

BIL 467 Görüntü İşleme

Ödev #3

Kağan Hamzaçebi

151101064

06/07/2021

1. Genel Açıklama

Ödev klasörü içerisinde 2 farklı script dosyası, kullanılan görüntülerin bulunduğu images isimli bir klasör ve outputs isimli çıktıların bulunduğu bir klasör bulunmaktadır.

Ödev 2 farklı script dosyası içerisine dağıtılarak yapılmıştır. Bu bölümler;

- morphology.m (1 – 2 - 3 numaralı sorular)
- sobel.m (4 numaralı soru)

1.1. Kullanım

Ödevde istenilen işlevler fonksiyonlar olarak tanımlanmıştır ve kullanılan inputlar script dosyasının başında import edilmiştir. Test edilmesi istenilen fonksiyon hariç diğer fonksiyonları comment out yaparak ve istenilen image'ı input olarak vererek gerekli testler gerçekleştirilebilir. Gerekli ek açıklamalar script dosyalarının içerisinde yer almaktadır.

Not: 4. soruda istenilen video üzerinde edge detection yapma sorusuna ait olan video çıktı outputs dosyasının içerisinde bulunmaktadır.

2. Morphology Modülü (1 – 2 - 3 numaralı sorular)

Bu script dosyası grayscale image'lar üzerinde çalışmakta ve 1, 2 ve 3 numaralı soruların cevaplarını içermektedir.

- thresholder (Satır 33 – 45 arası)
- cornerDetection (Satır 48 – 132 arası)
- convexHull (Satır 135 – 224 arası)
- connectedComponents (Satır 227 – 245 arası)
 - ✓ labelObj (Satır 248 – 287 arası)
 - ✓ isIncludes (Satır 290 – 302 arası)
 - ✓ labelCounter (Satır 305 – 317 arası)

2.1. Sonuç

Thresholder ilk ödevde yazılmış olan bir metoddur ve işlevini düzgün bir şekilde yerine getirebiliyor.

CornerDetection metodu istenildiği gibi çalışmaktadır ve verilen image'daki köşeleri düzgün bir şekilde tespit etmektedir.

ConvexHull metodu da düzgünce çalışmaktadır ancak bazı durumlar objelerin convexHull'ini tespit ederken fazla büyüme yaşanıldığı gözlemlenmiştir.

ConnectedComponents metodu düzgün bir şekilde çalışmaktadır. Bir image içerisinde yer alan tüm objeleri tespit edip 1'den başlayarak etiket(label) atamaktadır ve çıktısında da düzgün bir şekilde hangi etkiye ait kaç adet pixel bulunmaktadır gösteriyor. Bir kısıt verilmediğinden ötürü çok küçük (3-5 Pixel büyüklüğünde) olan objeleri de çıktıda belirttim. Connected Components metodu birkaç fonksiyonun birleşimi olarak çalışmaktadır. Modülün açıklamasının yapıldığı kısımda bu alt fonksiyonlar belirtilmiştir. Ek olarak script dosyası içerisinde de yorum satırları ile çalışma mantığı açıklanmıştır.

3. Sobel Modülü (4 numaralı Soru)

Bu script dosyası grayscale image'lara Sobel Edge Detection uygulamaktadır. Sobel edge detection aşağıda belirtilmiş olan fonksiyonlar ile sağlanmaktadır.

- thresholder (Satır 22 – 34 arası)
- fiterImg (Satır 37 – 72 arası)
- sobelEdgeDetection (Satır 75 – 108 arası)

3.1. Sonuç

Bu bölümde önceki ödevde (HW#2) tanımlamış olduğum fonksiyonları kullandım.

Thresholder ilk ödevde yazılmış olan bir metoddur ve işlevini düzgün bir şekilde yerine getirebiliyor.

FilterImg ikinci ödevde yazmış olduğum parametre olarak vermiş olduğumuz maskeyi kullanarak avarage fitering işlevini yapmaktadır. Düzgün bir şekilde çalışmaktadır.

SobelEdgeDetection ikinci ödevde yazmış olduğum bir metoddur ve FilterImg metoduna parametre olarak özel sobel filtrelerini vererek edge detection gerçekleştirmektedir. Düzgün bir şekilde çalışmaktadır. Verilen video framelerini düzgün bir şekilde tesipt edebilmektedir.

Not: Bu script dosyası istenilen işlevi yerine getirebilmektedir ancak yazılan kodda optimizasyon sıkıntıları olduğundan dolayı çıktı üretmesi uzun sürmektedir. Bu problemi daha ayrıntılı olarak karşılaşılan problemler bölümünde açıkladım.

4. Karşılaşılan Problemler – Zorluklar

Corner Detection olan ilk soruyu derste anlatıldığı şekilde implement ettim ve istenildiği gibi çalıştığını gördüm. Bu soruda bir problem ile karşılaşmadım.

Convex Hull sorusunda genel olarak istenildiği gibi çalışmasına rağmen bazı objelerde fazla büyüme gözlemledim sebebini tespit edemedim.

