

BIL 467 Görüntü İşleme

Ödev #2

Kağan Hamzaçebi

151101064

23/06/2021

1. Genel Açıklama

Ödev klasörü içerisinde 3 farklı script dosyası, kullanılan görüntülerin bulunduğu inputs isimli bir klasör ve outputs isimli çıktıların bulunduğu bir klasör bulunmaktadır.

Ödev 3 farklı script dosyası içerisine dağıtılarak yapılmıştır. Bu bölümler;

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

Not: Girdi ve Çıktılar bölümüne videoyu ekleyemediğim için outputs klasörü içerisinde filtrelenmiş videoyu bulabilirsiniz.

1.1. Kullanım

Çalıştırılması istenilen fonksiyonu çağırıp diğer fonksiyonları comment out yaparak testleri gerçekleştirebilirsiniz. Ek olarak script dosyaları içerisinde yorum satırları ile gerekli açıklamalar yapılmıştır.

2. Sobel Modülü (1 -2 - 4 numaralı Sorular)

Bu script dosyasının içerisinde **thresholder** isminde ilk ödevde yazmış olduğumuz metod, **filterImg** isimli image'ları parametre olarak verilen maske ile filtreleyen bir metod, **filterImg** metodunu kullanarak Sobel Edge Detection yapan **sobelEdgeDetection** isimli bir metod ve son olarak **sobelEdgeDetection** değerlerini kullanarak Hough Space oluşturan **houghTransform** isimli bir metod bulunmaktadır. Ayrıca **filterImg** metoduna parametre olarak verebilmek için bir maske bulunmaktadır.

- thresholder (Satır: 24 – 36 arası)
- filterImg (Satır 39 – 74 arası)
- maske (7 – 9 arası)
- sobelEdgeDetection (Satır 79 – 108 arası)
- houghTransform (Satır 125 – 190 arası)

2.1. Sonuç

Thresholder metodu zaten ilk ödevde kodlamış olduğum bir metod olup düzgün çalışmaktadır ve filterImg içerisinde kullanmak gerektiği için bu ödevde de dahil edilmiştir.

FilterImg metodu istenilen smooth işlemini son derece iyi bir şekilde yerine getirmiştir. 3x3, 5x5, 9x9, 15x15'lik filtreler ile test edilip beklenen sonuca ulaşılmıştır.

Hough Transform beklenene oldukça yakın sonuç verdi istenildiği gibi çalıştı.

3. Median Filtering Modülü (3 numaralı Soru*)

Bu scriptin amacı 3 numaralı soruda istenilen video filtreleme için tanımlanmış olan median filtreleme fonksiyonlarını image'lar üzerinde test edebilmektir. Script içerisinde **medianFilterImgGRAY** ve **medianFilterImgRGB** metodları yer almaktadır.

- medianFilterImgGRAY (Satır: 22 – 58 arası)
- medianFilterImgRGB (Satır 62 – 112 arası)

3.1. Sonuç

MedianFilterImgGray metodu gri image'lar için istenilen şekilde çalışmaktadır. Ödev dökümanında belirtilmemiş ancak emin olamadığımdan dolayı mask'e boyutunu da parametre olarak alacak şekilde kodlandı. maskSize parametresi integer olup verilen değer N ise NxN olacak şekilde bir maske oluşturup gerekli işlemleri gerçekleştirmektedir.

MedianFilterImgRGB bir önceki metodun aynısı olup tek farkı RGB Image'lar için median filterin işlemini gerçekleştirmektedir. Düzgün bir şekilde çalışıp istenilen sonuçları üretmektedir.

Yukarıda belirtilen iki metod da 3x3, 5x5, 9x9 ve 15x15 maskeler ile test edilip beklenen sonuçlara ulaşılmıştır.

**: 3. soruda istenilen median filtering metodlarını image üzerinde test edebilmek için oluşturulmuştur.*

4. Vid Filtering Modülü (3 numaralı Soru)

Bu script dosyası bir video'yu frame'lerine ayırarak Median Filtering modülünde belirtilmiş fonksiyonlar ile gerekli filtreleme işlemini gerçekleştirip filtrelenmiş bir videoyu çıktı olarak sağlamaktadır. Bu modülde Median Filtering modülüne ek olarak sadece video'yu framelerine ayırıp tekrardan birleştirmek amaçlı VideoWriter kullanılmıştır.

4.1. Sonuç

Bu modülde Median Filtering Modülünde test edilmiş olan filtreleme fonksiyonları kullanılmıştır. Sadece 3x3'lik maske ile test edilebilmiş olup çıktıda beklenen sonuç gözlemlenebilmiştir. Çok uzun bekleme süresinden dolayı 15x15 gibi daha büyük maskeler ile test yapılamamıştır.

5. Karşılaşılan Problemler – Zorluklar

Sobel Edge Detection kısmında $\sqrt{G_x^2 + G_y^2}$ işlemini gerçekleştirirken elimdeki değerlerin uint8 olması sonucu matlab karekök almama izin vermedi bundan dolayı elimdeki değerleri double'a çevirip işlemi gerçekleştirdikten sonra tekrardan uint8 yaparak sorunu çözdüm.

```
sx = double(sobelX(i,j))^2;  
sy = double(sobelY(i,j))^2;  
mag = (sx + sy)^(1/2);  
magnitude(i,j) = uint8(mag);
```

FilterImg ve Median Filtering kısımlarında bir problem ile karşılaşmadım.

Video Filtering bölümünde fazla sayıda image ile çalıştığım için ve yazmış olduğum medianFilteringImgRGB metodu da optimize olmadığından dolayı işlem çok uzun sürmektedir. Tek iterasyonda R, G ve B değerlerini işlememe rağmen bu süreyi kısaltamadım. Ek olarak Zero Padding uygularken de matrisi tekrardan gezmeme gerektiğinden bu aşamada da işlem süresi uzatmaktadır. Ek olarak okuduğum frame'leri anlık olarak değiştirip Video'ya geri yazdıramadığım için elimdeki Video'yu değiştirmek yerine filtrelenmiş yeni bir video oluşturmayı tercih ettim. Bunun için VideoWriter'dan yararlandım.

```
open(filteredVideo)  
while (hasFrames(filteredVideo)):  
    writeVideo(filteredVideo, frame)  
end  
close(filteredVideo)
```

