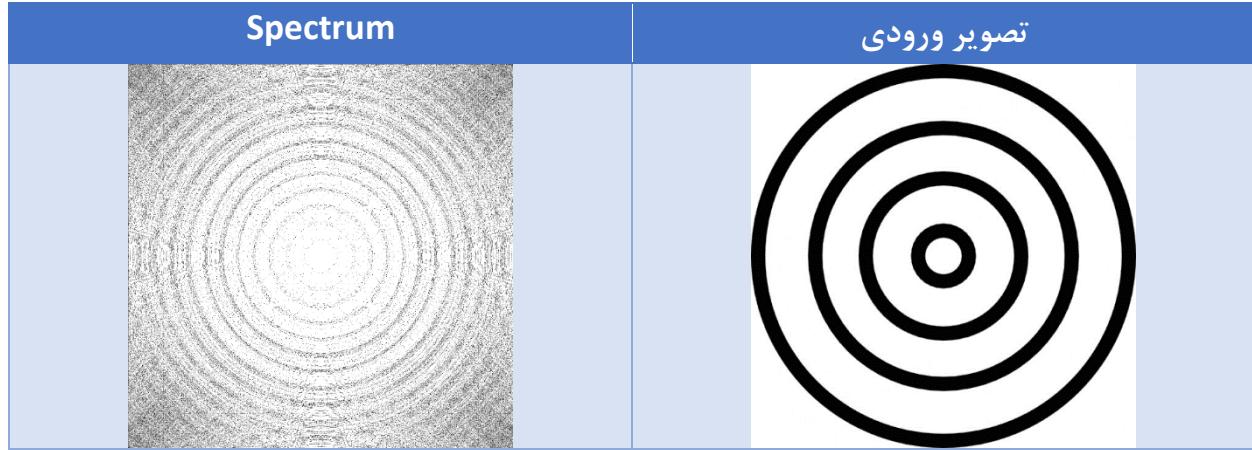
زيرقسمت (a5)

تصویر ورودی چند خط عمودی سياه در پس زمینه‌ی سفيد است. spectrum نسبت به انتقال ثابت است و تغييری حاصل نمی‌شود از آنجايی که در قسمت‌های قبل مشخص شده که Spectrum يک خط افقی سياه يک خط عمودی سفيد است، Spectrum چند خط موازي افقی نيز يک خط افقی است. برای اين کار کد p3a_a5.m را نوشته‌ام که يک تصویر خط سياه رنگ عمودی را دریافت می‌کند و Spectrum آن را رسم می‌کند. در زير خروجی‌های کد را مشاهده می‌کنيد (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایيد):



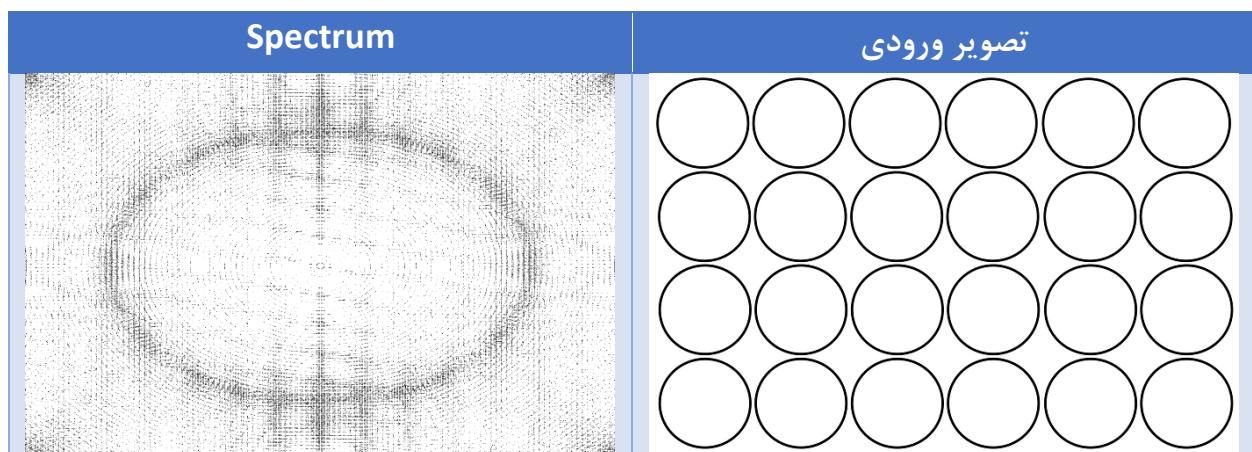
زيرقسمت (a6)

برای اين کار کد p3a_a6.m را نوشته‌ام که يک تصویر حاوي چند دايره‌ی هم مرکز را دریافت می‌کند و Spectrum آن را رسم می‌کند. در زير خروجی‌های کد را مشاهده می‌کنيد (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایيد). Spectrum يک دايره، شکلی شبیه به دايره‌های متعدد المرکز است، با زياد شده دايره‌ها در تصویر ورودی و اضافه کردن چندين دايره‌ی متعدد المرکز مانند اين است که فركانس رخ داد دايره‌ها در Spectrum يک دايره بيشرتر شود به همين دليل در Spectrum تصویر با چند دايره‌ی متعدد المرکز، تعداد رخ داد دايره‌ها بيشرتر می‌شود:



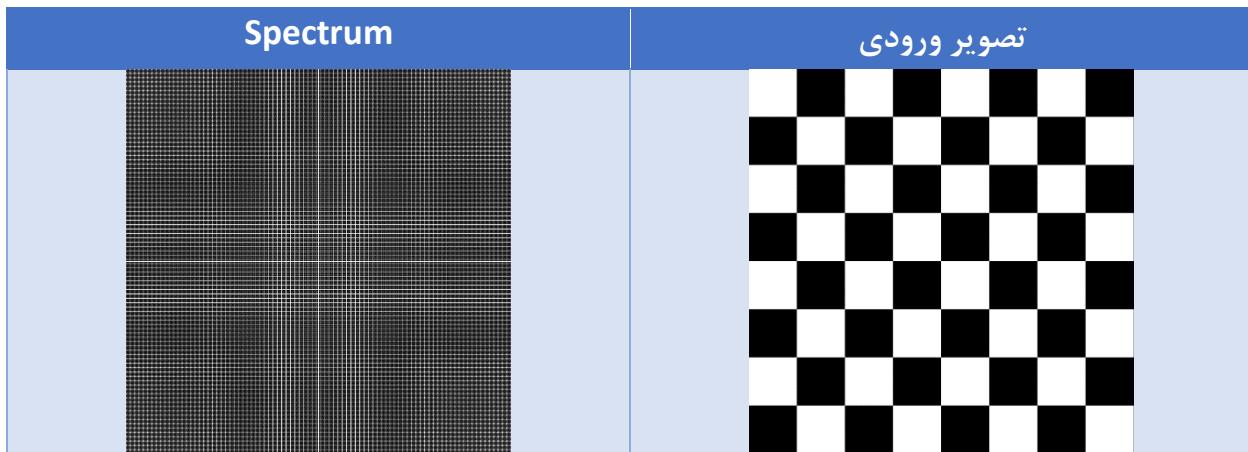
زیرقسمت (a7)

برای این کار کد p3a_a7.m را نوشته‌ام که یک تصویر حاوی دایره‌هایی چیده شده در یک شبکه را دریافت می‌کند و Spectrum آن را رسم می‌کند. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوششی p3/a مراجعه فرمایید). یک دایره، شکلی شبیه به دایره‌های متعدد المركز است، و از آنجایی که Spectrum نسبت به انتقال تاثیر ناپذیر است، تصویر ورودی دایره‌هایی متعدد المركز می‌شوند:



زیرقسمت (a8)

برای این کار کد p3a_a8.m را نوشتہام که یک تصویر صفحه‌ی شترنج را دریافت می‌کند و Spectrum را رسم می‌کند. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایید). خانه‌های سفید و سیاه در بازه‌های خاصی به صورت متناوب تکرار می‌شوند و لبه‌ها عمودی و افقی هستند در نتیجه grid یک Spectrum می‌شود:



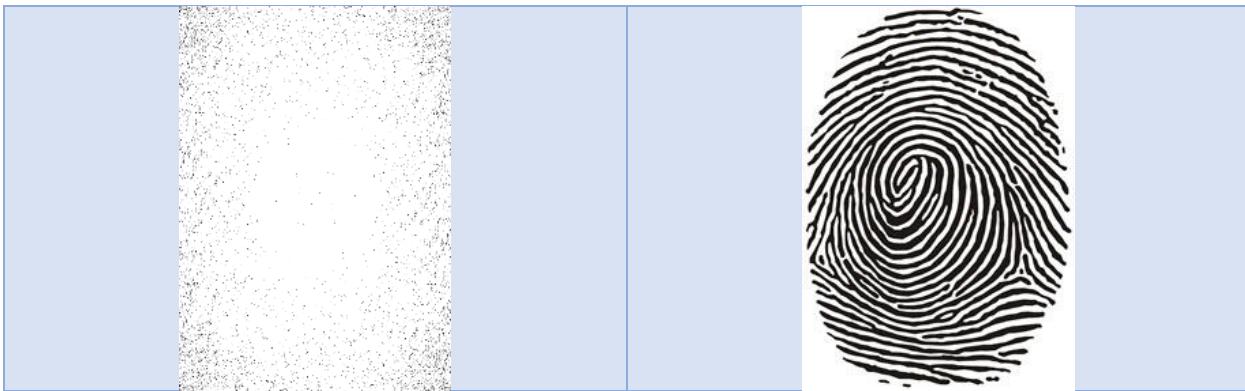
زیرقسمت (a9)

مجموعه‌ای از اشیا شبیه به هم، از آنجا که Spectrum به جایه جایی حساس نیست، تا حدودی شبیه به Spectrum شبیه یکی از آن‌ها می‌شود.

زیرقسمت (a10)

برای این کار کد p3a_a10.m را نوشتہام که یک تصویر اثر انگشت را دریافت می‌کند و Spectrum رسم می‌کند. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصاویر در سایز اصلی به پوشه‌ی p3/a مراجعه فرمایید). از آنجایی که اثر انگشت دارای لبه‌های زیادی در جهت‌های مختلف است، باید Spectrum آن تقریباً سفید باشد و تعدادی نقطه‌ی سیاه در آن قرار داشته باشد.:





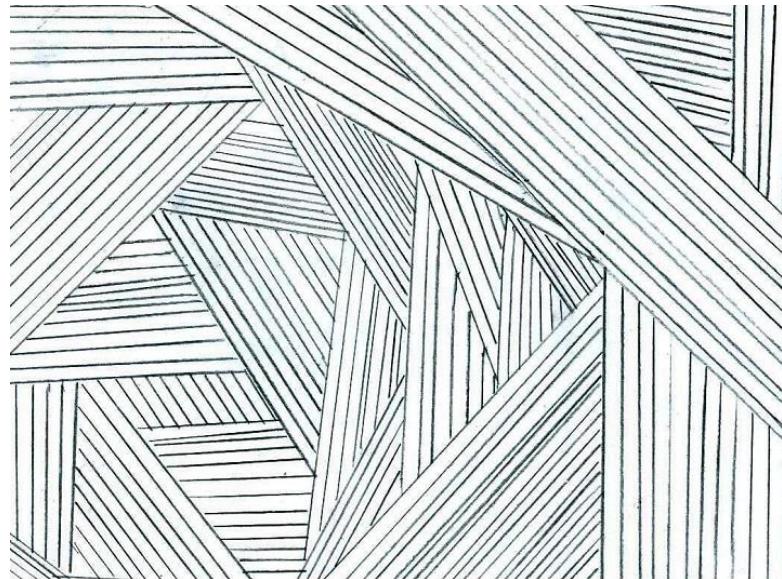
قسمت (b)

کد این قسمت در p3b.m قرار دارد. در این کد یک حلقه وجود دارد که به ازای ۴ تصویر ورودی مختلف عمل زیر را انجام می‌دهد:

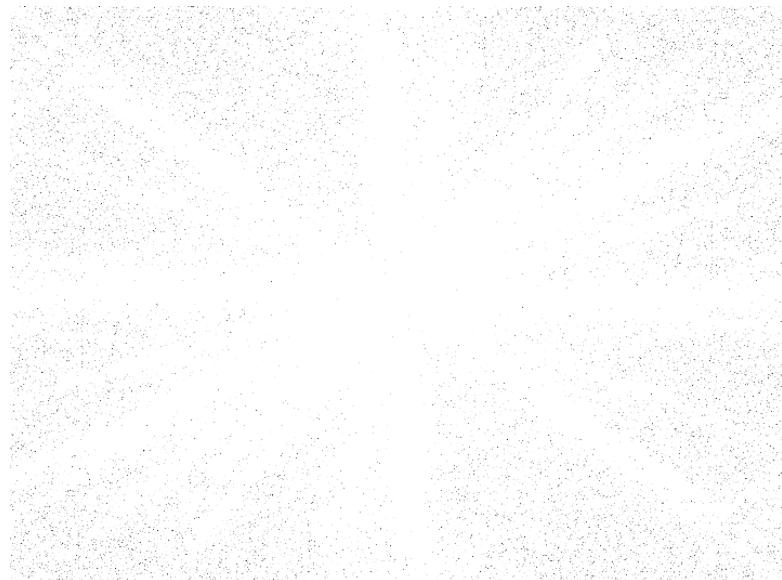
ابتدا تصویر خوانده می‌شود، سپس تصویر به صورت gray-level در می‌آید. سپس angle spectrum را حساب می‌کنیم، سپس مختصات صفر را به وسط شیفت می‌دهیم. در ادامه Inverse phase را محاسبه می‌کنیم. بعد صفر مختصات را به گوشه سمت چپ بالا منتقل می‌کنیم و در ادامه با Fast Fourier Transform تصویر اصلی را دوباره به دست می‌آوریم. در زیر خروجی‌های مختلف کد را مشاهده می‌کنید(برای دیدن تصویرها در سایز اصلی، به پوشه‌ی P3 مراجعه کنید):

- از \log_{10} برای نمایش Spectrum استفاده شده است.

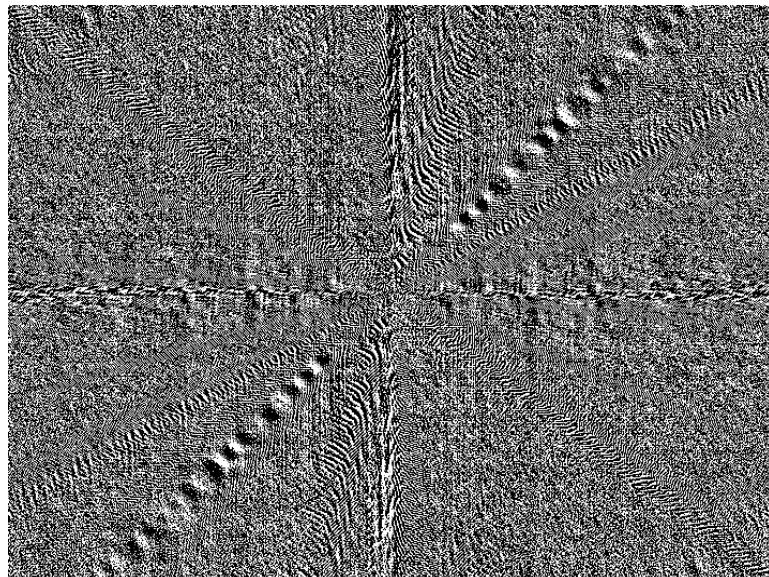
تصویر :sketch.jpg



: (p3b-sketch.jpg-spectrum.png) sketch.jpg تصویر Spectrum



: (p3b-sketch.jpg-phase.png) sketch.jpg تصویر Phase Angle

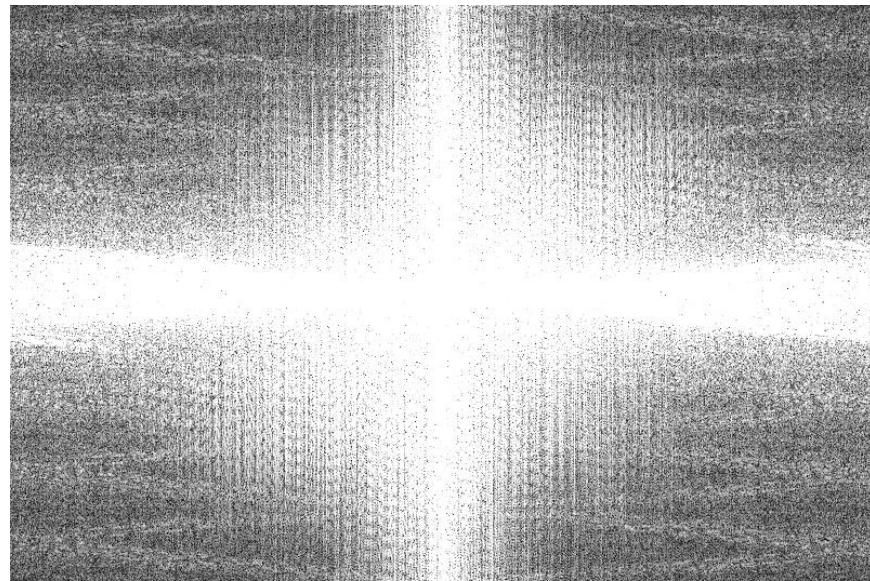


در تصویر ورودی خطها و لبهایی با زاویه‌های مختلف وجود دارند که اگر به Spectrum با دقت نگاه کنیم، خطهایی با زاویه‌های مختلف که از وسط صفحه عبور کرده‌اند مشاهده می‌شود. به دلیل عبور خطها و لبهای با زاویه‌های مختلف در تصویر ورودی، مقدار هر نقطه در Spectrum بالاست به همین دلیل spectrum بیشترش سفید رنگ است. تاثیر مشابه‌ای هم در Phase Angle دیده می‌شود.

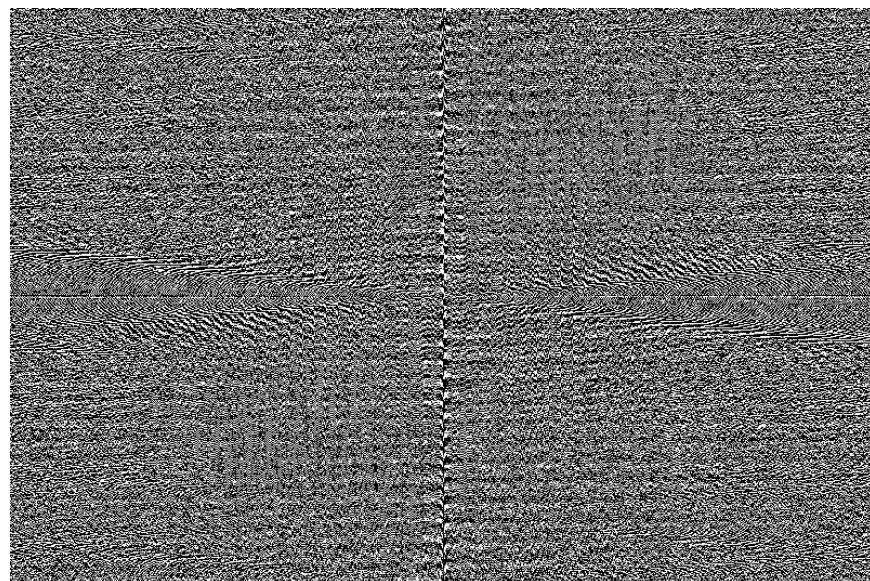
تصویر piano_keys.jpg



:(p3b- piano_keys.jpg-spectrum.png) piano_keys.jpg تصویر Spectrum



: (p3b- piano_keys.jpg-phase.png) piano_keys.jpg Phase Angle تصویر

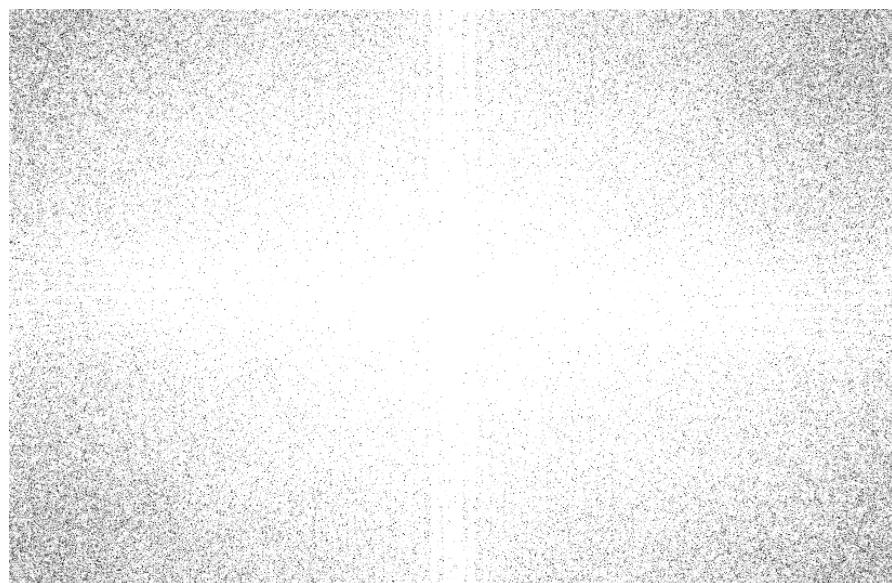


تصویر پیانو حاوی خطها و لبه‌های افقی و عمودی (کلیدهای) زیادی است به همین دلیل در Spectrum شاهد خطهای عمودی و افقی زیادی هستیم. تاثیر مشابه‌ای هم در Phase Angle دیده می‌شود.

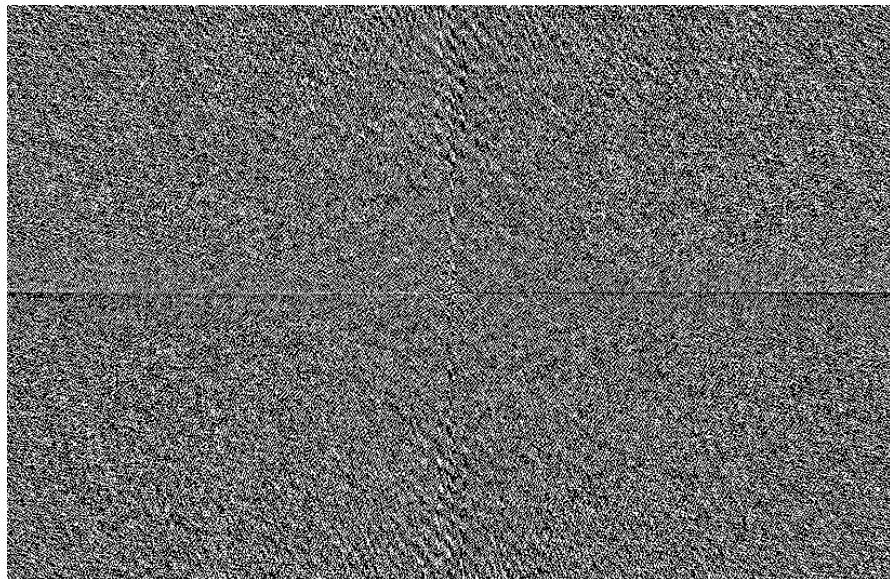
تصویر :chessboard.jpg



:
:(p3b-chessboard.jpg-spectrum.png)chessboard.jpg تصویر Spectrum



:
:(p3b-chessboard.jpg-phase.png) chessboard.jpg تصویر Phase Angle

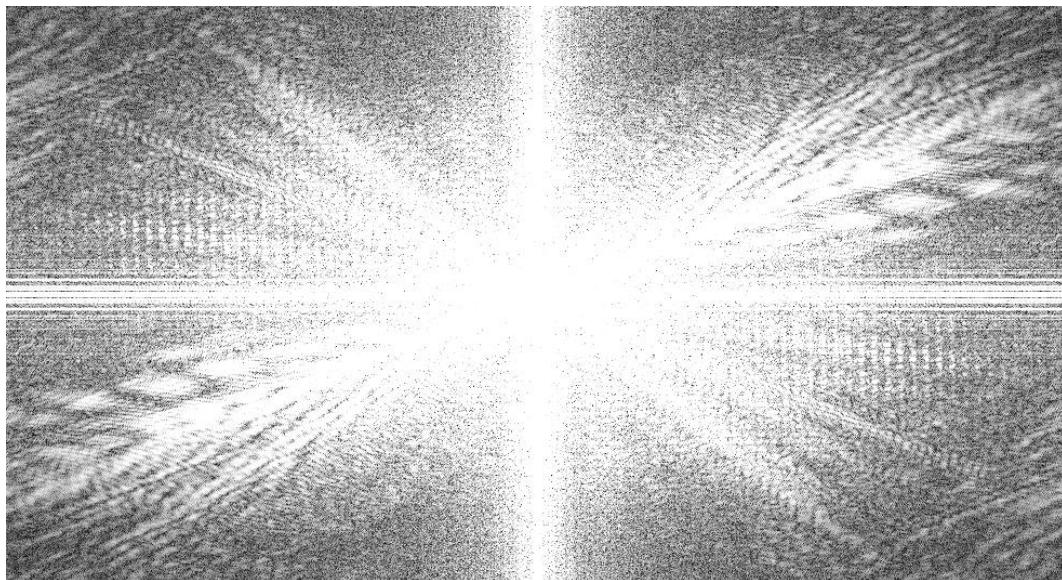


در تصویر صفحه‌ی شطرنج ورودی، لبه‌ها و خطهای با زاویه‌ها مختلف وجود دارد ولی خطها و لبه‌های افقی و عمودی (۶۴ خانه‌ی شطرنج) بیشتر هستند. به همین دلیل در spectrum تصویر خطهایی که در جهت‌های مختلف رسم از مرکز عبور می‌کنند دیده می‌شود ولی خطهای افقی و عمودی در اکثریت هستند. تاثیر مشابه‌ای هم در Phase Angle دیده می‌شود.

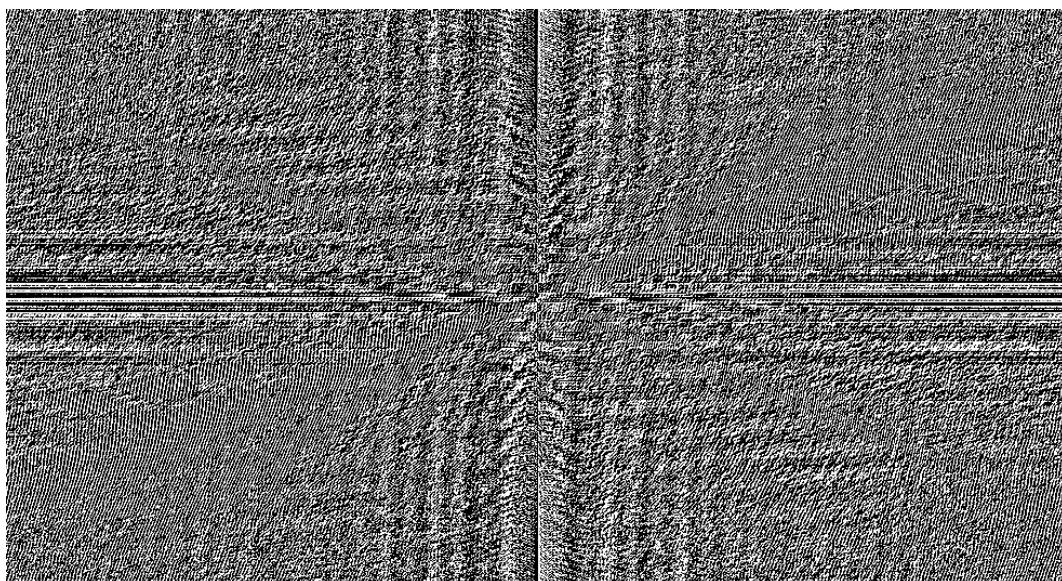
تصویر :trump_and_flag.jpg



: (p3b-trump_and_flag.jpg-spectrum.png) trump_and_flag.jpg تصویر Spectrum



: (p3b-trump_and_flag.jpg-phase.png) trump_and_flag.jpg تصویر Phase Angle

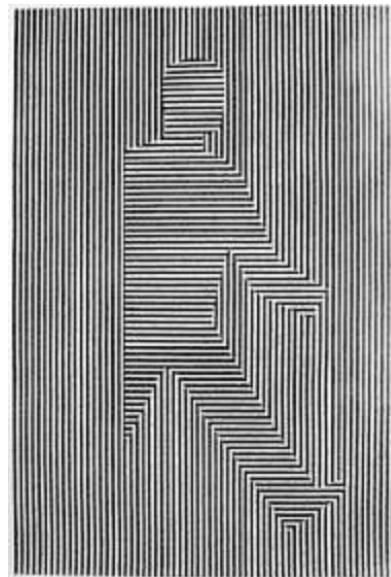


در تصویر ورودی پرچم آمریکا، خطها و مرزهایی افقی و عمودی و با زوایه‌های ۲۵، ۴۵ و ۷۵ درجه در قسمت سفید و قرمز و ستاره‌های سفید دیده می‌شود که به همین دلیل در تصویر ورودی خطهای سفیدی به صورت افقی، عمودی و به صورت قطری دیده می‌شود. تاثیر مشابه‌ای هم در Phase Angle دیده می‌شود.