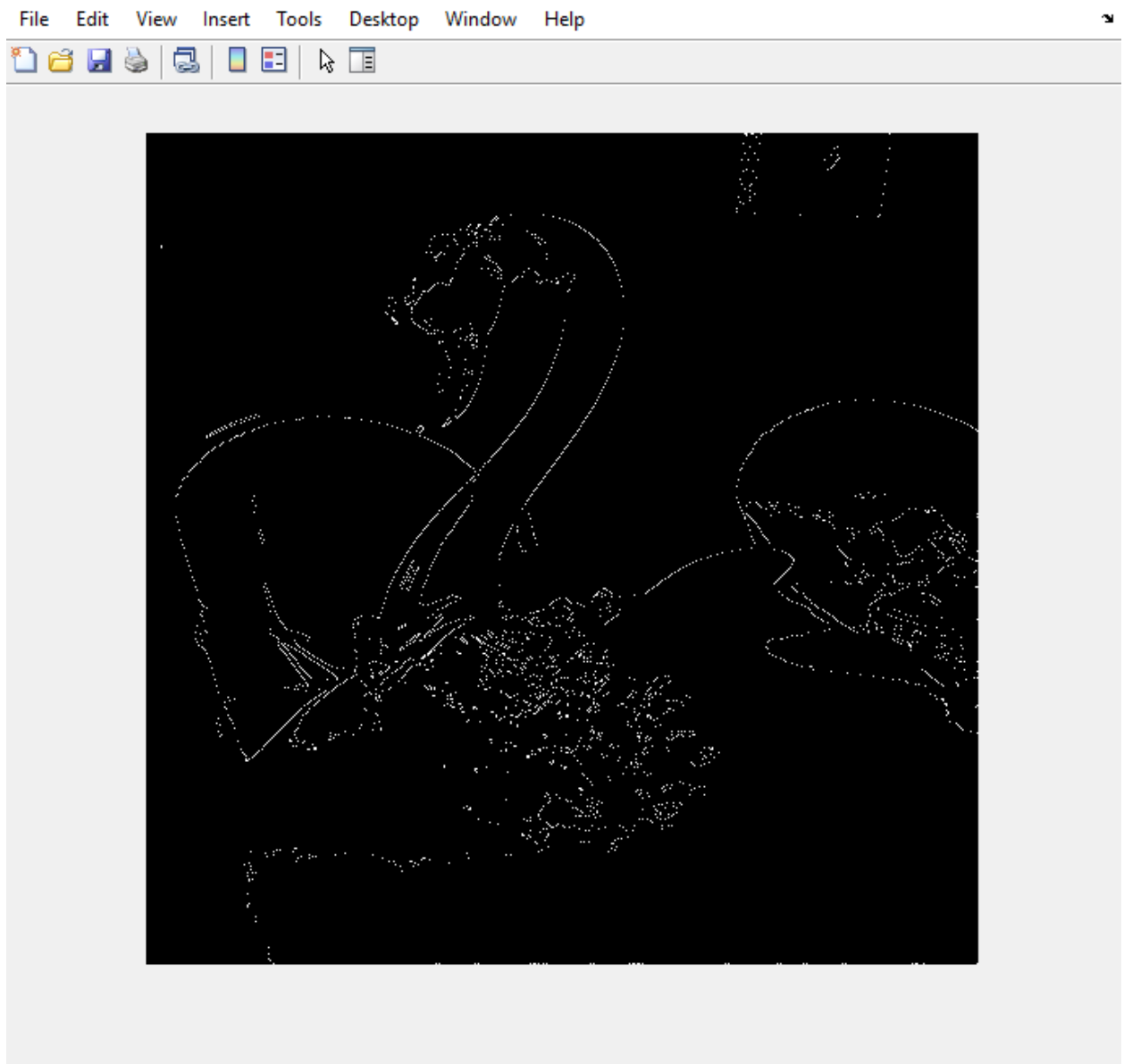
Connected Components yani üçüncü soruyu yaparken bir problem ile karşılaşmadım ve sonucunda istenildiği gibi olduğunu gördüm. Örnek olarak verilen 11 adet bozuk paranın olduğu bir image’da çıktının da 11 farklı birbirine yakın pixel sayısı içeren label olduğunu gördüm bu sayede doğru çalıştığı kanısına vardım. Ek olarak image içerisinde yer alan küçük pixellere de etiketleme işlemi uyguladım bundan dolayı çıktıda 5 – 10 pixel içeren labellar da bulunmaktadır. Bunun bir problem olup olmadığını bilmediğimden ötürü çıktıya müdahale etmedim.

Son soru olan Video Sobel Edge Detection’da çıktı çok başarılı olmaktadır ancak önceki ödevde de yaşamış olduğum optimizasyon problemlerinden ötürü tek frame (900x540) üzerinde edge detection işlemi yaklaşık 7 - 8 saniye sürmektedir (kendi bilgisayarımda) ve input olarak kullanmış olduğum video 125 frame’e sahip. Bu sebeple render işlemi yaklaşık olarak $(125 * 8) / 60 = 16$ dakika civarında bir vakit almaktadır. Optimizasyon problemine ek olarak Video’yu rgb’den grayscale’e kendim dönüştürdüm ancak videoda hala 3 Channel bulunmadığında VideoReader’da videoyu okuma hatası aldım. Bundan dolayı 3 Channel (R,G,B) yapısını bozmadan videomu grayscale yapıp iterasyon yapımı buna göre ayarladım.

