

برای رفتن به بعدی در پنجره های اوپن سی وی یک کلید از صفحه کلید را بفشارید.

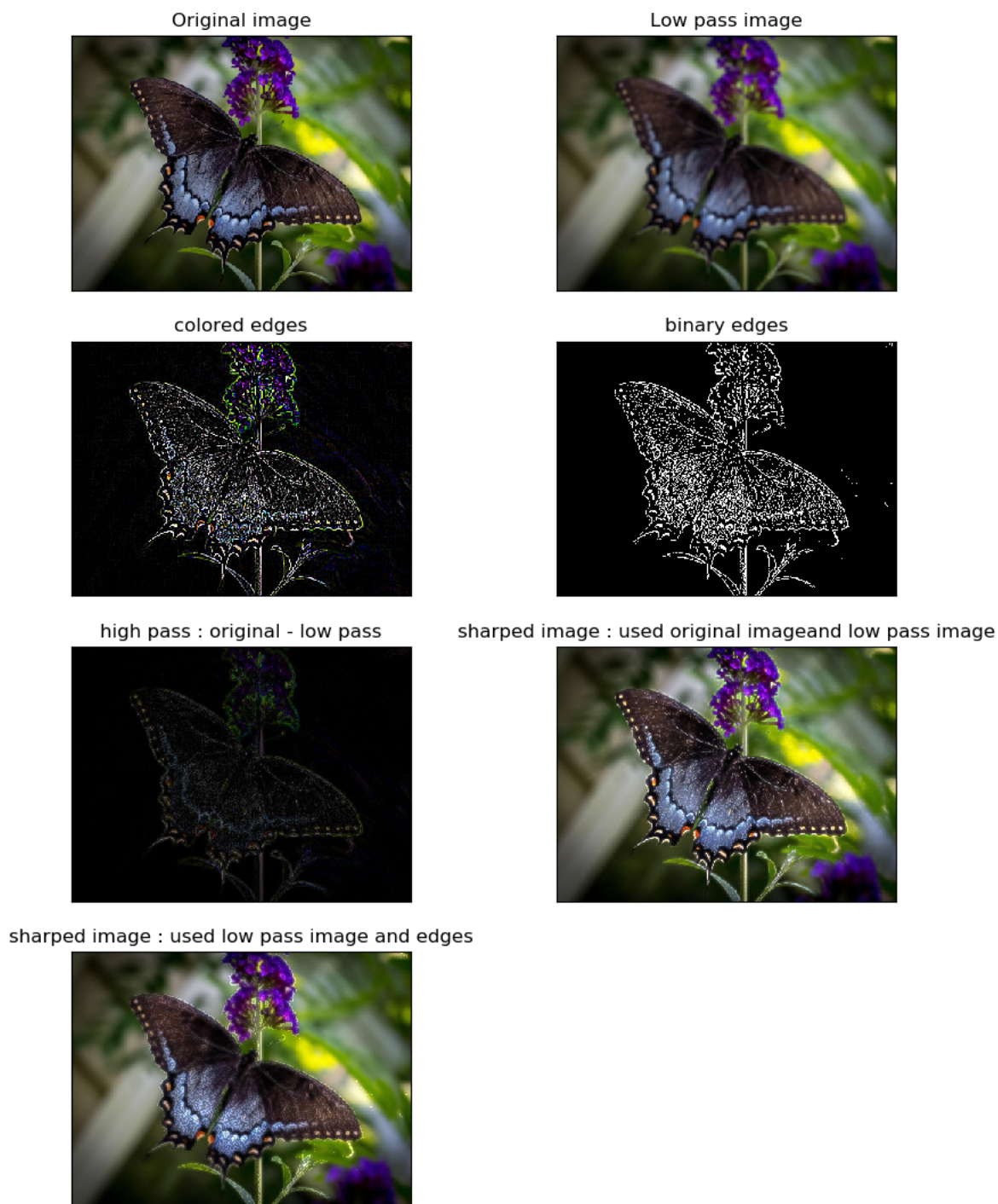
## الف

۳

در این سوال در ابتدا با تابع `cv.resize()` عکس را به ابعاد خواسته شده بردم. برای فیلتر پایین گذر یک آرایه ۵ در ۵ که همه ی المان هایش یک است تعریف و آن را تقسیم بر ۲۵ میکنیم. این کرنل پایین گذر ماست که با کانوالو با عکس به وسیله تابع `cv.filter2D` تصویر پایین گذر حاصل میشود که در شکل زیر آمده است.

