

دانشکده مهندسی پزشکی

دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده مهندسی پزشکی  
گروه بیوالکتریک



درس پردازش تصویر

## تمرین ۱

### عملیات‌های هندسی و شدتی

استاد درس:

دکتر حامد آذرنوش

تدریس‌یاران ارشد:

امیرمحمد مولایی، میلاد طهرانی، سیدمعین طیبی، پیمان باقری

نیم‌سال اول سال تحصیلی ۱۴۰۲-۱۴۰۳

## بخش تشریحی

### سوال ۱ (10%)

تصویر زیر را با نام  $f(x, y)$  در نظر بگیرید. تابع درونیابی دوخطی (Bilinear) توصیف کننده‌ی مقدار روشنایی تصویر در نقطه‌های بین پیکسل‌های با مختصات  $(1, 2)$  و  $(2, 3)$  را بیابید. توجه کنید که محورهای مختصات را بر اساس جهت‌های تعریف شده در درس و کتاب گنزالز در نظر بگیرید.

0	2	2	7
5	7	2	2
7	1	7	0
4	5	3	7

### سوال ۲ (10%)

الف) تمام صفحات بیتی تصویر زیر را از لایه پرارزش به لایه کم ارزش به دست آورید. (50%)

0	1	8	6
2	2	1	1
1	15	14	12
3	6	9	10

ب) اگر بخواهیم حجم فایل این تصویر را کم کنیم طوری که کیفیت آن کمترین کاهش را داشته باشد، چه رویکردی را پیشنهاد می‌کنید؟ راهنمایی: به بخش الف دقت کنید. (50%)

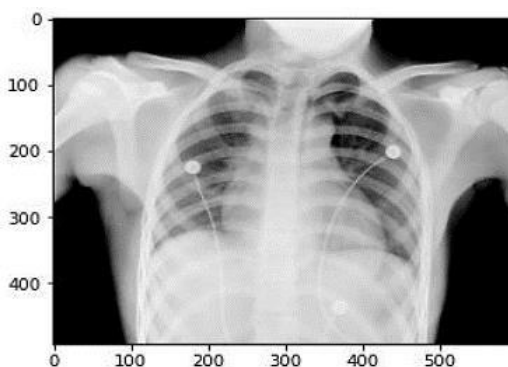
### سوال ۳ (5%)

عملیات Contrast Stretching را روی تصویر ۸ بیتی که شدت پیکسل‌های آن در بازه ی  $[87, x]$  بوده اعمال کرده ایم و رابطه‌ی  $S(r) = 17r - 147$  بین تصویر جدید و قدیم برقرار است. مقدار  $x$  را بیابید.

## بخش کدنویسی

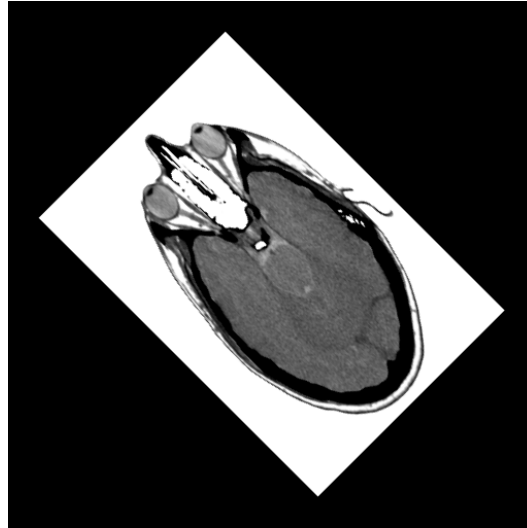
### ویژگی‌های نمایش استاندارد خروجی‌ها برای تمام سوال‌های کدنویسی:

۱. در تابع `imshow` مربوط به آن، آرگومان `cmap` متناسب با تصویر و هدف مساله انتخاب شود. در درس حاضر، تقریباً در تمام موارد همان `'gray'` انتخاب می‌شود. دقت شود که این مورد مستقل از رنگی (سه بعدی) یا خاکستری بودن آن است.
  ۲. در تابع `imshow` آن، آرگومان‌های `vmin` و `vmax` با توجه به کران تصویر که به تعداد بیت آن وابسته است مقدارهای مناسب بگیرد. مثلاً برای یک تصویر ۸ بیتی، این دو آرگومان به ترتیب برابر ۰ و ۲۵۵ هستند.
  ۳. پنجره‌ی اصلی نمایش و تک تک تصاویر نمایش داده شده عنوان (`title`) مناسب (نه لزوماً خیلی کامل و طولانی) داشته باشند. آرگومان‌های `color` و `backgroundcolor` آن نیز مقدار دهی مناسبی شوند تا عنوان چشم نواز باشد.
  ۴. ویژگی `axis` از `plt` برای این تصاویر خاموش باشد تا در کنار تصاویر محور اعداد را نبینیم.
  ۵. ابعاد پنجره و تصاویر مناسب باشد.
- مثال: تصویر سمت راست به دلیل داشتن محور اعداد و نداشتن عنوان استاندارد نیست ولی تصویر سمت راست استاندارد است.