5. Girdi ve Çıktılar



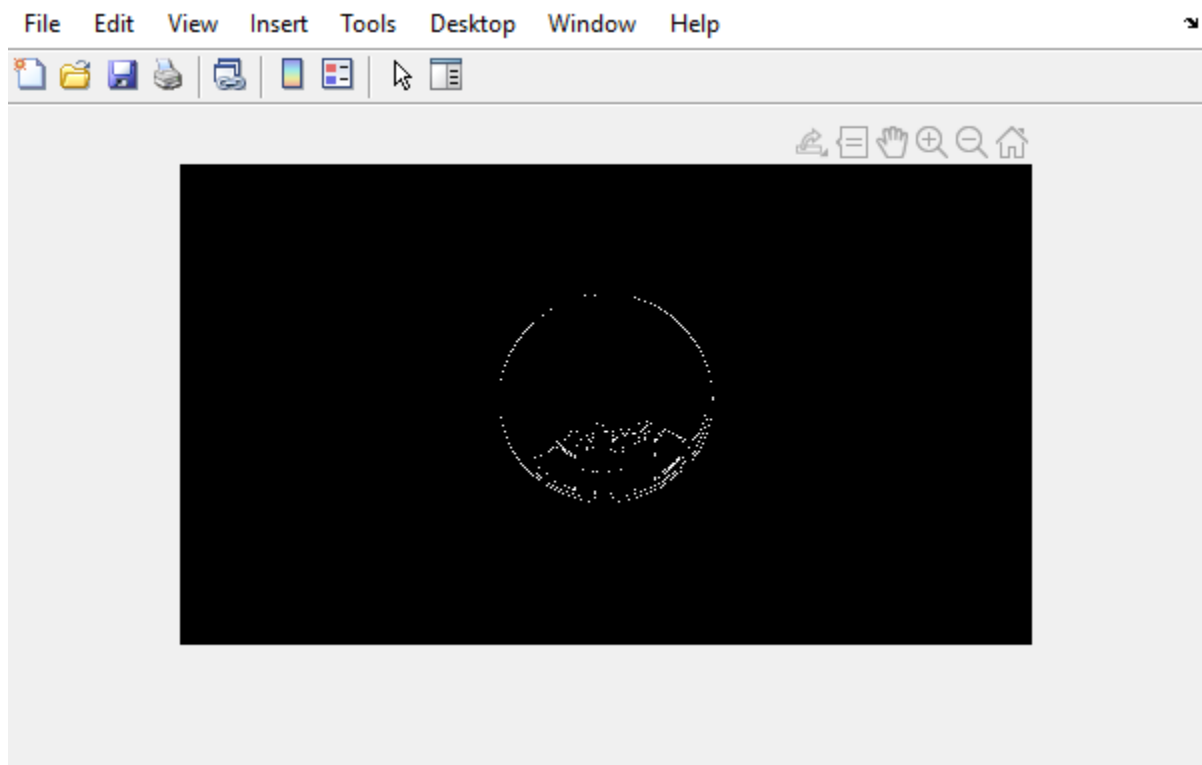
Input 1 – Grayscale (512x512)



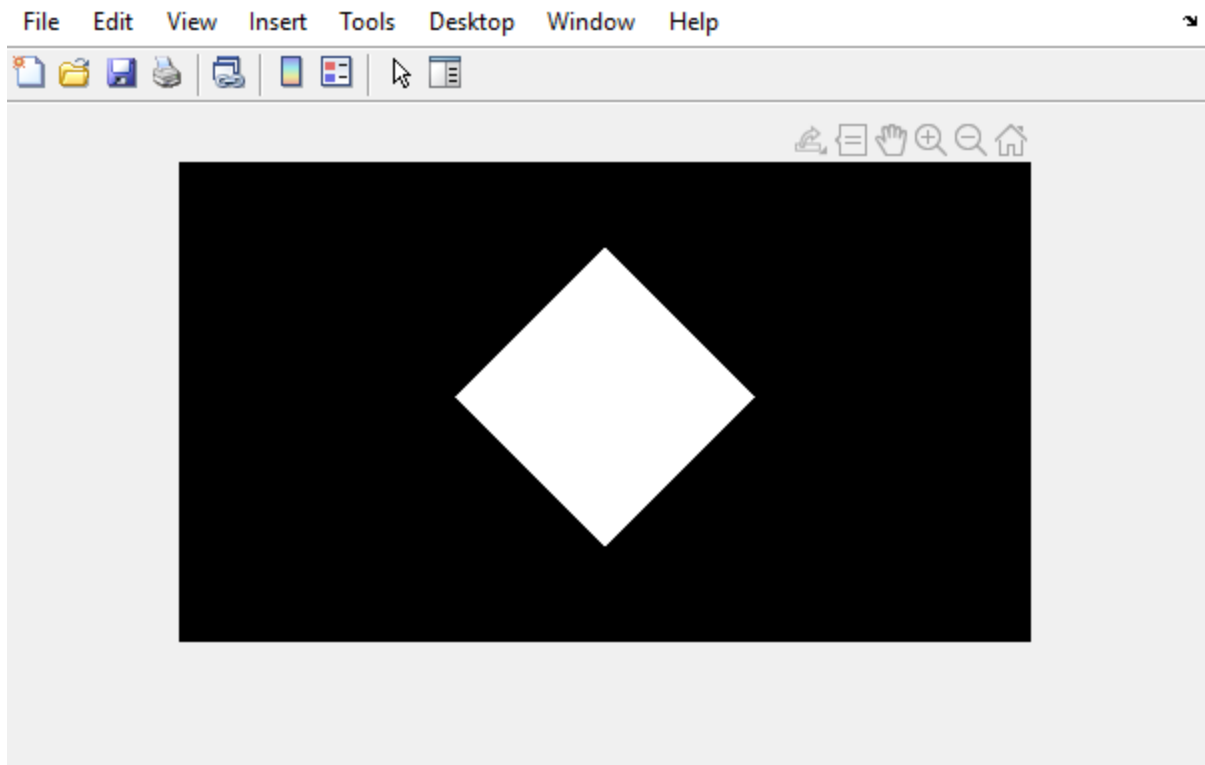
Input 1 – Corner Detection



Input 2 – Grayscale (426x240)