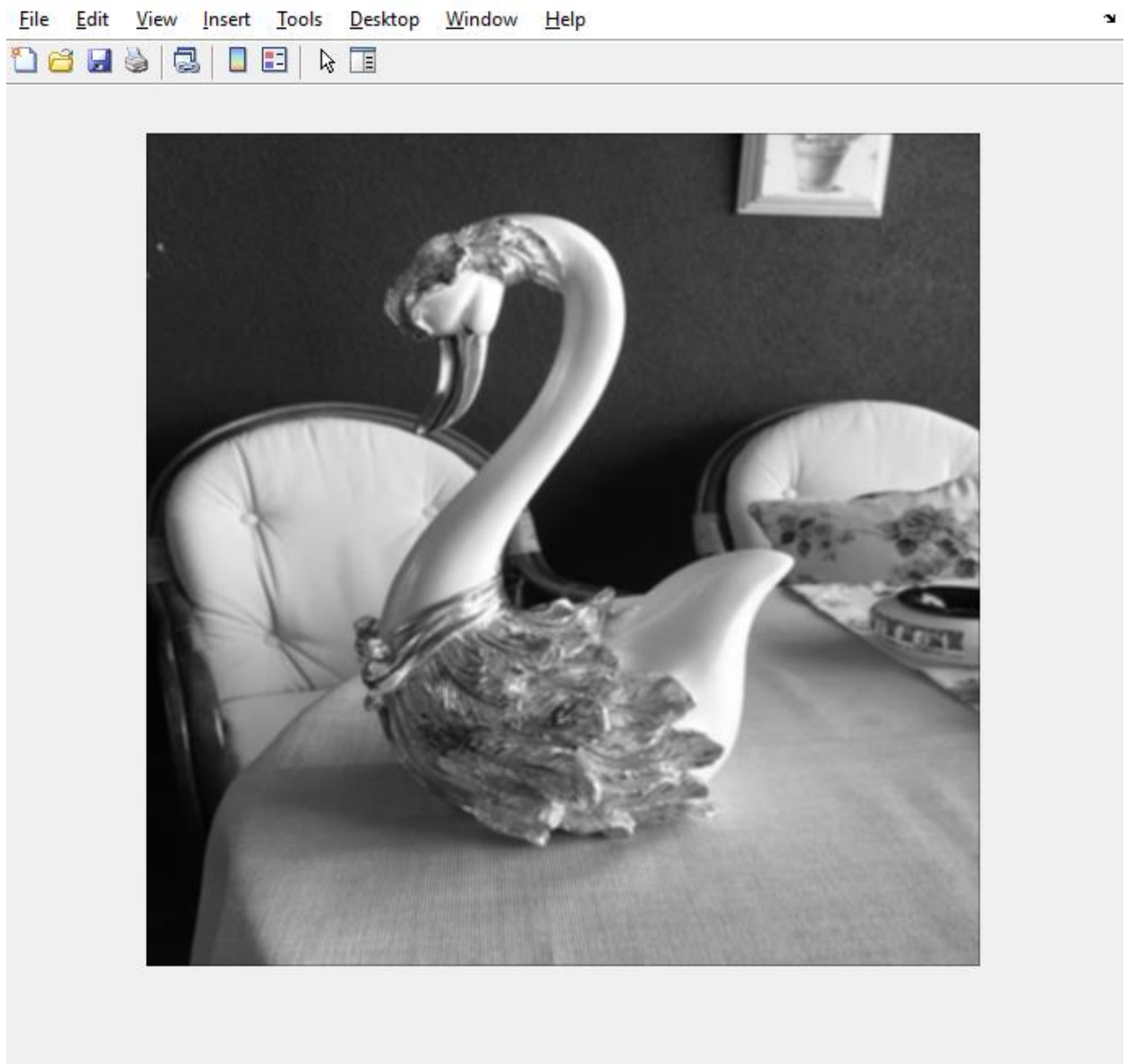
VideoFiltering aşamasında yapılmış olan filtreleme işlemi filtrede kullanılan maskenin boyutunun artması ile çok fazla uzadığından dolayı sadece 3x3 çıktı elde edebildim. 3x3 Maske ile filtrelenmiş video ile orijinal video arasındaki farkı göz ile farketmek biraz zor oluyor ancak fark yaratacak büyüklükteki filtreleri test edemediğimden dolayı büyük filtreler ile filtrelenmiş image çıktılarını ekledim. Filtrelenmiş image'ları Girdi ve Çıktılar bölümünde ve outputs kısmı içerisinde bulabilirsiniz.

Hough Transform bölümünde bir problem ile karşılaşmadım.

