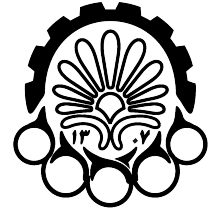


به نام او



دانشگاه صنعتی امیرکبیر  
دانشکده مهندسی پزشکی  
گروه بیوالکتریک



دانشگاه صنعتی امیرکبیر  
( پلی تکنیک تهران )

## پردازش تصویر

تمرین شماره ۴  
حوزه فرکانسی تصویر، نگاهی بیشتر به شیوه نمایش تصویر و  
آشنایی با فرمت تصویری DICOM

زمان ارسال: ۰۰/۰۲/۰۴

مهلت تحویل: ۰۰/۰۲/۲۵

استاد درس:  
دکتر حامد آذرنوش

تدریس یاران:

حسین قاسم دامغانی

حمیدرضا ابوتی مهریزی

یلدا ظفری قدیم

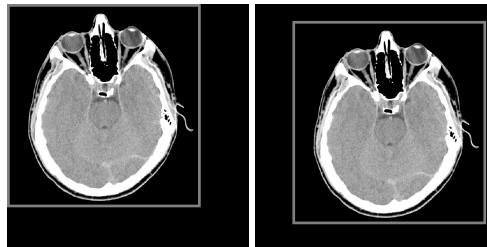
نیم سال بهار ۹۹-۰۰

## ۱ (تشریحی) ۱۰%

روابط مربوط به محاسبه مشتق اول و دوم در تصاویر دو بعدی دیجیتال را نوشته و برابر هم‌ارز آن‌ها را در حوزه فرکانس به دست آورید.

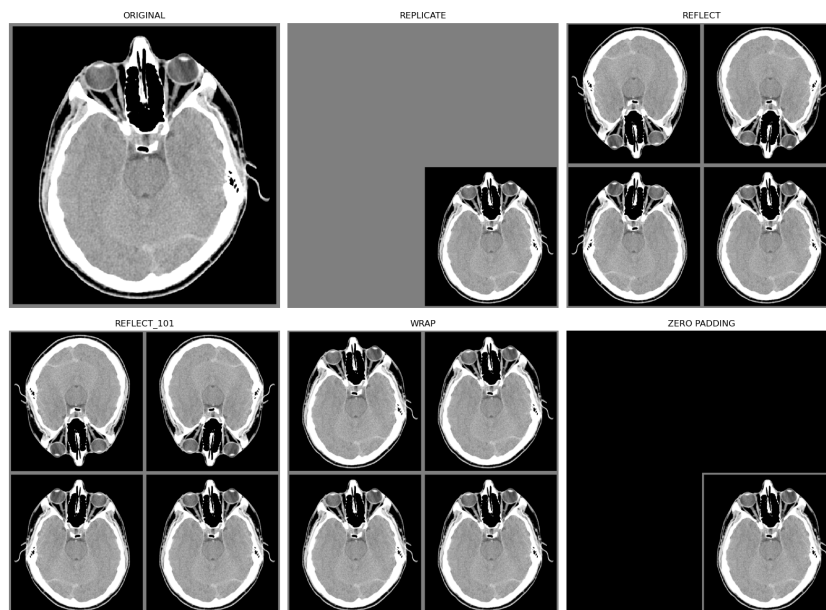
## ۲ (تشریحی) ۲۰%

آ) ضرورت استفاده از padding قبل از اعمال فیلترهای فرکانسی در کلاس بیان شده است. با فرض ثابت بودن تعداد پیکسل‌های اضافه‌شده در مرحله padding در هر محور، با روابط ریاضی بیان کنید آیا نوع آرایش مکانی این پیکسل‌ها تأثیری در نتیجه نهایی خواهد داشت یا خیر. (۵۰٪)



شکل ۱: padding با دو نوع آرایش مکانی مختلف

ب) اگر بخواهیم یک فیلتر فرکانسی را عیناً در حوزه مکان اعمال کنیم، در هنگام convolution (یا cross-correlation) از کدام نوع padding باید استفاده شود؟ با ذکر روابط ریاضی، علت آن را بیان کنید. (۵۰٪)



شکل ۲: مثال از انواع padding در کنار تصویر اصلی

## توضیحاتی در مورد فرمت تصویری DICOM

نوع فایل (Digital Imaging and Communications in Medicine (DICOM با پسوند فایل dcm. نوعی از فایل تصویری است که در تصویربرداری پزشکی استفاده می‌شود. برای خواندن این تصاویر در پایتون، باید بسته pydicom را نصب نمایید. راهنمای نصب آن در [این لینک](#) موجود است اما به شکل خلاصه همانند نصب numpy، دستور

```
pip install pydicom
```

باید اجرا شود و به راحتی روی دستگاه شما نصب می‌شود.