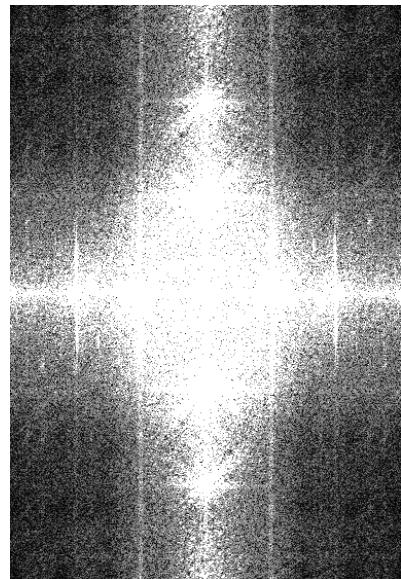
قسمت (C)

کد این قسمت در p3c.m قرار دارد. ابتدا تصویر خوانده می‌شود، سپس تصویر به صورت gray-level در می‌آید. سپس Fast Fourier Transform تصویر را حساب می‌کنیم، سپس مختصات صفر را به وسط شیفت می‌دهیم. در ادامه spectrum و angle phase را محاسبه می‌کنیم. از آنجایی که خط‌های عمودی را می‌خواهیم حذف کنیم باید نقطه‌های Peak افقی در تبدیل فوریه را حذف کنیم. به همین دلیل از فیلتر Ideal Notch Reject استفاده می‌کنیم. سپس فیلتر Horizontal Ideal Notch Reject ایجاد می‌شود، بعد فیلتر در تبدیل فوریه ضرب می‌شود، بعد صفر مختصات را به گوشه سمت چپ بالا منتقل می‌کنیم و در ادامه با Inverse Fast Fourier Transform تصویر فیلتر شده را به دست می‌آوریم. در زیر خروجی‌های مختلف کد را مشاهده می‌کنید(برای دیدن تصویرها در سایز اصلی، به پوشه‌ی P3 مراجعه کنید):

تصویر ورودی (parallel_man.jpg):

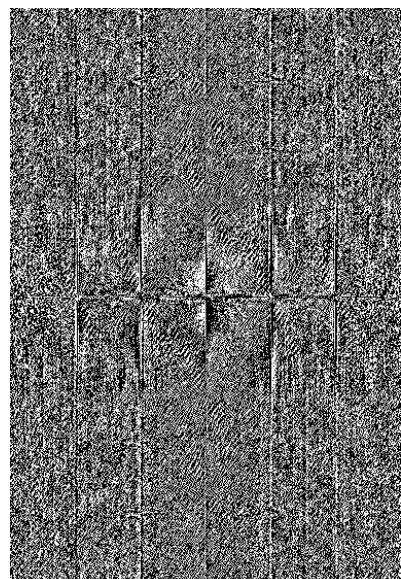


: (p3c_parallel_man_spectrum.png) Spectrum تصویر ورودی

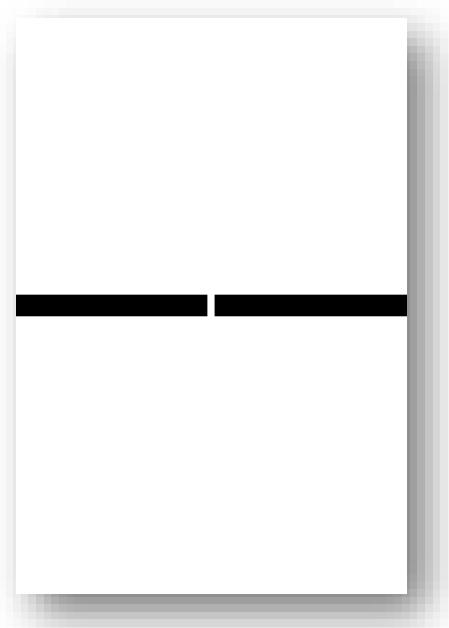


برای رسم Spectrum ابتدا Log10 گرفته شده است و سپس از Histeq استفاده شده است.

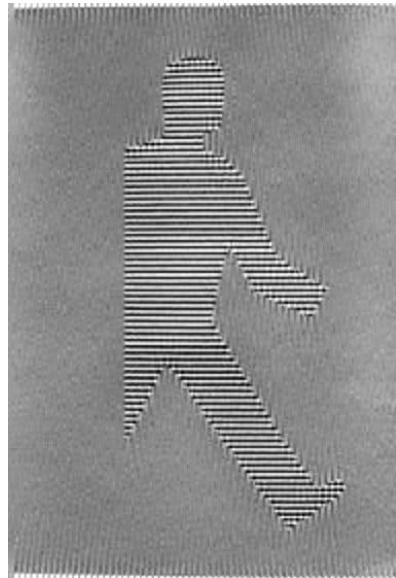
: (p3c_parallel_man_phase.png) تصویر ورودی Phase Angle

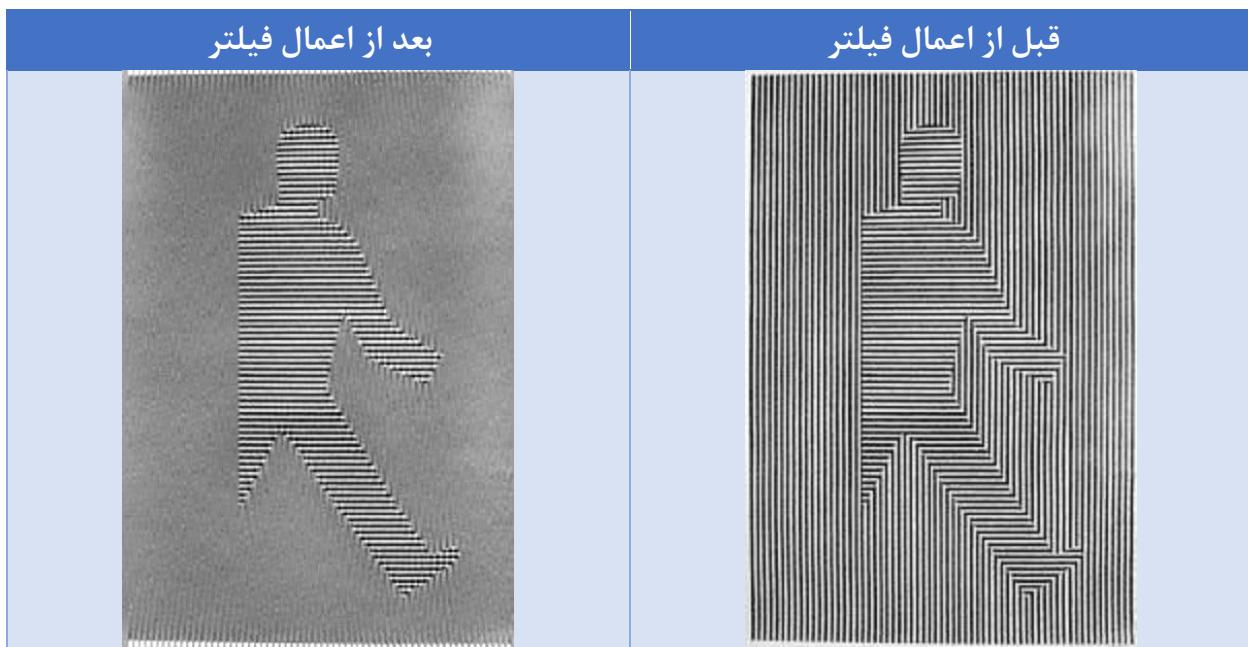


فیلتر مرکز ۳ با عرض ۱۰ و حاشیهی Band Reject (p3c_filter.png) پیکسل



حاصل اعمال فیلتر بر تصویر(p3c_filtered_image.png)





همین طور که مشاهده می‌شود خطهای عمودی بعد از اعمال فیلتر از میان رفته‌اند.

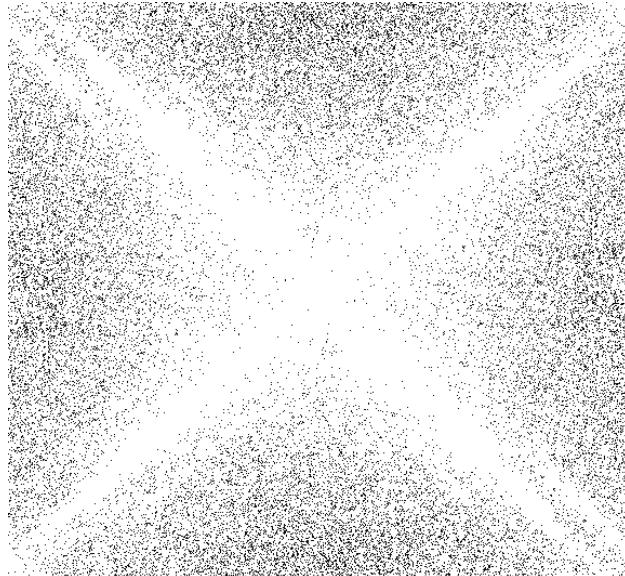
(D) قسمت

کد این قسمت در p3d.m قرار دارد. ابتدا تصویر خوانده می‌شود، سپس تصویر به صورت gray-level در می‌آید. سپس تصویر را حساب می‌کنیم، سپس مختصات صفر را به وسط شیفت می‌دهیم. در ادامه angle spectrum و spectrum را محاسبه می‌کنیم. از آنجایی که خطهای موازی قطر اصلی را می‌خواهیم حذف کنیم باید نقطه‌های خطهای موازی قطر اصلی در تبدیل فوریه را نگه داریم، در قسمت های قبل دیدیم که خطهای موازی یک قطر به در spectrum به صورت خطهای موازی قطر مخالف دیده می‌شوند. به همین دلیل از فیلتری استفاده می‌کنیم که با عرض مشخصی روی قطر اصلی برابر با یک است و سایر نقطه‌ها صفر است. سپس فیلتر ایجاد می‌شود، بعد فیلتر در تبدیل فوریه ضرب می‌شود، بعد صفر مختصات را به گوشه سمت چپ بالا منتقل می‌کنیم و در ادامه با Inverse Fast Fourier Transform تصویر فیلتر شده را به دست می‌آوریم. در زیر خروجی‌های مختلف کد را مشاهده می‌کنید(برای دیدن تصویرها در سایز اصلی، به پوششی P3 مراجعه کنید):

تصویر ورودی (phoenix.jpg):

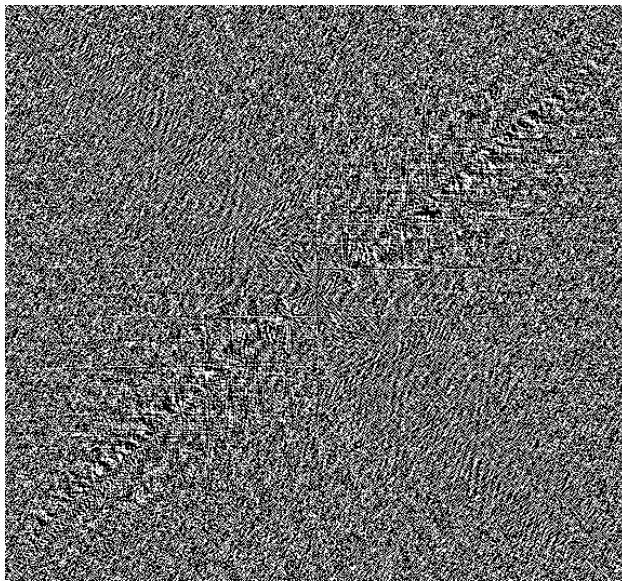


: (p3d_phoenix_spectrum.png) تصویر ورودی Spectrum



برای رسم Log10 ابتدا گرفته شده است و سپس از Histeq استفاده شده است.

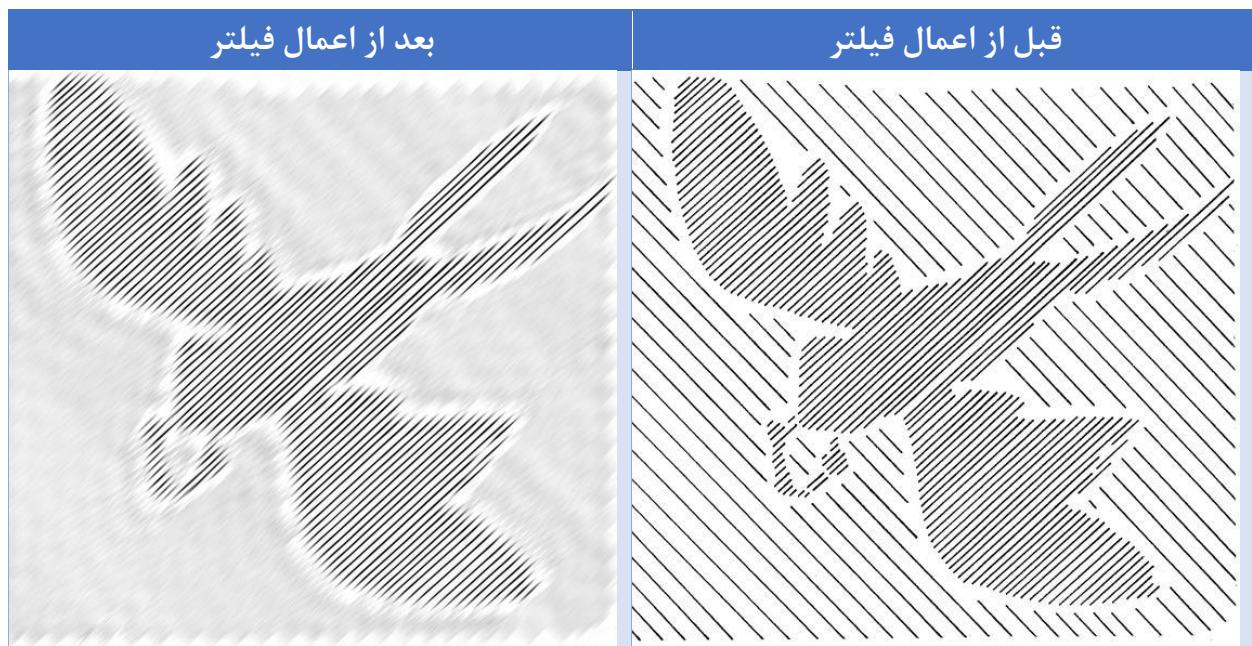
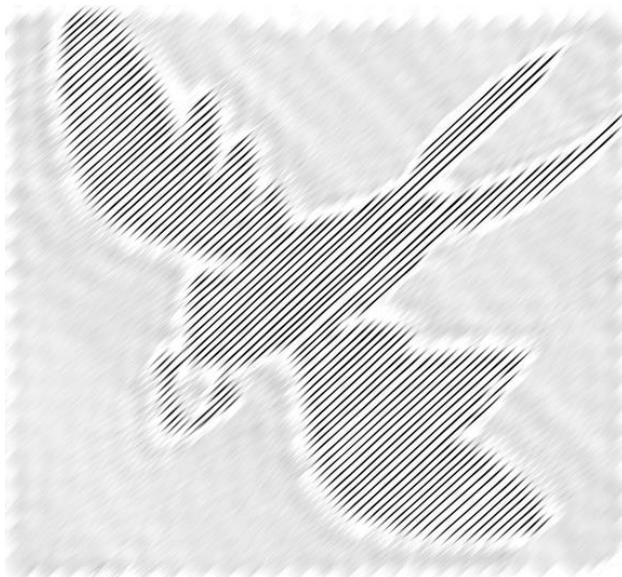
: (p3d_phoenix_phase.png) تصویر ورودی Phase Angle



فیلتر قطری با عرض ۵۰ : (p3d_filter.png)



حاصل اعمال فیلتر بر تصویر : (p3d_filtered_image.png)



همین طور که مشاهده می‌شود خطهای موازی قطر اصلی بعد از اعمال فیلتر از میان رفته‌اند، کمی پدیده‌ی Ringing دیده می‌شود که به دلیل Ideal بودن تصویر است.

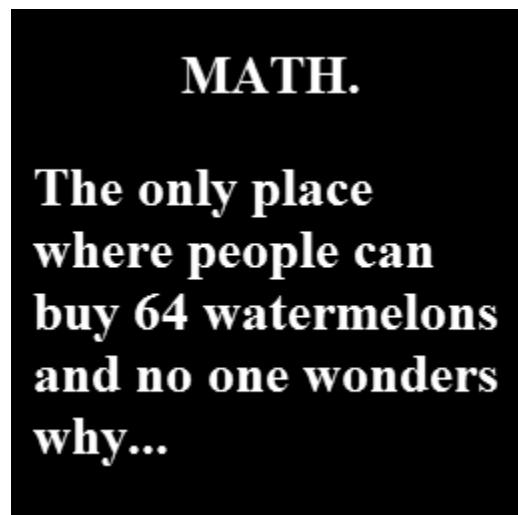
تمرین ۴

کدهای این قسمت در p4d.m قرار دارد.

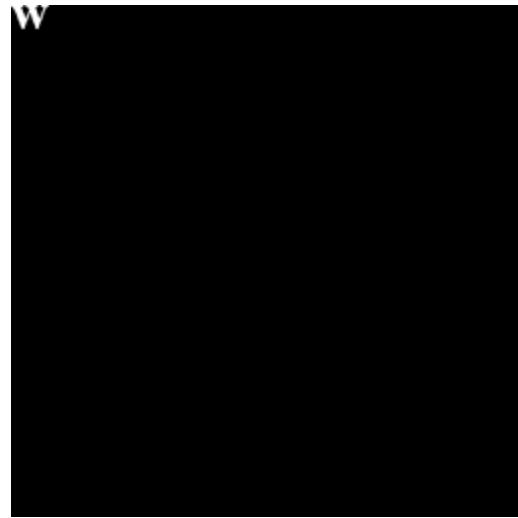
قسمت (A)

در ابتدا تصویر ورودی و تصویر حرف W را می‌خوانم، سپس فوریه ترانسفورم هر کدام را به دست می‌آورم. فوریه ترانسفورم‌ها را طوری محاسبه می‌شوند که مرکز آن‌ها وسط صفحه باشد و هر دو تصویر Pad شده باشند(در این مثال نیازی به Pad کردن نبود ولی برای اصولی کار کردن آن را انجام دادم)، سپس فوریه ترانسفورم تصویر و فوریه ترانسفورم حرف W را در هم ضرب نظیر به نظیر می‌کنم. در ادامه ضمن برگرداندن صفر مختصات از مرکز صفحه به سمت چپ بالا، تبدیل معکوس فوریه را حساب می‌کنم. Thresholding را بر تصویر حاصل اعمال می‌کنم و نقطه‌های پیک را محاسبه می‌کنم. سپس تصویر اصلی را به گونه‌ای تغییر می‌دهم که در اطراف نقطه‌های Peak مقدارشان حفظ می‌شود و در سایر نقطه‌ها مقدارشان صفر می‌شود. در زیر خروجی‌های کد را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به p4 مراجعه فرمایید):

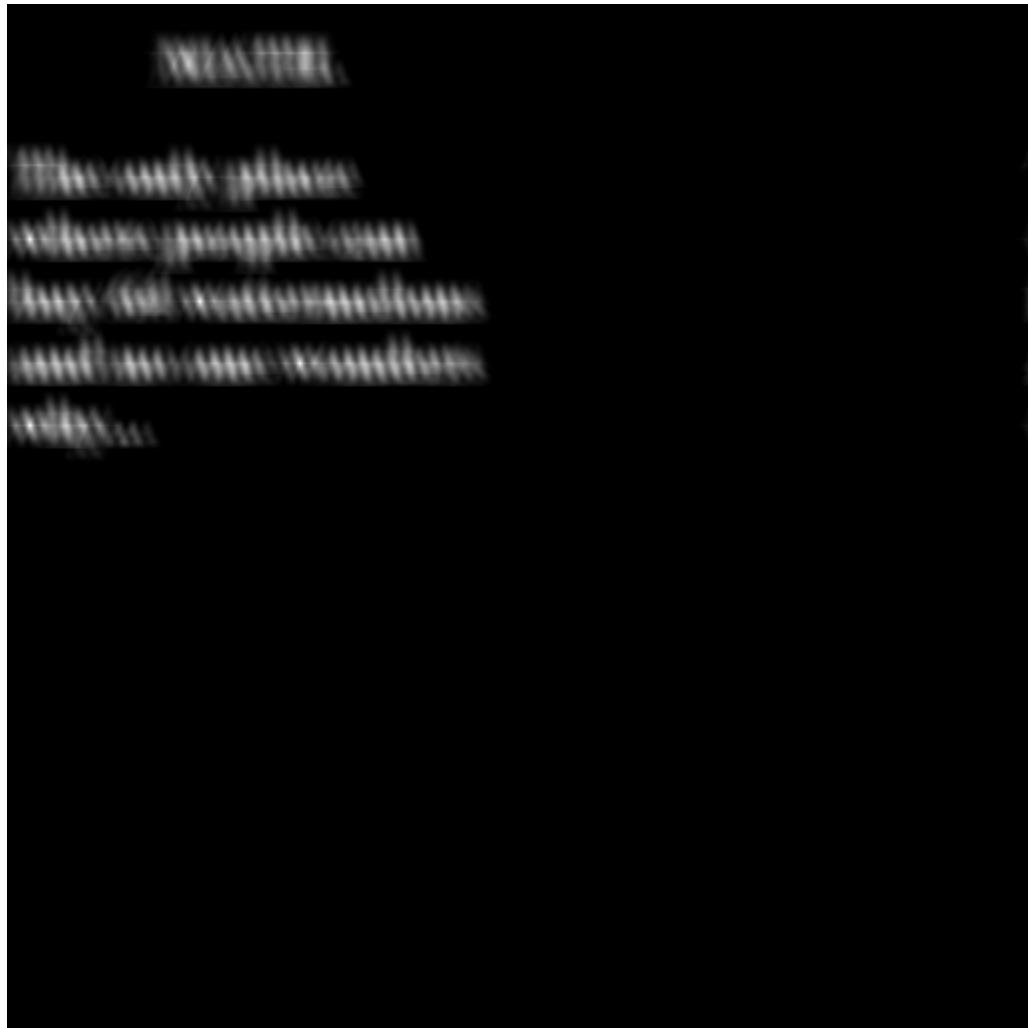
تصویر ورودی:



تصویر حرف W:

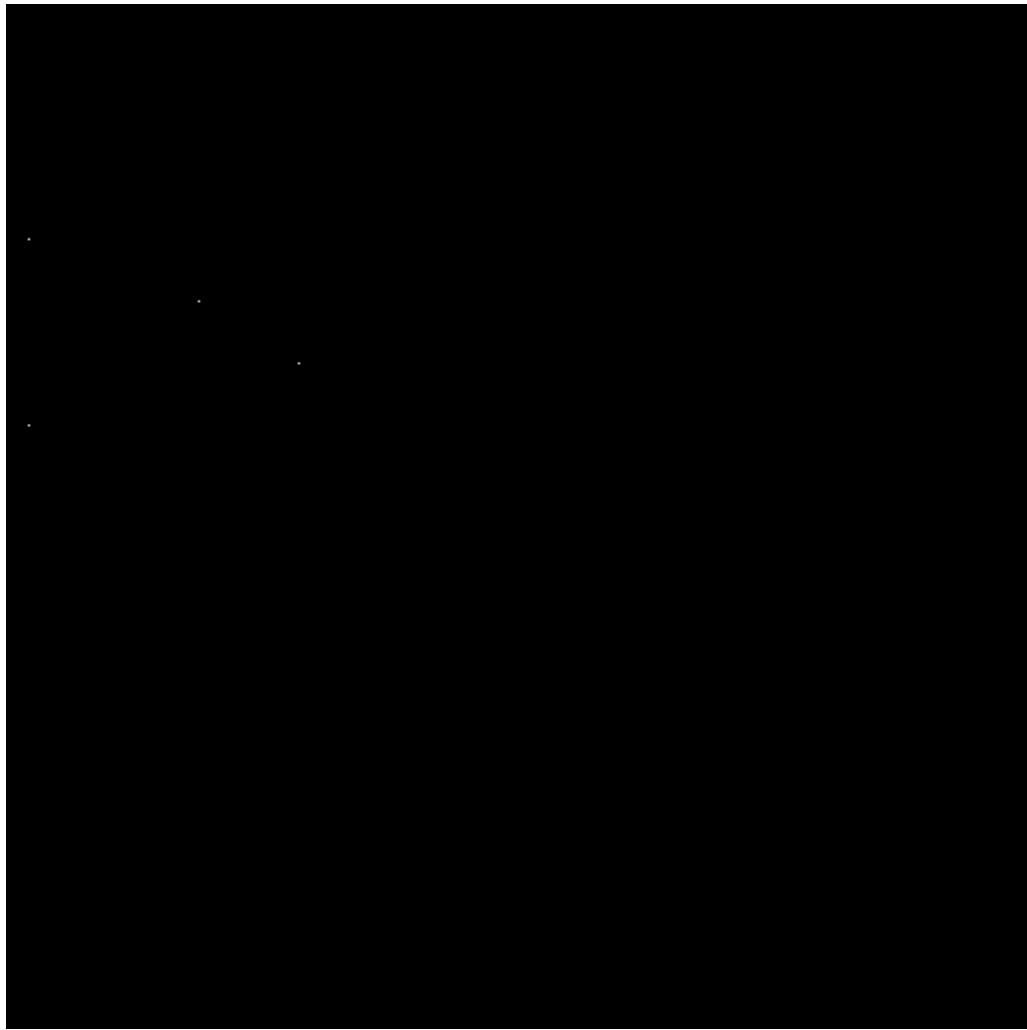


تصویر بعد از ضرب در حوزه‌ی فوریه(p4a_filtered_image_before_threshold.png)



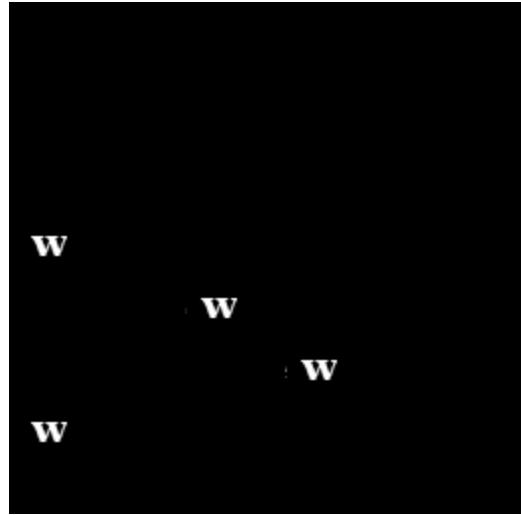
*بزرگی تصویر به خاطر Padding است.

: (p4a_filtered_image_thresholded.png) Peak نقطه‌های



در تصویر بالا **۴ نقطه‌ی Peak** با **رنگ سفید** قرار دارد. این تصویر با Thresholding از تصویر حاصل از تبدیل معکوس فوریه به دست می‌آید.

خروجی نهایی: همسایگی نقطه‌های Peak در تصویر اصلی ورودی (p4a_highest_value.png)



همین طور که دیده می‌شود ۴ حرف W به درستی تشخیص داده شده اند و حرفی غیر از W به اشتباه تشخیص داده نشده است.

قسمت (B)

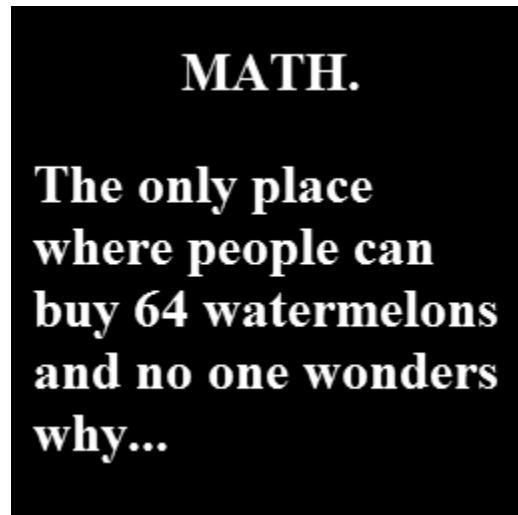
در ابتدا تصویر ورودی و تصویر حرف W را می‌خوانم، سپس فوريه ترنسفورم هر کدام را به دست می‌آورم. فوريه ترنسفورم‌ها را طوری محاسبه می‌شوند که مرکز آن‌ها وسط صفحه باشد و هر دو تصویر Pad شده باشند(در اين مثال نيازی به Pad کردن نبود ولی برای اصولی کار کردن آن را انجام دادم)، سپس فوريه ترنسفورم فیلتر را به عمل Thresholding می‌کنم. بر روی Phase Angle و Spectrum را انجام می‌دهم، و بعد فوريه ترنسفورم را بر اساس فرمول زیر، بر اساس Phase Angle و Spectrum می‌سازم:

$$G(k) = |G(k)| e^{i\theta_k}$$

از آنجایی که

سپس فوريه ترنسفورم تصویر و conjugate فوريه ترنسفورم حرف W را در هم ضرب نظير به نظير می‌کنم. در ادامه ضمن برگرداندن صفر مختصات از مرکز صفحه به سمت چپ بالا، تبدیل معکوس فوريه را حساب می‌کنم. نقاط پیک را پیدا می‌کنم و سپس تصویر اصلی را به گونه‌ای تغيير می‌دهم که در اطراف نقطه‌های Peak مقدارشان حفظ می‌شود و در سایر نقطه‌ها مقدارشان صفر می‌شود. در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به p4 مراجعه فرمایید):

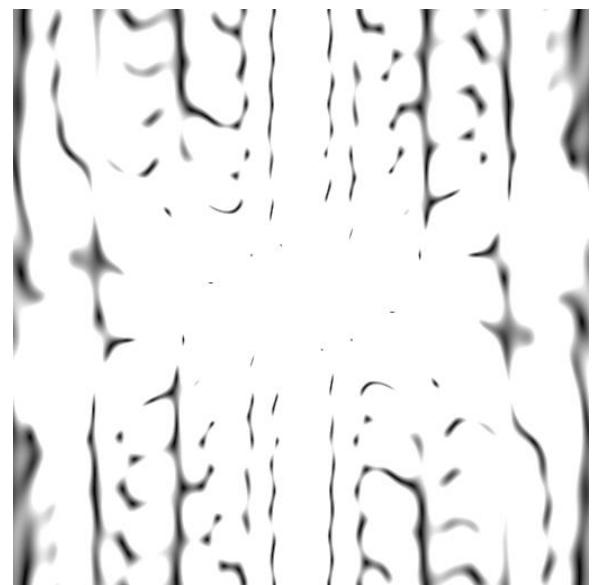
تصویر ورودی:



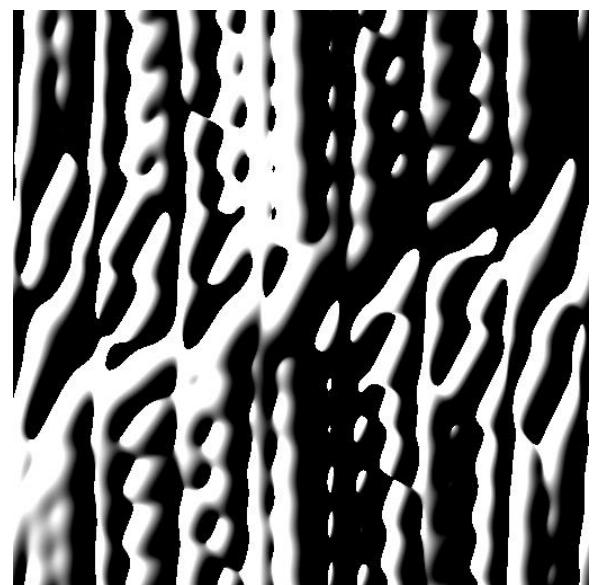
تصویر حرف W:



تصویر Spectrum Thresholding فیلتر پیش از :(p4b_spectrum_of_filter_before_threshold.png)



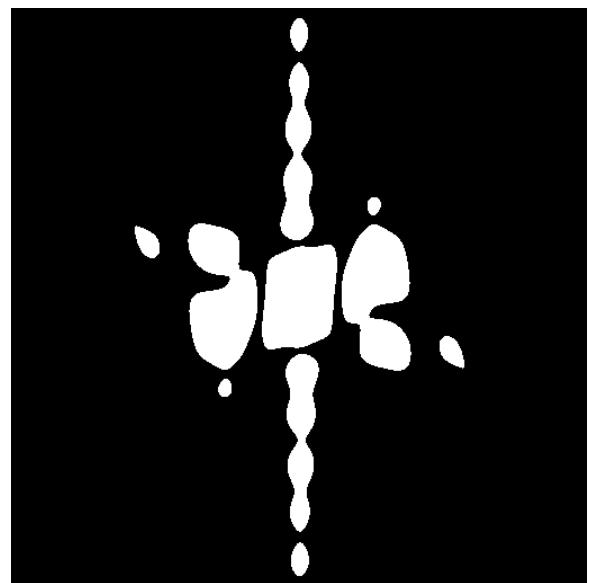
تصویر فیلتر پیش از Thresholding Phase Angle
(p4b_phase_of_filter_before_threshold.png)