برای یافتن لبه های عمودی و افقی (طبق فرموده تی ای در سی دبلیو مبنی بر استفاده یک فیلتر) از فیلتر لاپلاسی ۳ در ۳ استفاده میکنیم که همه داریه هایش به جز عنصر وسط که ۸ است؛  $-1$  هستند استفاده میکنیم که با کانوالو در عکس رنگی با همان تابع فیلتر ذکر شده تصویر `colored edges` حاصل میشود. اگر تصویر را ابتدا به سیاه سفید تبدیل و سپس لبه یابی را انجام و نتیجه را با تابع `cv.threshold` به تصویر باینری تبدیل کنیم تصویر `binary edges` حاصل میشود.

برای یافتن تصویر بالاگذر میتوان تصویر اصلی را از تصویر پایین گذر توسط `cv.absdiff()` کم کرد که در زیر آمده است. برای ایجاد تصاویر با وضوح بیشتر (`sharp`) از دو راه میتوان استفاده کرد یکی اینکه از تصویر اصلی و پایین گذر استفاده کرد و تفاضل این دو را یافت و تفاضل را با بازی با ضرایب توسط تابع `cv.addWeighted` به تصویر اولیه اضافه کرد و تصویر با وضوح بیشتر را ساخت. یک راه دیگر این است تصویر لبه های باینری را در تصویر پایین گذر ضرب و تبدیل به رنگی کرد و آن را به تصویر پایین گذر افزود که این کاری نیازی به تصویر اولیه ندارد و فقط لبه ها و تصویر پایین گذر را میخواهد. تصویر حاوی همه مراحل در صفحه بعد آمده است.