با خواندن این تصاویر به اطلاعات مفیدی می‌توانید دسترسی پیدا کنید؛ برای مثال خود آرایه تصویر، تاریخ و زمان تصویربرداری، تعداد بیت‌های اختصاص داده شده (برای هر پیکسل)، تعداد بیت‌های ذخیره شده (برای هر پیکسل)<sup>۱</sup>، بخش از بدن که تصویربرداری شده، دسته‌بندی روش تصویربرداری (MRI, CT, etc.)، سن بیمار، کد شناسایی بیمار، جنسیت بیمار و اطلاعات لازم دیگر.

برای دسترسی به این اطلاعات، بعد از خواندن و ذخیره کردن فایل دایکام در یک متغیری مثلاً به نام ds، کافی است attribute مورد نظر را بخوانید.

```
>>> ds.Modality
'CT'
```

از این روش برای دستیابی به یک مورد خاص استفاده می‌شود، اما برای دیدن اطلاعات کلی فایل می‌توانید خود ds را چاپ کنید تا به فرم استاندارد خودش، جدولی از اطلاعات فایل را برایتان نمایش دهد.

```
>>> print(ds)
```

```
Dataset.file_meta -----
```

```
(0002, 0000) File Meta Information Group Length  UL: 194
```

```
(0002, 0001) File Meta Information Version        OB: b'\x00\x01'
```

```
(0002, 0002) Media Storage SOP Class UID         UI: CT Image Storage
```

```
(0002, 0003) Media Storage SOP Instance UID      UI: 1.3.6.1.4.1.14519.5.2.1.7777.9002.1798755185374
```

```
(0002, 0010) Transfer Syntax UID                 UI: Implicit VR Little Endian
```

```
(0002, 0012) Implementation Class UID            UI: 1.2.40.0.13.1.1.1
```

```
(0002, 0013) Implementation Version Name         SH: 'dcm4che-1.4.31'
```

```
.
```

```
.
```

```
.
```

<sup>۱</sup> فرق بیت‌های اختصاص داده با بیت‌های ذخیره شده این است که مثلاً برای تصاویر ۶ بیتی، ساختار آرایه‌ای مختص آن در نظر گرفته نشده، پس آن را درون یک ساختار ۸ بیتی ذخیره می‌کنند. ۸ تعداد بیت‌های اختصاص داده شده است و ۶ تعداد بیت‌های ذخیره شده است.

## توضیحاتی در مورد کران تصویر

کران یک تصویر خاکستری، حد فاصل بین مقدار منسوب به سیاه و مقدار منسوب به سفید است. برای مثال در یک تصویر ۸ بیتی، مقدار ۰ معادل سیاه و مقدار ۲۵۵ معادل سفید است. اعداد بین این‌ها به شکل خطی در طیف خاکستری قرار می‌گیرند. پس بنابراین کران یک تصویر، لزوماً بیشترین و کمترین مقدار آرایه تصویر نیست؛ برای مثال مقادیر یک تصویر ۸ بیتی می‌تواند بین ۳۰ تا ۲۰۰ باشد، اما همچنان کران آن  $[0, 255]$  است.