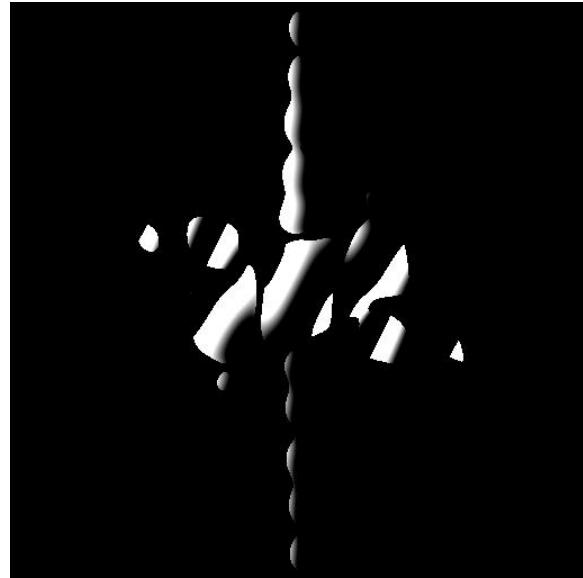
تصویر تبدیل فوریه پیش از Thresholding (p4b_fourier_of_filter.png)



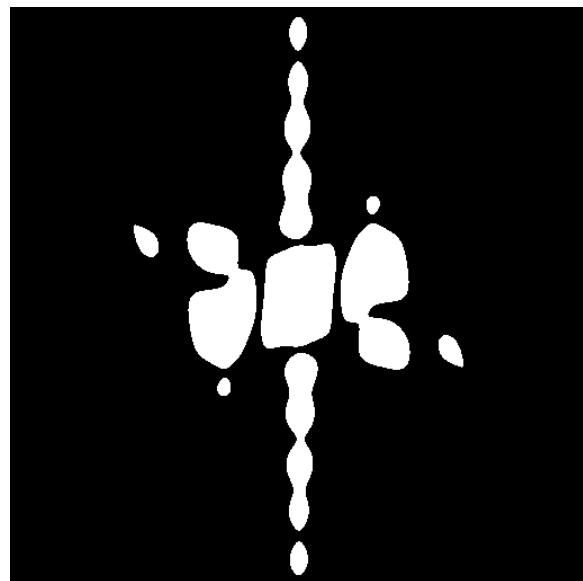
تصویر فیلتر بعد از Thresholding Spectrum
(p4b_spectrum_of_filter_after_threshold.png)



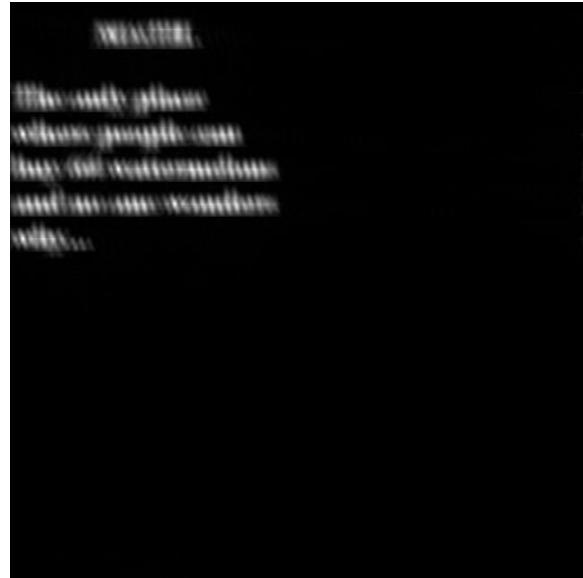
تصویر فیلتر بعد از Thresholding Phase Angle
(p4b_phase_of_filter_before_threshold.png)



تصویر تبدیل فوریه بعد از Thresholding : (p4b_filtered_image.png)

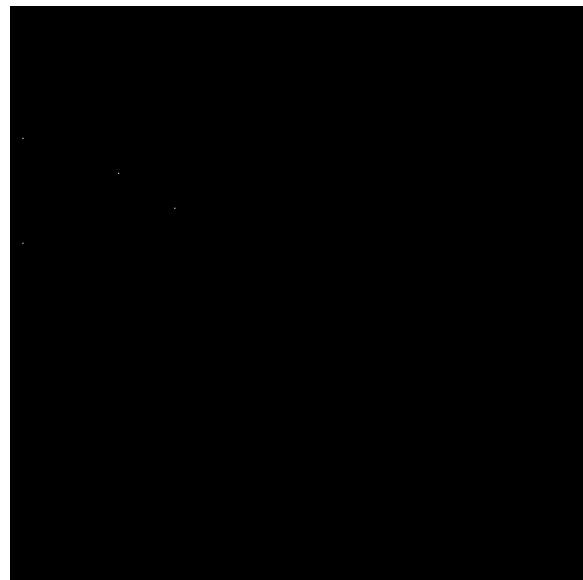


تصویر بعد از ضرب در حوزه‌ی فوریه (p4a_filtered_image_before_threshold.png)



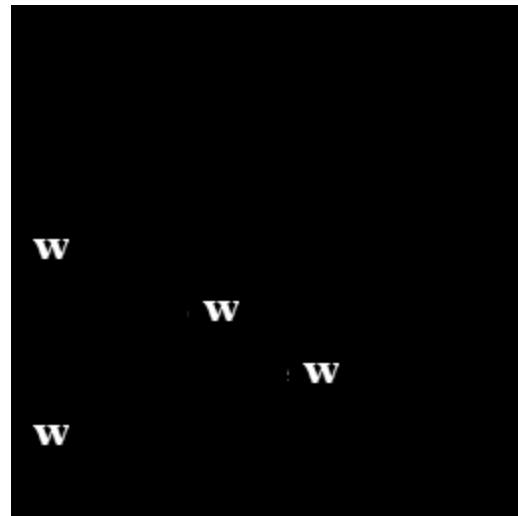
*بزرگی تصویر به خاطر Padding است. (برای دیدن سایز اصلی به P4 مراجعه نمایید)

: (p4a_filtered_image_thresholded.png) Peak نقطه‌های

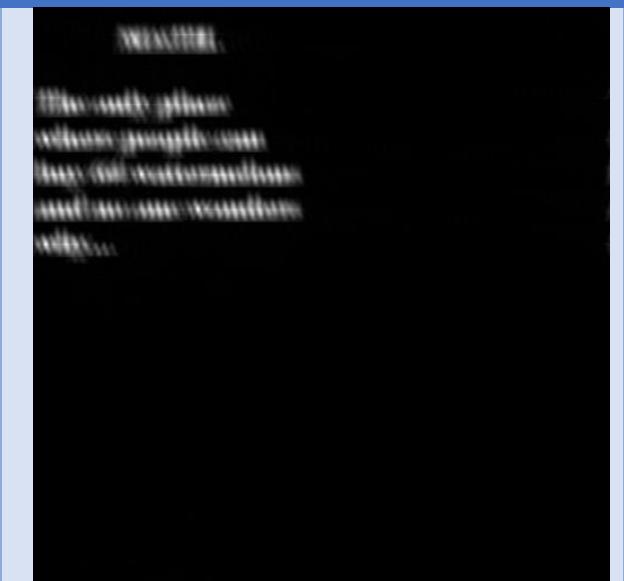
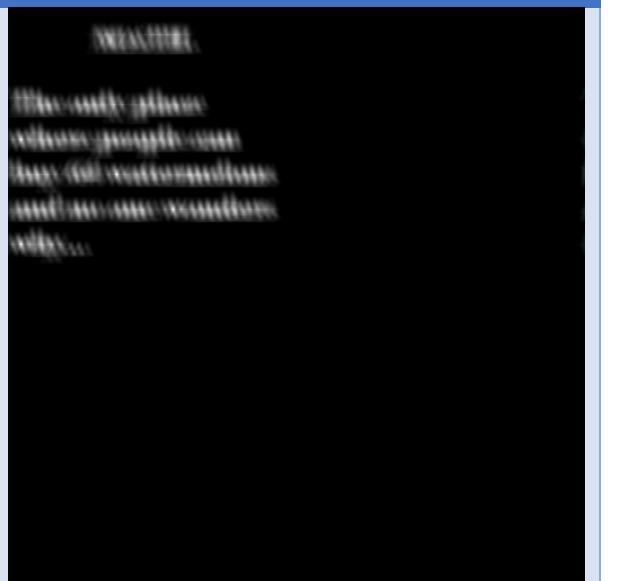


در تصویر بالا ۴ نقطه‌ی Peak با رنگ سفید قرار دارد.

خروجی نهایی: همسایگی نقطه‌های Peak در تصویر اصلی ورودی (p4b_highest_value.png)



همین طور که دیده می‌شود ۴ حرف W به درستی تشخیص داده شده اند و حرفی غیر از W به اشتباه تشخیص داده نشده است.

حاصل اعمال فیلتر در حوزه فرکانس، از فیلتر در حوزه‌ی فرکانس Threshold گرفته شده است	حاصل اعمال فیلتر در حوزه فرکانس، از فیلتر در حوزه‌ی فرکانس Threshold گرفته شده نشده است
	

همین طور که در دو تصویر بالا دیده می‌شود، تصویر حاصل از اعمال فیلتر توسط فیلتری که در حوزه‌ی فرکانس از آن **Threshold** گرفته شده است شارپ‌تر است و در نقطه‌های غیر صفر از شدت روشنایی بیشتری برخوردار است.

تمرین ۵

کدهای این قسمت در p5 قرار دارد.

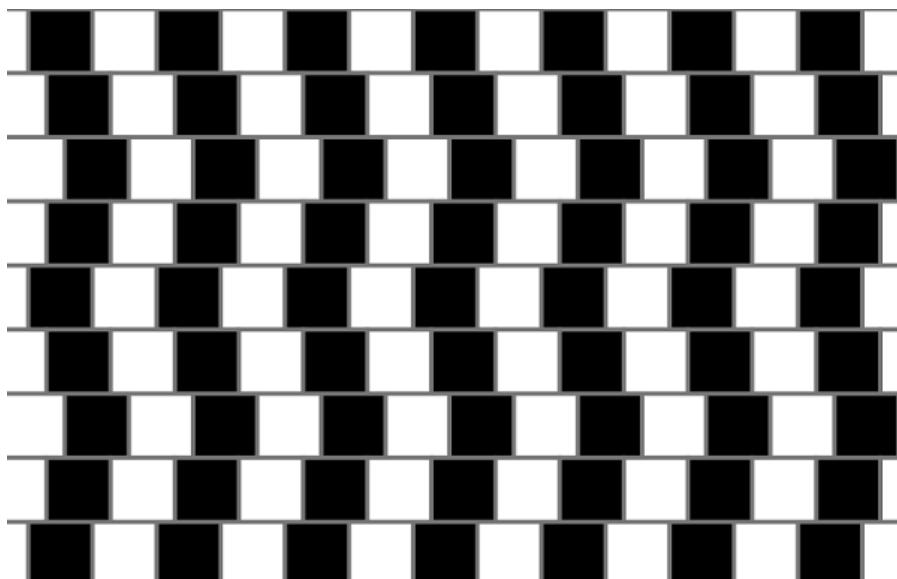
قسمت (a)

زیر قسمت (a1)

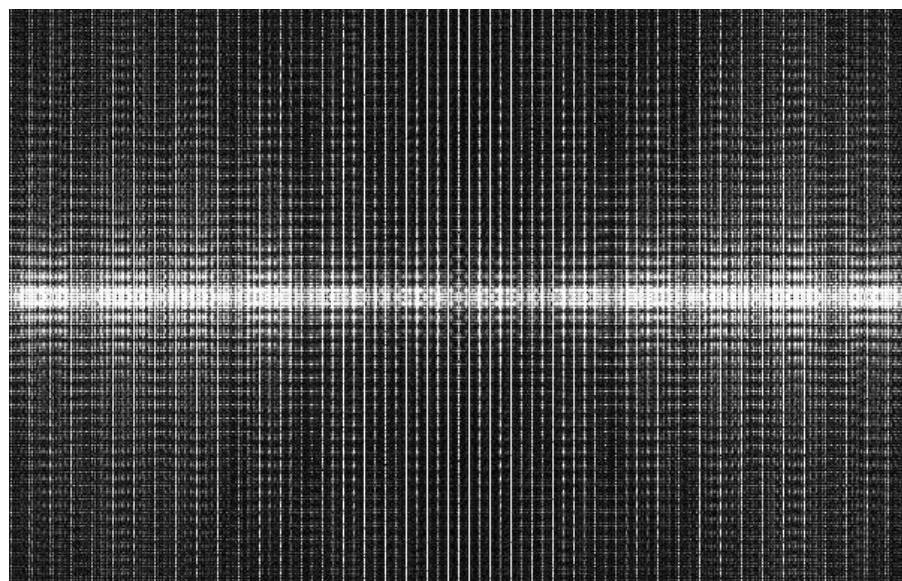
کد این قسمت در m_p5a_a1.m قرار دارد. در ابتدا تصویر را خوانده‌ام، سپس مقدار آن را به صورت double درآورده‌ام و در ادامه تبدیل فوریه‌ی تصویر را محاسبه کرده‌ام. مرکز مختصات به وسط صفحه منتقل می‌شود. در ادامه Phase Angle و Spectrum محاسبه می‌شود.

بعد از محاسبه‌ی Spectrum تصویر، تصویر را Reflect می‌کنم و در ادامه مانند آنچه در بالا گفته شده را محاسبه می‌کنم. در خروجی‌های کد را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به P5 مراجعه کنید):

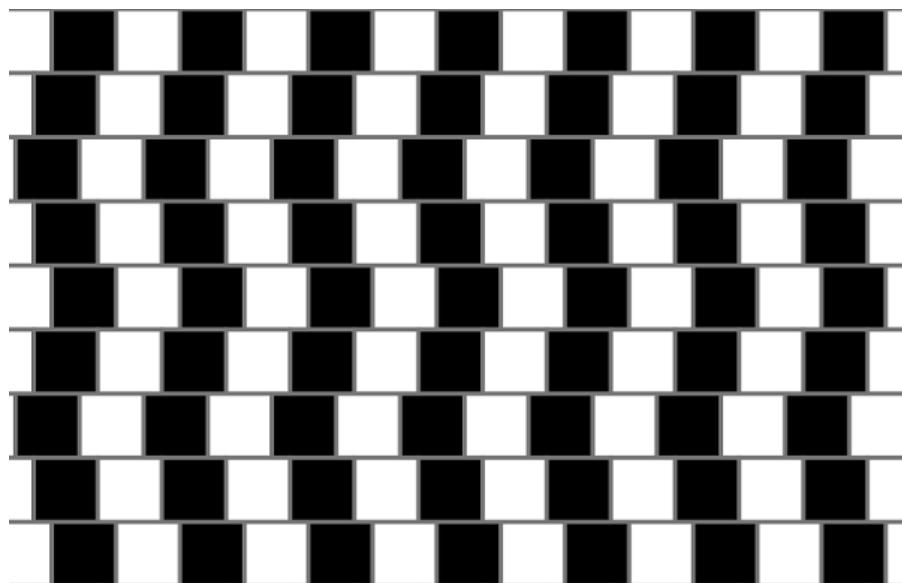
تصویر ورودی:



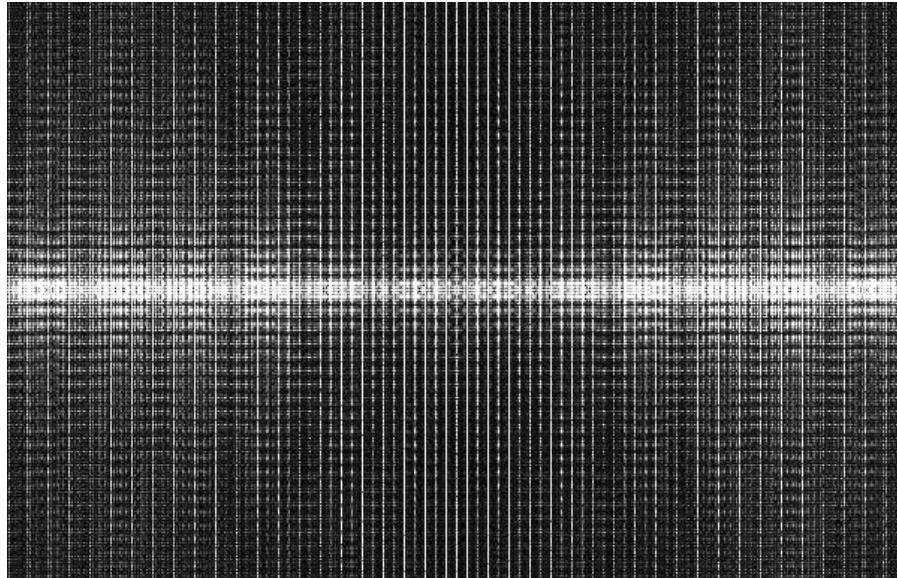
: (a-a1-spectrum.png) تصویر ورودی Spectrum



تصویر بعد از (a-a1-reflected.png) Reflection



تصویر بعد از (a-a1-spectrum-reflected.png) Spectrum Reflection



تصویر را نسبت به محور عمودی Reflect کردیم، همین طور که دیده می‌شود Spectrum نسبت به محور افقی Reflect شده است.

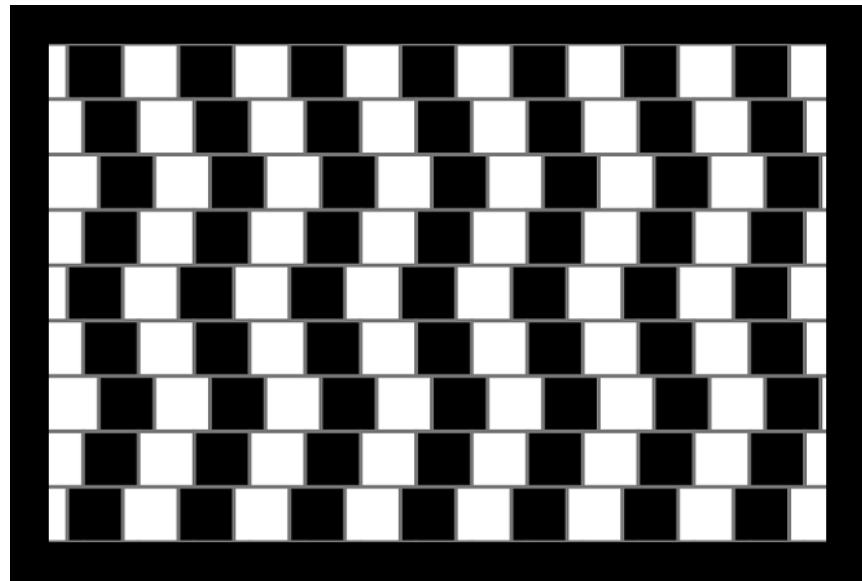
(زیر قسمت 2)

کد این قسمت در p5a_a2.m قرار دارد. در ابتدا تصویر را خوانده‌ام، سپس مقدار آن را به صورت double درآورده‌ام و در ادامه تبدیل فوریه‌ی تصویر را محاسبه کرده‌ام. مرکز مختصات به وسط صفحه منتقل می‌شود. در ادامه Phase Angle و Spectrum محاسبه می‌شود.

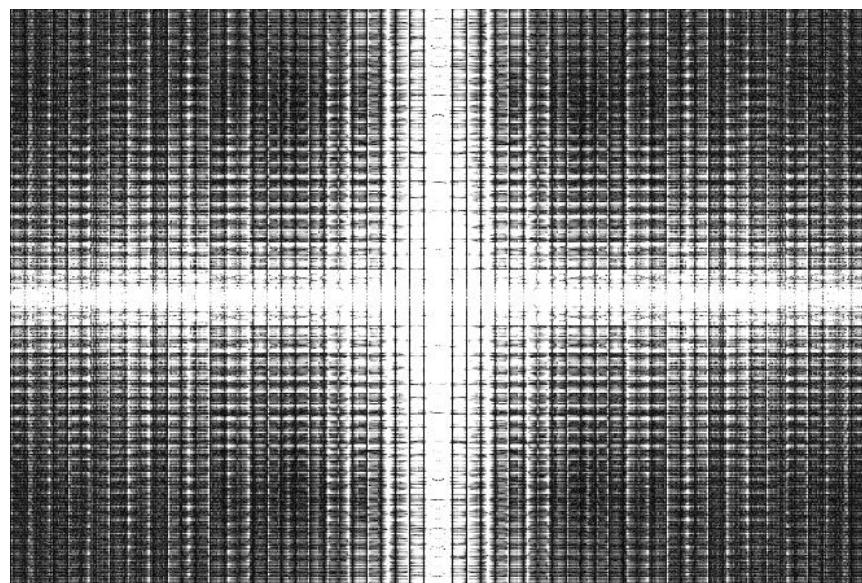
*برای اینکه نتیجه‌گیری تحت عوامل اضافه‌ای مانند قطع شدن بخشی از تصویر و اضافه شدن قسمتی سیاه رنگ بر اثر Translation قرار نگیرد، تصویر را به اندازه‌ی ۳۰ پیکسل با مقدار صفر Pad کرده‌ام.

بعد از محاسبه‌ی Spectrum تصویر، تصویر را به میزان ۱۰ پیکسل به سمت راست منتقل می‌کنم و در ادامه مانند آنچه در بالا گفته شده Spectrum را محاسبه می‌کنم. در خروجی‌های کد را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به P5 مراجعه کنید):

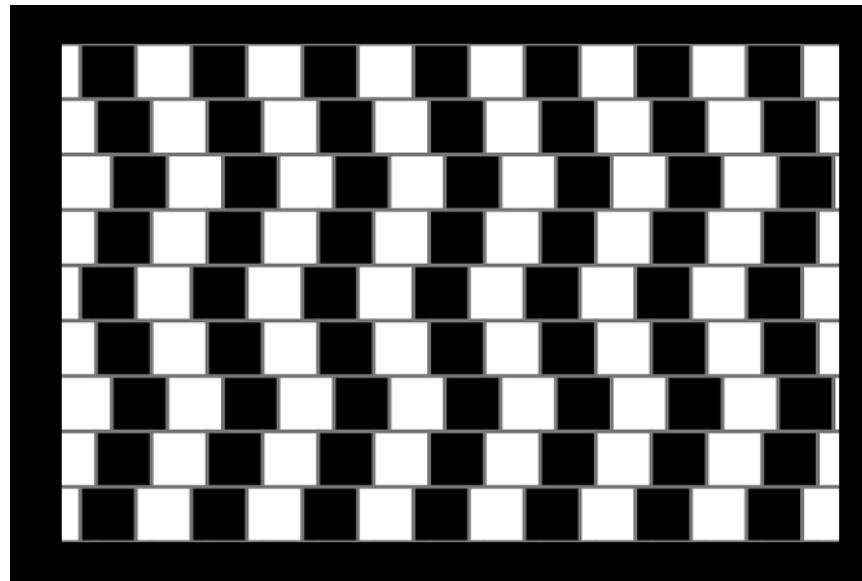
تصویر ورودی (a-a2-input.png)



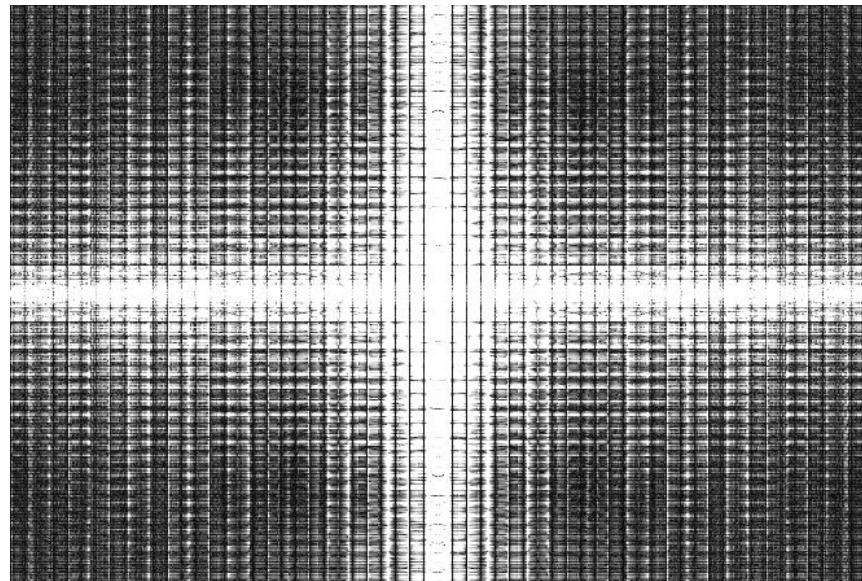
تصویر ورودی (Spectrum) : (a-a2-spectrum.png)



تصویر بعد از Translation : (a-a2-translation.png)



: (a-a2-spectrum-translation.png) Translation Spectrum تصویر بعد از



همین طور که در کتاب توضیح داده است و در اینجا مشاهده می‌شود، Spectrum نسبت به انتقال تاثیرپذیر نیست و با انتقال یک تصویر تغییری در spectrum رخ نمی‌دهد.

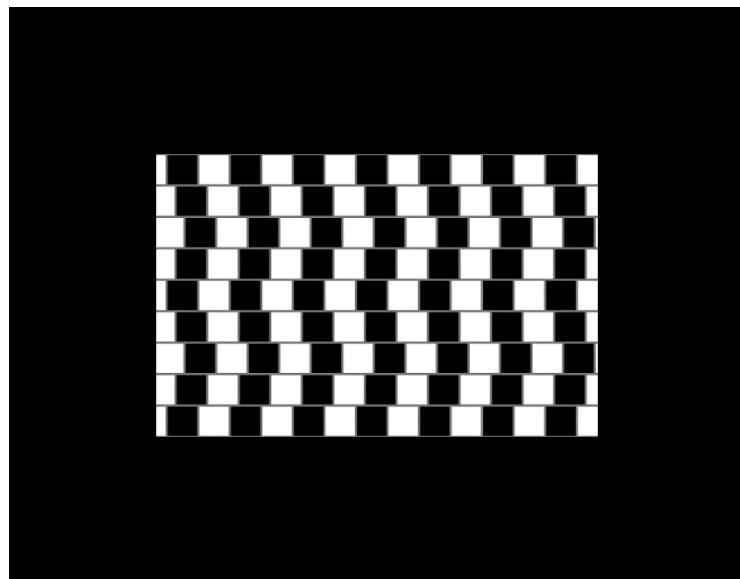
زیر قسمت (a3)

کد این قسمت در p5a_3.m قرار دارد. در ابتدا تصویر را خوانده‌ام، سپس مقدار آن را به صورت double درآورده‌ام و در ادامه تبدیل فوریه‌ی تصویر را محاسبه کرده‌ام. مرکز مختصات به وسط صفحه منتقل می‌شود. در ادامه Phase Angle و Spectrum محاسبه می‌شود.

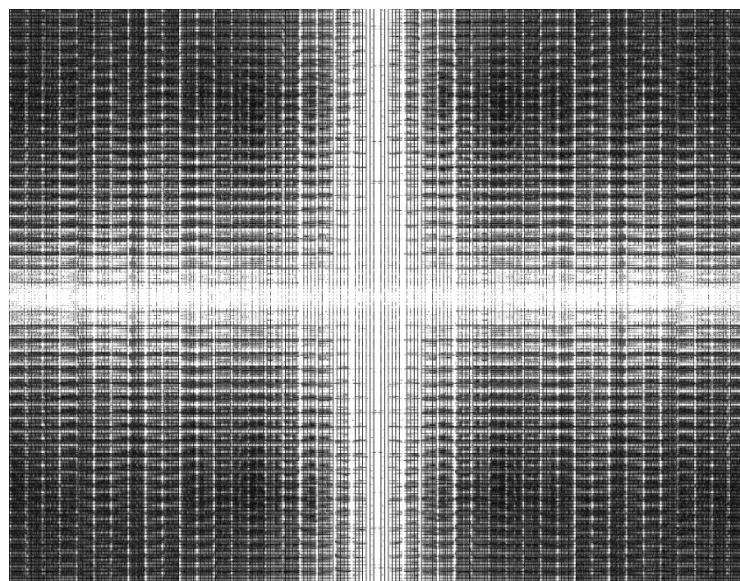
*برای اینکه نتیجه‌گیری تحت عوامل اضافه‌ای مانند قطع شدن بخشی از تصویر و اضافه شدن قسمتی سیاه رنگ بر اثر Rotation قرار نگیرد، تصویر را با مقدار صفر Pad کرده‌ام.