#### سوال ۴ (20%)

الف) تصویر transformed.png را بخوانید و به صورت استاندارد نمایش دهید.. ماتریس تبدیل هندسی‌ای که روی آن اعمال شده است را حدس بزنید و آنها را نام ببرید. (تصویر در حالت اولیه تمام نقاط را پوشانده بود و مستطیل سیاه رنگی نداشت.) (25%)



Transformed.png

ب) با استفاده از حدس خود و تبدیل‌های هندسی، تصویر را به حالت اولیه خود برگردانید. (40%)

راهنمایی: می‌توانید به صفحه‌ی [مبحث مرتبط در مستندات OpenCV](#) رجوع کنید.

پ) تصویر خروجی به دست آمده از مرحله قبل را به اندازه ۱۰۰ پیکسل در هر جهت Padding اضافه کنید. سپس تصویر را با استفاده از تبدیل هندسی جابجایی (Translation) به اندازه ۸۰ پیکسل به پایین چپ منتقل کنید. (35%)

## سوال ۵ (20%)

الف) تصویر spine.tif را به صورت خاکستری بخوانید (با آرگومان `cv2.IMREAD_GRAYSCALE` برای تابع `cv2.imread`). همچنین نوع دادگان و ابعاد آن را چاپ کنید. سپس نوع دادگان آن را به `np.float64` تغییر داده و بعد آن را به صورت استاندارد نمایش دهید. (10%)



دقت: در نوشتن توابع زیر، اگر پارامتر یا مقداری به جز ورودی‌های تابع برای محاسبات لازم است، آن را از خود تصویر استخراج کنید و مقدار ثابتی را در توابعتان قرار ندهید ( **hard code** نکنید). طوری که اگر ابعاد یا تعداد بیت تصویر تغییر کنند، توابع همچنان کار کنند. مثلاً اگر برای یک عملیات شدتی، لازم است تعداد بیت‌های تصویر را داشته باشیم، آن را به عنوان یک ورودی جدا برای تابع قرار ندهید و به جای این کار، درون تابع، تعداد بیت‌های تصویر را از تصویر ورودی تابع استخراج کنید.

ب) یک تابع به نام `log_transform` بنویسید که یک تصویر به عنوان ورودی دریافت کند و روی آن تبدیل لگاریتمی شدت اعمال کند خروجی تبدیل به صورت یک آرایه از جنس مناسب خروجی داده شود ولی نیازی به چاپ نیست. لگاریتم این سوال را لگاریتم طبیعی (با پایه‌ی عدد نپر) در نظر بگیرید. پس از نوشتن تابع، سپس تصویر اولیه را به این تابع بدهید و خروجی اش را نمایش دهید. (10%)

پ) یک تابع به نام `contrast_stretching` بنویسید که یک تصویر به عنوان ورودی دریافت کند و روی آن تبدیل کشیدگی کنتراست را اعمال کند. سپس تصویر اولیه را به این تابع بدهید و خروجی را نمایش دهید. (30%)

ت) یک تابع `power_law_transform` بنویسید که یک تصویر و یک مقدار گاما به عنوان ورودی دریافت کند و روی آن تبدیل توانی با توان گامای داده شده اعمال کند. سپس تصویر اولیه را به ترتیب با گاماهای ۰.۵ و ۳ به این تابع بدهید و خروجی ها را در کنار هم نمایش دهید. (15%)

ث) در نهایت تصویر اولیه و هیستوگرام آن و همچنین هر ۴ خروجی توابع و هیستوگرامشان را در یک پنجره نمایش بدهید. برای این کار پنجره را به ۵ ستون و ۲ ردیف تقسیم کنید که ردیف بالایی برای خود عکس و ردیف پایینی برای هیستوگرام عکس باشد. هنگام رسم هیستوگرام، هر ۴ شدت در یک دسته قرار گیرند. برای تنظیم کردن بزرگی و کوچکی پنجره از `figsize` با مقدار مناسب استفاده کنید. (15%)

ج) تصاویر بخش ث را با هم مقایسه کرده و مشاهدات و نتیجه گیری خود را بنویسید. (20%)

