

بِنَامِ خَدا و نَبْلَهُ شَدَه بِحَشَابِ يَسِّرٍ

سیسمونی چند رسانه ای

تمرين سري اول

عَصْرِي مهیار 9632093

سوال اول:

(الف)

برای اینکه بتوانیم تصویر را به صورت horizontal flip داشته باشیم باید جای ستون های آن عوض شود به گونه ای که ستون اول به جای ستون آخر برود و ستون دوم به جای ستون یکی مانده به آخر و به همین ترتیب جلو برویم.

(ب)

همان گونه که در متن سوال گفته شده برای داشتن این تصویر باید جای سطرها و ستون ها را عوض کنیم.

(ج)

برای این کار ابتدا ابعاد ماتریس تصویر را حساب کرده و بر 4 تقسیم کردم و ceil گرفتم. حاصل محاسبات قبلی برابر حذفیات از طرفین عکس می باشد و چون از هر طرف 25 درصد عکس کسر شده است لذا ماتریس تصویر حاصله در این بخش

داراي ابعادی به اندازه نصف تصویر اصلی خواهد بود. به کمک

حلقه ها تصویر اصلی را به تصویر بریده شده نگاشت می کنيم.

(د، ۵، و)

در اين بخش ها ما الگوريتم يكسانی را در پيش ميگيريم. روند کلي به اين صورت است که در سطر يا ستون مرزي در حلقه شرط می گذاريم و دو عبارت متناسب با منطق سوال را در آن می نويسیم.

مثالاً برای بخش د، هنگامی که میخواهیم Right Shift داشته باشیم ستون اول تصویر اصلی ستون سوم تصویر جواب، ستون دوم تصویر اصلی ستون چهارم تصویر جواب و ... خواهند بود و دو ستون اول تصویر جواب دو ستون آخر تصویر اصلی هستند. پس باید در حلقه و به کمک مرزها این شرایط را پياده سازی کنيم.

در این دو بخش من الگوریتم پیاده شده در بخش های (د) و (ه) را برای تصاویر حاصل شده در بخش (و) استفاده کردم.

به عبارت دیگر شیفت قطری را ترکیب دو شیفت به جهت های مختلف در نظر گرفتم. مثلا شیفت قطری به سمت بالا و راست با ترکیب دو شیفت به بالا و راست برابر است. از این رو الگوریتم شیفت به راست در بخش (د) را روی خروجی شیفت به بالا در بخش (و) اجرا کردم.

سوال دوم:

در این تابع، ابتدا ابعاد تصاویر دریافتی را محاسبه می کنیم و چون دو تصویر هم اندازه هستند تفاوتی ندارد در دستور `size` کدام یک از تصاویر را قرار دهیم.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

pixels of the original image
 pixels of the modified image
 Number of pixels in the image

طبق رابطه فوق ابتدا ماتریس دو تصویر را از یک دیگر کم میکنیم، سپس ماتریس حاصله را به توان دو می رسانیم و در نهایت جمع درایه های آن را حساب کرده و بر حاصل ضرب ابعاد تصاویر دریافتی تقسیم می کنیم تا `MSE` به دست بیاید.

در فایل `q2firsttest` دو تصویر هم اندازه دلخواه را برای تست کردن صحت عملکرد این تابع به عنوان ورودی داده ام تا `MSE` را محاسبه کند.

سوال سوم:

(الف)

تصویر داده شده در صورت تمرین را می خوانیم و سپس طبق گفته سوال 20 واحد از آن کم کرده و بعد به کمک یک شرط و قرار دادن حلقه، مقادیر پیکسل هایی که از صفر کمتر شده بودند را برابر صفر قرار می دهیم.

هر دو تصویر را به تابع سوال دو می دهیم و میبینیم MSE برابر 376.8883 شد.

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2$$

pixels of the original image
 pixels of the modified image
 Number of pixels in the image

با توجه به رابطه فوق اگر هر درایه تصویر دوم از درایه نظری خود در تصویر اول 20 واحد کمتر باشد رابطه فوق به صورت زیر در می آید:

$$MSE = \frac{450 \times 450 \times 20^2}{450 \times 450} = 20^2 = 400$$

اما در اینجا بعضی از درایه های تصویر دوم که بعد از تفریق 20 واحد از تصویر اول منفی شده بودند مقدارشان برابر صفر قرار داده شد (این یعنی در بعضی از درایه های نظیر اختلافی کمتر از 20 واحد داریم) و چون تعداد این درایه ها نسبت به کل پیکسل ها زیاد نبود انتظار داشتیم MSE با 400 تفاوت زیادی نداشته باشد که حاصل 376.8883 اثباتی بر حرف ماست.

(ب)

از الگوریتم استفاده شده در سوال اول استفاده می کنیم و دو تصویر خواسته شده را می سازیم. سپس سه تصویر را با پیکسل به پیکسل با هم جمع و تقسیم بر 3 می کنیم تا ماتریس تصویر میانگین ساخته شود. با کمکتابع سوال دو MSE را حساب می کنیم که برابر 90.7533 می شود و سپس تصاویر را نمایش می دهیم.

سوال چهار:

برای نوشتن تابع ابتدا به کمک دستور `imresize` ابعاد تصویر را 0.8 برابر می کنیم و همزمان الگوریتم های مورد نظر خواسته شده در صورت سوال را روی آن اجرا می کنیم، سپس بار دیگر با کمک این دستور ابعاد تصویر قبلی را 1.25 برابر میکنیم ($1.25 * 0.8 = 1$) و سه بار از تابع سوال دوم استفاده میکنیم (هر بار تصویر اصلی را با تصویر یکی از الگوریتم ها به تابع سوال دوم ارسال می کنیم). در انتهای تابع تصاویر خواسته شده را با عنوان مناسب نمایش می دهیم.

در کد 94C هم مقدمات لازم مانند خواندن تصویر موردنظر و فرآخوانی این تابع را انجام داده ام.