بعد از محاسبه‌ی Spectrum تصویر، تصویر را به میزان ۴۵، ۹۰ و ۱۸۰ درجه چرخش می‌دهم و در ادامه مانند آنچه در بالا گفته شده Spectrum را محاسبه می‌کنم. در خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به P5 مراجعه کنید):

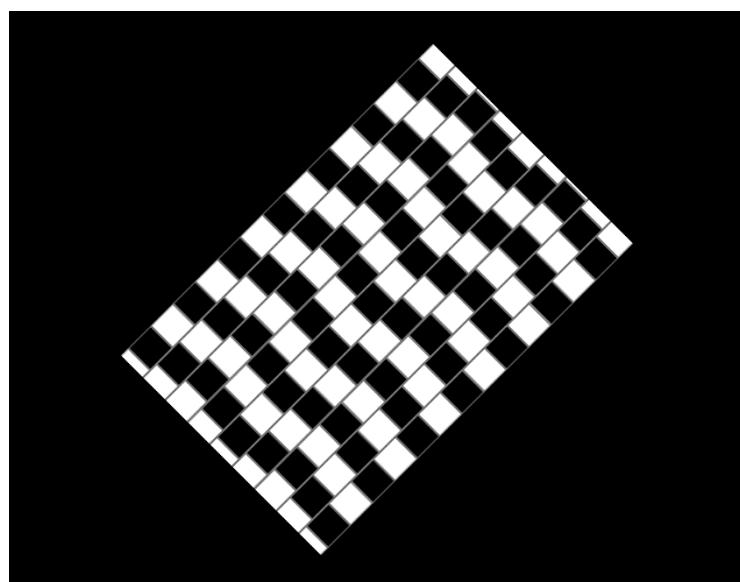
تصویر ورودی (a-a3-input.png)



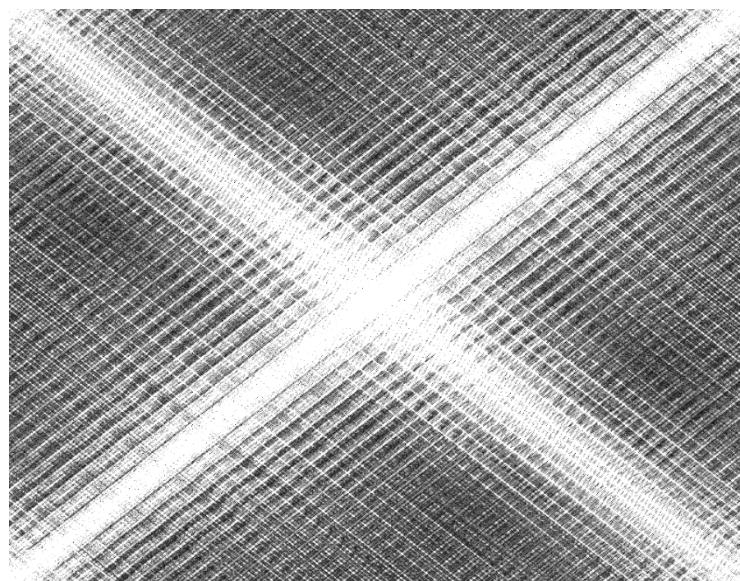
تصویر ورودی Spectrum (a-a3-spectrum.png)



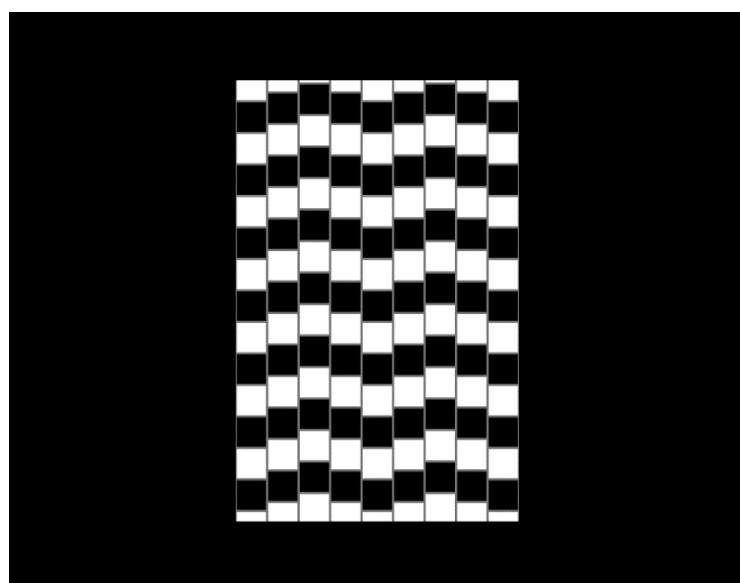
تصویر بعد از Rotation به اندازه‌ی 45 درجه : (a-a3-rotate-45.png)



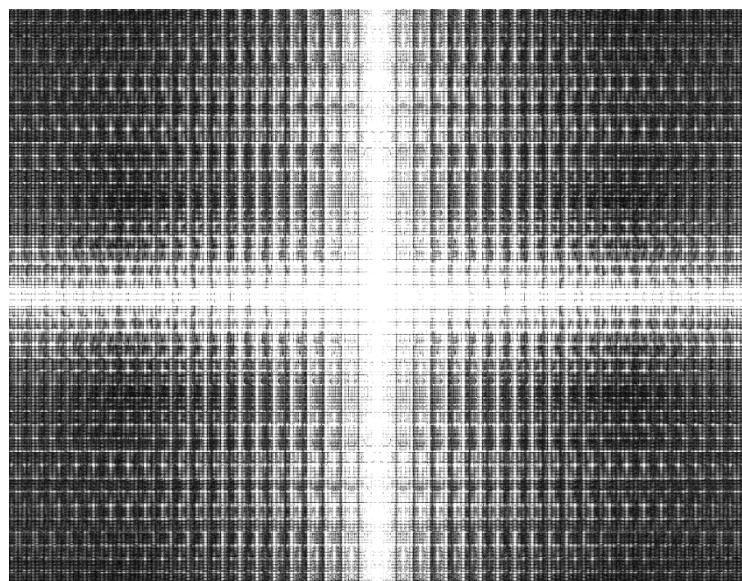
Spectrum تصویر بعد از Rotation به اندازه‌ی 45 درجه : (a-a3-spectrum-rotate-45.png)



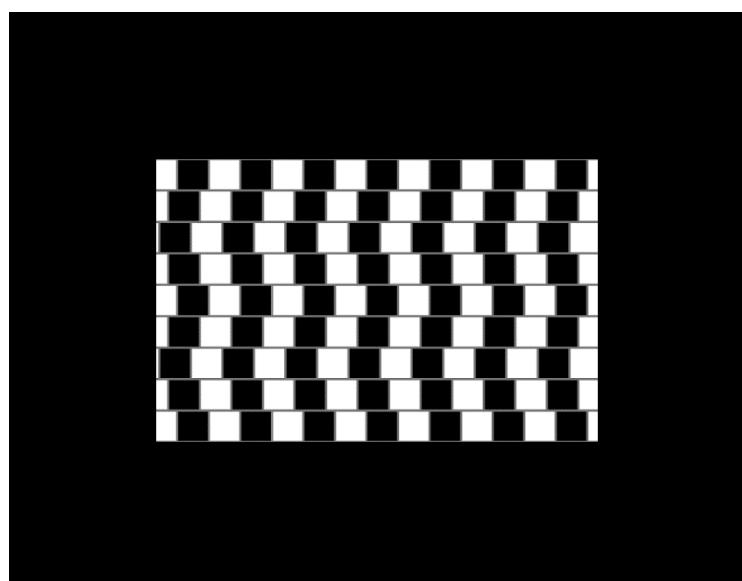
تصویر بعد از Rotation به اندازه‌ی 90 درجه (a-a3-rotate-90.png)



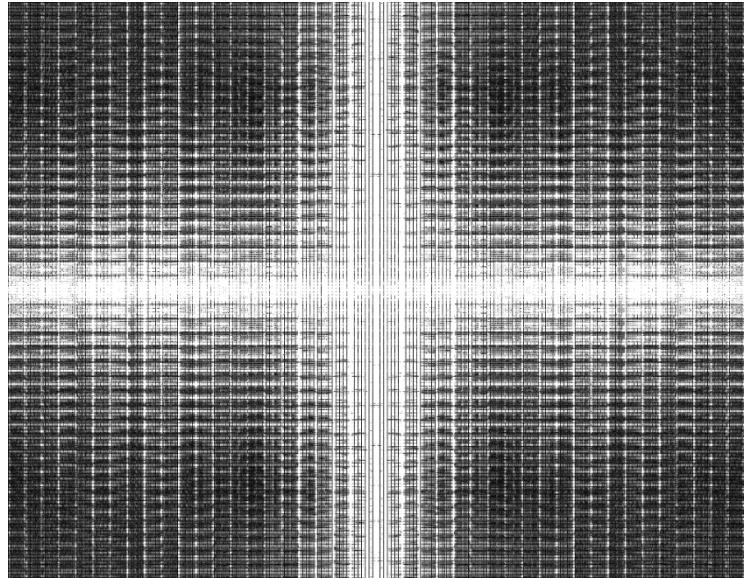
تصویر بعد از Rotation به اندازه‌ی 90 درجه (a-a3-spectrum-rotate-90.png) Spectrum



تصویر بعد از Rotation به اندازه‌ی 180 درجه : (a-a3-rotate-180.png)



Spectrum تصویر بعد از Rotation به اندازه‌ی 180 درجه : (a-a3-spectrum-rotate-180.png)



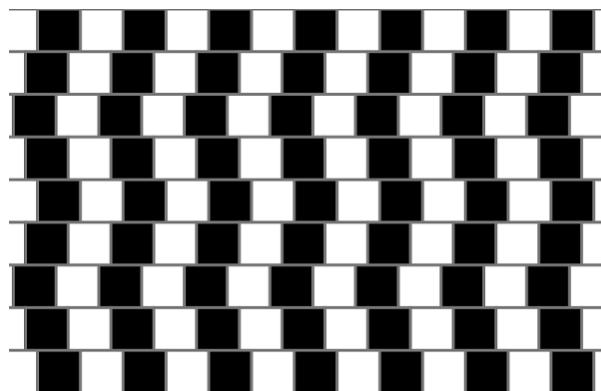
نتیجه‌گیری: همین طور که در کتاب توضیح داده است و در نتیجه‌ها مشاهده می‌شود، تصویر را با هر زاویه‌ای چرخش دهیم، Spectrum نیز با همان میزان چرخش پیدا می‌کند.

زیر قسمت (a4)

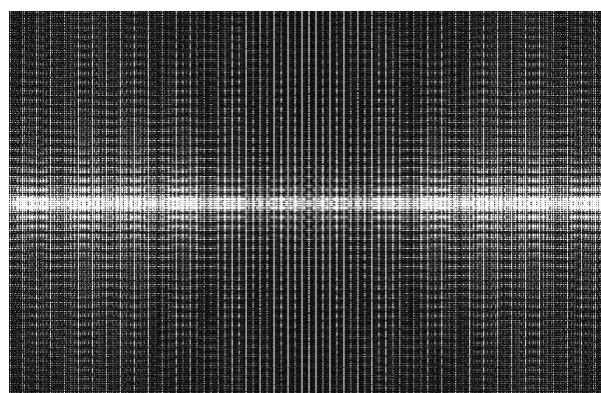
کد این قسمت در p5a_4.m قرار دارد. در ابتدا تصویر را خوانده‌ام، سپس مقدار آن را به صورت double درآورده‌ام و در ادامه تبدیل فوريه‌ی تصویر را محاسبه کرده‌ام. مرکز مختصات به وسط صفحه منتقل می‌شود. در ادامه Phase Angle و Spectrum محاسبه می‌شود.

بعد از محاسبه‌ی Spectrum تصویر، تصویر را به میزان $0, 5$ و 2 Scale می‌کنم و در ادامه مانند آنچه در بالا گفته شده Spectrum را محاسبه می‌کنم. در خروجی‌های کد را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به P5 مراجعه کنید):

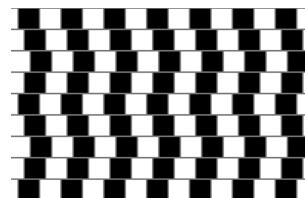
تصویر ورودی:



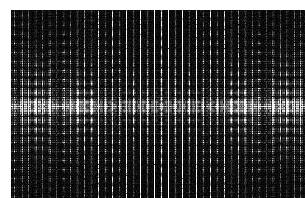
: (a-a3-spectrum.png) تصویر ورودی Spectrum



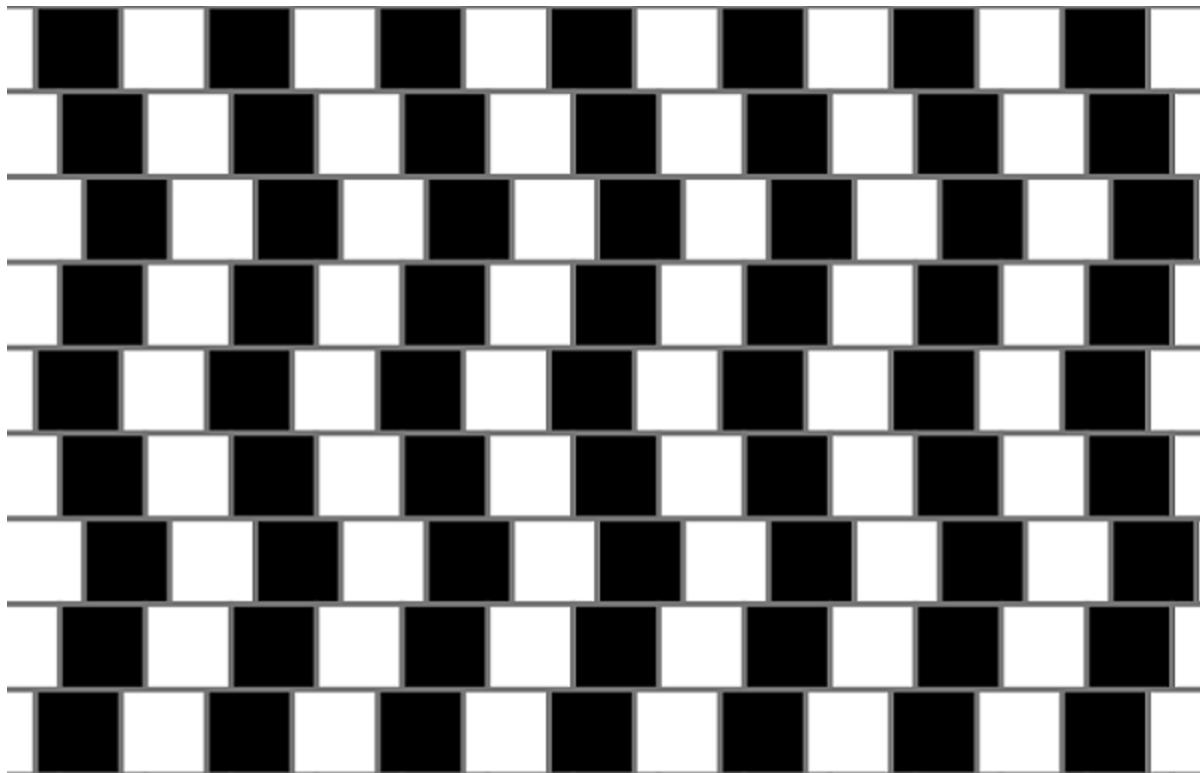
: (a-a4-scaled-0.5.png) تصویر بعد از scale به اندازه‌ی 0.5



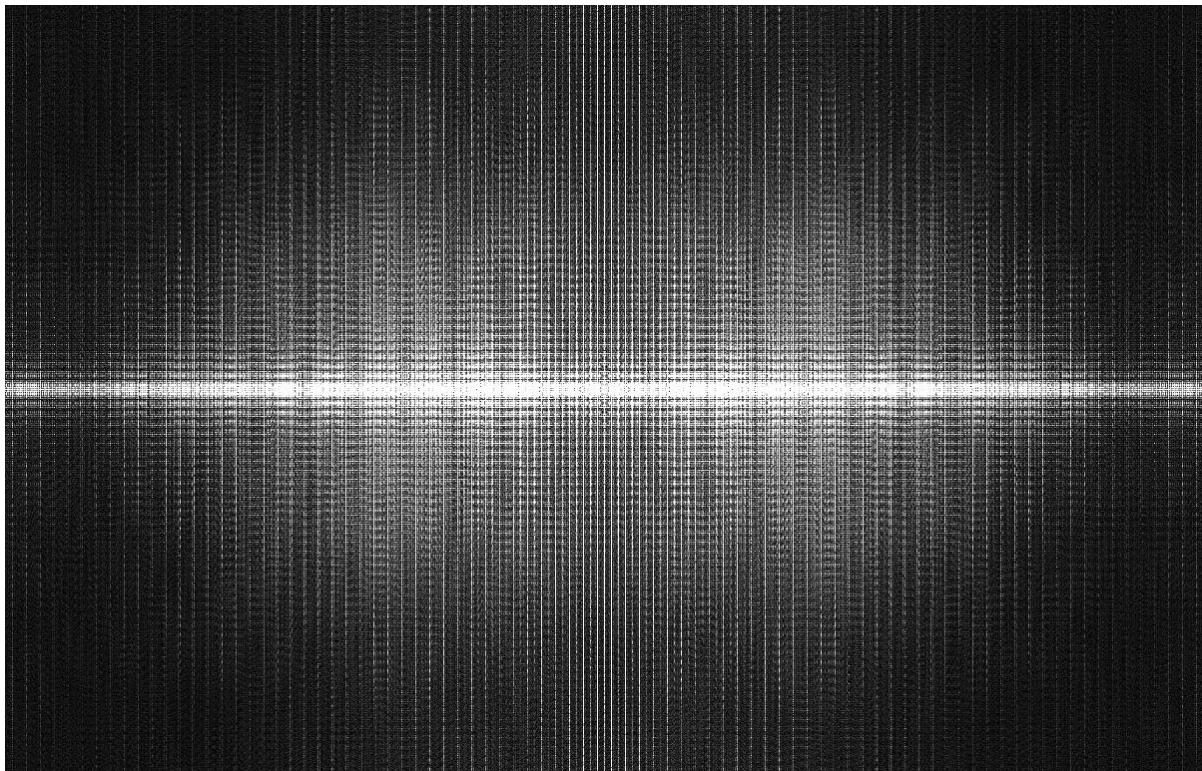
: (a-a4-spectrum-scaled-0.5.png) تصویر بعد از scale به اندازه‌ی 0.5 Spectrum



تصویر بعد از scale به اندازه‌ی 2 : (a-a4-scaled-2.png)



تصویر بعد از scale به اندازه‌ی 2 : (a-a4-spectrum-scaled-2.png) Spectrum



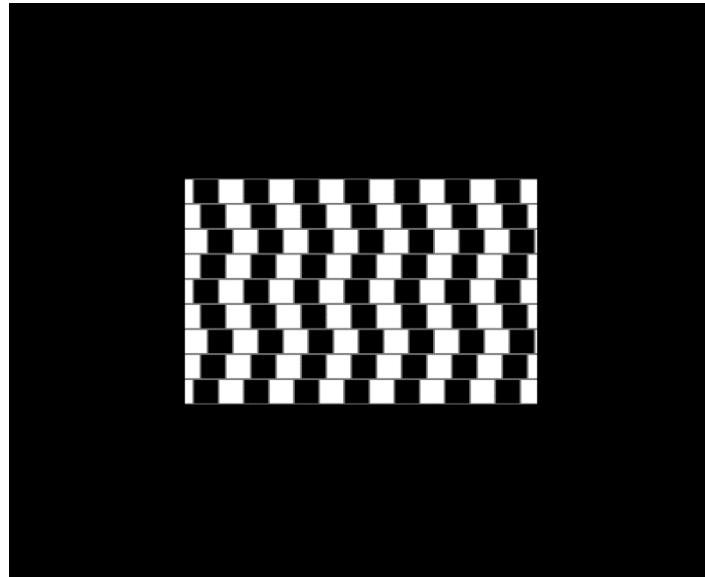
نتیجه‌گیری: از نظر اندازه طول و عرض، به اندازه‌ای که تصویر اسکیل شده است، تبدیل فوریه‌ی آن نیز تغییر کرده است. هر اندازه‌ای تصویر `scale` شده است با معکوس آن اندازه تبدیل فوریه تغییر پیدا کرده است. همان طور که مشاهده می‌شود در تصویر با میزان `scale` به اندازه‌ی 0.5 فاصله‌ی بین خطها ۲ برابر شده است.

زیر قسمت (a5) کد این قسمت در `p5a_a5.m` قرار دارد. در ابتدا تصویر را خوانده‌ام، سپس مقدار آن را به صورت `double` درآورده‌ام و در ادامه تبدیل فوریه‌ی تصویر را محاسبه کرده‌ام. مرکز مختصات به وسط صفحه منتقل می‌شود. در ادامه `Phase Angle` و `Spectrum` محاسبه می‌شود.

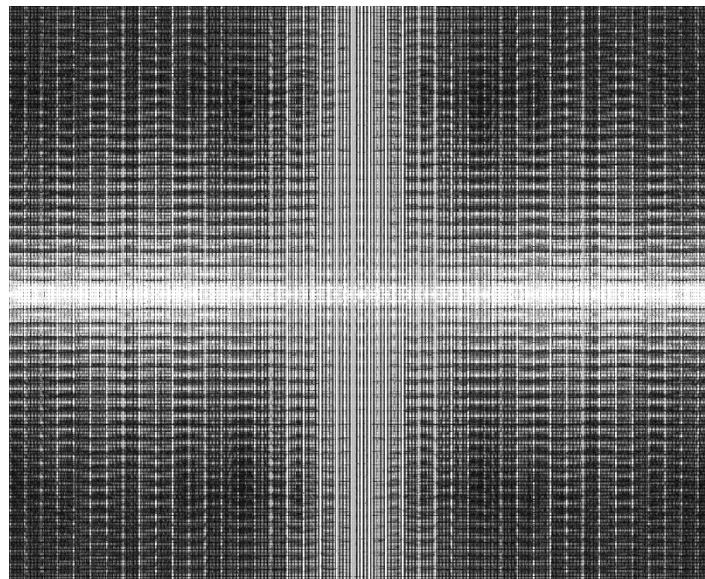
*برای اینکه نتیجه‌گیری تحت عوامل اضافه‌ای مانند قطع شدن بخشی از تصویر و اضافه شدن قسمتی سیاه رنگ قرار نگیرد، تصویر را با مقدار صفر `Pad` کرده‌ام.

بعد از محاسبه Spectrum تصویر، تصویر را به میزان 45 درجه Shearing کرده‌ام و در ادامه مانند آنچه در بالا گفته شده Spectrum را محاسبه می‌کنم. در خروجی‌های کد را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به P5 مراجعه کنید):

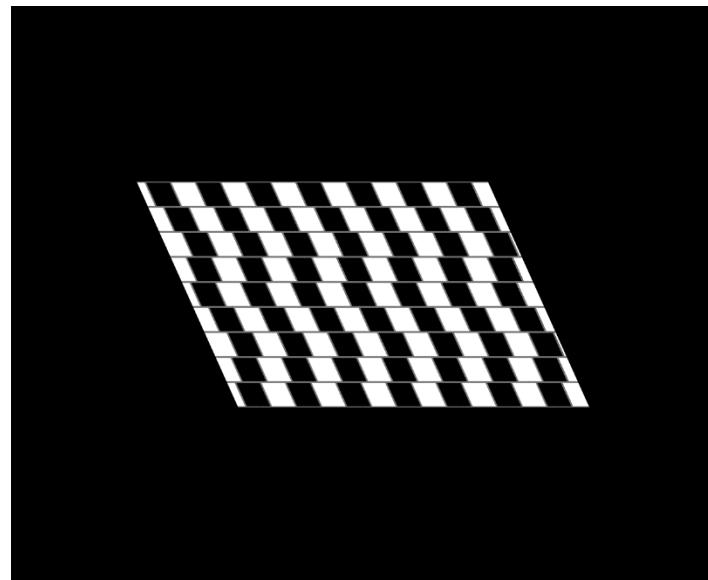
تصویر ورودی (a-a5-input.png):



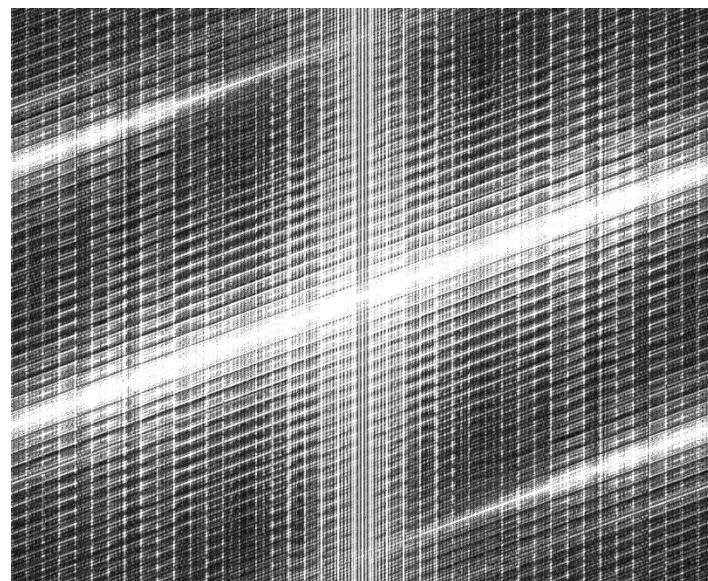
Spectrum تصویر ورودی (a-a5-spectrum.png):



تصویر بعد از Shearing : (a-a5-shearing.png)



تصویر بعد از Shearing Spectrum : (a-a5-spectrum-shearing.png)



نتیجه‌گیری: همین طور که در Spectrum بعد از Shearing مشاهده می‌شود، جدیدی نیز نسبت به قبلي 45 درجه دچار Shearing شده است.

قسمت (b)

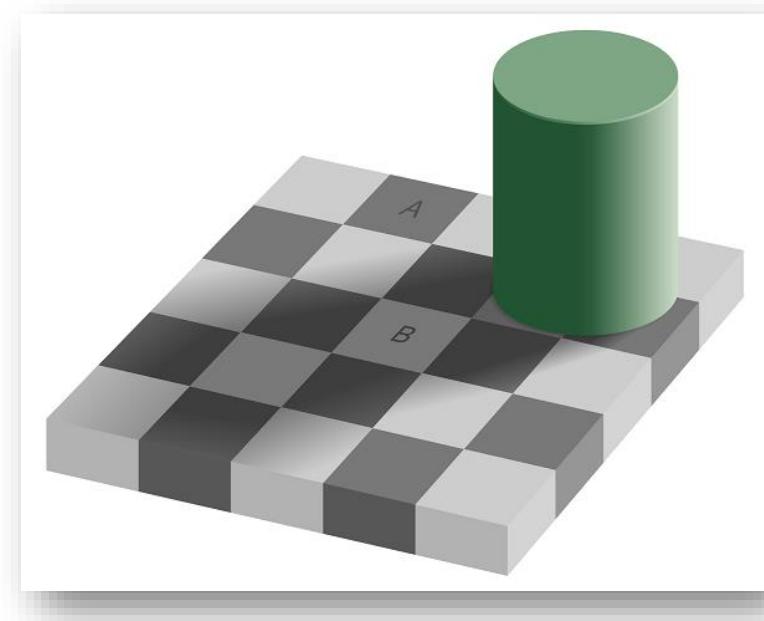
زیر قسمت (b1)

کد این قسمت در p5b_b1.m قرار دارد. ابتدا تصویر را می‌خوانم، سپس دیتابایپ آن را به صورت double در می‌آورم، بعد از آن با استفاده از تابع fftshift و fft2 تبدیل فوریه را ضمن انتقال مرکز به مرکز صفحه انجام می‌دهم. سپس Spectrum آن را محاسبه می‌کنم و نمایش می‌دهم.

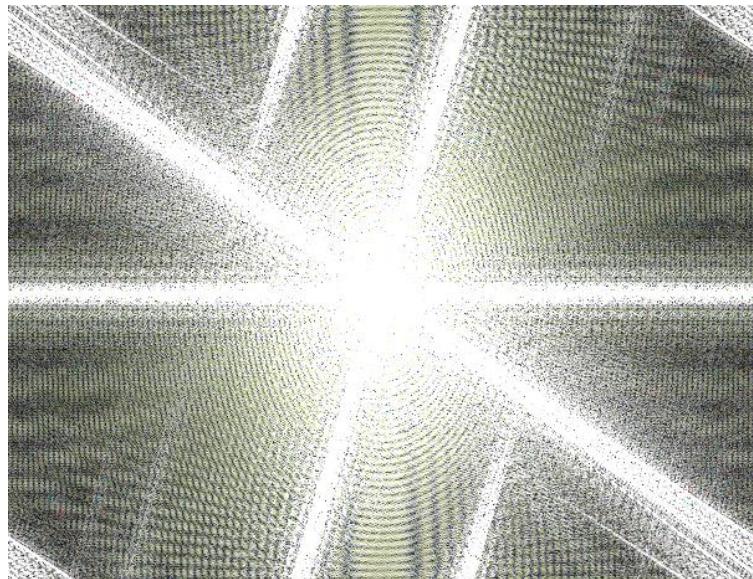
در ادامه تغییر را بر روی Fourier Transform اعمال می‌کنم و آن را به اندازه‌ی ۶۰ پیکسل به سمت راست Translate می‌کنم.

در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به p5 مراجعه فرمایید):

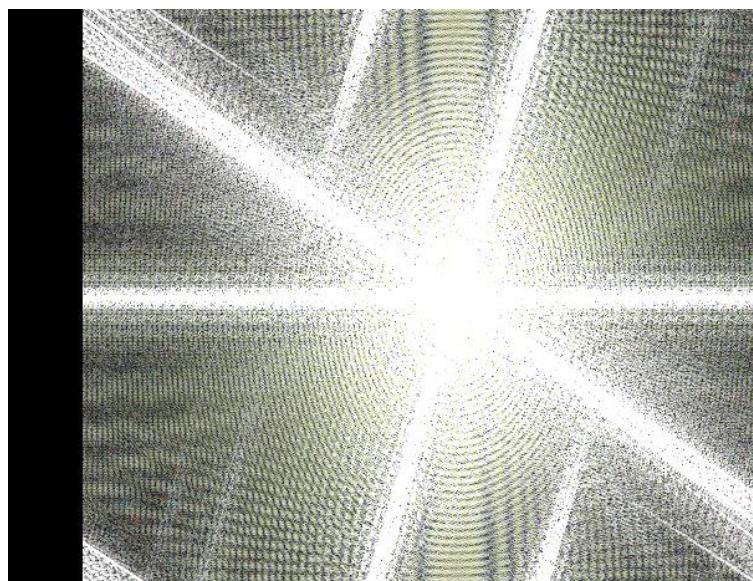
تصویر ورودی:



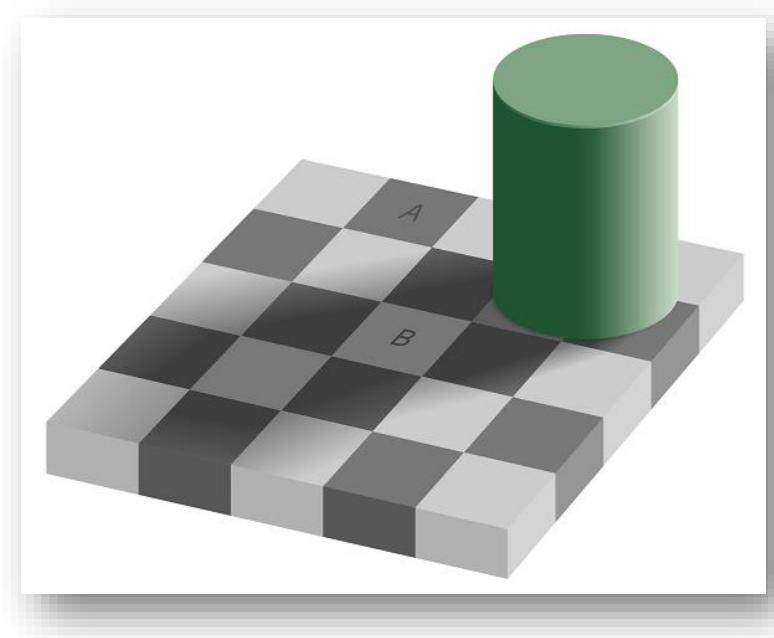
:<(p5b_b1_spectrum_of_input.png) تصویر ورودی Spectrum



: (p5b_b1_spectrum_after_change.png) Translate Spectrum تصویر بد از



: (p5b_b1_reconstructed_image.png) جدید Fourier Transform تصویر ساخته شده با استفاده از



نتیجه‌گیری: همین طور که دیده می‌شود، تصویر بر اثر **Translate** دچار هیچ گونه تغییری نشده است.