شکل ۱: تصویر مربوط به سوال الف) ۳

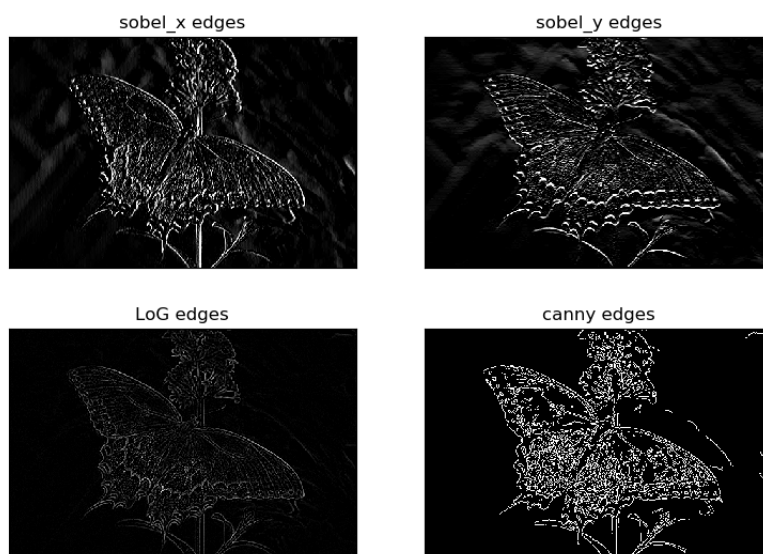
۴

برای تصویر ۱ و ۲ تصاویر لبه یابی به ۳ روش خواسته شده در ادامه آمده اند. در هر دو تصویر برای فیلتر sobel در  $dx = 1, dy = 1$  لبه های افقی تصویر تشخیص داده شده است و در حالت  $dx = 0, dy = 1$  لبه های عمودی معلوم میشوند؛ در فیلتر LoG ابتدا کرنل گاوسی با سیگما ۲ اعمال و سپس کرنل لاپلاسی بر آن اعمال میشود؛ که لبه ها را در راستای تشخیص میدهد. برای بهتر شدن تصویر عکس را بر ماکس آن تقسیم کرده ایم. در نهایت فیلتر canny اعمال میشود که بهترین عملکرد را به نظر من در لبه یابی دارد همانگونه که انتظار میرفت؛ البته با ترشولد های آن مقداری بازی شده که بهترین تصویر حاصل میشود. برای بهبود کیفیت میتوان مثلاً برای حذف نویز اول تصویر را با فیلتر گاوسی مقدار تار کرد و یا کنتراست را با روش های تغییر

کنتراست عوض کرد تا بهترین خروجی در لبه یابی حاصل شود.



شکل ۲: لبه یابی در تصویر ۳



شکل ۳: لبه یابی در تصویر ۲

در این قسمت از تابع `cv.SimpleBlobDetector-create(params)` یک تشخیص دهنده ساخته که با متد `detect`. آن لکه ها تشخیص داده میشوند؛ پارامترهای ورودی به صورت زیر هستند.

ترشولد مینیمم و ماکسیمم و پله افزایشی آن به این صورت هستند که تصویر در بازه مین تا ماکس با پله داده شده باینری شده و تصاویر متعدد ایجاد میشوند که لکه یابی در آنها انجام میشود هر چه این بازه وسیع تر باشد بیشتر لکه پیدا میشود.

فیلتر کردن بر اساس مساحت نیز فعال شده است و لکه هایی که مساحت آنها از ۶۰۰ داده شده بیشتر باشند باقی میماند؛ هر چه بیشتر کمتر لکه یافت میشود.

میزان دایره ای بودن و محدب بودن و اینرسی لکه های یافت شده که فرمول محاسبه برای هر خم را دراند فعال شده و اگر لکه ای این پارامتر هایش کمتر از مقادیر داده شده بود حذف شده و در غیر این صورت باقی میماند.

این پارامتر ها در بهترین حالت ست شده و لکه های یافت شدند.

```
params = cv.SimpleBlobDetector_Params()

params.minThreshold = 0
params.maxThreshold = 256
params.thresholdStep = 1

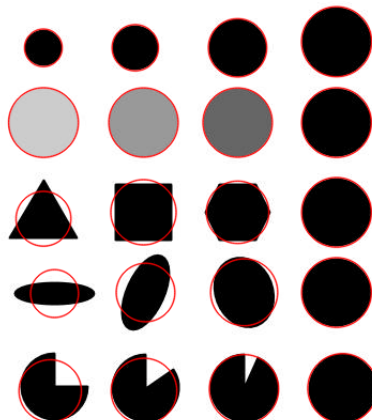
params.filterByArea = True
params.minArea = 600

params.filterByCircularity = True
params.minCircularity = 0.5

params.filterByConvexity = True
params.minConvexity = 0.5

params.filterByInertia = True
params.minInertiaRatio = 0.01
```