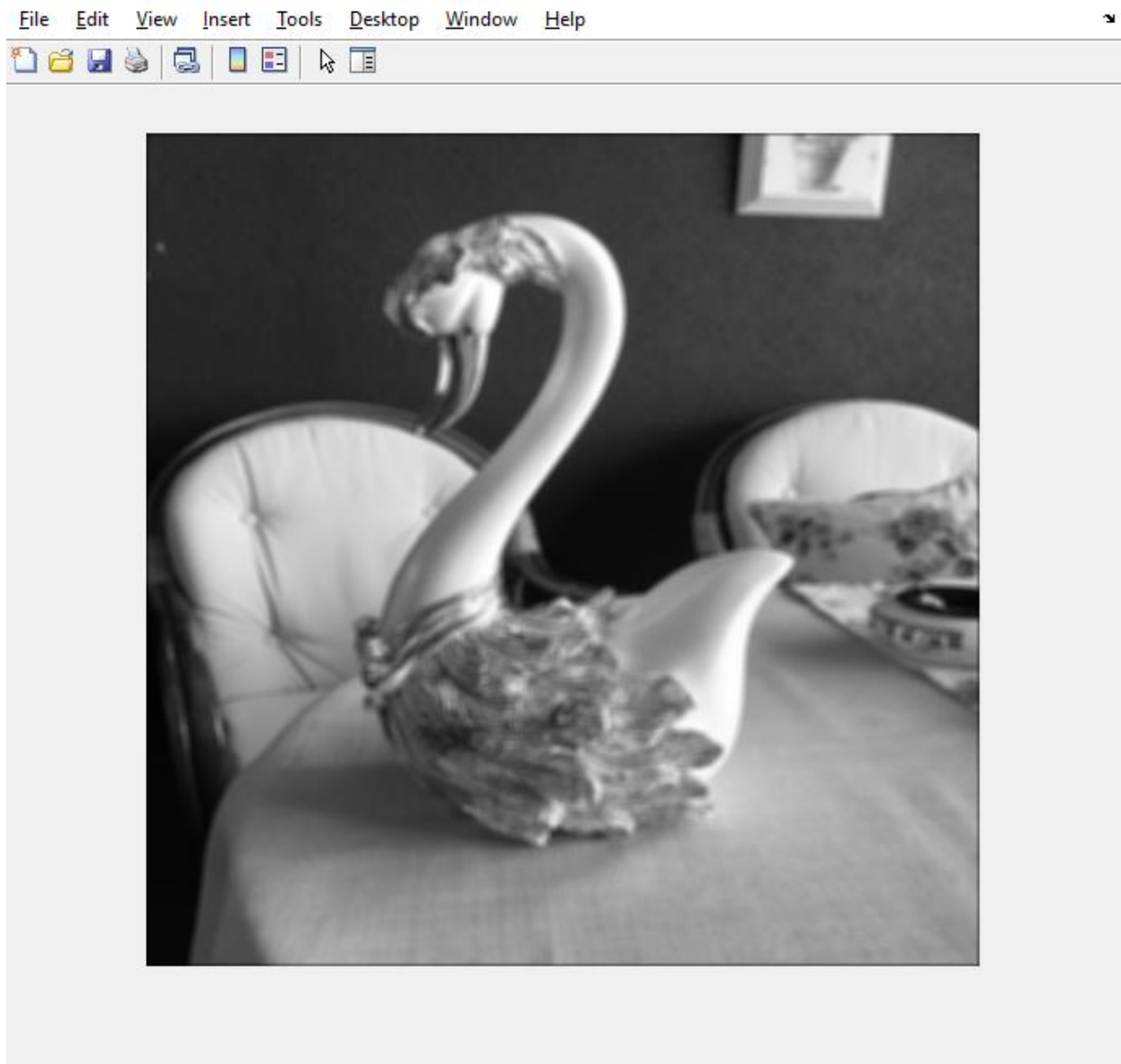
6. Girdi ve Çıktılar



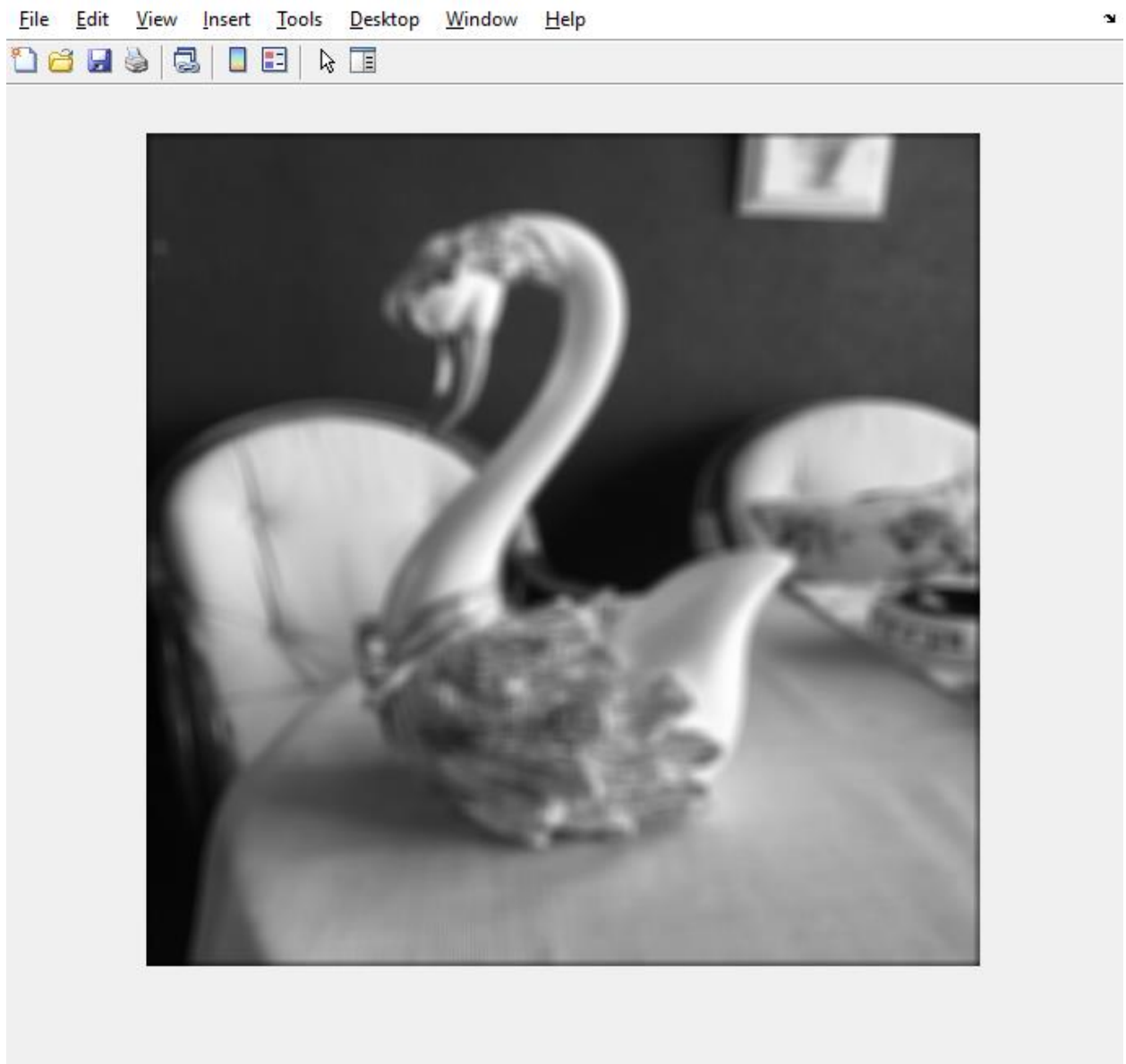
Original Grayscale Image (512x512)



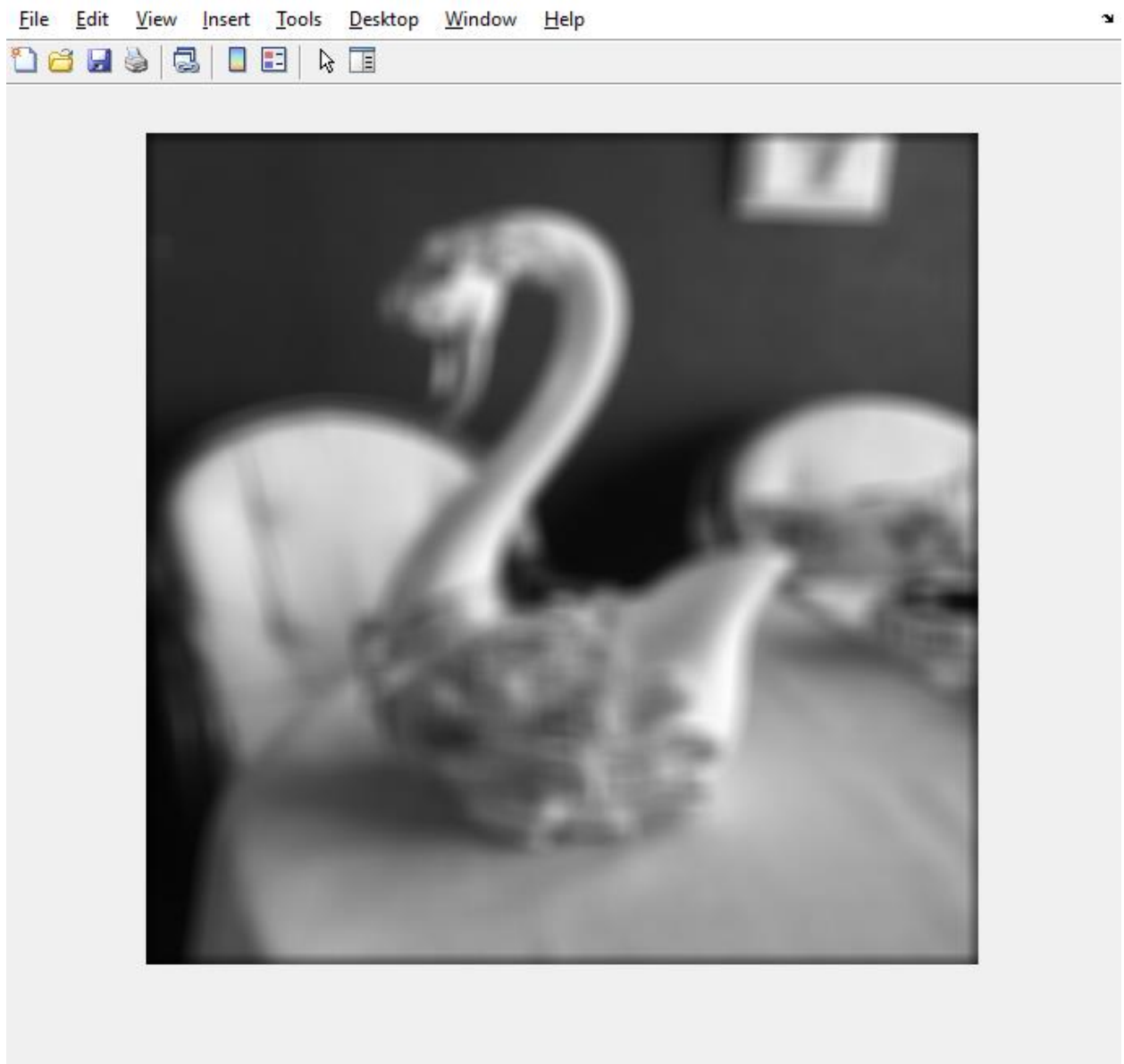
Filtered Image (3x3 Box Filter) (Average Filter)