هنگامی که یک نگاشت (عملیات) را به دادگان تصویر اعمال می‌کنیم، کران آن هم نگاشت می‌شود. برای مثال وقتی از مقادیر تصویر ۸ بیتی جذر می‌گیریم، کران تصویر  $[0, \sqrt{255}]$  می‌شود. از آنجایی که تمایل داریم کران تصویر را ثابت نگه داریم تا بتوانیم تصویر را قبل و بعد از عملیات‌ها مقایسه کنیم، با عملیات‌های جابه‌جایی و ضرب کردن کران تصویر را به حالت اولیه برمی‌گردانیم، برای مثال در اینجا کافی است پس از جذرگیری، مقادیر را در  $\sqrt{255}$  ضرب کنیم تا کران تصویر دوباره  $[0, 255]$  شود. با اینکه کران تصویر ثابت مانده، اما به مانند مثال‌های مبحث تبدیلات شدت روشنایی، تصویر کاملاً تغییر کرده. تعیین کران تصویر برای نمایش آن ضروری است (vmin, vmax در plt.imshow). بدین منظور به شکل ساده، کافی است همان عملیات‌هایی که روی پیکسل‌ها انجام می‌دهید را روی بازه کران تصویر نیز انجام دهید، اگر برای هر پیکسل نیز کران متفاوتی حساب شد، کران کل تصویر از اجتماع کران‌ها بدست می‌آید (برای مثال اگر جذر نیمه راست تصویر ۸ بیتی را حساب کنید و نصف قرینه نیمه چپ تصویر را حساب کنید، کران تصویر  $[-\frac{255}{\sqrt{2}}, \sqrt{255}] \cup [-\frac{255}{\sqrt{2}}, 0]$  می‌شود؛ یا اینکه باید هر تکه را جدا با کران خود نمایش بدهیم، که معمولاً مطلوب نیست).

## ۳ ۷۰%

در این مسأله شما یک کد پایه با نام 3.py در اختیار دارید که یا باید به فراخور نیاز بعضی جاهایش را در کد پر کنید، تغییر دهید، یا به سوالاتی از آن در گزارش پاسخ دهید که به شرح زیر است:

۱. در خط ۶ و ۸ باید به ترتیب از فایل‌های Thoracic CT 1.dcm و Thoracic CT 2.dcm تصاویر را بخوانید. برای راهنمایی به این لینک و این لینک یا باقی آموزش‌های این صفحه مراجعه کنید. (۴٪)

۲. با مراجعه به اطلاعات فایل‌های دایکام، در خط ۱۰ بیان کنید چرا با وجود نوع داده (dtype) uint16 برای تصاویر، اما آن‌ها را ۱۲ بیتی در نظر گرفتیم. (۴٪)

۳. در خط ۱۱، کد لازم را برای محاسبه  $L - 1$  این تصاویر ۱۲ بیتی بنویسید. ( $L$  نماد تعداد سطوح شدت روشنایی است). (۴٪)

۴. در خط ۱۴ بیان کنید mask چه آرایه‌ای است و با توجه به مفاهیم این بخش از درس، به چه منظوری در ادامه کد از آن استفاده شده. (۱۲٪)

۵. از خط ۱۹ تا ۳۹: نتیجه را مشاهده کنید و گزینه مناسب را از داخل پرانتز برای هر تصویر انتخاب کنید و باقی را حذف کنید.<sup>۴</sup> منظور از clipped این است که مقادیر داده به یک کران انتخابی محدود شده‌اند، مانند مثال‌های کتاب و اسلاید درسی که برای نمایش برخی تصاویر، مقادیر منفی به صفر نگاشته می‌شوند؛ و منظور از stretched این است که کران بالا و پایین برای تعیین سفید و سیاه، وسیع‌تر از کرانی است که باید انتخاب می‌شد. (۱۶٪)

۶. در خط‌های ۴۲، ۴۴، ۴۷ و ۴۹، اندازه و فاز IMG1 و IMG2 را محاسبه کنید. (۸٪)

۷. در خط ۵۲، با روابط ریاضی توضیح دهید چرا بیشترین شدت روشنایی ممکن برای اندازه طیف فوریه این تصاویر، از ضرب تعداد کل پیکسل‌ها در  $L - 1$  بدست آمده. (۱۲٪)