شکل ۴: لبه یابی در تصویر ۲

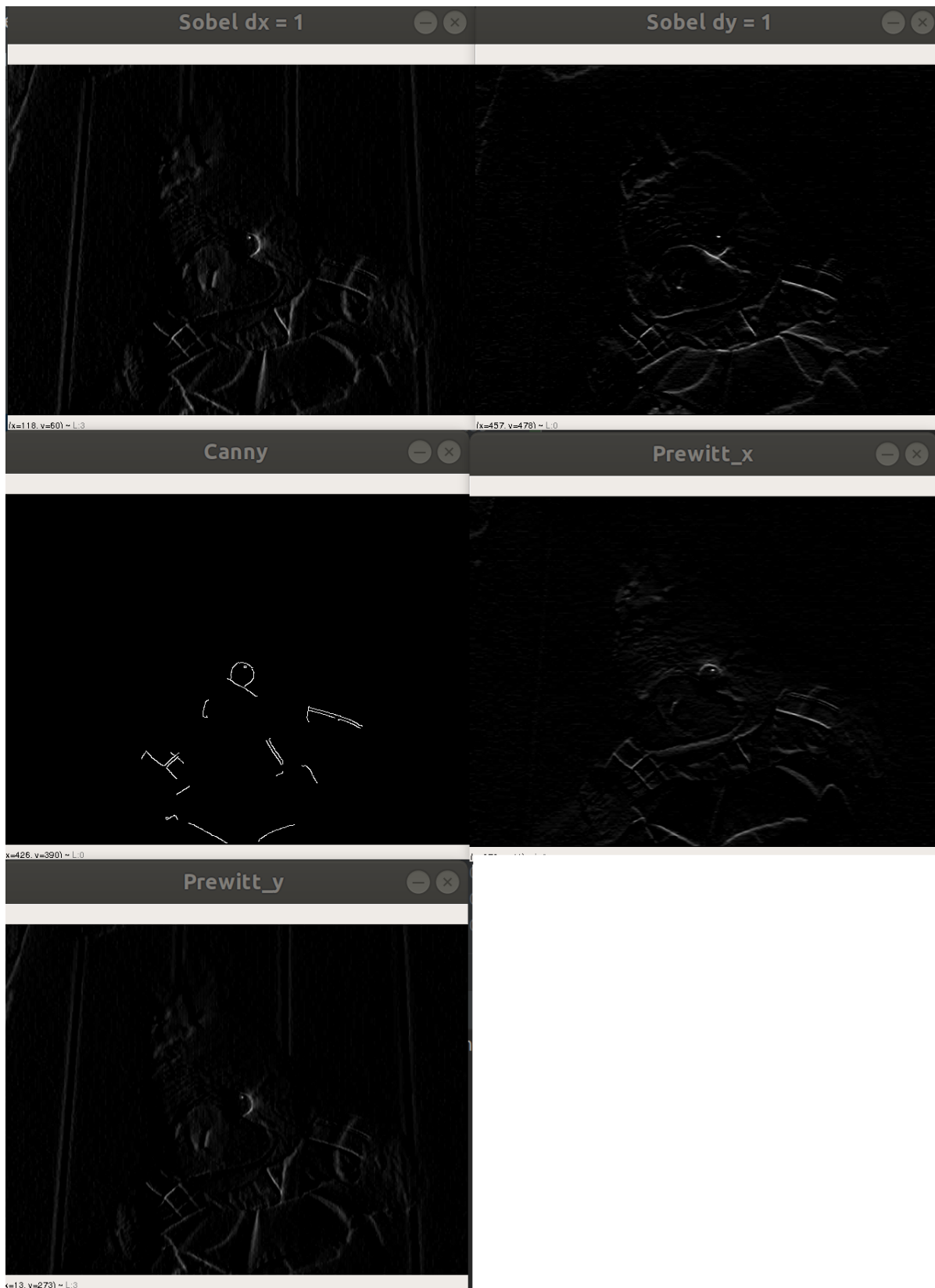


شکل ۵: لکه های یافت شده

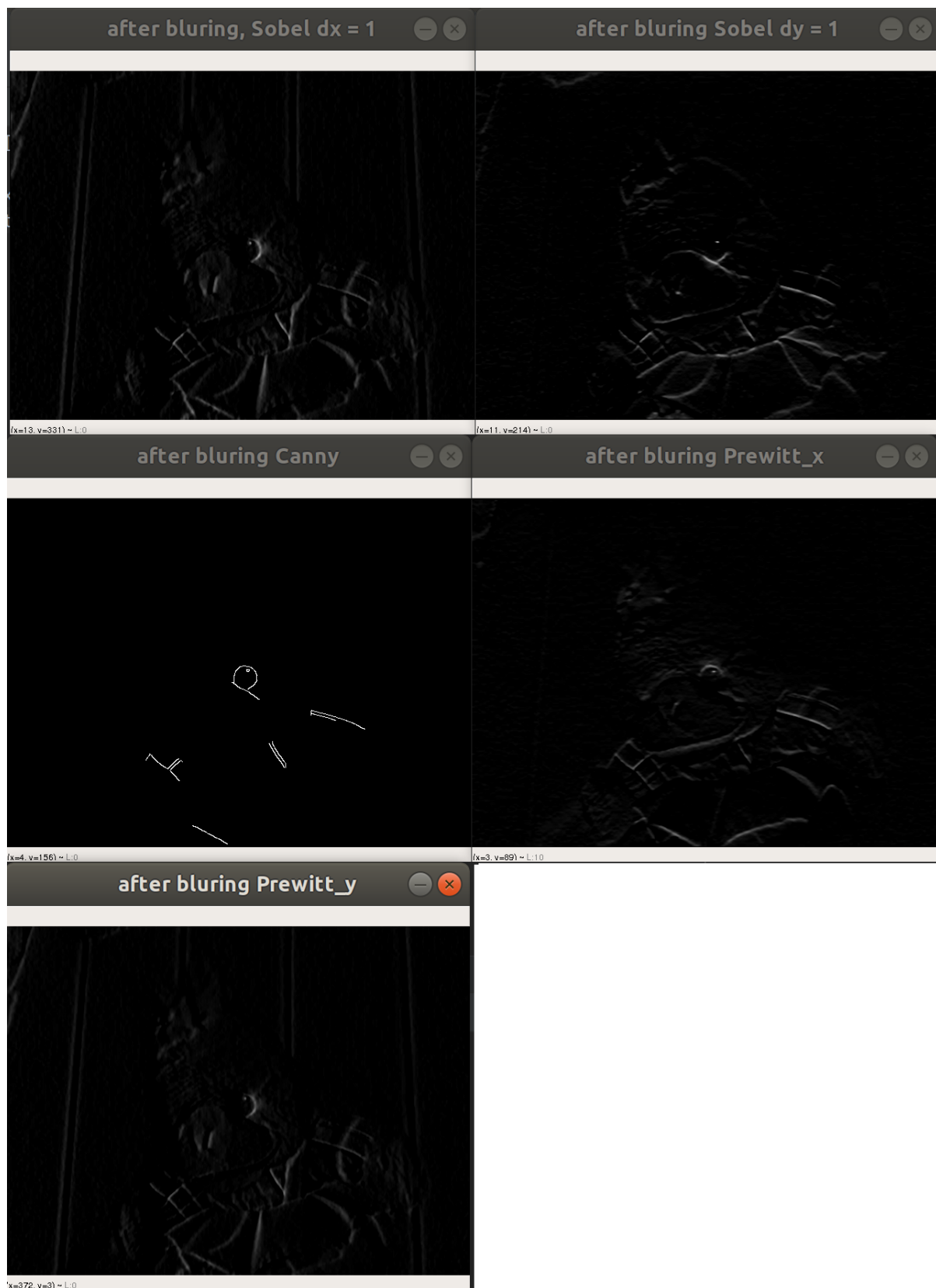
ویدئو ذخیره شده با نام saved.avi موجود است. فریم آخر آن و نتایج لبه یابی روی آن در ادامه آورده شده است. خود ویدئو ها با ران کردن کد حاصل میشوند؛ لطفا پس از اتمام نمایش یک ویدئو با فشردن کلیدی از صفحه کلید به ویدئو بعدی بروید. سوبل و پرویت در جهت های خاصی لبه را تشخیص داده اما کیفیت سوبل همانگونه که انتظار میرود بهتر است. کنی اما در این حالت خوب جواب نمیدهد. در هنگامی که فیلتر گاوسی اعمال میشود لبه ها نرم تر شده و کمتر تشخیص داده میشوند.



شکل ۶: فریم آخر



شکل ۷: لبه یابی



شکل ۸: لبه یابی بعد از فیلتر گاوسی