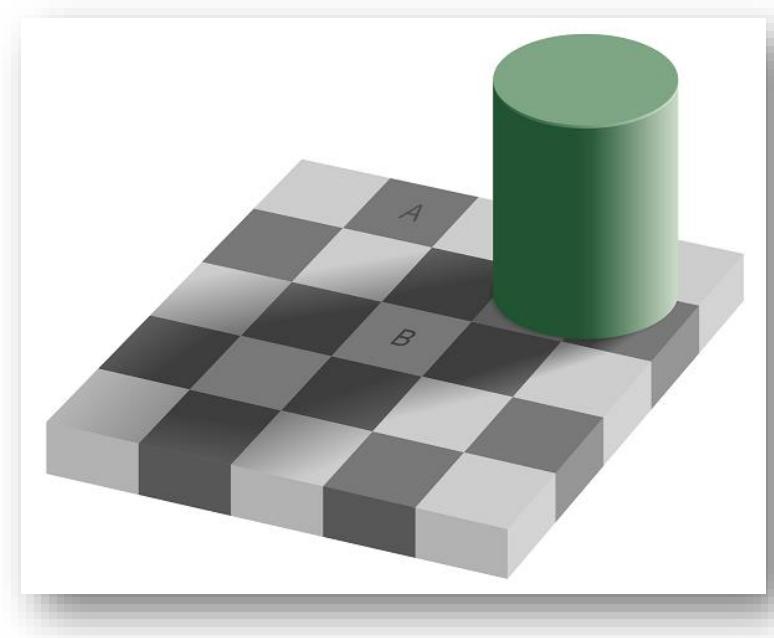
زیر قسمت (b2)

کد این قسمت در p5b_b2.m قرار دارد. ابتدا تصویر را می‌خوانم، سپس دیتابایپ آن را به صورت **double** تبدیل می‌کنم، بعد از آن با استفاده از تابع **fft2** و **fftshift** تبدیل فوریه را ضمن انتقال مرکز به مرکز صفحه انجام می‌دهم. سپس **Spectrum** آن را محاسبه می‌کنم و نمایش می‌دهم.

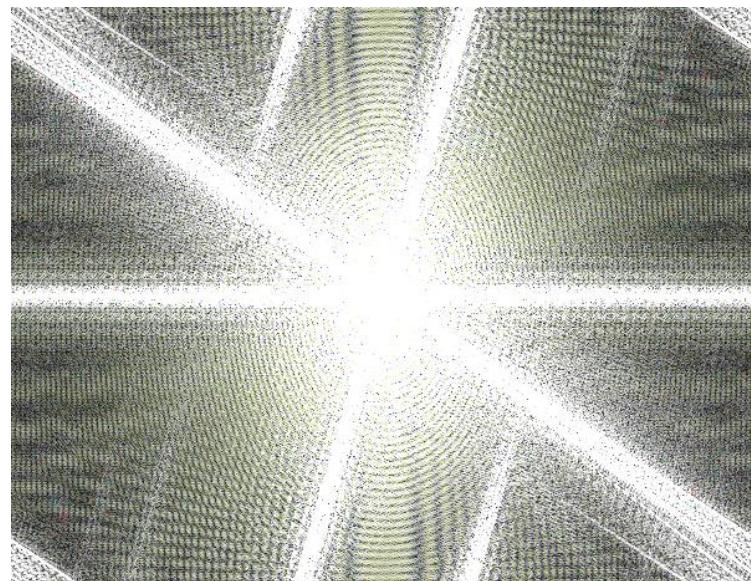
در ادامه تغییر را بر روی **Fourier Transform** اعمال می‌کنم و آن را با اندازه‌های ۹۰ و ۱۸۰ درجه **Rotate** می‌کنم.

در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به p5 مراجعه فرمایید):

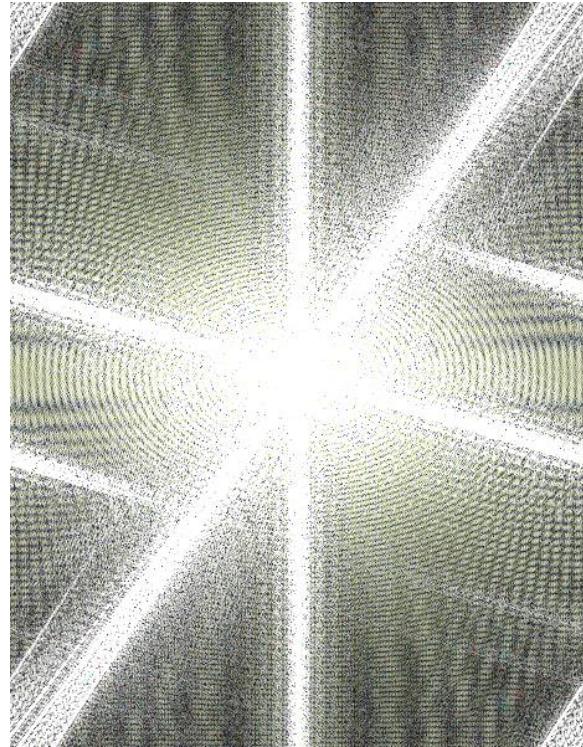
تصویر ورودی:



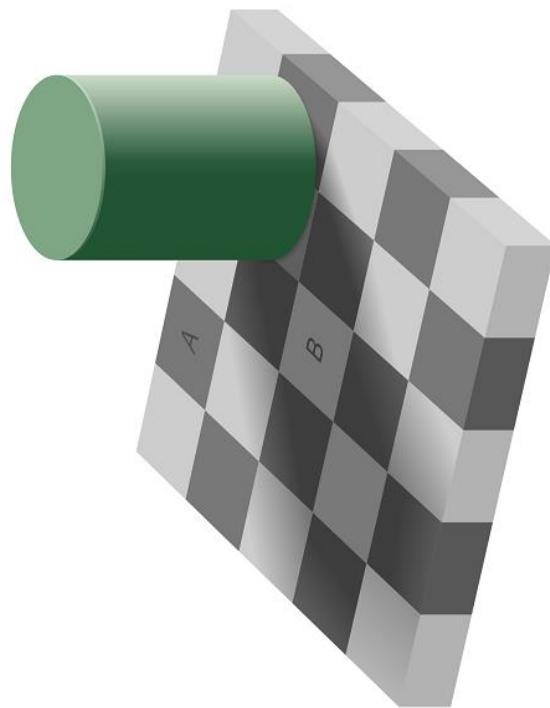
:*(p5b_b2_spectrum_of_input.png)* تصویر ورودی Spectrum



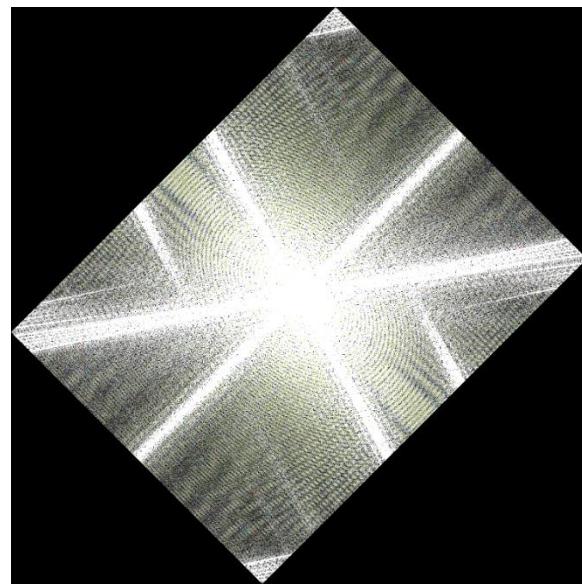
p5b_b2_spectrum_after_change-) ۹۰ به اندازه‌ی Rotate تصویر بد از Spectrum :*(degree90.png*



تصویر ساخته شده با استفاده از Fourier Transform جدید Rotate به اندازه‌ی ۹۰ : (p5b_b2_reconstructed_image-degree-90.png)



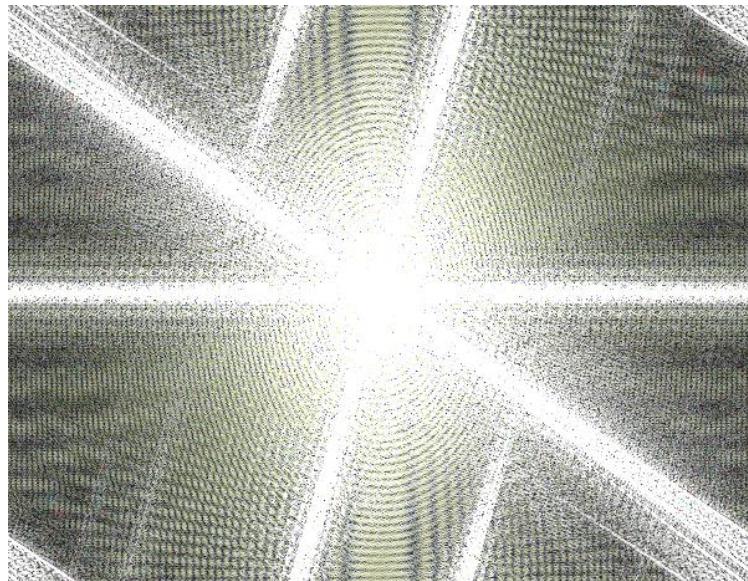
p5b_b2_spectrum_after_change-) ۴۵ به اندازه‌ی Rotate از Spectrum : (degree45.png



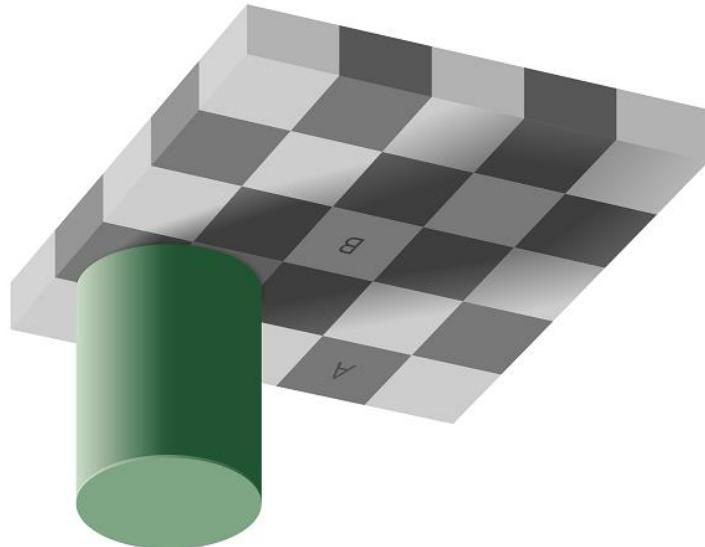
تصویر ساخته شده با استفاده از Fourier Transform جدید Rotate به اندازه‌ی ۴۵ : (p5b_b2_reconstructed_image-degree-45.png)



p5b_b2_spectrum_after_change-(۱۸۰ به اندازهی Rotate از Spectrum :degree180.png



تصویر ساخته شده با استفاده از Fourier Transform جدید Rotate به اندازهی ۱۸۰ : (p5b_b2_reconstructed_image-degree-180.png)



نتیجه‌گیری: همین طور که دیده می‌شود، با چرخش Fourier Transform تصویر بر نیز به همان میزان دچار چرخش می‌شود. این مورد به خوبی در تصویرهای به دست آمده بر اثر چرخش با ۹۰ و ۱۸۰ درجه به خوبی دیده می‌شود . ولی در چرخش با ۴۵ درجه، تصویر به دست آمده در حوزه‌ی spatial کمی مشکل دارم آن هم به خاطر چرخش با زاویه‌ای است که مضربی از ۹۰ نیست و پیکسل‌ها به یک نقطه به صورت دقیق نگاشت نمی‌شوند.

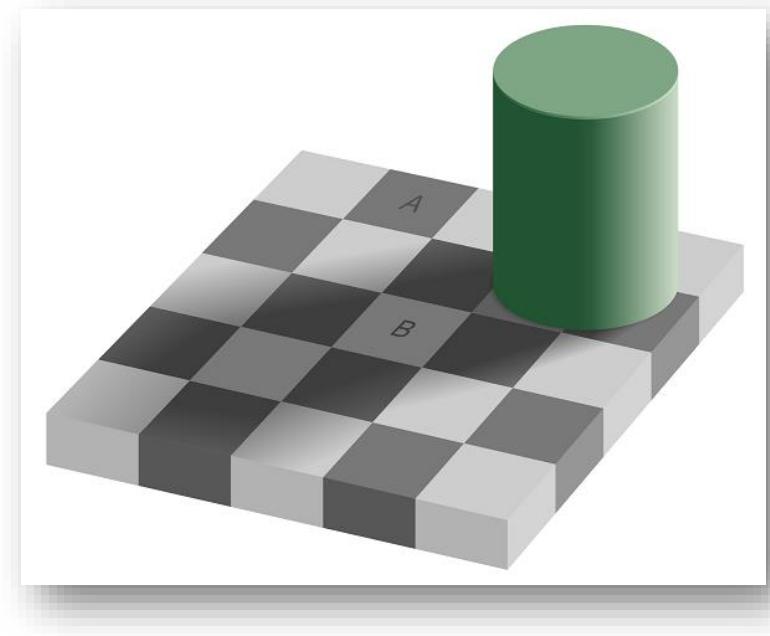
زیر قسمت (b3)

کد این قسمت در p5b_b3.m قرار دارد. ابتدا تصویر را می‌خوانم، سپس دیتاتایپ آن را به صورت double تبدیل می‌کنم، بعد از آن با استفاده از تابع fft2 و fftshift تبدیل فوریه را ضمن انتقال مرکز به مرکز صفحه انجام می‌دهم. سپس Spectrum آن را محاسبه می‌کنم و نمایش می‌دهم.

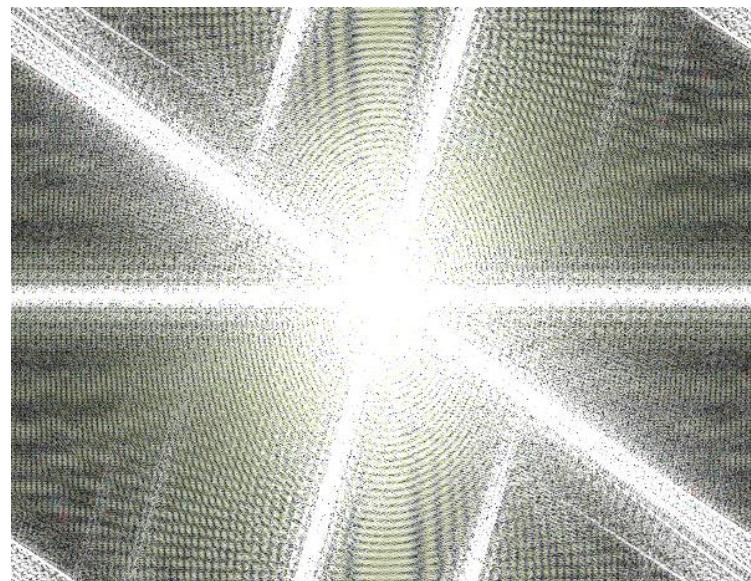
در ادامه تغییر را بر روی Fourier Transform اعمال می‌کنم و آن را با ۰.۵ و ۲ Scale می‌کنم.

در زیر خروجی‌های کد را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به p5 مراجعه فرمایید):

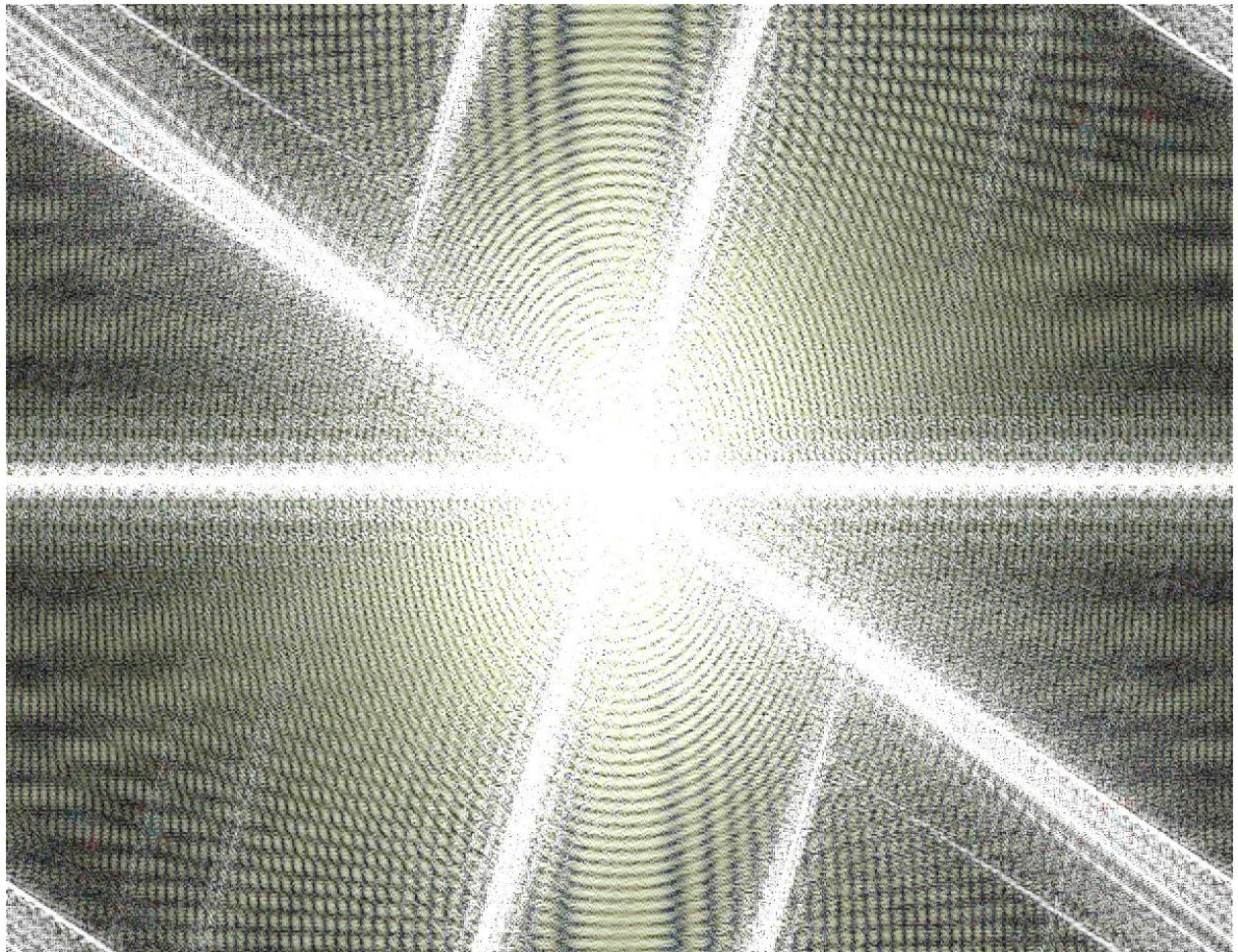
تصویر ورودی:



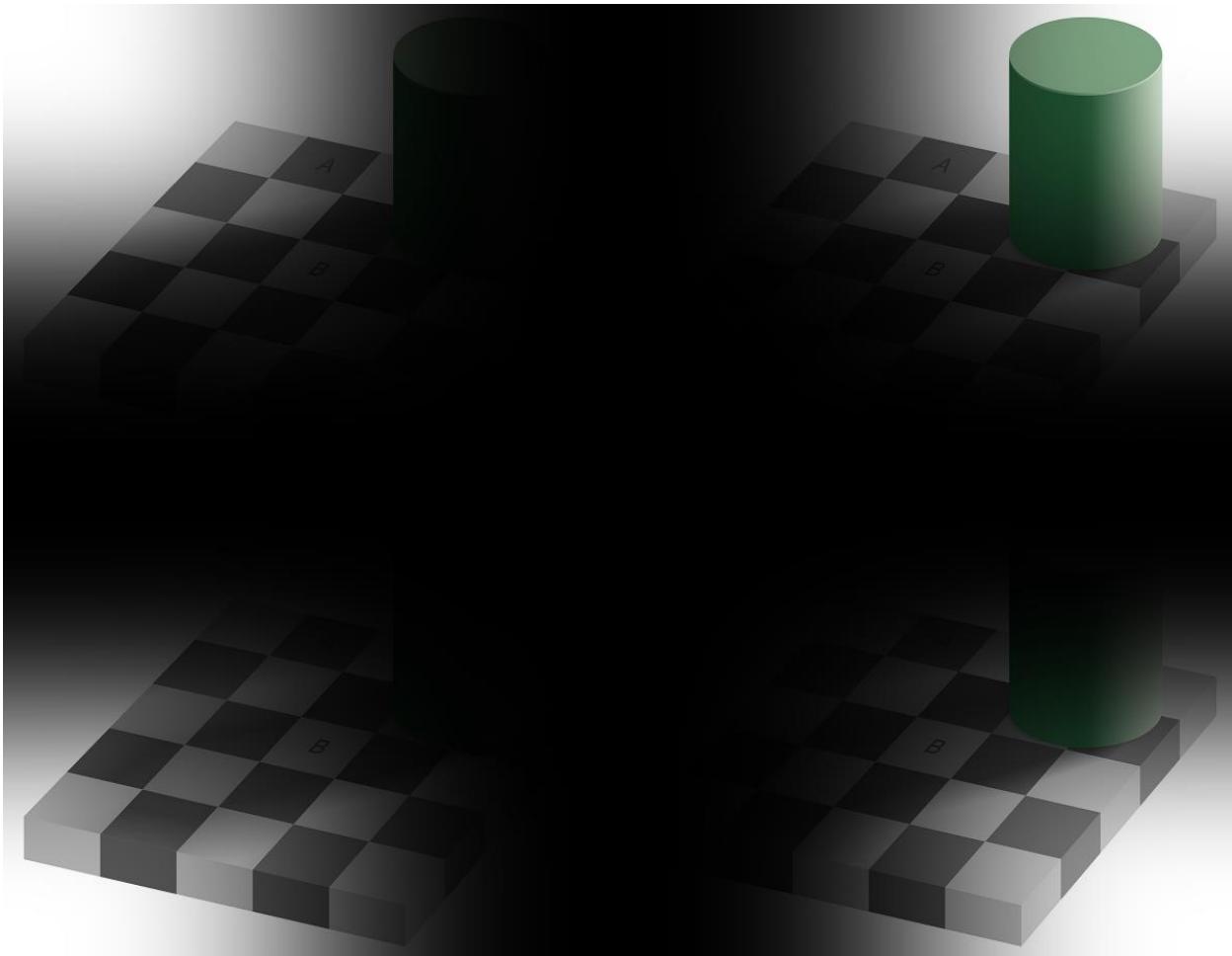
:*(p5b_b3_spectrum_of_input.png)* تصویر ورودی Spectrum



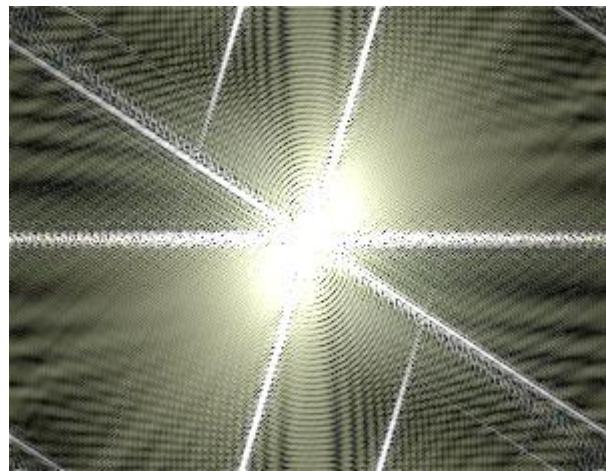
:*(p5b_b3_spectrum_after_change-scale2.png)* تصویر بد از scale 2 به اندازه‌ی Spectrum



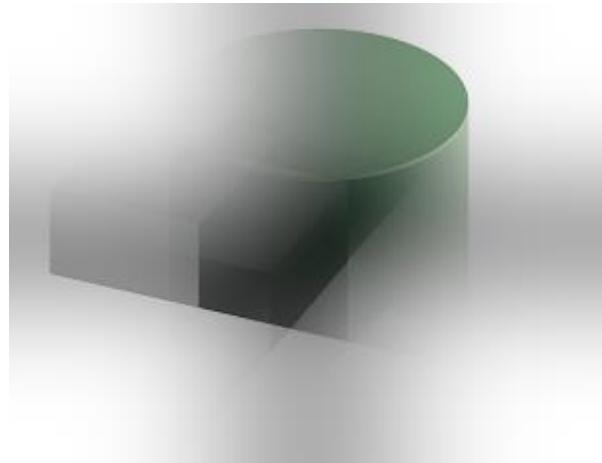
تصویر ساخته شده با استفاده از Fourier Transform جدید به اندازهی 2
:(p5b_b3_reconstructed_image-degree-2.png)



p5b_b3_spectrum_after_change-(۰,۵ به اندازه‌ی scale از Spectrum تصویر بعد :scale0.5.png



تصویر ساخته شده با استفاده از Fourier Transform جدید Scale به اندازه‌ی ۰,۵ : (p5b_b3_reconstructed_image-degree-0.5.png)



نتیجه‌گیری: با اسکیل کردن Fourier Transform، از نظر طول و عرض، تصویر هم به آن اندازه کوچک و بزرگ می‌شود. ولی اگر به خروجی‌ها نگاه کنیم متوجه می‌شویم که موقعیت مکانی نسبت به تصویر اصلی حفظ نشده است.

تمرین ۶

کدها و نتیجه‌های این بخش در پوشه‌ی p6 قرار دارد.

(a) قسمت

کدهای این قسمت در p6a_func.m و p6a_run.m قرار دارد.

در p6a_func.m تابعی قرار دارد که یک تصویر، سایز فیلتر (n) و width را به عنوان آرگمان می‌گیرد. ابتدا تصویر را از سمت راست و پایین به اندازه‌ی n با صفر pad می‌کند. سپس آن را از نوع double به Uint تبدیل می‌کند. در ادامه با استفاده از fftshift و fft2 ضمن انقال مرکز از گوشی سمت چپ بالا به مرکز، تبدیل فوریه‌ی آن را حساب می‌کند. در ادامه در حوزه‌ی فرکانس فیلتر Ideal Low Pass را با توجه به Width ایجاد می‌کند. با ضرب نقطه‌ی فیلتر با تبدیل فوریه‌ی تصویر، عمل فیلتر انجام می‌شود. با استفاده از تبدیل معکوس فوریه تصویر در حوزه‌ی Spatial به دست می‌آید.

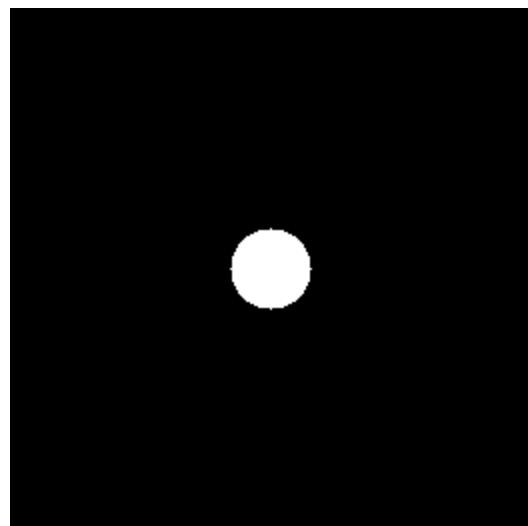
در p6a_run.m تابع بالا برای مقدارهای مختلف فراخوانده می‌شود و نتیجه در فایل ذخیره می‌شود.

خروجی‌ها را در ادامه مشاهده می‌کنید:

تصویر ورودی:



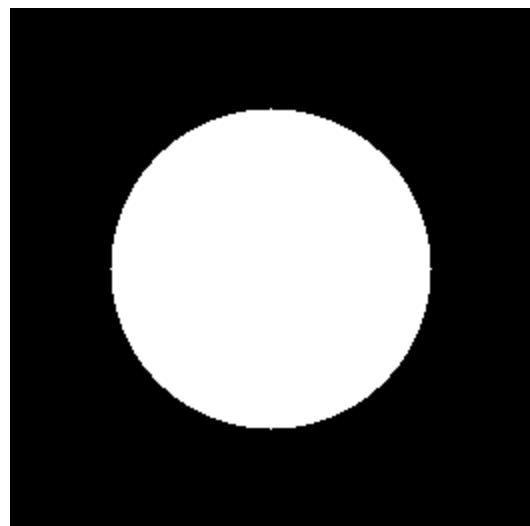
: (p6a_filter_n_260_width_20.png) width=20 , n=260 با Ideal Low Pass فیلتر



: (p6a_filtered_img_n_260_width_20.png) تصویر حاصل از اعمال فیلتر بالا



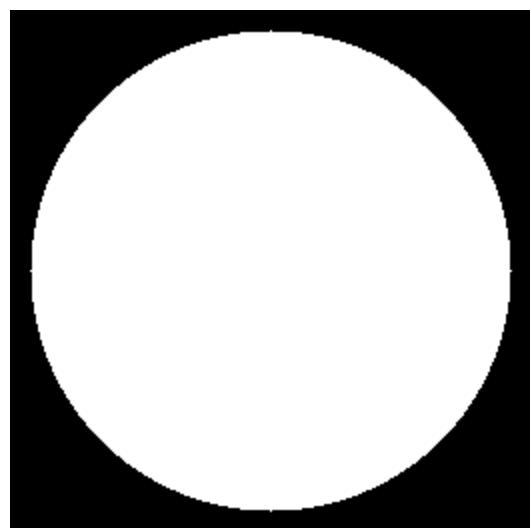
: (p6a_filter_n_260_width_80.png) width= 80, n=260 با Ideal Low Pass فیلتر



: (p6a_filtered_img_n_260_width_80.png) تصویر حاصل از اعمال فیلتر بالا



: (p6a_filter_n_260_width_120.png) width=120 و n=260 با Ideal Low Pass فیلتر



تصویر حاصل از اعمال فیلتر بالا (p6a_filtered_img_n_260_width_120.png)



نتیجه‌گیری: هر چه میزان width افزایش پیدا کرده است، تصویر کمتر Blur شده است.

(c) قسمت

کدهای این قسمت در p6b_func.m و p6b_run.m قرار دارد.

در p6b_func.m تابعی قرار دارد که یک تصویر، سایز فیلتر (n) و width را به عنوان آرگمان می‌گیرد. ابتدا تصویر را از سمت راست و پایین به اندازه‌ی n با صفر pad می‌کند. سپس آن را از نوع double Uint به double تبدیل می‌کند. در ادامه با استفاده از fft2 و fftshift ضمن انقال مرکز از گوشه‌ی سمت چپ بالا به مرکز، تبدیل فوریه‌ی آن را حساب می‌کند. در ادامه در حوزه‌ی فرکانس فیلتر Gaussian Low Pass را با توجه به Width ایجاد می‌کند. با ضرب نقطه‌ی فیلتر با تبدیل فوریه‌ی تصویر، عمل فیلتر انجام می‌شود. با استفاده از تبدیل معکوس فوریه تصویر در حوزه‌ی Spatial به دست می‌آید.