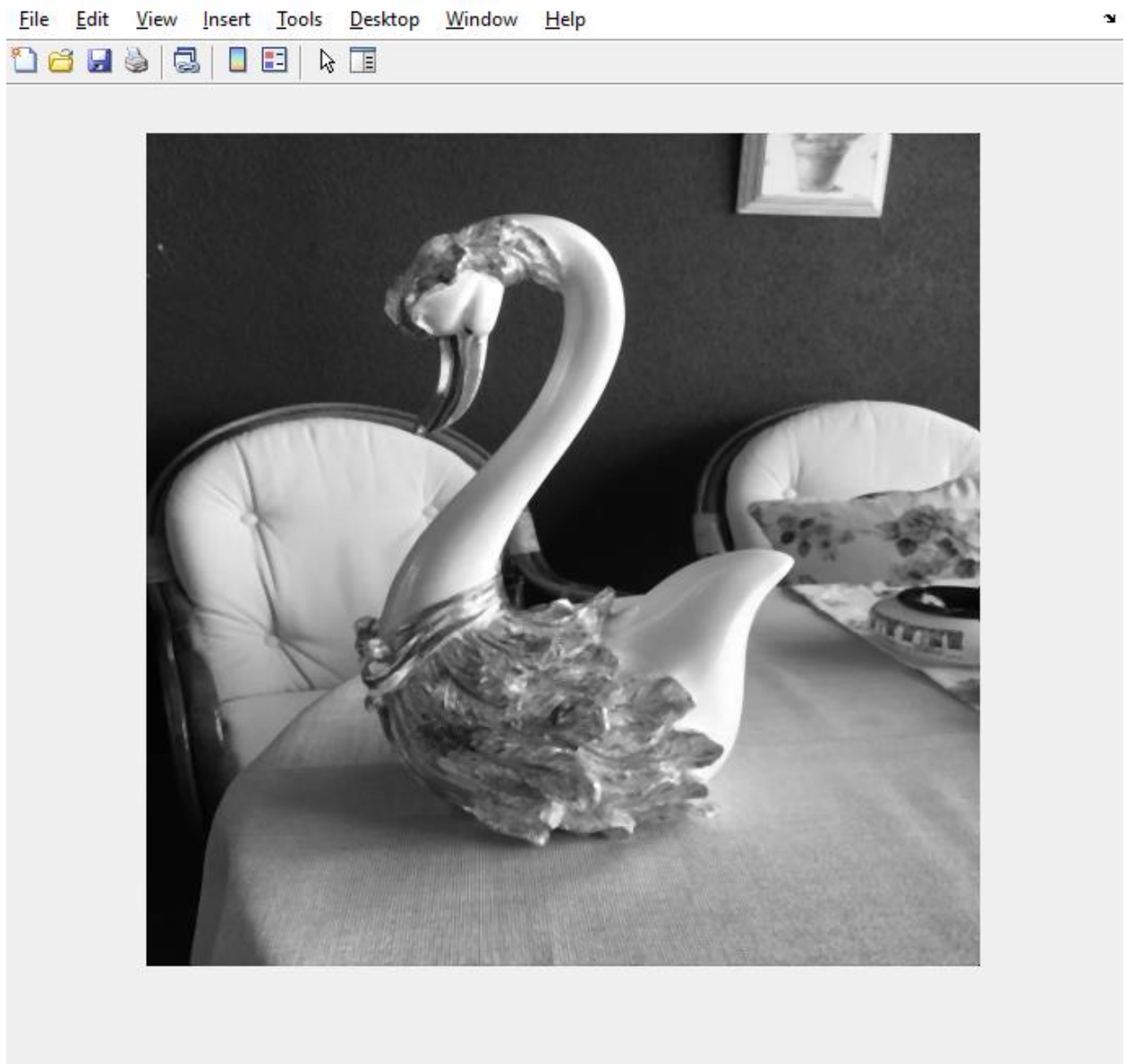
Filtered Image (5x5 Box Filter) (Average Filter)



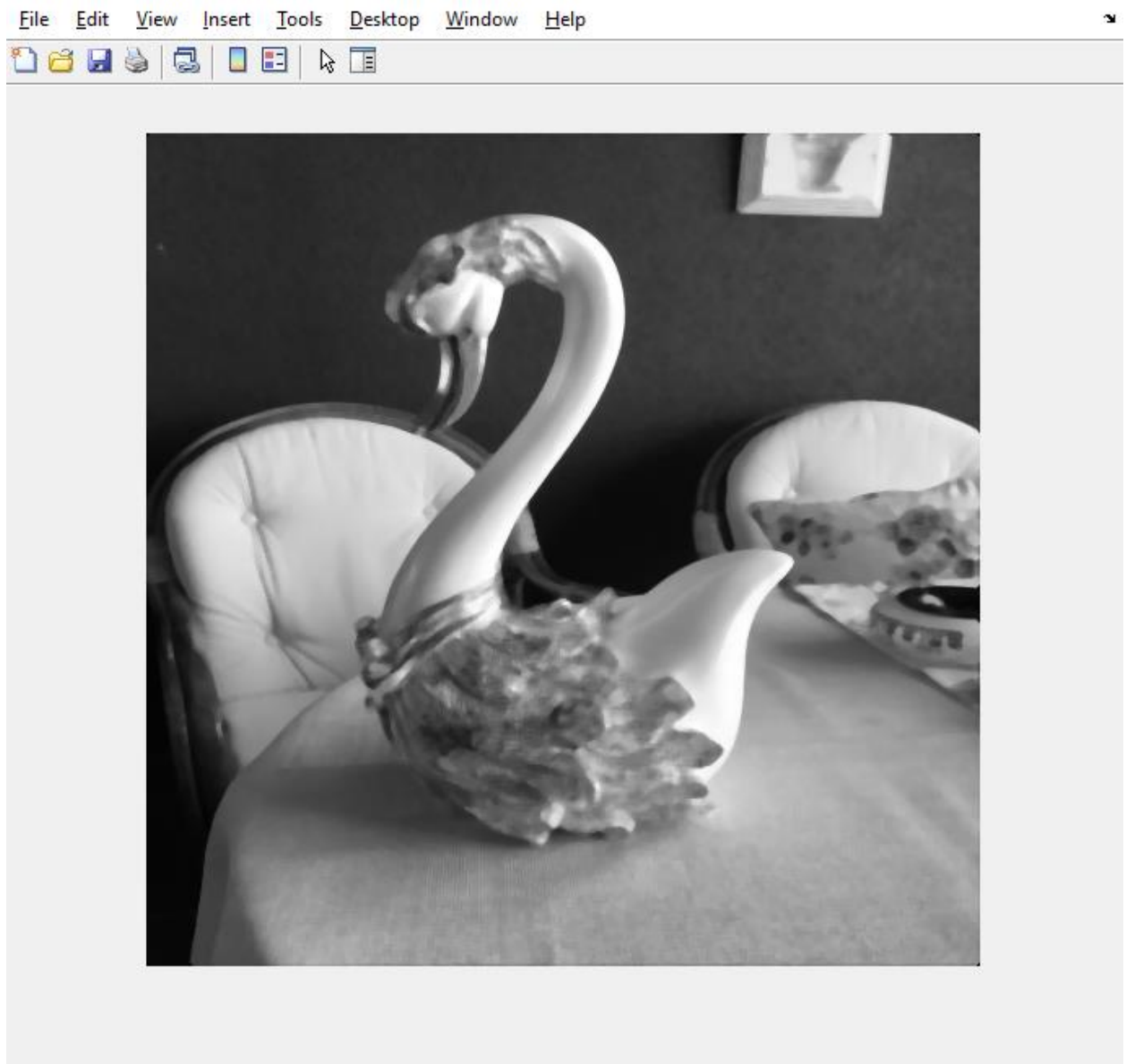
Filtered Image (9x9 Box Filter) (Smoothing)