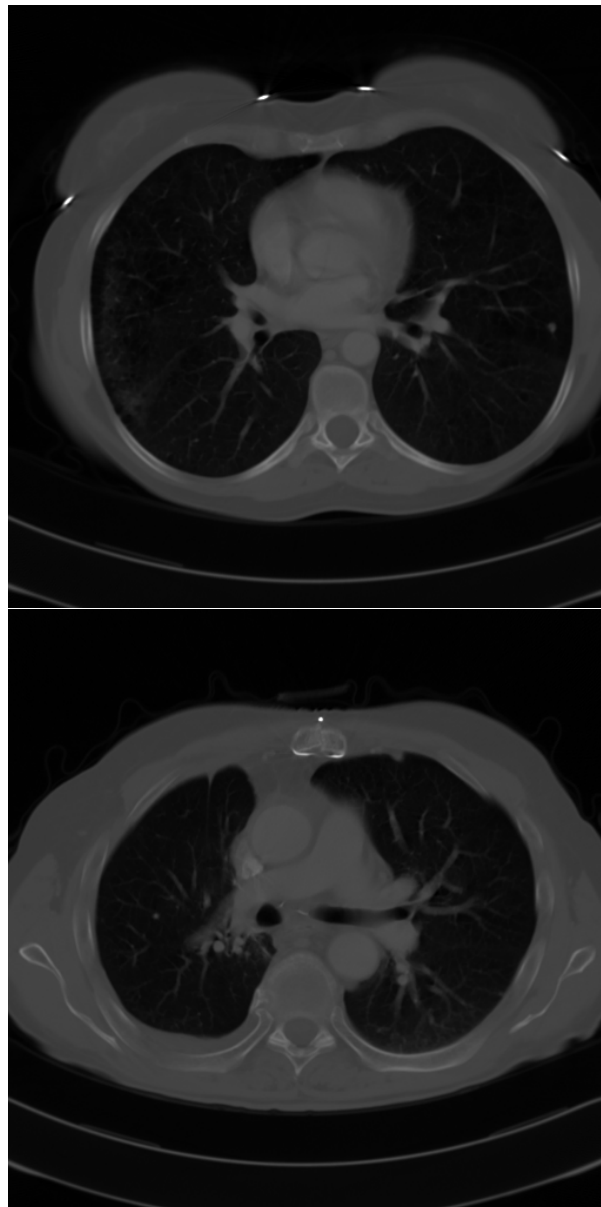
۸. از خط ۵۶ تا ۸۹: نتیجه را مشاهده کنید و با استفاده از مفاهیم مبحث تبدیل شدت روشنایی بیان کنید چرا تبدیل لگاریتمی در نمایش اندازه تبدیل فوریه تصویر به ما کمک می‌کند.<sup>۵</sup> همچنین فاز تبدیل فوریه تصاویر هم برای مشاهده آورده شده؛ بیان کنید چه اطلاعات دیداری‌ای از تصویر در فاز تبدیل فوریه هر تصویر موجود است. (۱۲٪)

۹. خط ۹۲: اگر مزدوج یک عدد مختلط، بخش موهومی آن را قرینه می‌کند؛ قرینه مزدوج یک عدد مختلط چه کار مشابهی انجام می‌دهد؟ (۴٪)

۱۰. از خط ۱۰۱ تا ۱۲۳: نتیجه را مشاهده کنید و با توضیح روند بیان کنید عملیات‌هایی که انجام دادیم، چه تغییراتی روی تصاویر اعمال کردند و چرا. (۲۴٪)

<sup>۴</sup> از خط ۴۰ تا ۱۲۲ را کامنت کنید تا با خطا مواجه نشوید و همچنین نمودارتان نمایش داده شود. برای کامنت کردن چندین خط کد، همه آن‌ها را انتخاب کنید و از میانبر  $\text{Ctrl} + /$  یا  $\text{Shift} + \text{Alt} + \text{A}$  استفاده کنید، میانبر این دو نوع کامنت کردن (خطی و بلوکی) ممکن است در محیطی غیر از Visual Studio Code متفاوت باشد.

<sup>۵</sup> نیازی به درک کدهای نوشته شده برای عنوان‌ها نیست، برای مطالعه علاقه‌مندان.



[1]

**نحوه‌ی ارسال:** فایل گزارش را به همراه کدهای نوشته شده در قالب یک فایل فشرده zip به اسم HW4\_Num باشد که Num شماره‌ی دانشجویی شما هست، مانند HW4\_9433001. **فقط از طریق سامانه مدیریت یادگیری Moodle ارسال بفرمایید.** موفق باشید.

مراجع

- [1] Albertina, B., Watson M. Holback C. Jarosz R. Kirk S. Lee Y. ... Lemmerman J. Radiology Data from The Cancer Genome Atlas Lung Adenocarcinoma [TCGA-LUAD] collection, 2016.