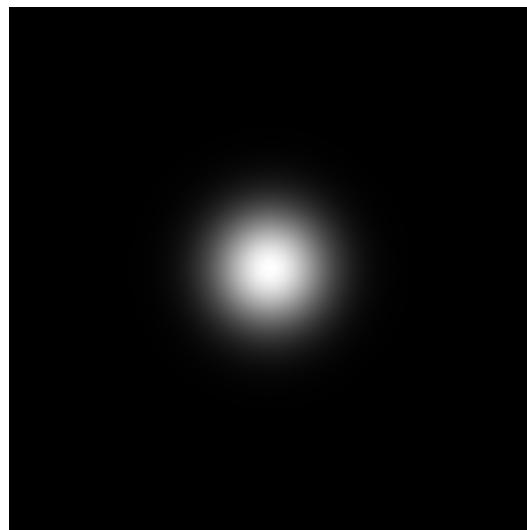
در p6b_run.m تابع بالا برای مقدارهای مختلف فراخوانده می‌شود و نتیجه در فایل ذخیره می‌شود.

خروجی‌ها در ادامه مشاهده می‌کنید:

تصویر ورودی:



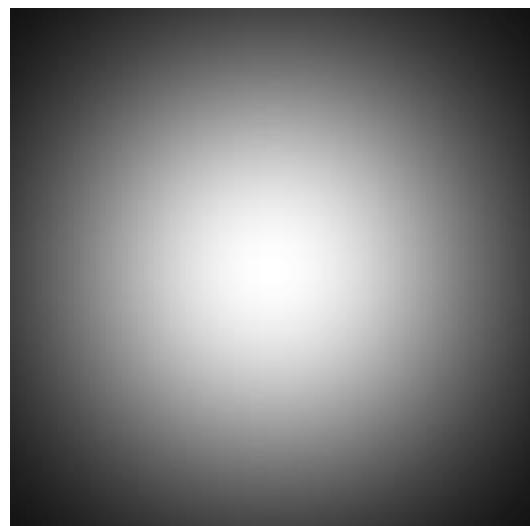
: (p6b_filter_n_260_width_20.png) width=20 و n=260 با Gaussian Low Pass فیلتر



تصویر حاصل از اعمال فیلتر بالا (p6b_filtered_img_n_260_width_20.png)



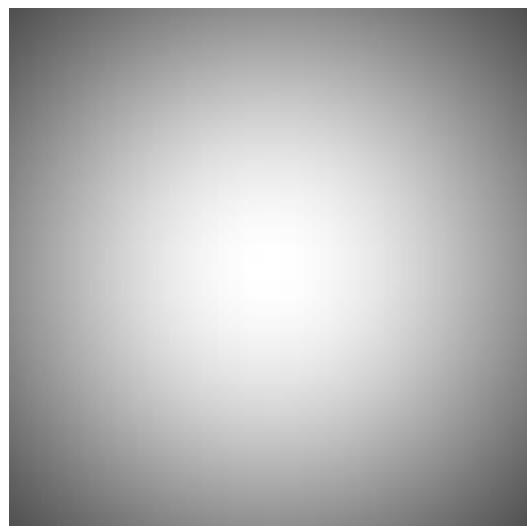
: (p6b_filter_n_260_width_80.png) width= 80 و n=260 با Gaussian Low Pass فیلتر



تصویر حاصل از اعمال فیلتر بالا (p6b_filtered_img_n_260_width_80.png)



: (p6b_filter_n_260_width_120.png) width=120 با Gaussian Low Pass فیلتر



تصویر حاصل از اعمال فیلتر بالا (p6b_filtered_img_n_260_width_120.png)



نتیجه‌گیری: هر چه میزان width افزایش پیدا کرده است، تصویر کمتر Blur شده است. همچنین بر خلاف Ideal Low Pass Filter تصویر دچار Ringing نشده است.

قسمت ۵

در فیلتر Gaussian Low Pass شاهد پدیده‌ی Ringing هستیم. این پدیده در Ideal Low Pass فیلتر دیده نمی‌شود.

دلیل این اتفاق این است که در Ideal Low Pass Filter درون یک ناحیه را برابر با یک قرار می‌دهیم و خارج از آن را برابر با صفر قرار می‌دهیم و انتقال از مقدار یک به صفر به صورت ناگهانی رخ می‌دهم و مرز مشخصی بین صفر و یک است. وقتی انتقال از یک به صفر به صورت ناگهانی رخ می‌دهد، تبدیل معکوس فوریه‌ی آن به صورت تابعی مانند SINC می‌شود که دارای لبه‌های جانبی منفی و مثبت است. ضرب فیلتر در تصویر در حوزه‌ی فرکانس برابر با کانوال کردن فیلتر و تصویر در حوزه‌ی Spatial است و از آنجایی که فیلتر در حوزه‌ی Spatial دارای لبه‌های جانبی منفی و مثبت است، در تصویر فیلتر شده پدیده‌ی Ringing رخ می‌دهد. در حالی که در فیلتر Idea Low Pass Gaussian، مقدارها به آماری کاهش پیدا می‌کنند و پرشی بین مقدارها مانند آنچه در

وجود دارد، دیده نمی‌شود و تبدیل فوریه‌ی معکوس آن شکلی شبیه به تابع گوسی دارد و لبه‌های جانبی Pass مثبت و منفی در آن دیده نمی‌شود.

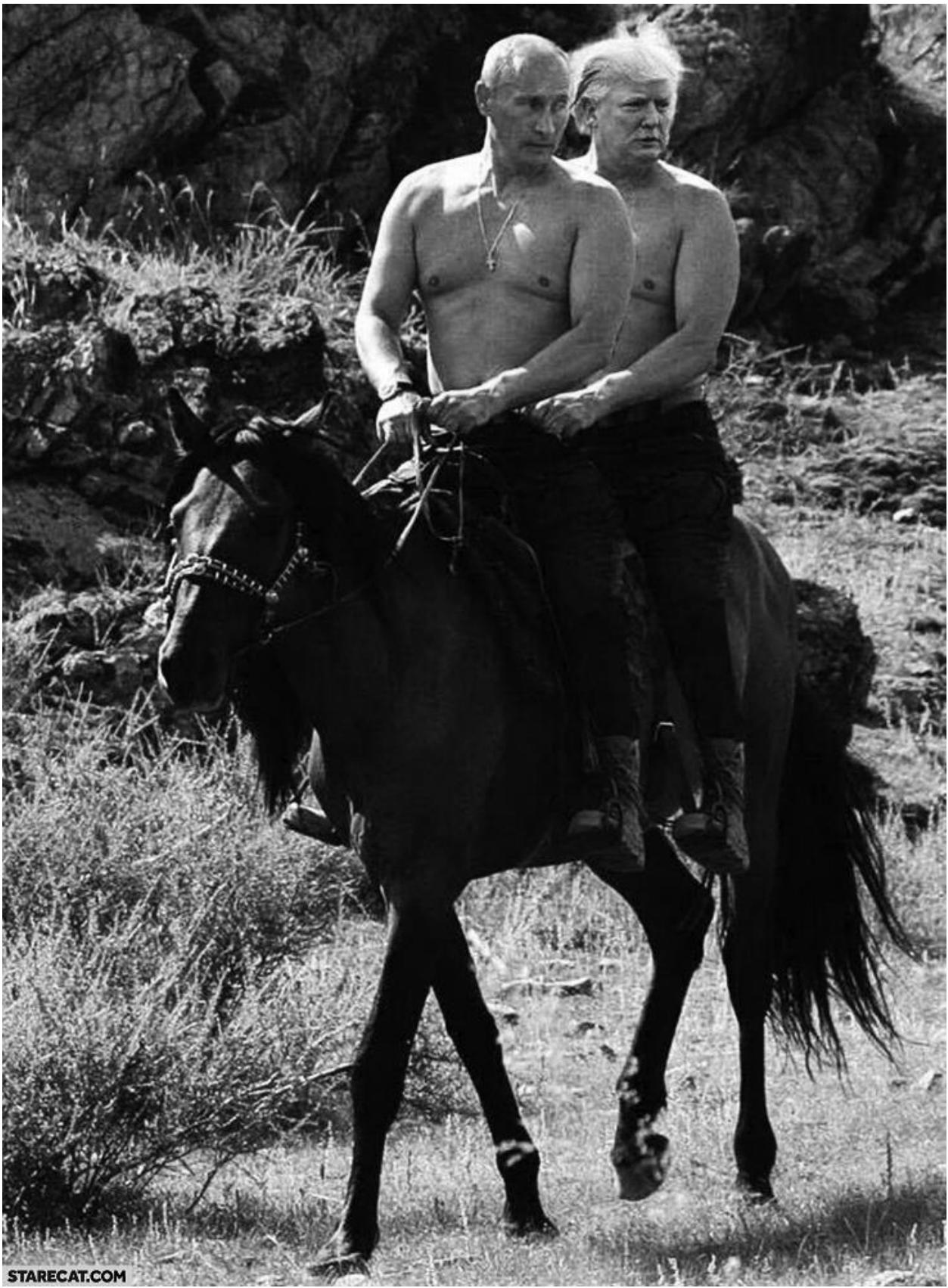
قسمت (d)

کدهای این قسمت در p6d_func.m و p6d_run.m قرار دارد.

در p6d_func.m تابعی قرار دارد که یک تصویر، سایز فیلتر (n) و width را به عنوان آرگمان می‌گیرد. ابتدا تصویر را از سمت راست و پایین به اندازه‌ی n با صفر pad می‌کند. سپس آن را از نوع double به Uint تبدیل می‌کند. در ادامه با استفاده از fftshift و fft2 ضمن انتقال مرکز از گوششی سمت چپ بالا به مرکز، تبدیل فوریه‌ی آن را حساب می‌کند. در ادامه در حوزه‌ی فرکانس فیلتر Ideal Band Pass را با توجه به Width و Center ایجاد می‌کند. با ضرب نقطه‌ی فیلتر با تبدیل فوریه‌ی تصویر، عمل فیلتر انجام می‌شود. با استفاده از تبدیل معکوس فوریه تصویر در حوزه‌ی Spatial به دست می‌آید.

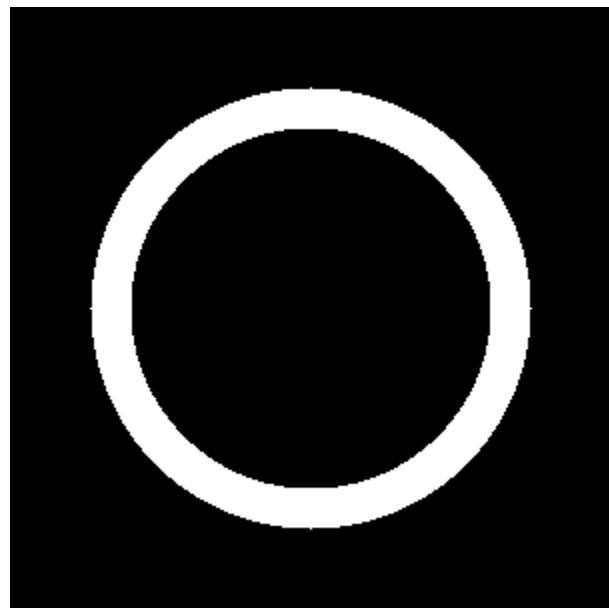
در p6d_run.m تابع بالا برای مقدارهای مختلف فراخوانده می‌شود و نتیجه در فایل ذخیره می‌شود. خروجی‌ها را در ادامه مشاهده می‌کنید (برای دیدن تصویرهای خروجی در اندازه‌ی واقعی به p6 مراجعه نمایید):

تصویر ورودی:



STARECAT.COM

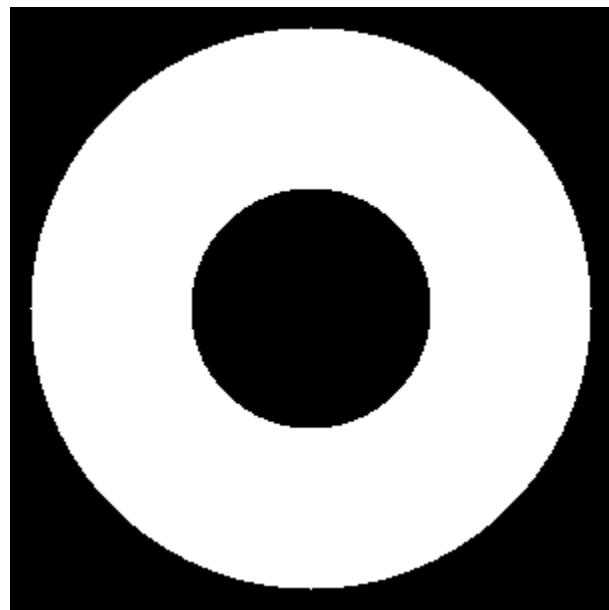
: (p6d_filter_n_260_width_20.png) width=20 n=260 با Ideal Band Pass فیلتر



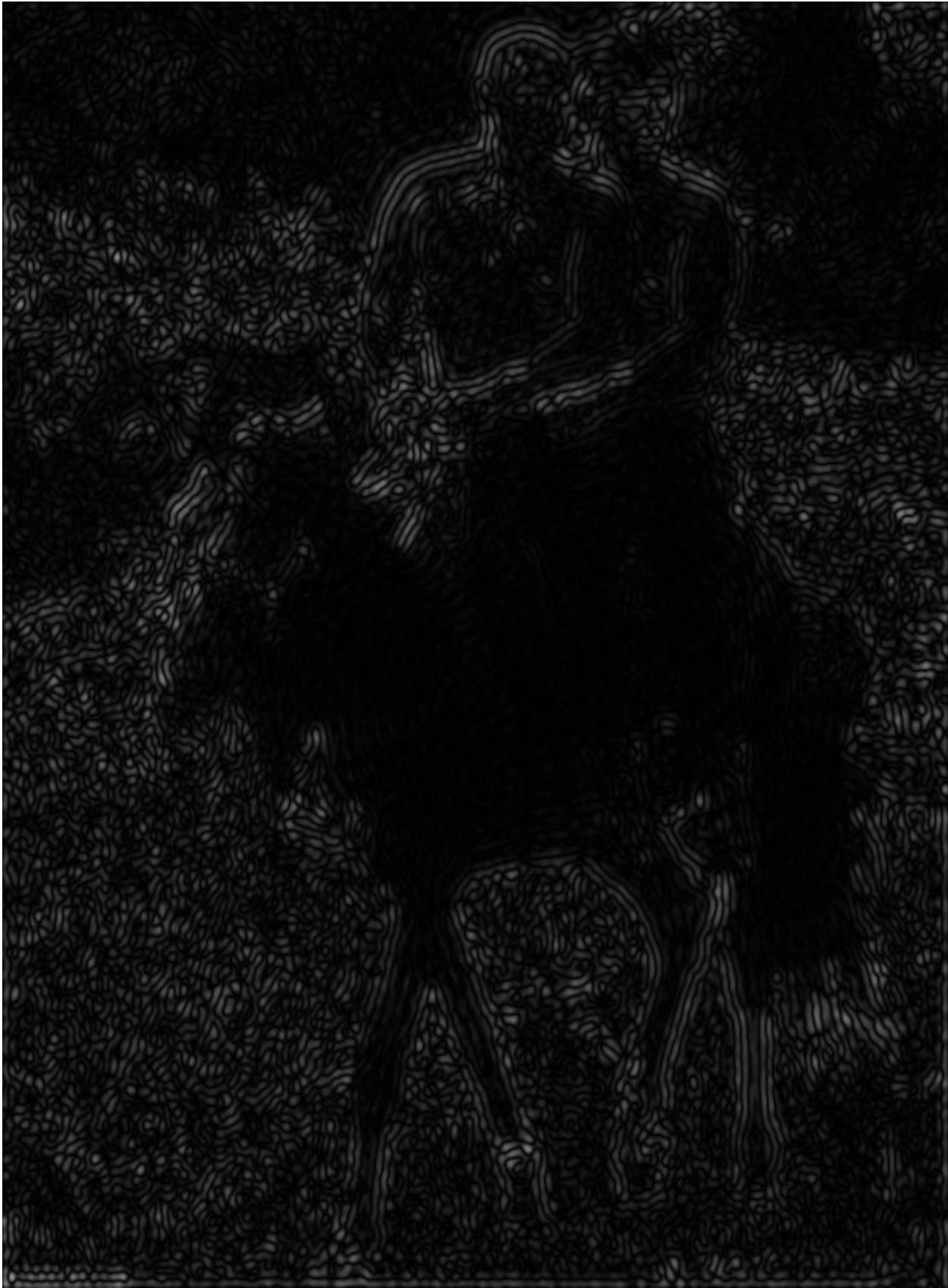
: (p6d_filtered_img_n_260_width_20.png) تصویر حاصل از اعمال فیلتر بالا



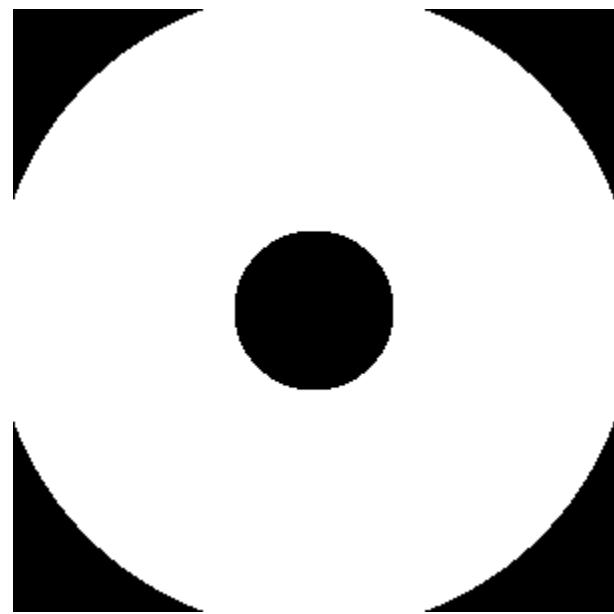
: (p6d_filter_n_260_width_80.png) width= 80, n=260 با Ideal Band Pass فیلتر



: (p6d_filtered_img_n_260_width_80.png) تصویر حاصل از اعمال فیلتر بالا



: (p6d_filter_n_260_width_120.png) width=120 n=260 با Ideal Band Pass فیلتر



: (p6d_filtered_img_n_260_width_120.png) تصویر حاصل از اعمال فیلتر بالا



(e) قسمت

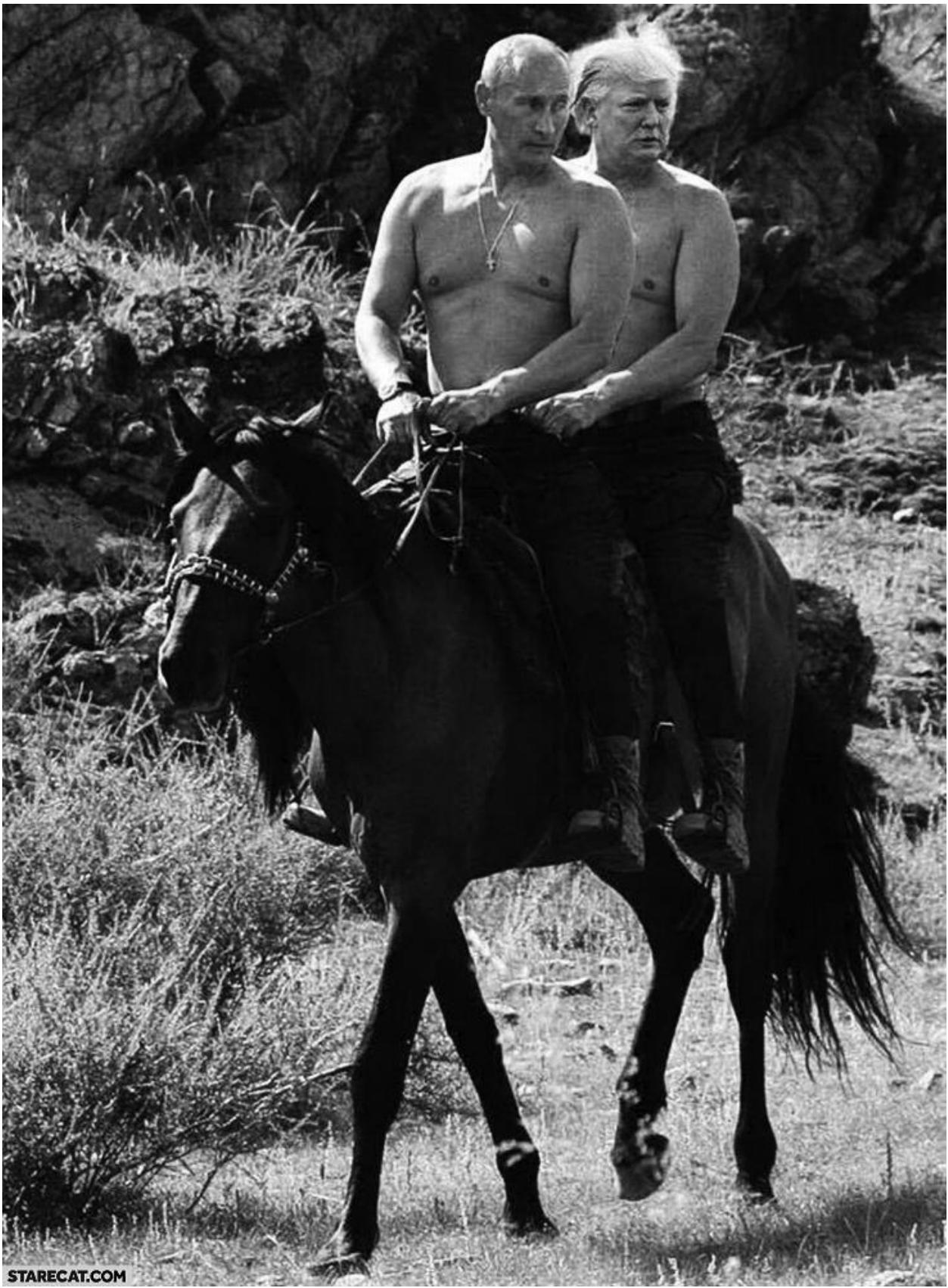
کدهای این قسمت در p6e_func.m و p6e_run.m قرار دارد.

در p6e_func.m تابعی قرار دارد که یک تصویر، سایز فیلتر (n) و width را به عنوان آرگمان می‌گیرد. ابتدا تصویر را از سمت راست و پایین به اندازه‌ی n با صفر pad می‌کند. سپس آن را از نوع Uint به double تبدیل می‌کند. در ادامه با استفاده از fft2 و fftshift ضمن انتقال مرکز از گوشی سمت چپ بالا به مرکز، تبدیل فوریه‌ی آن را حساب می‌کند. در ادامه در حوزه‌ی فرکانس فیلتر Gaussain Band Pass را با توجه به Width و Center ایجاد می‌کند. با ضرب نقطه‌ی فیلتر با تبدیل فوریه‌ی تصویر، عمل فیلتر انجام می‌شود. با استفاده از تبدیل معکوس فوریه تصویر در حوزه‌ی Spatial به دست می‌آید.

در p6e_run.m تابع بالا برای مقدارهای مختلف فراخوانده می‌شود و نتیجه در فایل ذخیره می‌شود.

خروجی‌ها را در ادامه مشاهده می‌کنید (برای دیدن تصویرهای خروجی در اندازه‌ی واقعی به p6 مراجعه نمایید):

تصویر ورودی:

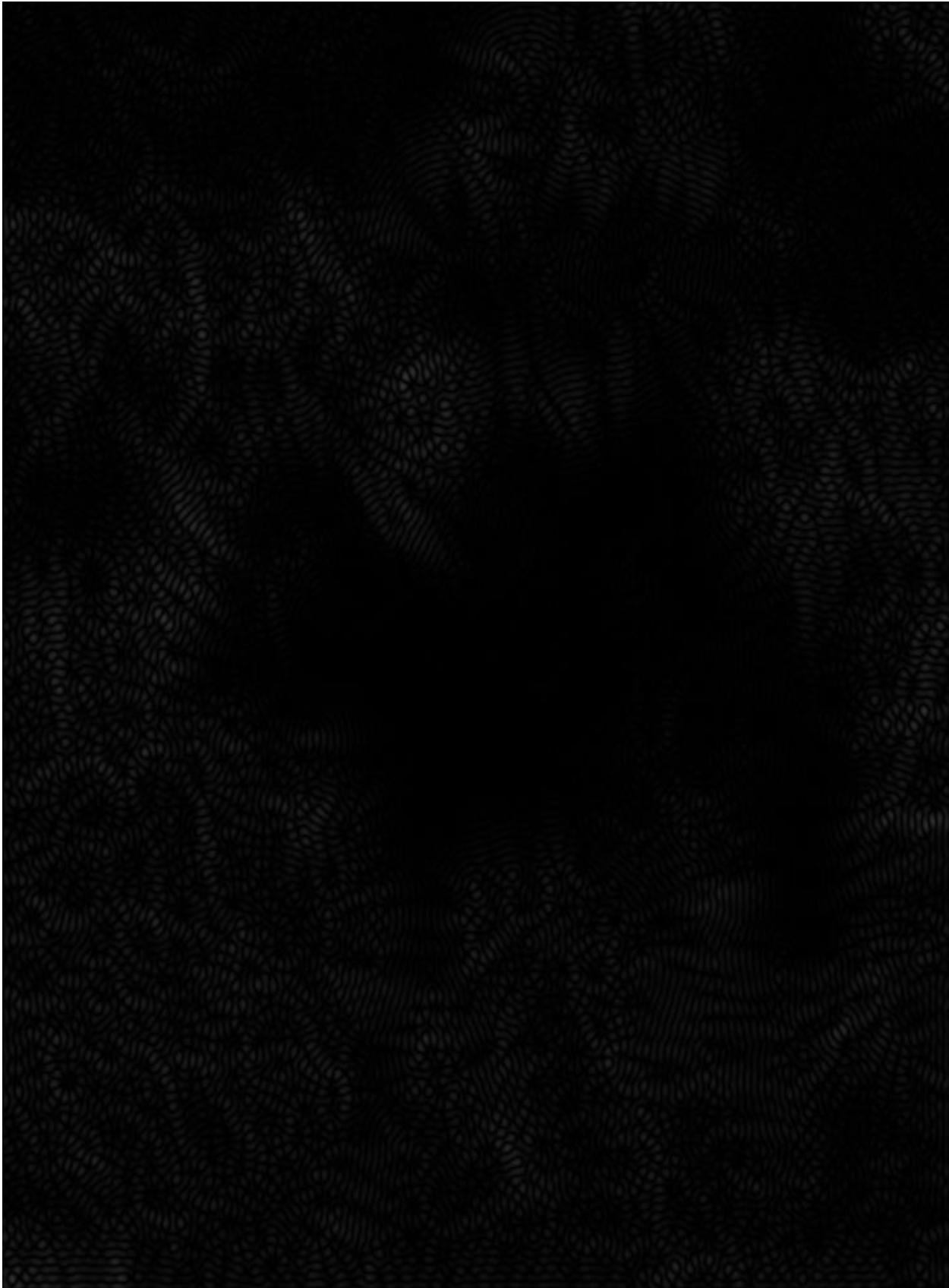


STARECAT.COM

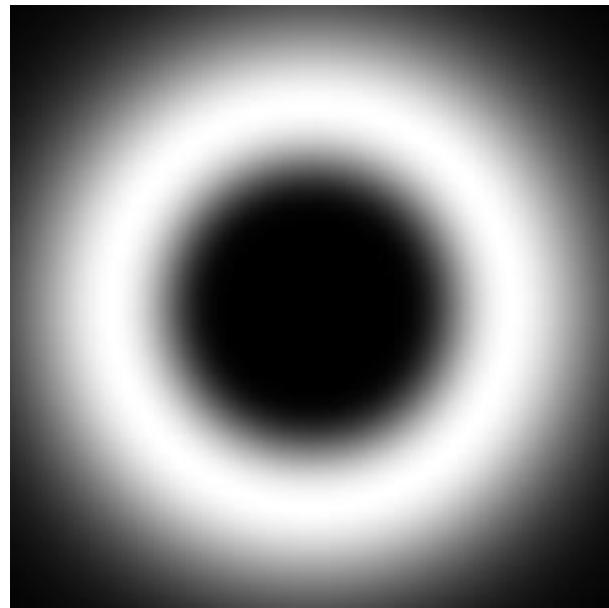
: (p6e_filter_n_260_width_20.png) width=20 و n=260 با Gaussian Band Pass فیلتر



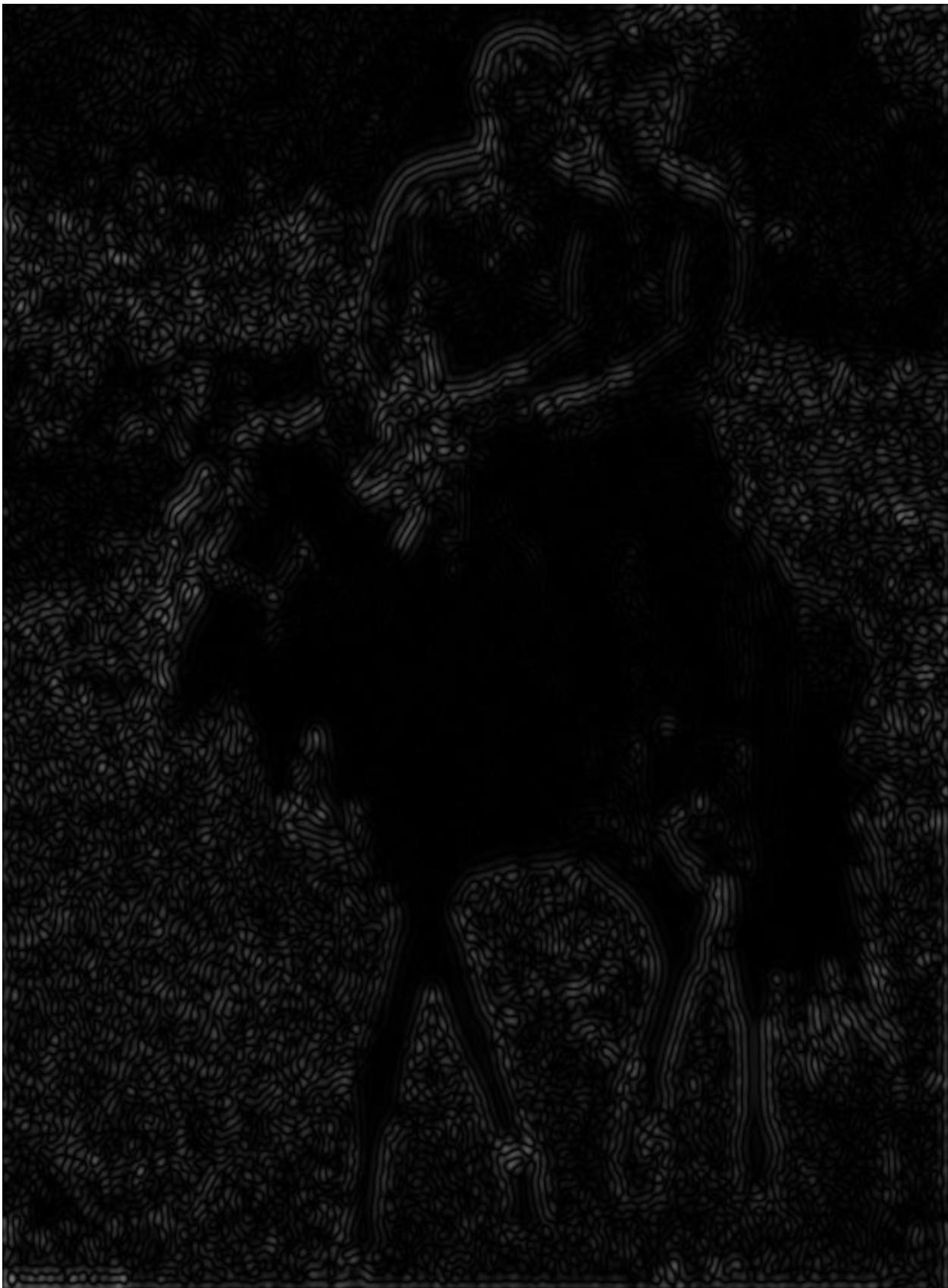
: (p6e_filtered_img_n_260_width_20.png) تصویر حاصل از اعمال فیلتر بالا