Input 2 – Corner Detection

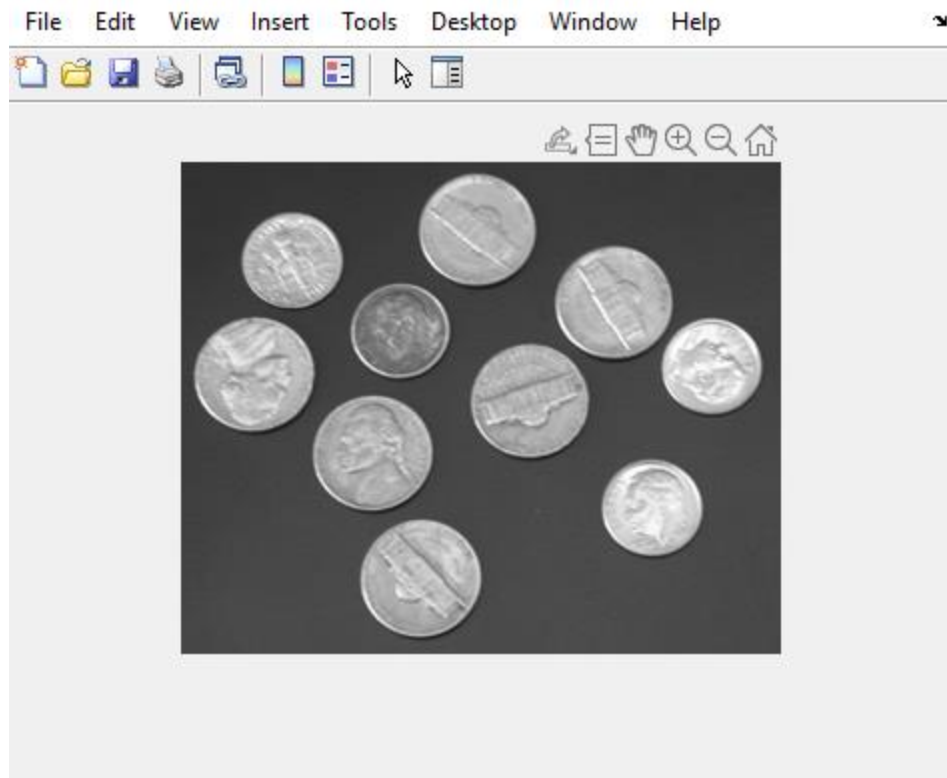


Input 2 – Convex Hull

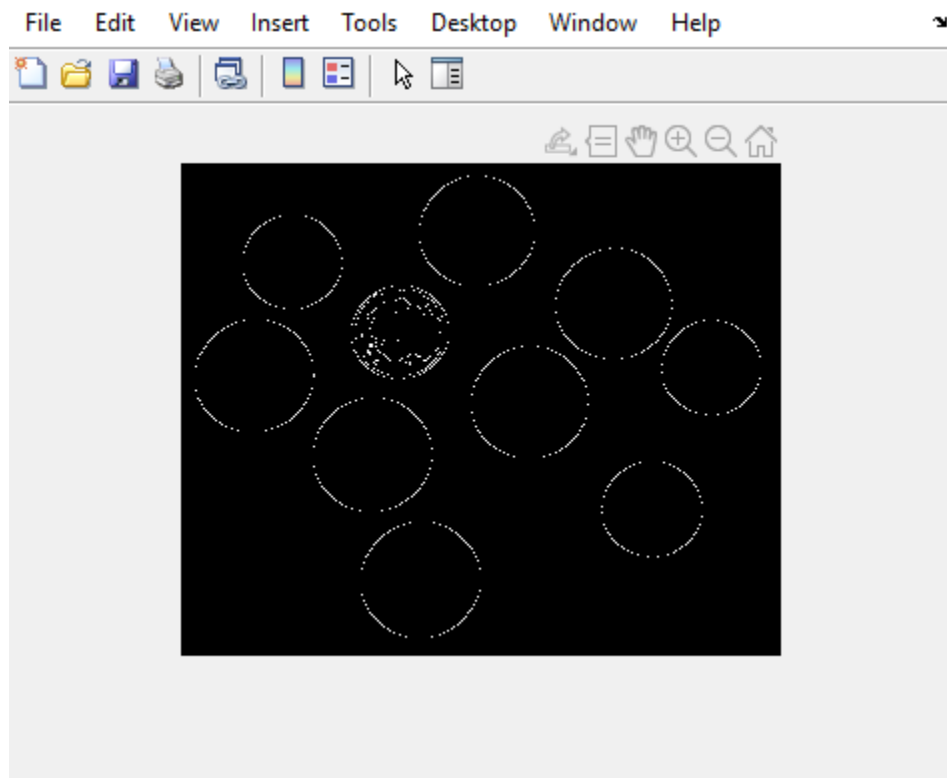
Label No	Number of Pixels
1	6811
2	19
3	21
4	4
5	1
6	1
7	12
8	1
9	1
10	1
11	1
12	1
13	19
14	3
15	5
16	4
17	4
18	1
19	1
20	1
21	9
22	1
23	1
24	7
25	2
26	5
27	2
28	2
29	14
30	3
31	2
32	2