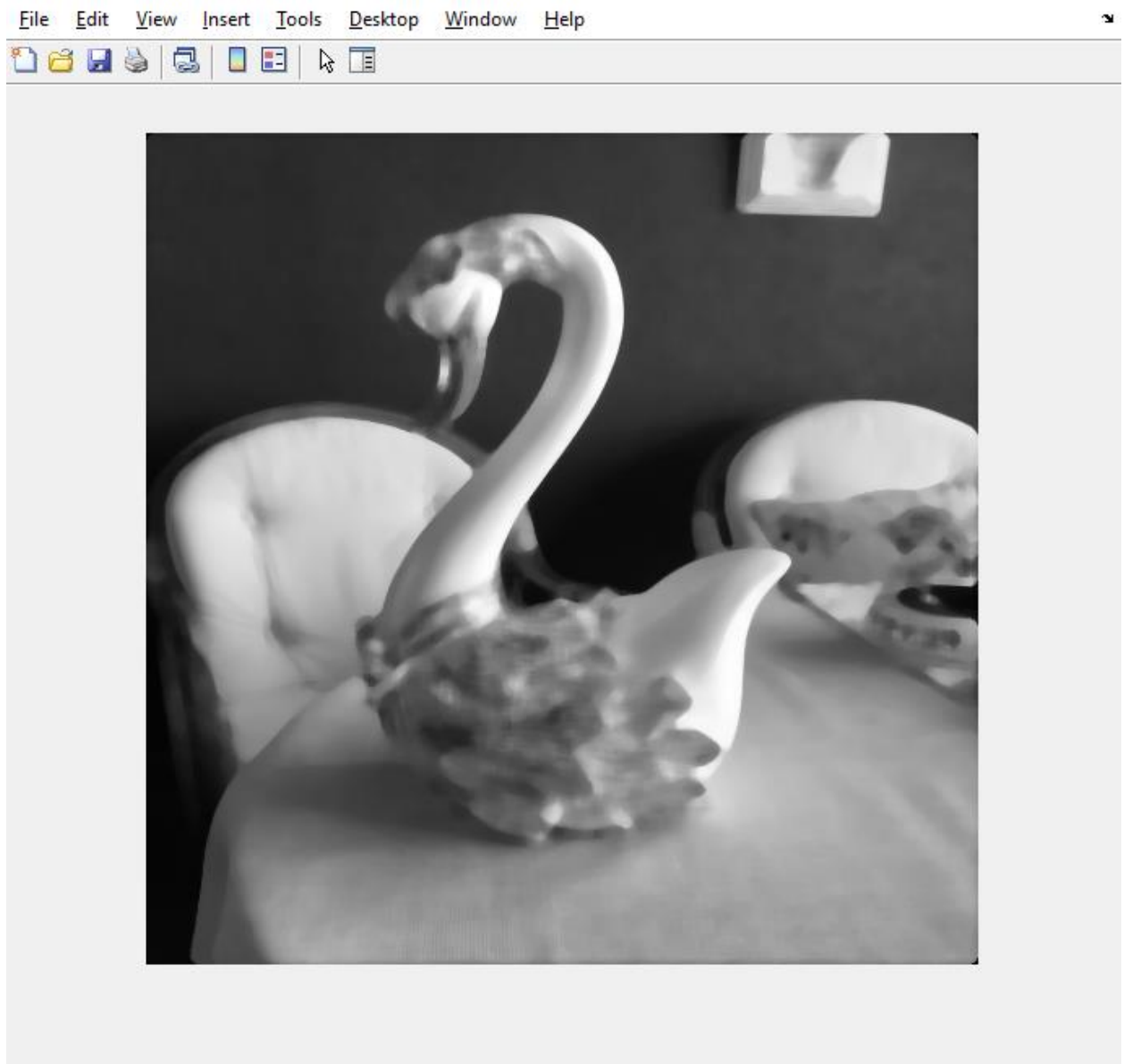
Filtered Image (15x15 Box Filter) (Smoothing)



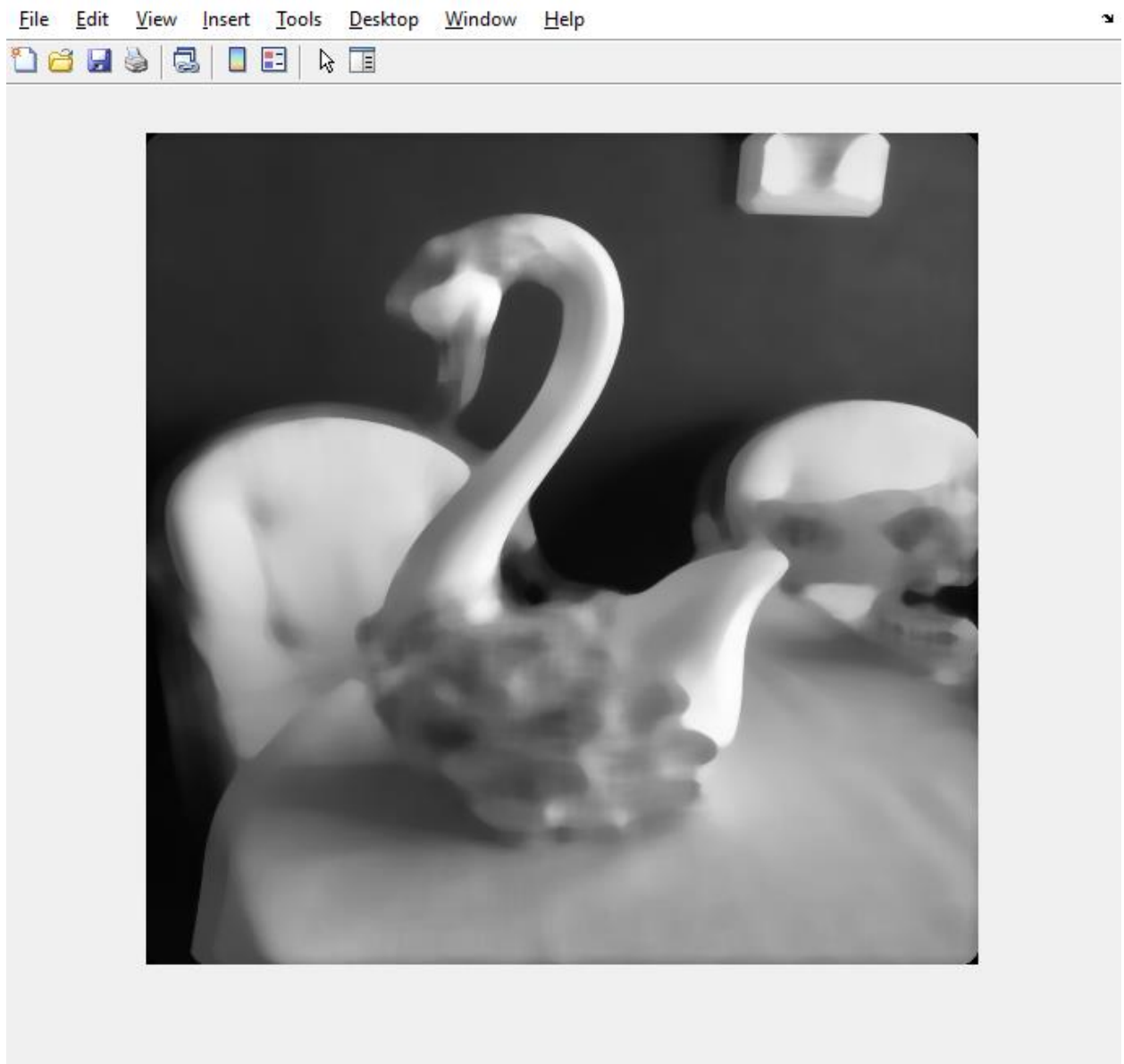
Filtered Gray Image (3x3) (Median Filter)