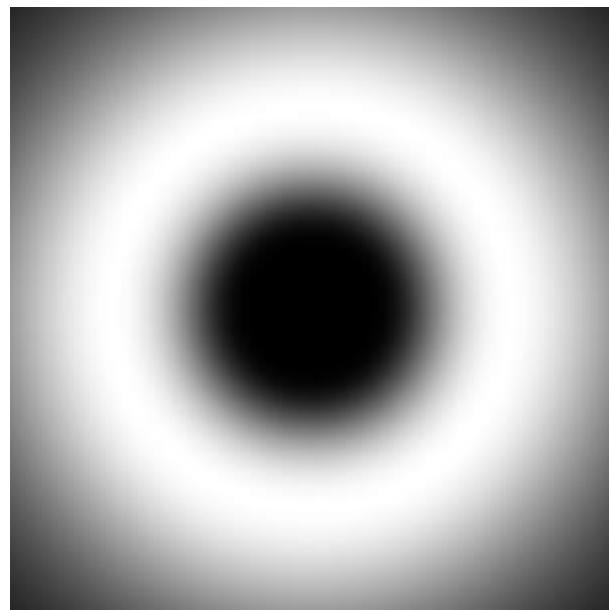
: (p6e_filter_n_260_width_80.png) width= 80 و n=260 با Gaussian Band Pass فیلتر



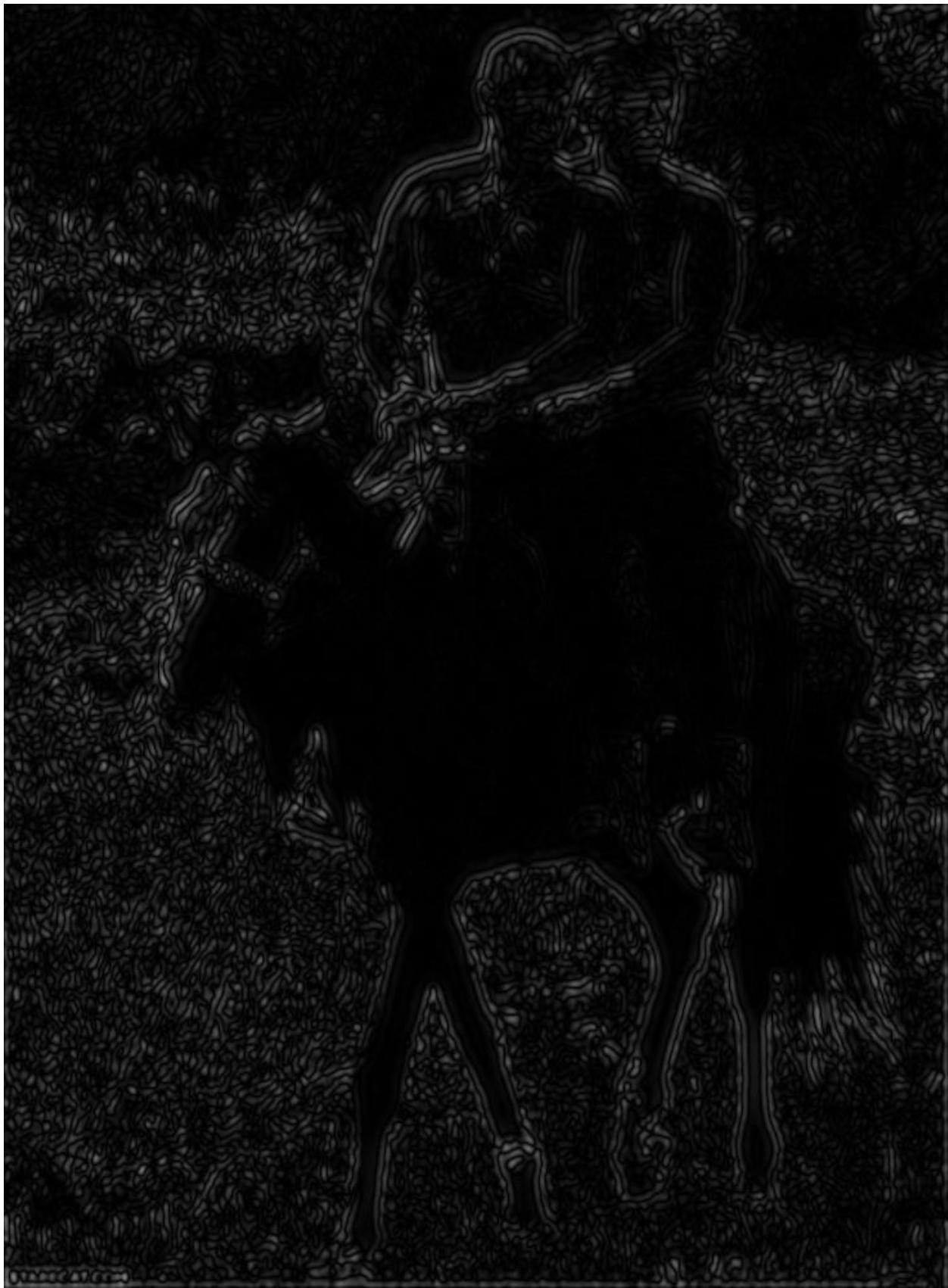
: (p6e_filtered_img_n_260_width_80.png) تصویر حاصل از اعمال فیلتر بالا



: (p6e_filter_n_260_width_120.png) width=120 n=260 با Ideal Band Pass فیلتر



: (p6e_filtered_img_n_260_width_120.png) تصویر حاصل از اعمال فیلتر بالا



تمرین (۷)

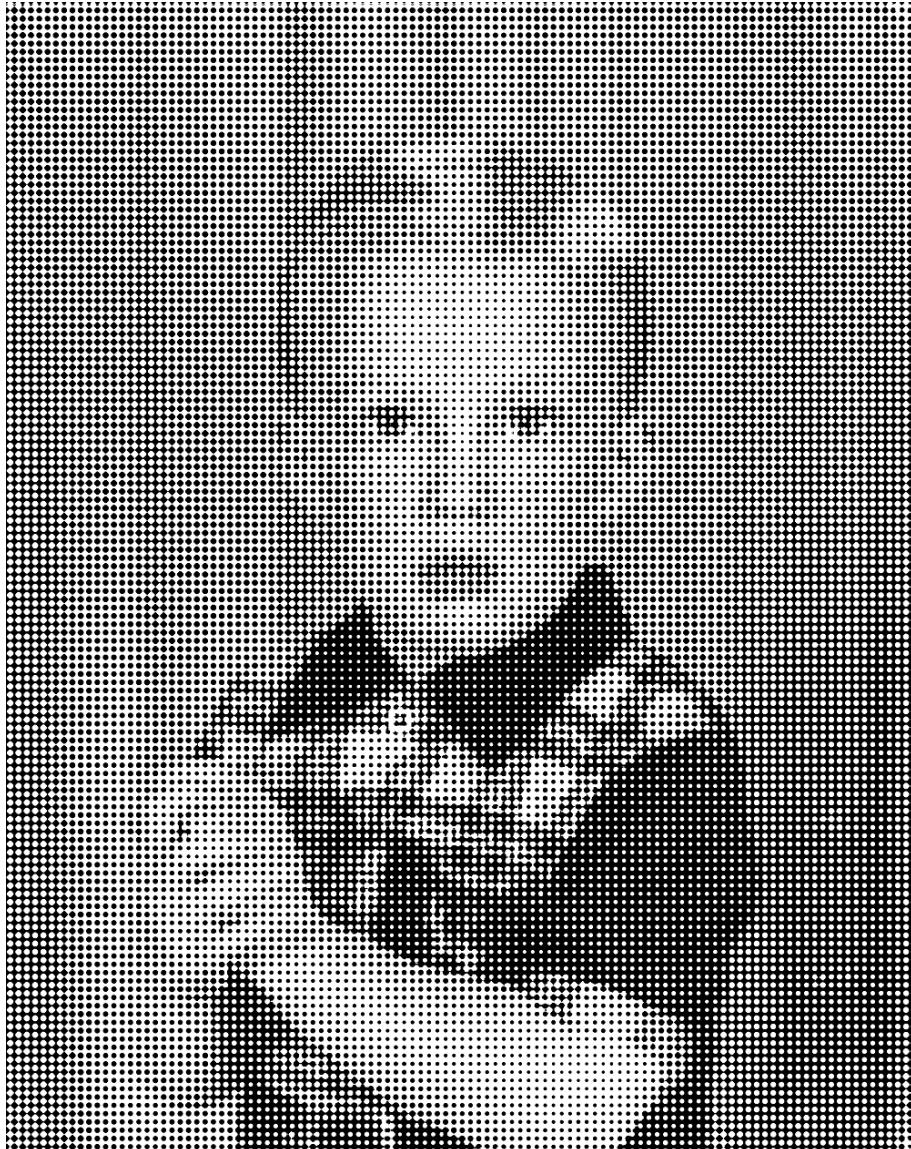
کدها و تصویرهای این قسمت در p7 قرار دارند.

قسمت (a)

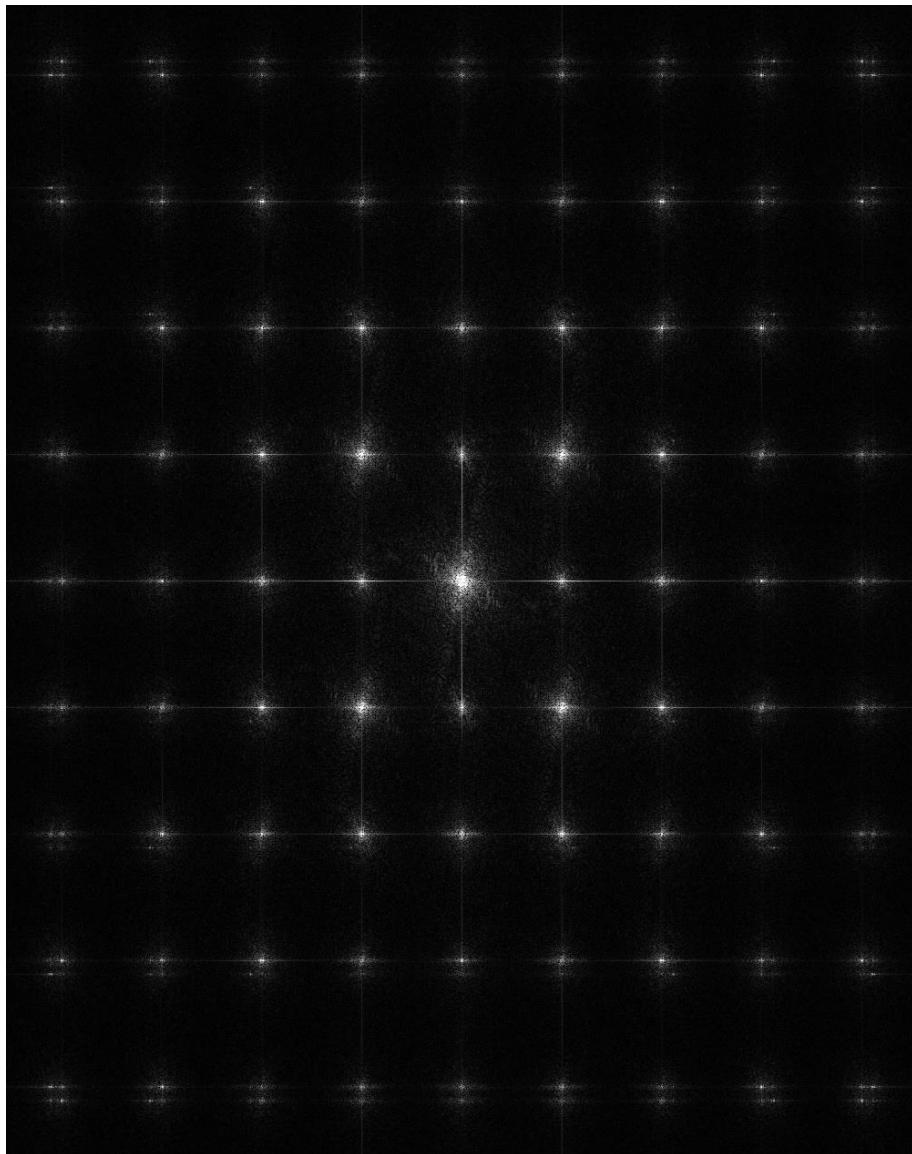
کد این قسمت از سوال در p7a.m قرار دارد. در ابتدا تصویر را می‌خوانم. سپس دیتا تایپ آن را به double تبدیل می‌کنم. در ادامه با استفاده از fft2 و fftshift ضمن انقال مرکز مختصات به مرکز صفحه، تبدیل فوریه‌ی تصویر را محاسبه می‌کنم. سپس Angle Phase و Spectrum را محاسبه می‌کنم و آن‌ها را نمایش می‌دهم. از آنجایی که پترن‌های تکراری در تصویر به صورت منظم تکرار شده‌اند، در Spectrum و تبدیل فوریه‌هم این پترن‌ها دیده می‌شوند. با ایجاد فیلتر Ideal Notch بر روی 80 نقطه‌ی Peak، و ضرب نظیر به نظیر فیلتر در تبدیل فوریه تصویر این پترن‌ها را حذف می‌کنم. در ادامه با استفاده از تبدیل فوریه‌ی معکوس تصویر اصلاح شده را به دست می‌آورم.

در زیر خروجی‌ها را مشاهده می‌کنید (برای دیدن تصویرها در اندازه‌ی واقعی به p7 مراجعه کنید):

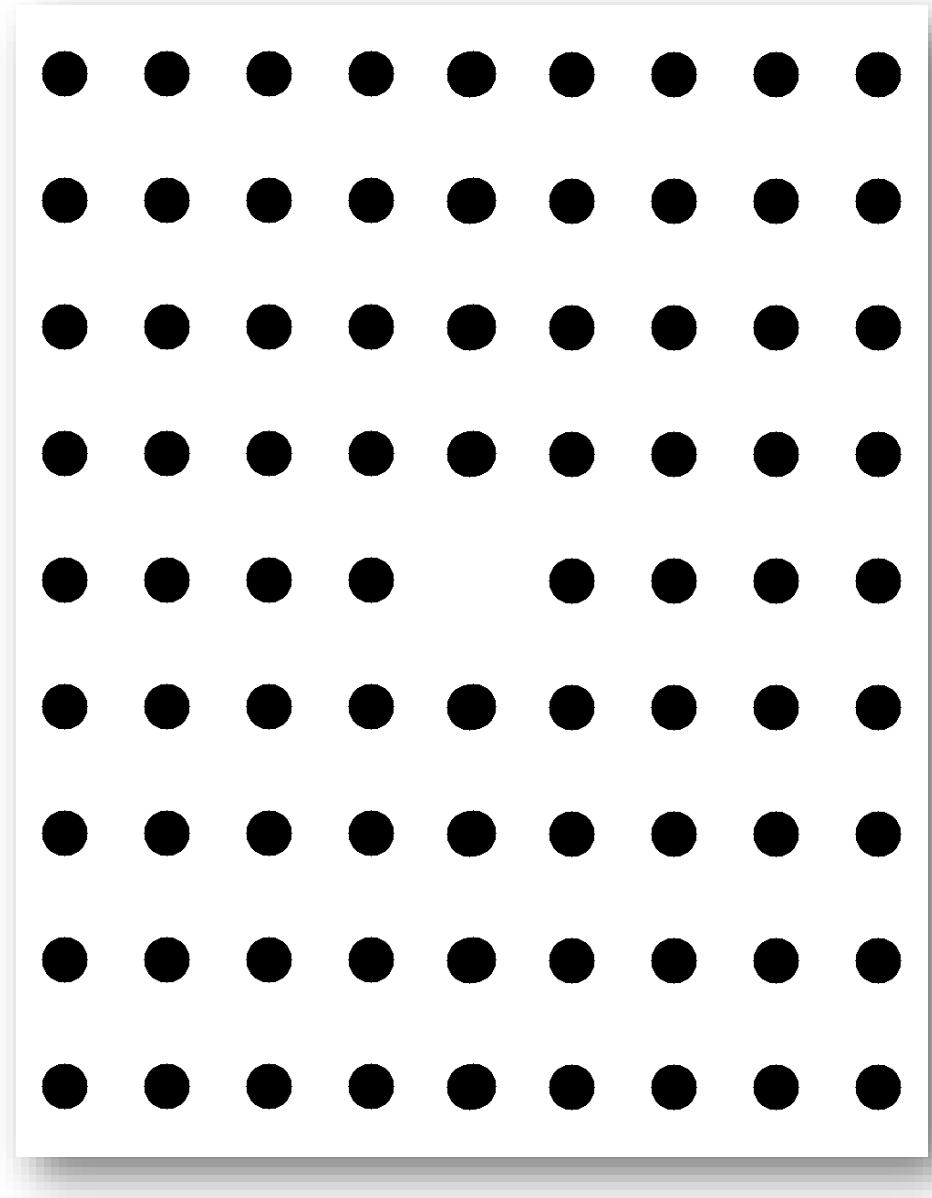
تصویر ورودی:



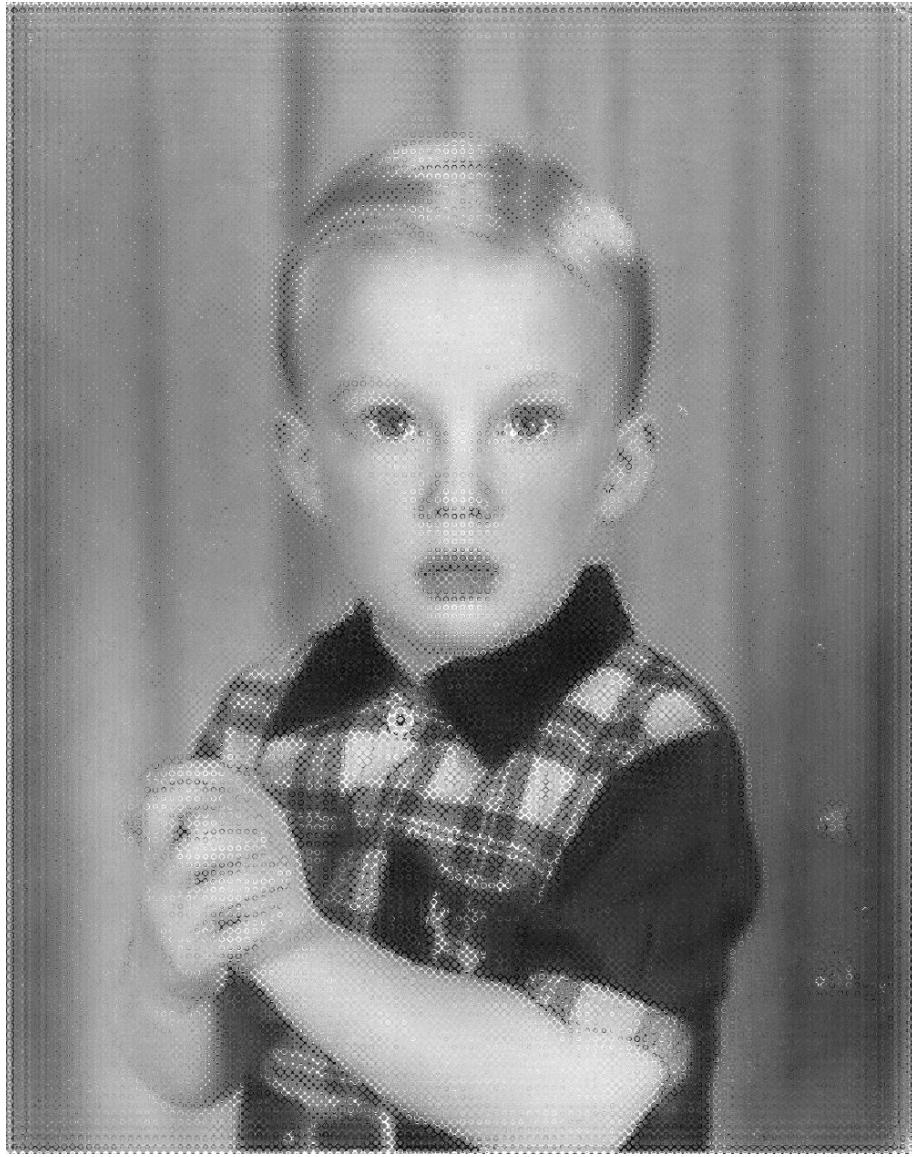
: (p7a_spectrum.png) تصویر ورودی Spectrum



: (p7a_filter.png) Pattern ایجاد شده برای حذف Filter



تصویر حاصل از اعمال فیلر (p7a_filtered_image.png)



همین طور که مشاهده می‌شود، خروجی نسبت به ورودی، بسیار بهبود یافته است.

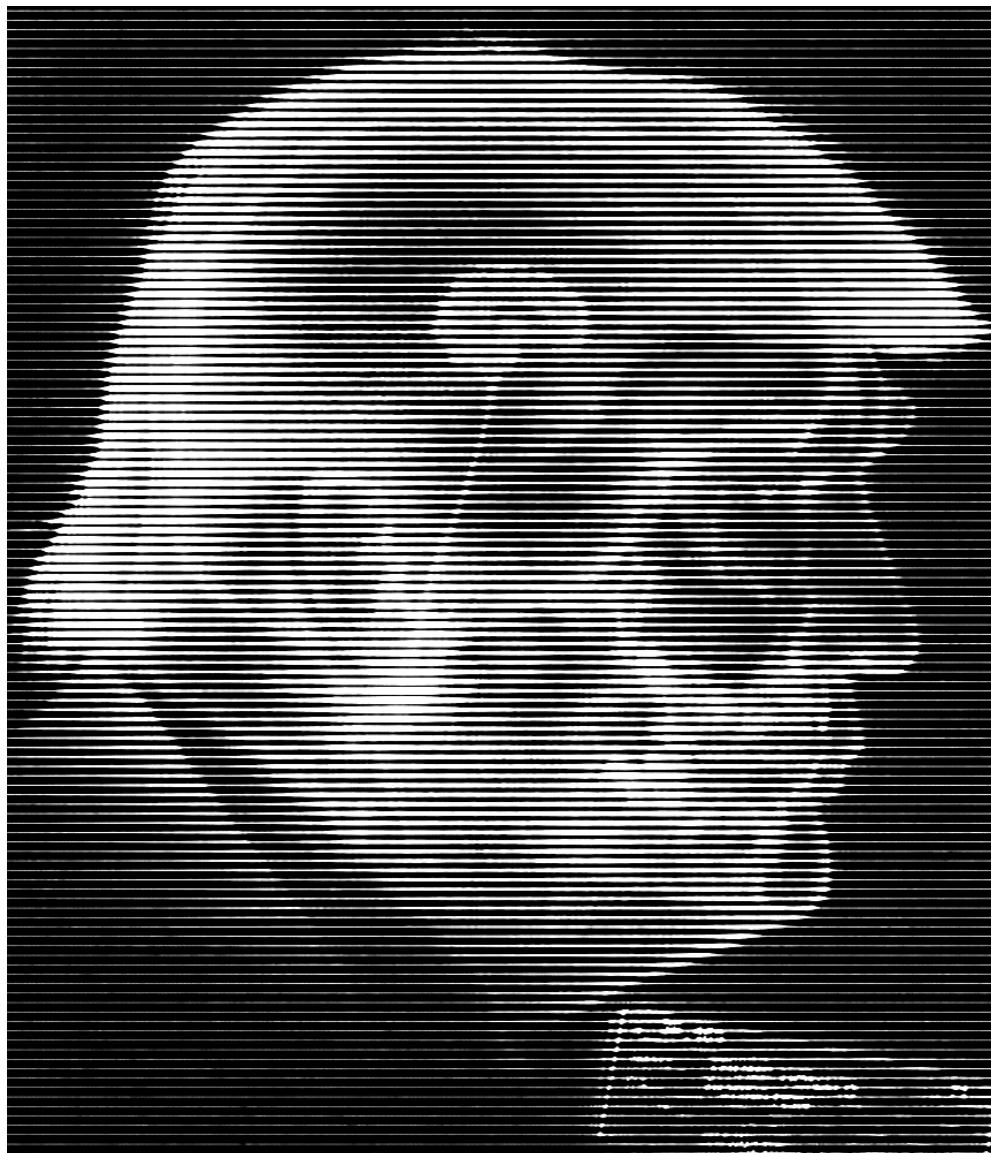
(قسمت b)

کد این قسمت از سوال در p7b.m قرار دارد. در ابتدا تصویر را می‌خوانم. سپس دیتابایپ آن را به double تبدیل می‌کنم. در ادامه با استفاده از fft2 و fftshift ضمن انتقال مرکز مختصات به مرکز صفحه، تبدیل فوریه‌ی تصویر را محاسبه می‌کنم. سپس Spectrum و Angle Phase را محاسبه می‌کنم و آن‌ها را نمایش می‌دهم. از آنجایی که پترن‌های تکراری در تصویر به صورت منظم تکرار شده‌اند، در Spectrum و تبدیل فوریه‌هم این پترن‌ها دیده می‌شوند. با ایجاد فیلتر Ideal Notch بر روی 8 نقطه‌ی Peak، و ضرب نظیر به نظیر فیلتر در

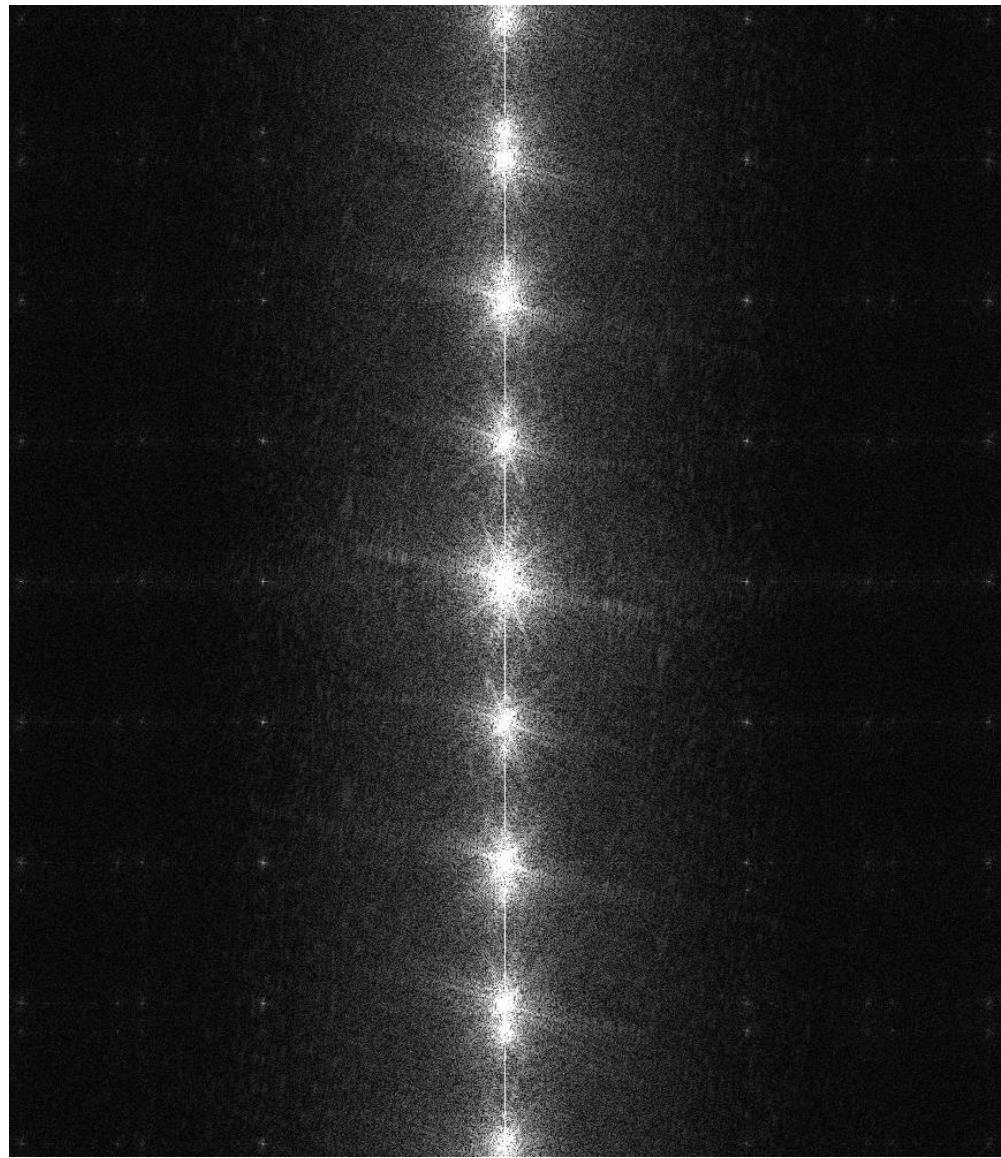
تبديل فوريه تصويراين پترنها را حذف می کنم. در ادامه با استفاده از تبدل فوريه معکوس تصوير اصلاح شده را به دست می آورم.