>>

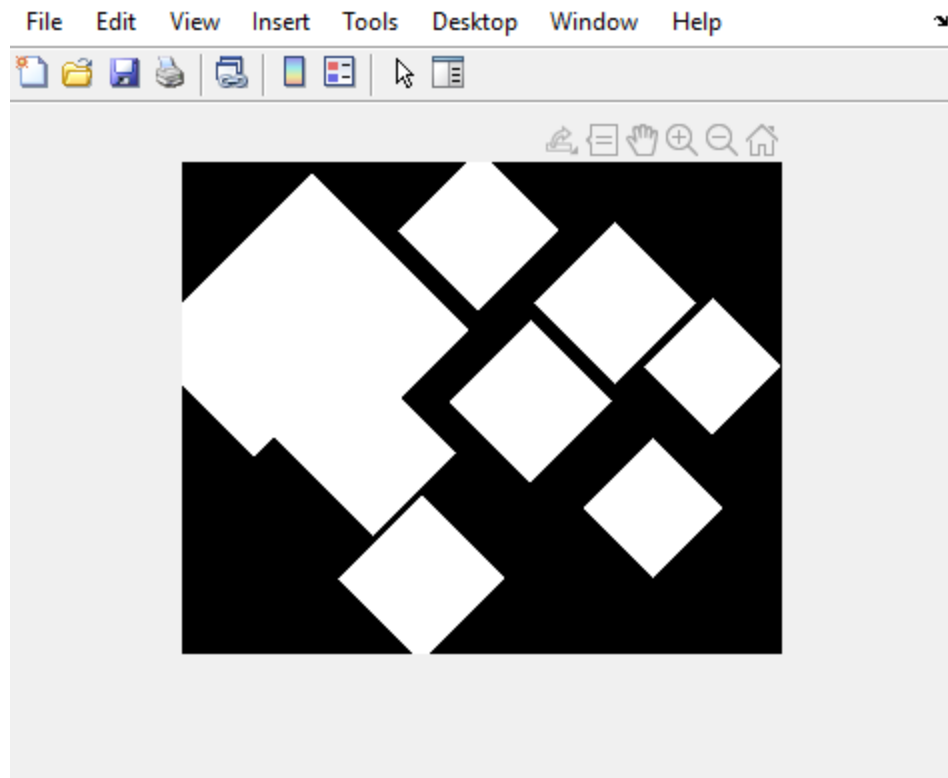
Input 2 – Connected Components



Input 3 – Grayscale (246x300)