Filtered Gray Image (5x5) (Median Filter)



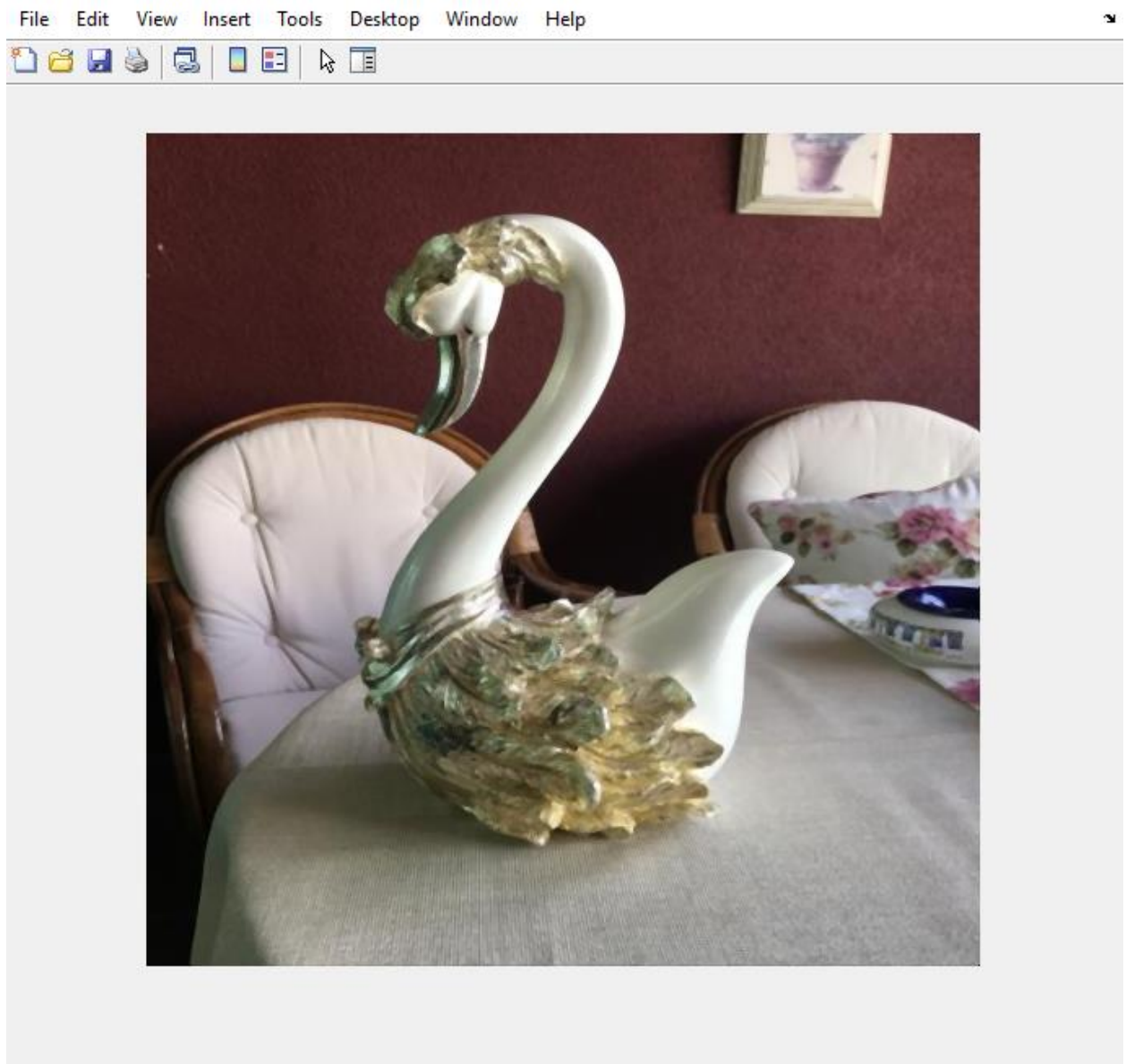
Filtered Gray Image (9x9) (Median Filter)