## سوال ۶ (35%)

الف) تصویر ۸ بیتی `spineXray.tif` را به صورت خاکستری بخوانید و آن را نمایش دهید (با آرگومان `cv2.IMREAD_GRAYSCALE` برای تابع `cv2.imread`). همچنین نوع داده‌گان و ابعاد آن را چاپ کنید. (5%)



دقت: در نوشتن توابع زیر، اگر پارامتر یا مقداری به جز ورودی‌های تابع برای محاسبات لازم است، آن را از خود تصویر استخراج کنید و مقدار ثابتی را در توابعتان قرار ندهید ( **hard code** نکنید). طوری که اگر ابعاد یا تعداد بیت تصویر تغییر کنند، توابع همچنان کار کنند. مثلاً اگر برای یک عملیات شدتی، لازم است تعداد بیت‌های تصویر را داشته باشیم، آن را به عنوان یک ورودی جدا برای تابع قرار ندهید و به جای این کار، درون تابع، تعداد بیت‌های تصویر را از تصویر ورودی تابع استخراج کنید.

ب) با استفاده از تابع `cv2.equalizeHist()` عملیات یکنواخت سازی هیستوگرام را روی تصویر انجام دهید و تصویر حاصل را ذخیره کنید. (5%)

پ) تابعی با اسم دلخواه بنویسید که عمل یکنواخت سازی هیستوگرام را بدون استفاده از تابع آماده‌ی `cv2.equalizeHist()` اجرا کند. ورودی این تابع، یک تصویر و خروجی آن، تصویر (آرایه‌ی) حاصل از یکنواخت سازی هیستوگرام است. (25%)

ت) با رجوع به مستندات **OpenCV** و مطالعه درباره‌ی نحوه‌ی کار با فیلتر **CLAHE**، از این فیلتر (با آرگومان‌های مناسب) استفاده کرده و خروجی را به صورت استاندارد نمایش دهید. درباره‌ی ورودی‌های فیلتر **CLAHE** توضیح دهید. (20%)

ث) در این بخش قصد داریم انواع داده‌های موجود را نمایش دهیم. نحوه‌ی نمایش خروجی به صورت یک پنجره‌ی  $3 \times 4$  است که ردیف اول مربوط به تصاویر (تصویر اولیه و خروجی هر تابع بخش‌های بالا)، ردیف دوم مربوط به هیستوگرام تصویر متناظر از ردیف بالا و ردیف سوم مربوط به نمودار توزیع تجمعی هیستوگرام است. ترتیب تصاویر از چپ: تصویر اولیه، خروجی بخش ب، خروجی تابع بخش پ، خروجی فیلتر بخش ت. ضمناً در رسم هیستوگرام‌ها، هر ۴ شدت در یک دسته قرار گیرند. (10%)

ج) خروجی بخش ث را تحلیل کنید. (10%)

چ) حال قصد داریم بخش ث و ج این سوال را برای تصویر `file.dcm` که یک تصویر با فرمت **DICOM** است تکرار کنیم. (به توضیحات صفحه بعد مراجعه کنید). برای این بخش نیز یک پنجره‌ی  $3 \times 4$  جدید تحویل دهید و تفاوت رابطه‌ی خروجی‌ها با خروجی‌های نمایش داده شده در بخش ث را توضیح دهید. (25%)



## فرمت فایل DICOM

فایل DICOM (مخفف Digital Imaging and Communication in Medicine) نوعی از فایل تصویری که در تصویربرداری‌های پزشکی استفاده می‌شود. برای خواندن این تصاویر در پایتون، باید ماژول pydicom را نصب نمایید که این موضوع در قالب گوگل کولب برای شما انجام شده است.

با خواندن این فایل‌ها می‌توان به اطلاعات جزئی‌تر و جامع‌تری نسبت به فرمت‌های معمول دسترسی پیدا کرد؛ مثلاً‌هایی از اطلاعات موجود در این فایل‌ها: خود آرایه‌ی تصویر، تاریخ و زمان تصویربرداری، تعداد بیت‌های ذخیره شده (برای هر پیکسل)، اطلاعات بخشی از بدن که تصویربرداری شده، تکنیک تصویربرداری به کار برده شده (مثلاً MRI یا CT). برای دسترسی به این اطلاعات، بعد از خواندن و ذخیره کردن فایل دایکام در یک متغیر مثلاً به نام ds کافی است attribute یا property مورد نظر از این متغیر را فراخوانی کنید. مثلاً ds.Modality به شما تکنیک تصویربرداری به کار برده شده را می‌دهد.

برای مشاهده‌ی اطلاعات بیشتر درباره‌ی این نوع فایل و شیوه‌ی کار با آن‌ها، به [صفحه‌ی مستندات آن](#) رجوع کنید.