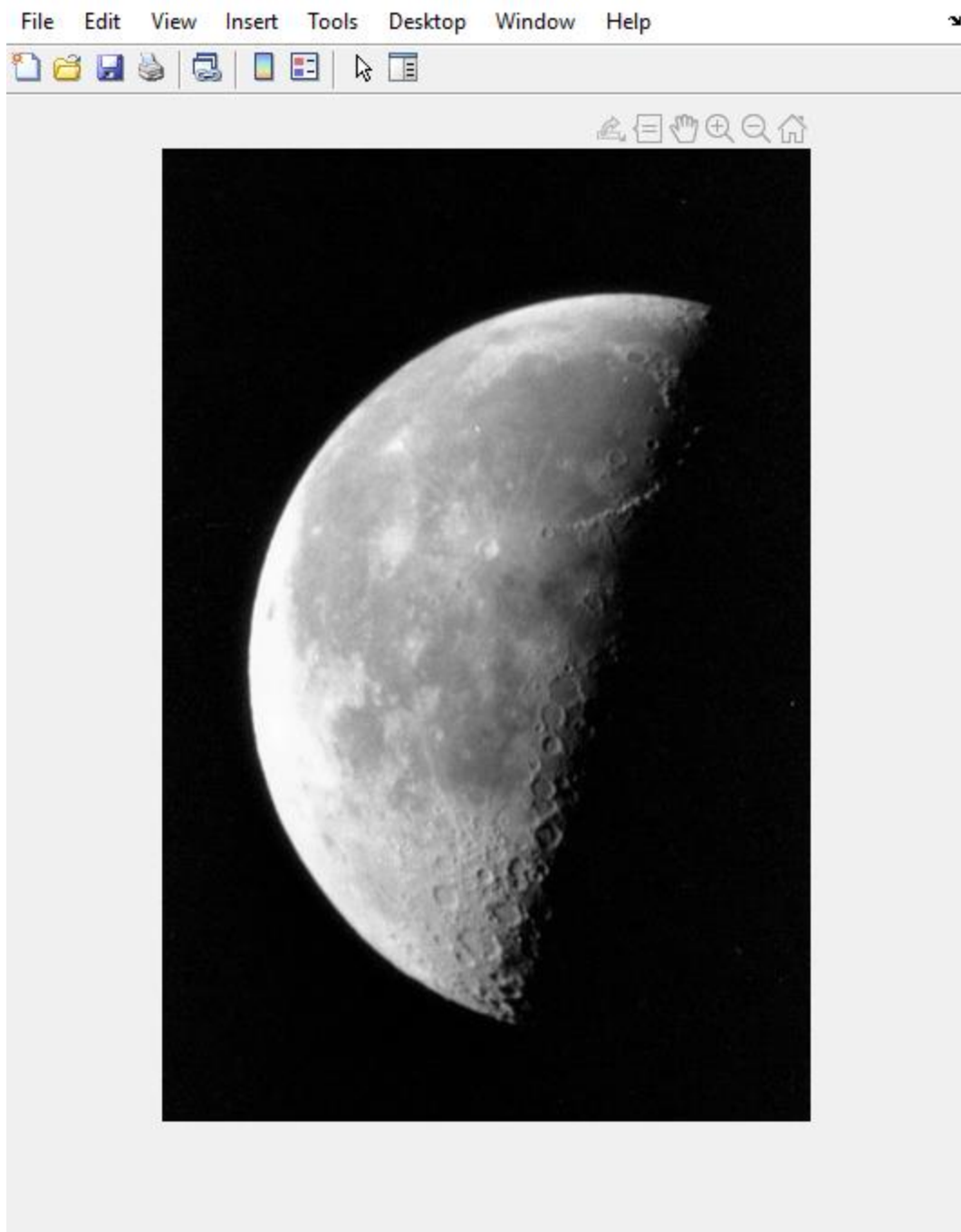
Input 3 – Corner Detection



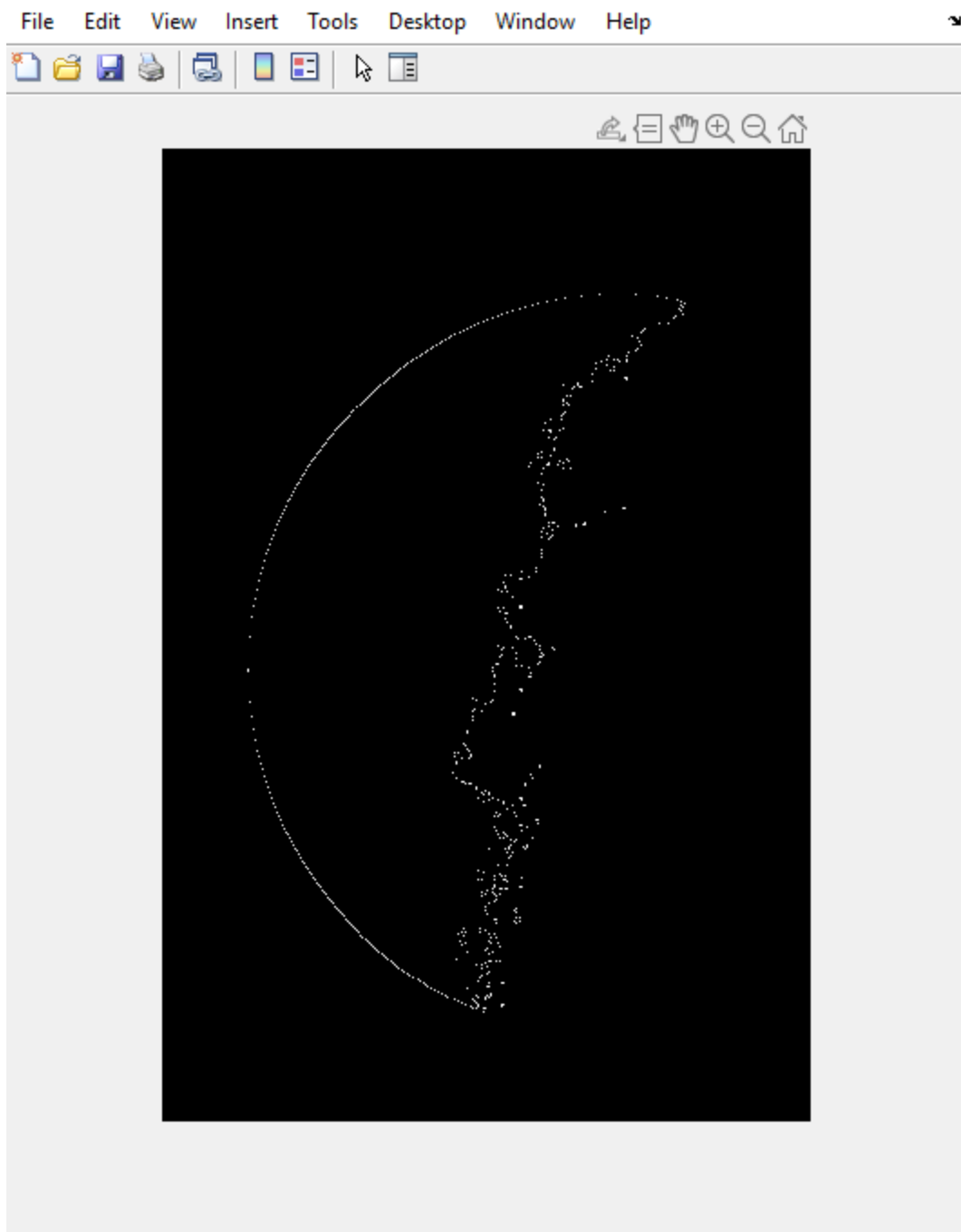
Input 3 – Convex hull

Label No	Number of Pixels
1	2532
2	1863
3	2560
4	1184
5	1
6	2658
7	1872
8	2
9	3
10	4
11	4
12	2607
13	7
14	2687
15	1907
16	2759
>>	