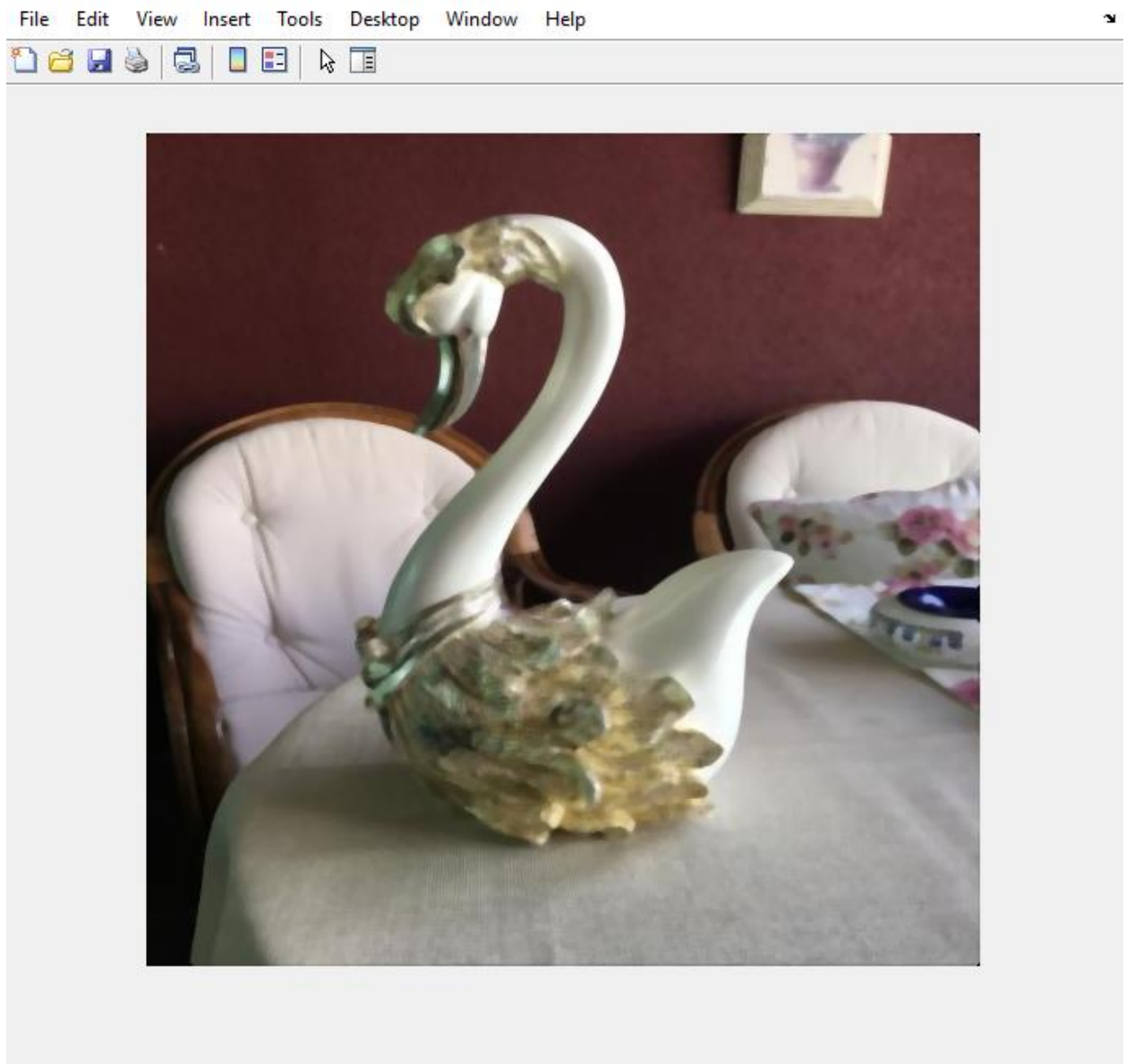
Filtered Gray Image (15x15) (Median Filter)



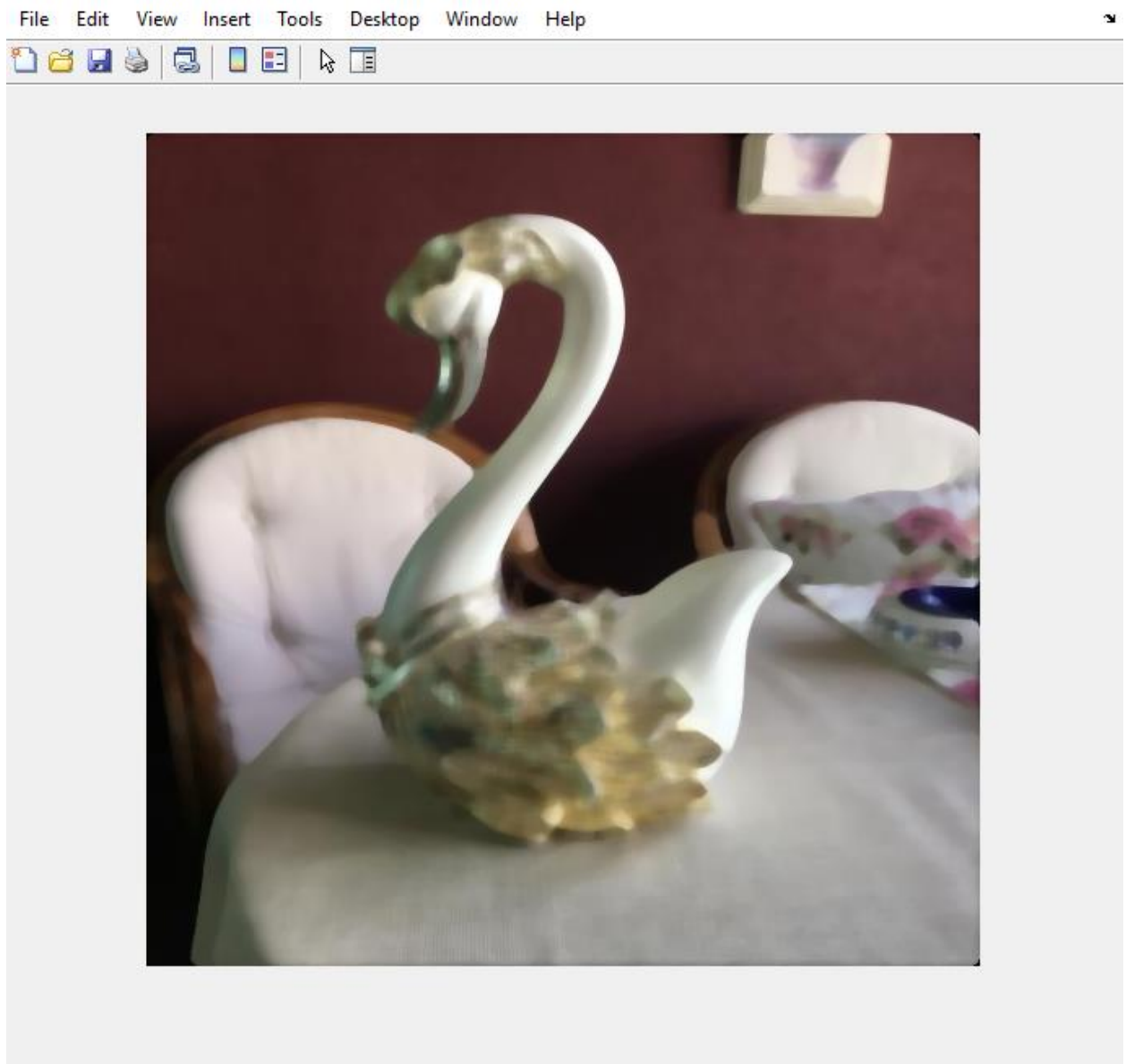
Original RGB Image (512x512)