در زير خروجيها را مشاهده می کنيد(براي ديدن تصويرها در اندازه‌ي واقعی به p7 مراجعه کنيد):

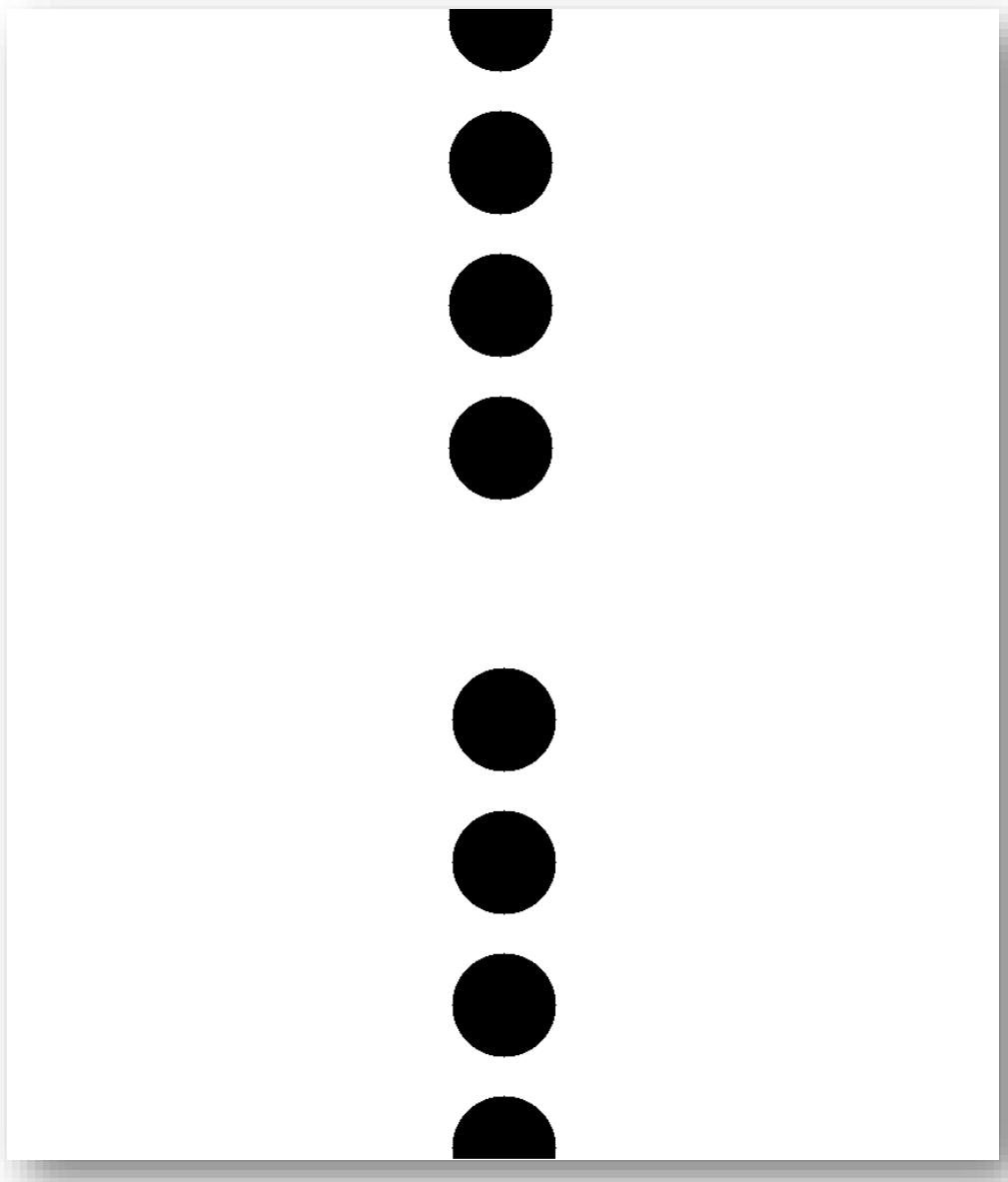
تصوير ورودی:



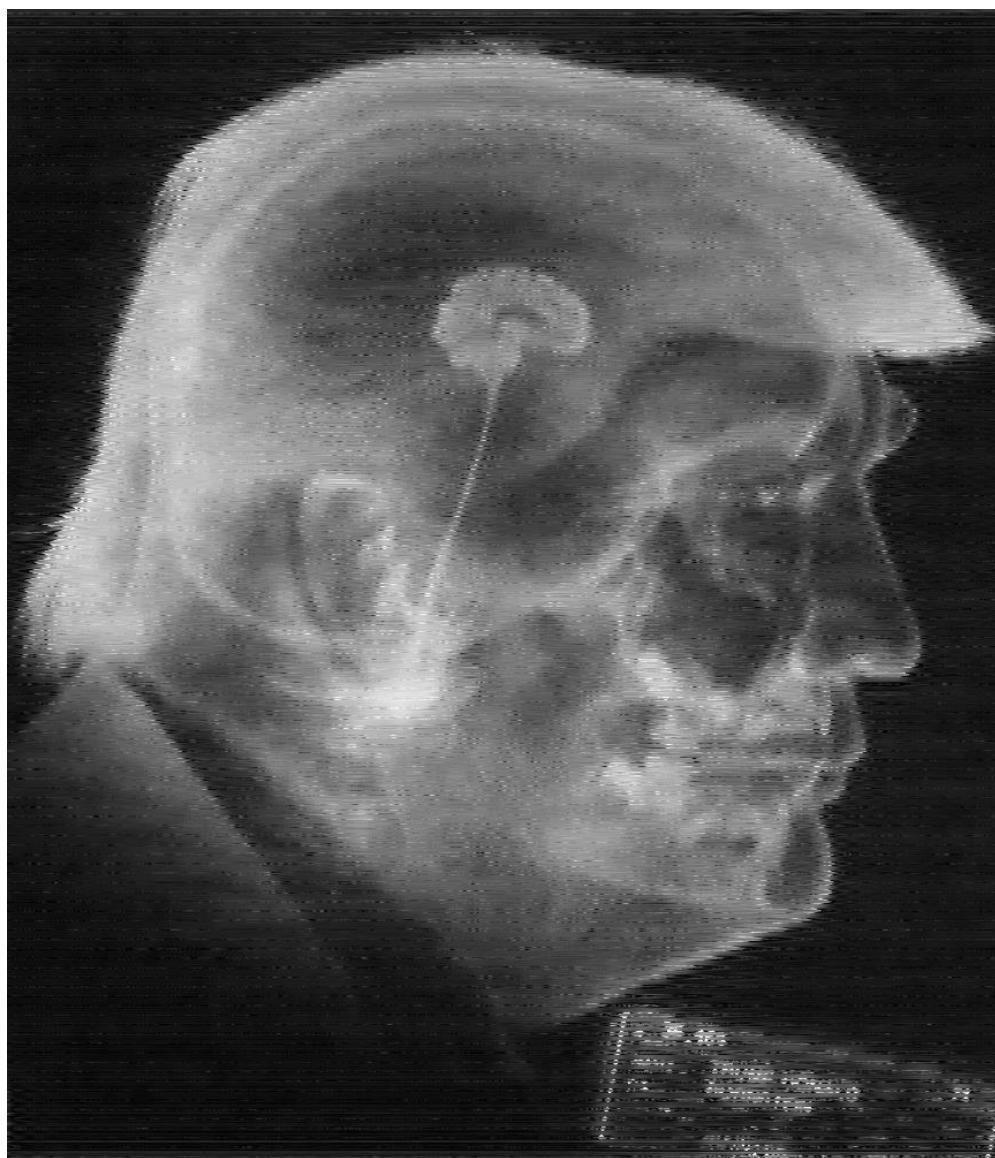
: (p7b_spectrum.png) تصوير ورودی Spectrum



: (p7b_filter.png) Pattern ایجاد شده برای حذف Filter



تصویر حاصل از اعمال فیلر : (p7b_filtered_image.png)



همین طور که مشاهده می‌شود، خروجی نسبت به ورودی، بسیار بهبود یافته است.

(قسمت C)

کد این قسمت از سوال در p7c.m قرار دارد. در ابتدا تصویر را می‌خوانم. سپس دیتاایپ آن را به double تبدیل می‌کنم. در ادامه با استفاده از fft2 و ifftshift ضمن انقال مرکز مختصات به مرکز صفحه، تبدیل فوریه‌ی تصویر را محاسبه می‌کنم. سپس Spectrum و Angle Phase را محاسبه می‌کنم و آن‌ها را نمایش می‌دهم. از آنجایی که پترن‌های تکراری در تصویر به صورت منظم تکرار شده‌اند، در Spectrum و تبدیل فوریه‌هم این

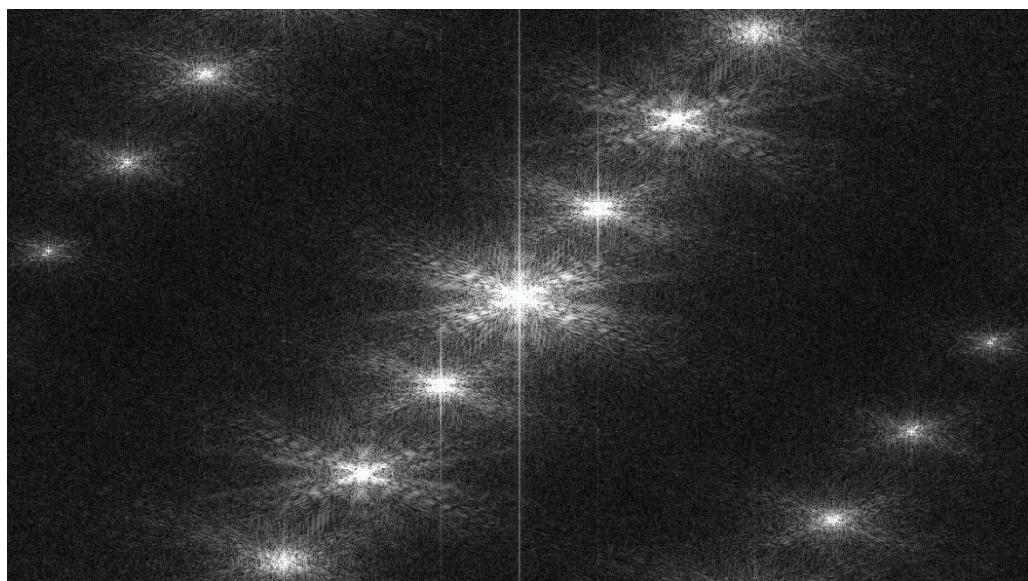
پترن‌ها دیده می‌شوند. با ایجاد فیلتر Ideal Notch بر روی 12 نقطه‌ی Peak، و ضرب نظیر به نظیر فیلتر در تبدیل فوریه تصویراین پترن‌ها را حذف می‌کنم. در ادامه با استفاده از تبدیل فوریه‌ی معکوس تصویر اصلاح شده را به دست می‌آورم.

در زیر خروجی‌ها را مشاهده می‌کنید(برای دیدن تصویرها در اندازه‌ی واقعی به p7 مراجعه کنید):

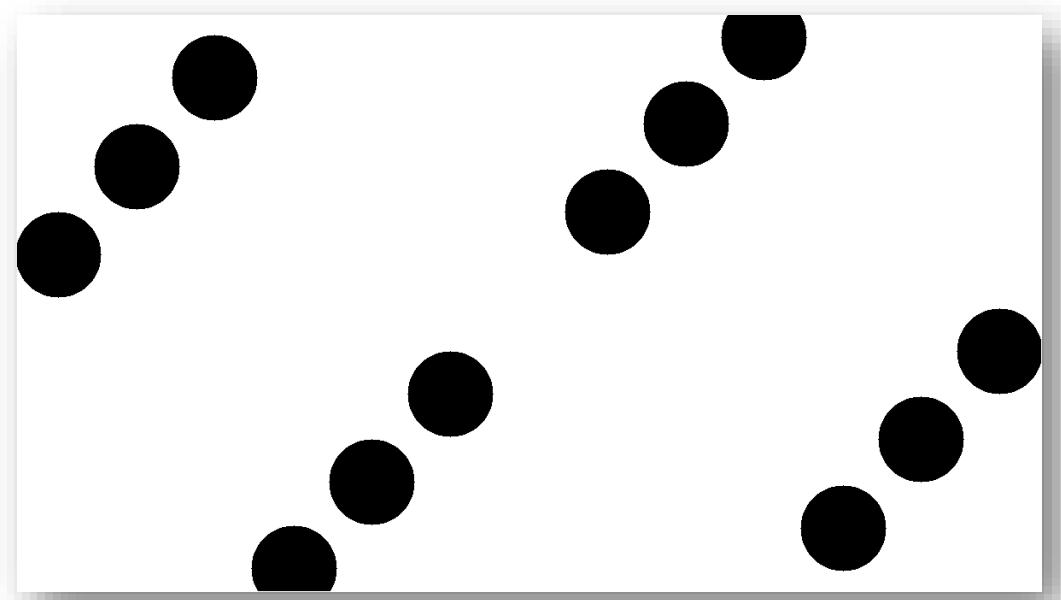
تصویر ورودی:



: (p7c_spectrum.png) Spectrum تصویر ورودی



: (p7c_filter.png) Pattern ایجاد شده برای حذف Filter



تصویر حاصل از اعمال فیلر (p7c_filtered_image.png)



همین طور که مشاهده می شود، خروجی نسبت به ورودی، بسیار بهبود یافته است.

تمرین ۸

قسمت a)

از آن جایی که تبدیل فوریه‌ی دو بعدی به صورت کلی، می‌توان آن را در مختصات قطبی نشان داد.

$$F(u, v) = |F(u, v)| e^{j\phi(u, v)}$$

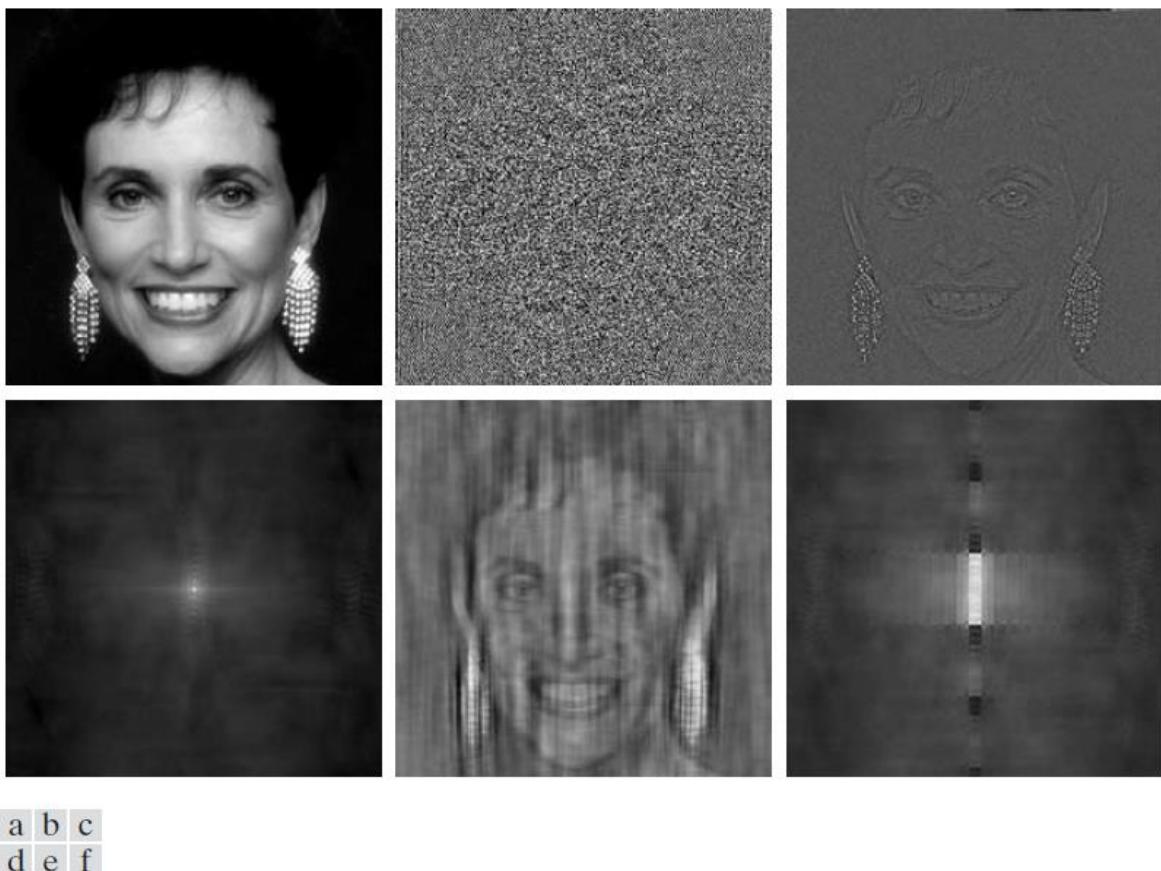
که برابر است با: Spectrum

$$|F(u, v)| = \left[R^2(u, v) + I^2(u, v) \right]^{1/2}$$

برابر می‌شود با: Phase Angle

$$\phi(u, v) = \arctan \left[\frac{I(u, v)}{R(u, v)} \right]$$

المنت‌های Spectrum، میزان Amplitude موج‌های سینوسی مختلف را مشخص می‌کند، هر در یک نقطه بیشتر باشد، میزان برجستگی بیشتر در موج سینوسی را نشان می‌دهد. جابه‌جایی موج‌های مختلف سینوسی را نسبت به مرکزشان نشان می‌دهد. در حالی که المنت‌ها Spectrum اطلاعاتی در مورد شدت روشنایی‌های تصویر می‌دهد، آرایه‌ای از زاویه‌های Phase Angle آرایه‌ای از زاویه‌های اطلاعاتی در مورد موقعیت قرارگیری اجزای مختلف تصویر می‌دهد. هم به اندازه‌ی Phase Angle مهم است ولی به صورت Visually کم تر قابل درک و بررسی است.



a b c
d e f

FIGURE 4.27 (a) Woman. (b) Phase angle. (c) Woman reconstructed using only the phase angle. (d) Woman reconstructed using only the spectrum. (e) Reconstruction using the phase angle corresponding to the woman and the spectrum corresponding to the rectangle in Fig. 4.24(a). (f) Reconstruction using the phase of the rectangle and the spectrum of the woman.

در تصویر بالا، بخشی از تصویر یک خانم را مشاهده می‌کنید. بخش ب Phase angle تصویر است، بخش c تصویر بازسازی شده با استفاده از phase angle است، تصویر d تصویر بازسازی شده با استفاده از Spectrum به تنهایی است. همین طور که مشاهده می‌شود، حاوی اطلاعات مهمی است ولی با بررسی با چشم اطلاعات مختصری می‌شود از آن برداشت کرد.

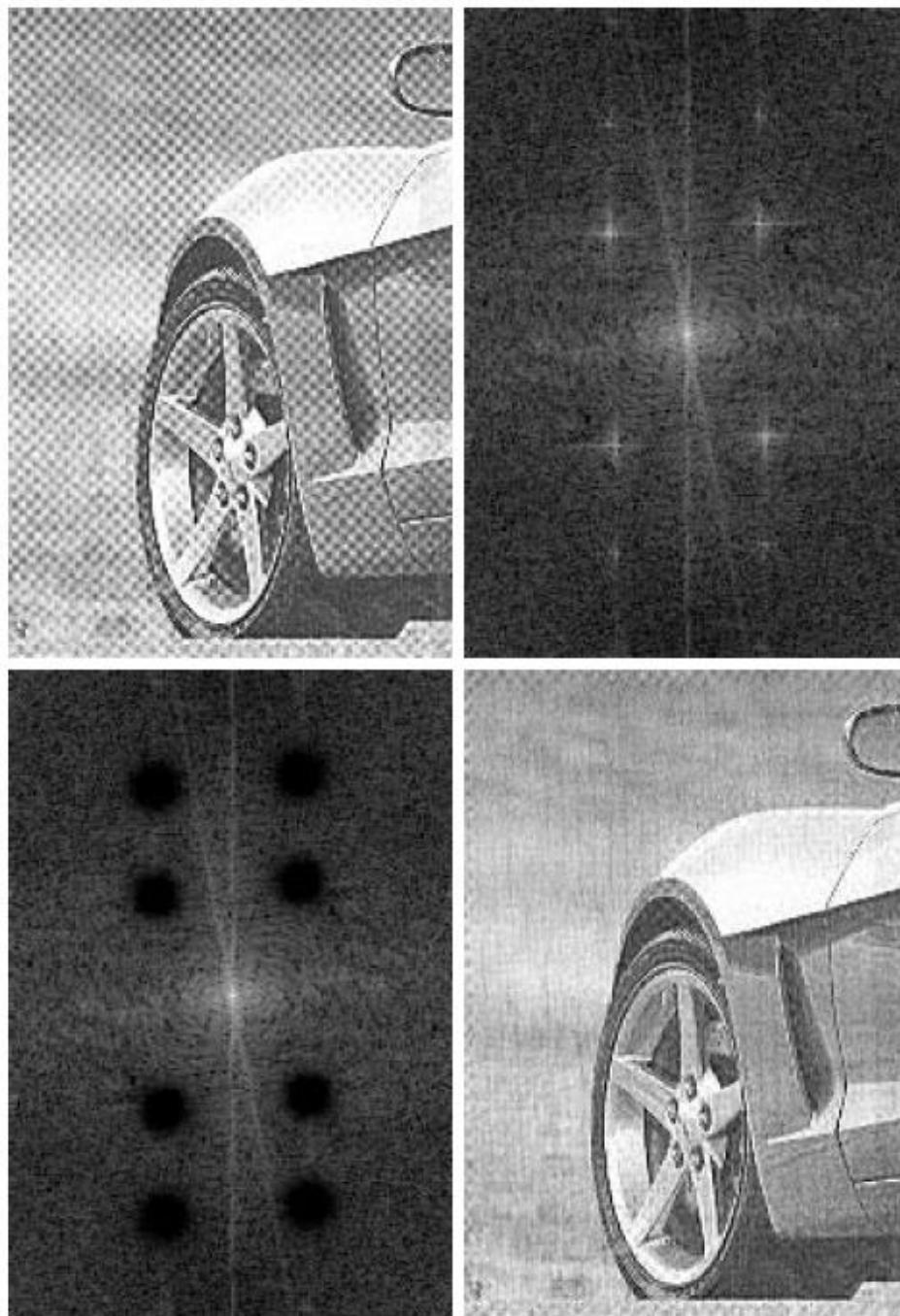
(b) قسمت

در بعضی از موردها فیلتر کردن در حوزه‌ی فرکانس بهتر است و در بعضی موارد فیلتر در حوزه‌ی Spatial عملکرد بهتری نشان می‌دهد.

در مواردی که در تصویر نویز متناوب ایجاد شده است و یا پترن هایی دارای ساختار و نظم در تصویر وجود دارد، بهترین گزینه استفاده از حوزه‌ی فرکانس است. مانند دو مثال زیر:

a
b
c
d

FIGURE 4.64
 (a) Sampled newspaper image showing a moiré pattern.
 (b) Spectrum.
 (c) Butterworth notch reject filter multiplied by the Fourier transform.
 (d) Filtered image.



در تصویر بالا، تصویر یک ماشین را مشاهده می کنید که دچار اثر Moire شده است و با استفاده از Notch فیلتر مشکلش برطرف شده است.

در تصویر زیر، تصویر کمربند زحل که با موجهای افقی متناوب ترکیب شده است را مشاهده می‌کنید که با استفاده از Vertical Notch برطرف شده است:

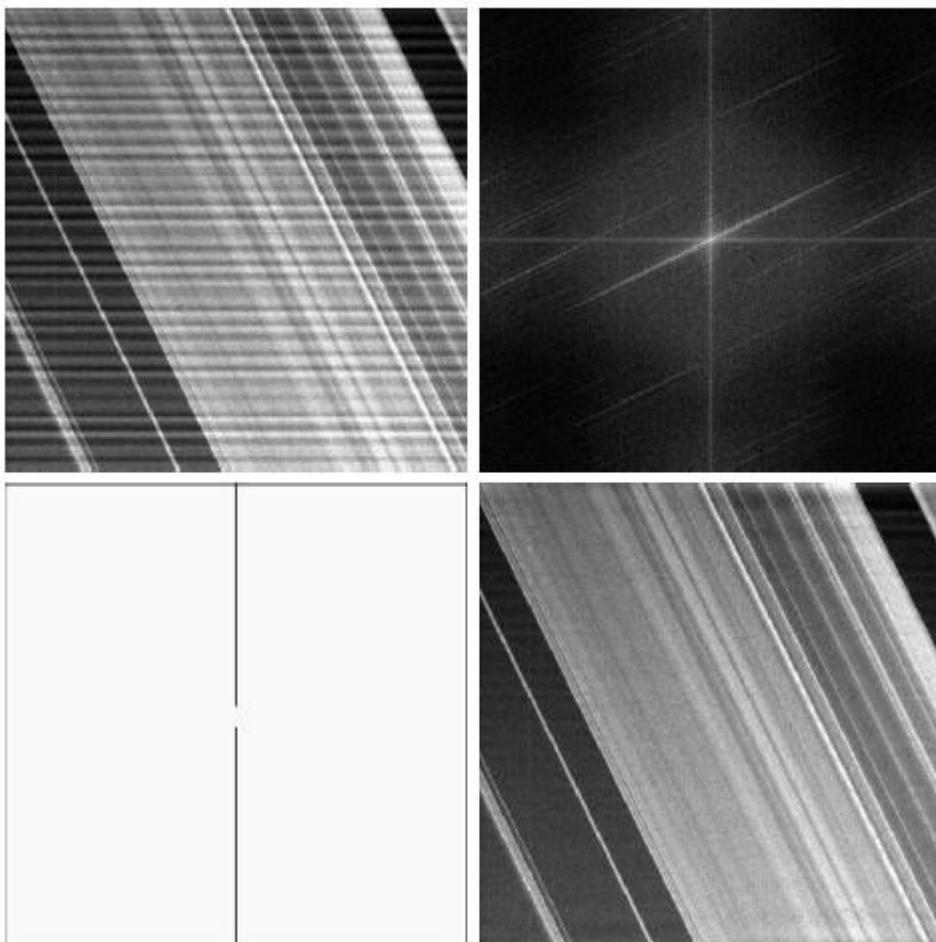


FIGURE 4.65
 (a) 674×674 image of the Saturn rings showing nearly periodic interference.
 (b) Spectrum: The bursts of energy in the vertical axis near the origin correspond to the interference pattern.
 (c) A vertical notch reject filter.
 (d) Result of filtering. The thin black border in (c) was added for clarity; it is not part of the data.
 (Original image courtesy of Dr. Robert A. West, NASA/JPL.)

همچنین در موردهای دیگر که هزینه‌ی محاسباتی در حوزه‌ی فرکانس نسبت به حوزه‌ی Spatial کمتر باشد، از فیلترها در حوزه‌ی فرکانس بهره می‌بریم.

() **قسمت ۵**
Low Pass فیلتر، فرکانس‌های کمتر از حد مشخصی را عبور می‌دهند. استفاده از آن برای حذف نویز موج بلور شدن تصویر می‌شود، همچنین با توجه به استفاده از نوع فیلتر می‌تواند موجب Ringing هم بشود.

4.7.3 Summary of Steps for Filtering in the Frequency Domain

The material in the previous two sections can be summarized as follows:

1. Given an input image $f(x, y)$ of size $M \times N$, obtain the padding parameters P and Q from Eqs. (4.6-31) and (4.6-32). Typically, we select $P = 2M$ and $Q = 2N$.
2. Form a padded image, $f_p(x, y)$, of size $P \times Q$ by appending the necessary number of zeros to $f(x, y)$.
3. Multiply $f_p(x, y)$ by $(-1)^{x+y}$ to center its transform.
4. Compute the DFT, $F(u, v)$, of the image from step 3.
5. Generate a real, symmetric filter function, $H(u, v)$, of size $P \times Q$ with center at coordinates $(P/2, Q/2)$.[†] Form the product $G(u, v) = H(u, v)F(u, v)$ using array multiplication; that is, $G(i, k) = H(i, k)F(i, k)$.
6. Obtain the processed image:

$$g_p(x, y) = \{\text{real}[\mathcal{F}^{-1}[G(u, v)]]\}(-1)^{x+y}$$

where the real part is selected in order to ignore parasitic complex components resulting from computational inaccuracies, and the subscript p indicates that we are dealing with padded arrays.

7. Obtain the final processed result, $g(x, y)$, by extracting the $M \times N$ region from the top, left quadrant of $g_p(x, y)$.

در صورتی که فیلتر را در حوزه‌ی فرکانس داشته باشیم به دین صورت عمل می‌کنیم:

برای انجام کانولوشن در حوزه‌ی فوریه، ابتدا فیلتر و تصویر را از سمت راست و پایین با استفاده از صفر Pad می‌کنیم به گونه‌ای که طول و عرض تصویر دوبرابر شود. این کار برای جلوگیری از Wraparound است.

در ادامه برای انتقال مختصات در حوزه‌ی فرکانس از گوشی سمت چپ بالا به مرکز صفحه، تمام پیکسل‌ها را در -1 به $x+1$ ضرب می‌کنیم. سپس تبدیل فوریه‌ی دو بعدی تصویر را محاسبه می‌کنیم. سپس تبدیل فوریه و فیلتر در حوزه‌ی فرکانس را به صورت نظیر در نظریه ضرب می‌کنیم. بعد از آن حاصل ضرب را با استفاده از تبدیل معکوس فوریه به دست می‌آوریم و قسمت Real آن را نگه می‌داریم و با ضرب در -1 به $x+1$ حذف Padding تصویر خروجی و نهایی را به دست می‌آوریم.

در صورتی که فیلتر در حوزه‌ی Spatial داده شده باشد:

مانند بالا عمل می‌کنیم با این تفاوت که تصویر و فیلتر را به صفر به گونه‌ای Pad می‌کنیم که طول و عرض تصویر و فیلتر برابر با مجموع طول فیلتر و تصویر و مجموع عرض تصویر و فیلتر شود. تصویر را از سمت راست و پایین pad می‌کنیم و فیلتر را از ۴ جهت Pad می‌کنیم. با استفاده از تبدیل فوریه تبدیل فوریه، فیلتر را به حوزه‌ی فرکانس می‌بریم. در ادامه مانند آنچه گفته شد عمل می‌کنیم.

(e) قسمت

وقتی که تصویر را بخواهیم از حوزه‌ی Spatial به حوزه‌ی فرکانس ببریم و با استفاده از Frequency Domain عمل کانولوشن انجام بدهیم، تصویر را سمت راست و پایین Pad می‌کنیم زیرا مختصات صفر و صفر در سمت چپ و بالای تصویر است. و فیلتر را از چهار جهت Pad می‌کنیم زیرا مختصات صفر و صفر فیلتر در مرکز است. در صورتی که برای هر کدام از دو مورد بالا از روش دیگر استفاده کنیم، مانند ای است که تصویر و یا فیلتر را شیفت داده‌ایم و نتیجه‌ی مطلوب به دست نمی‌آید.