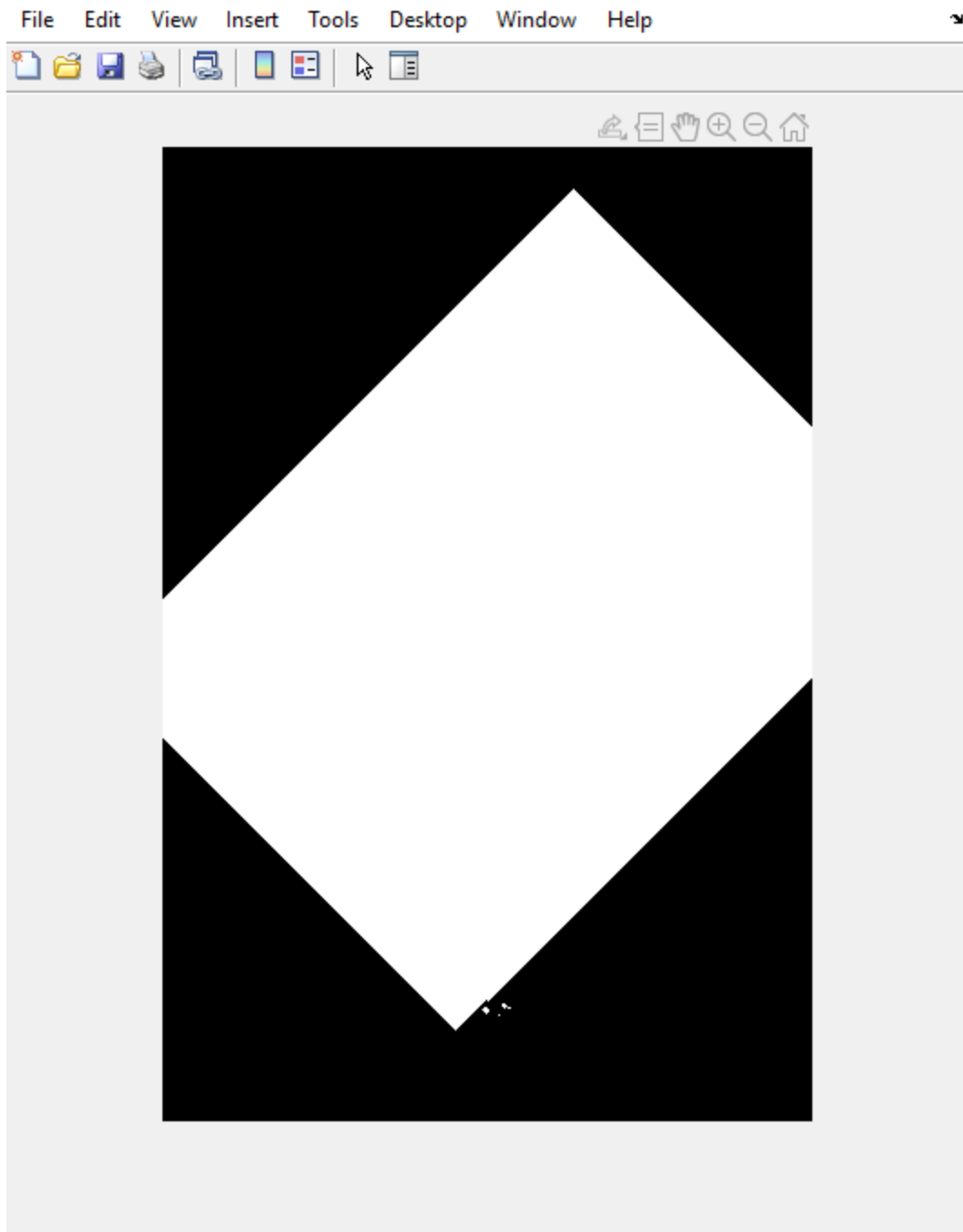
Input 3 – Connected Components



Input 4 – Grayscale (537x358)