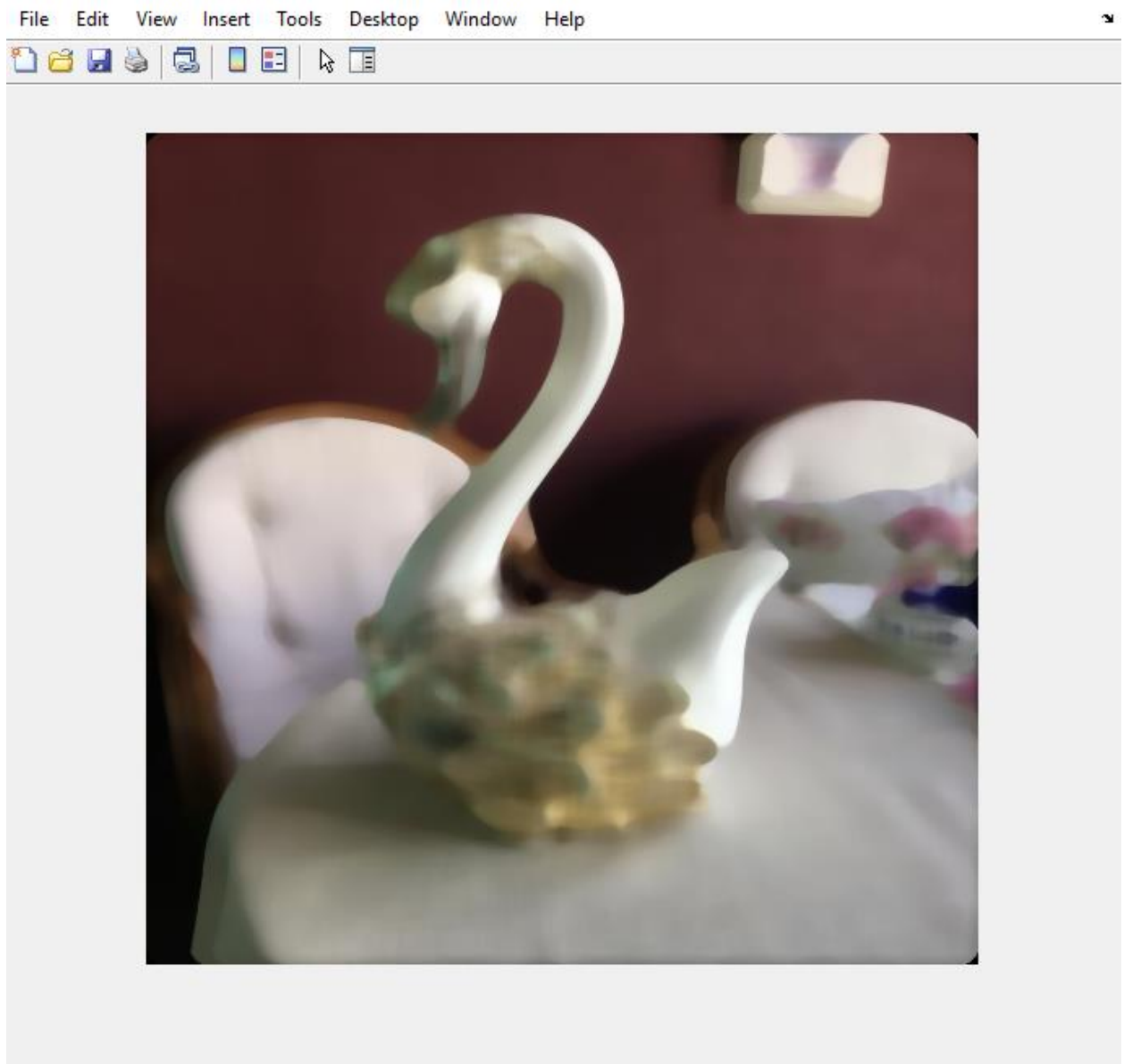
Filtered RGB Image (3x3) (Median Filter)



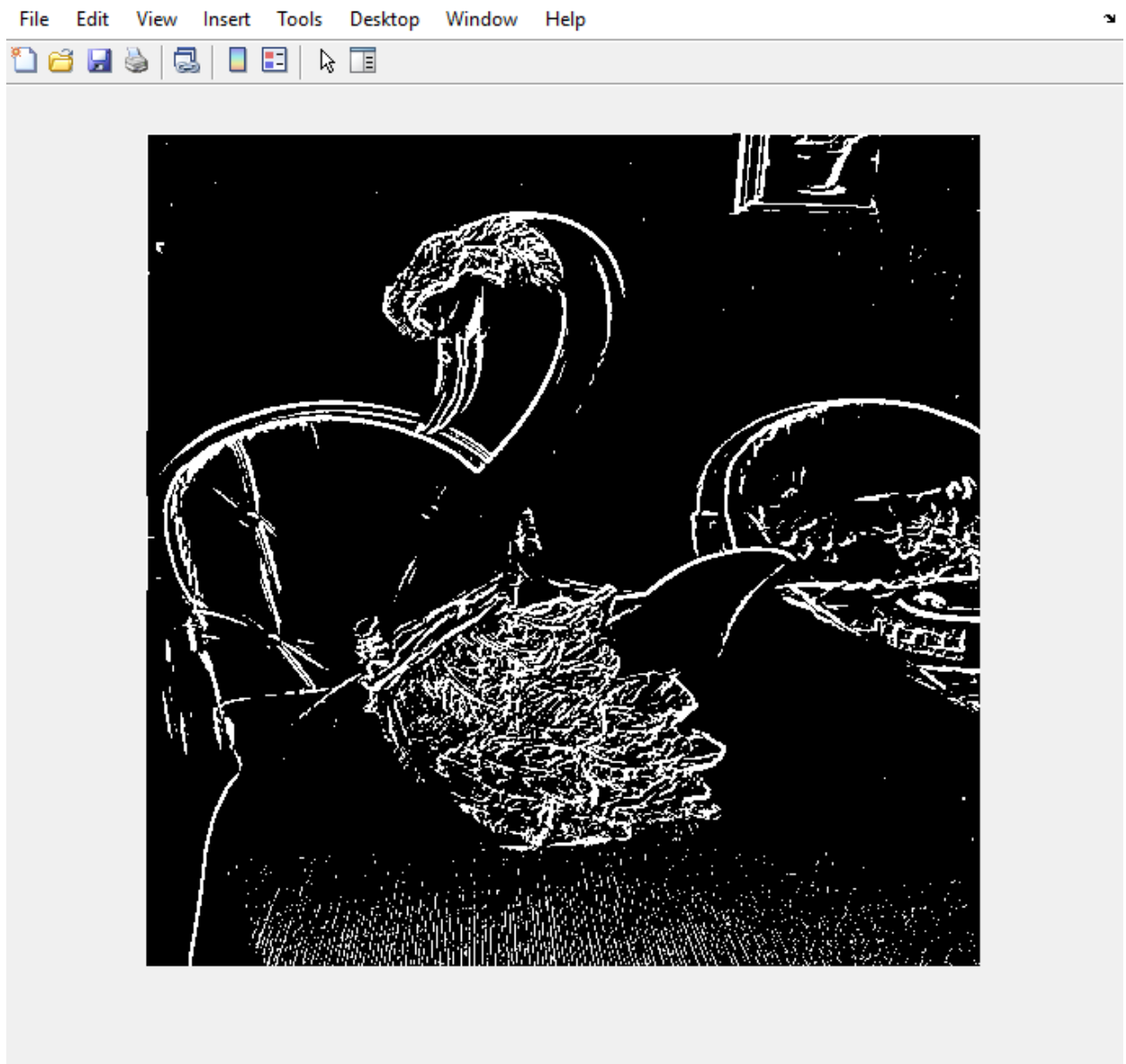
Filtered RGB Image (5x5) (Median Filter)



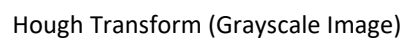
Filtered RGB Image (9x9)



Filtered RGB Image (15x15)



Sobel Edge Detection (Grayscale Image)



Hough Transform (Grayscale Image)