

### توضیح تابع :gaussian\_kernel

فرض کرده ایم که سایز کرنل های ورودی فرد است اما در صورت زوج بودن مشکلی پیش نمی‌اید. طبق رابطه تابع گاوسین، محاسبات را انجام داده و در کرنل آن را ذخیره می‌کنیم.

### توضیح تابع :convolve

عملیات کانولوشن را به کمک کرنل و عکس ورودی انجام داده. دقت کنید که عملیات را no\_padding انجام داده و سایز ماتریس خروجی کوچک‌تر است.

### توضیح تابع :sobel\_gradients

برای محاسبه  $\text{direction}$ , کاری که کرده ایم منفی مشتق در جهت عمود را به ورودی arctan داده ایم. این برای این است که کرنل vertical را جوری پیاده سازی کرده ایم که در جهت پایین افزایش میابد مقادیر  $y$ . اما در مختصات ریاضی این برعکس است. همانطور که میبینید خروجی‌ها درست به دست آمده اند.

### توضیح تابع :non\_maximum\_suppression

در این قسمت ابتدا جهت گرادیان را به دست آورده ایم و از روی آن تشخیص میدهیم که در کدام یک از جهات عملیات NMS را انجام دهیم. لیست dir برای همین کار است. ایندکس 0 آن نمایانگر این است که زاویه بین 0 تا 22.5 بوده و گرادیان افقی است و بقیه هم به همین ترتیب. دقت کنید که به 8 قسمت ربع اول و دوم را تقسیم کرده ایم برای بهتر نتیجه گرفتن.

همانطور که میبینید خروجی simple thresholding خیلی خوب نیست.

## توضیح تابع :**hysteresis**

برای پیاده سازی این قسمت از bfs کمک گرفته. به طوری که روی پیکسل های **strong** عملیات را انجام داده و اگر به راس **weak** برسیم، یعنی اینکه یک مسیری از آنها به پیکسل های **weak** وجود داشته و آنها را نیز لبه میکنیم. همانطور که در خروجی میبینید لبه ها به خوبی تشخیص داده شده و پیوسته است .**single thresholding**