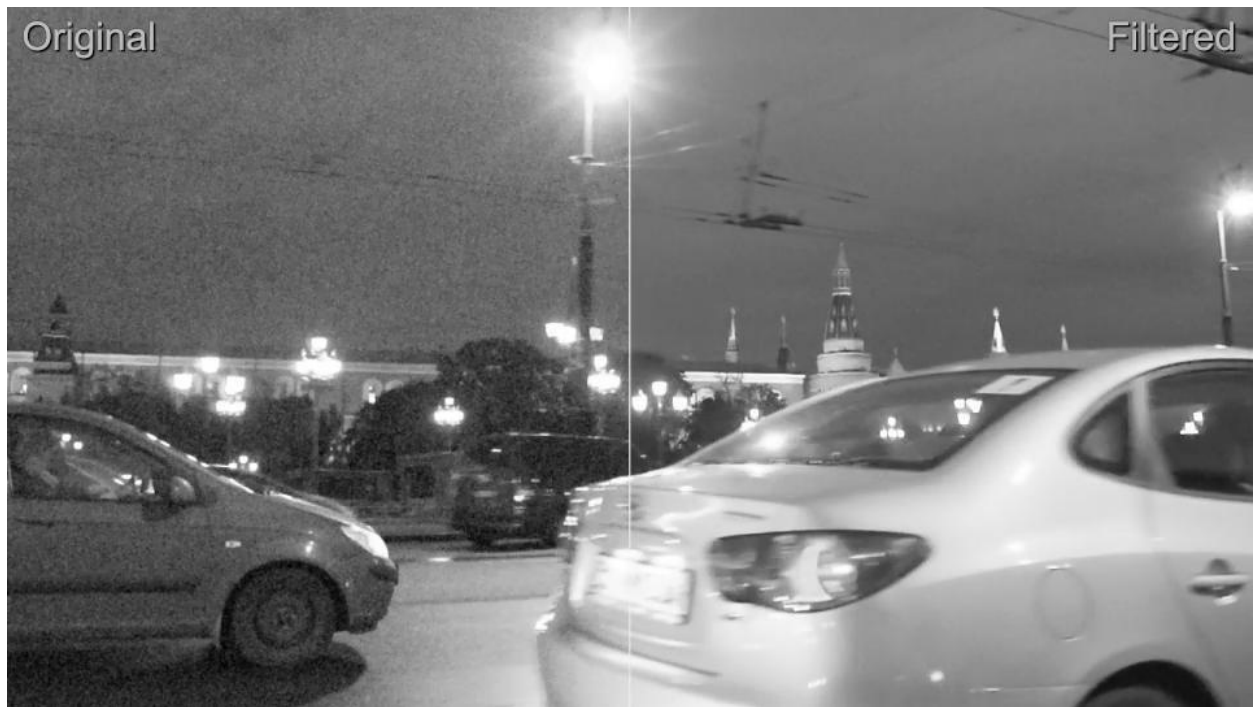
Input 4 – Corner Detection



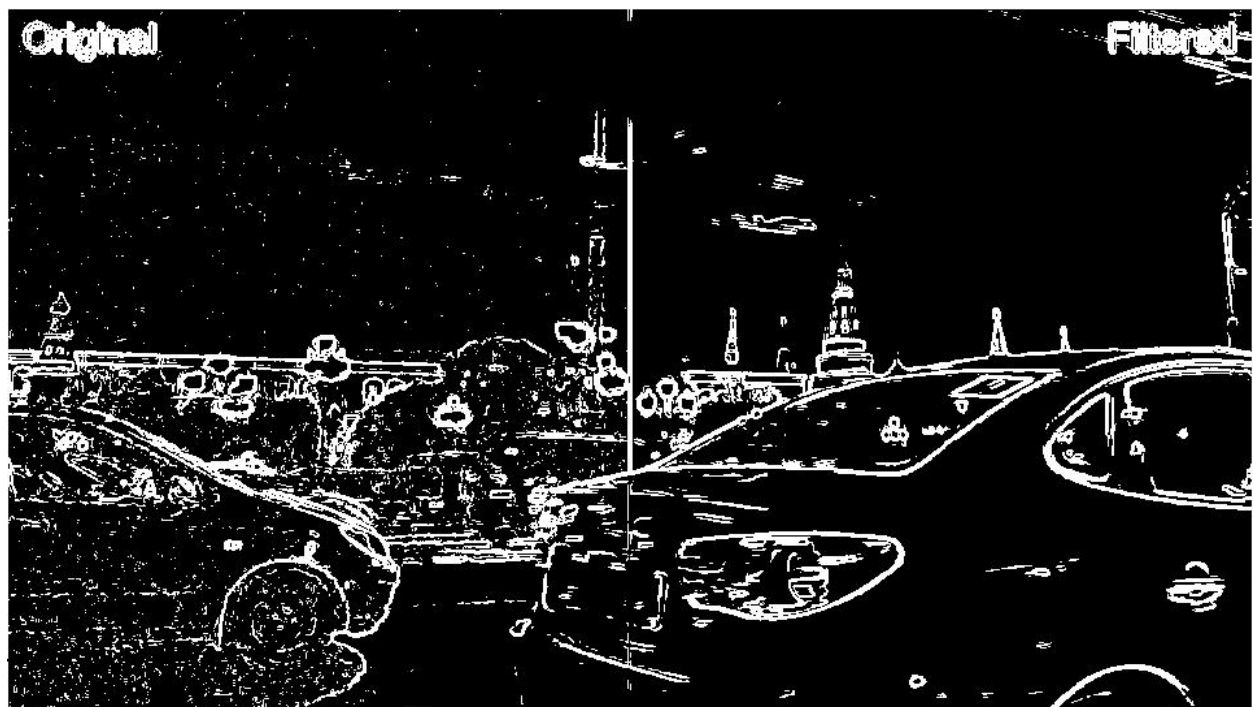
Input 4 – Convex Hull

Label No	Number of Pixels		
1	44874	32	4
2	2	33	1
3	1	34	3
4	1	35	1
5	3	36	6
6	5	37	1
7	2	38	8
8	2	39	4
9	1	40	4
10	1	41	7
11	1	42	8
12	1	43	4
13	38	44	9
14	4	45	2
15	3	46	3
16	6	47	1
17	2	48	3
18	4	49	1
19	1	50	4
20	1	51	9
21	4	52	2
22	1	53	7
23	1	54	14
24	6	55	2
25	1	56	7
26	1	57	7
27	4	58	1
28	5	59	12
29	1	60	8
30	4	61	8
31	10	62	1

Input 4 – Connected Components



Example Video Frame



Sobel Edge Detection to Frame