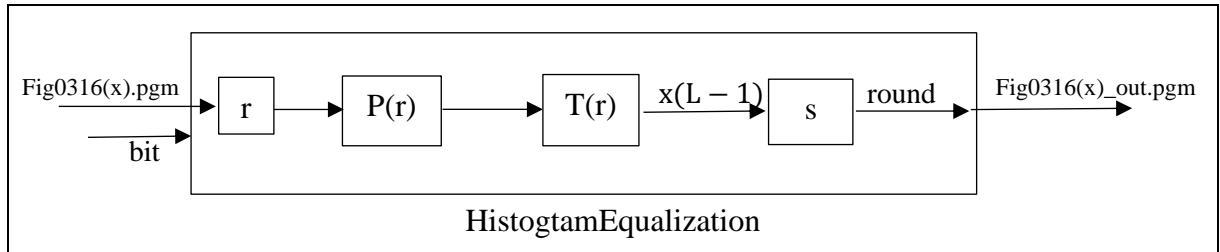


## LAB 02 RAPORU

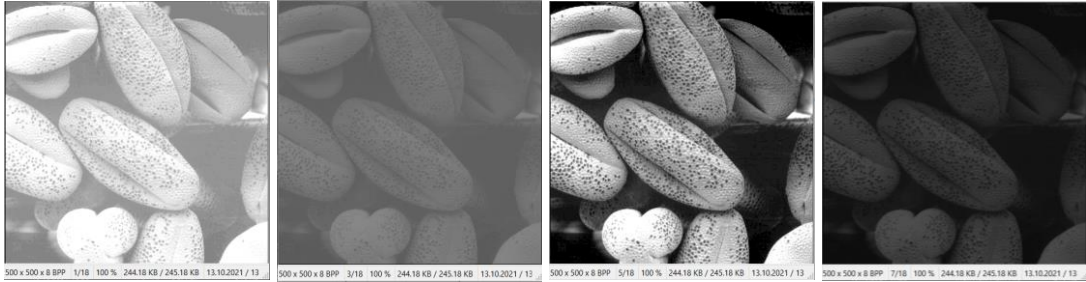
1. Histogram eşitlemede amaç, giriş görüntüsünün pdf değeri ne olursa olsun, çıkış görüntüsünün histogramının uniform dağılıma yakın elde edilmesidir. Bu şekilde görüntünün ton dağılımının homojen olması sağlanır.

Görüntünün histogramının eşitlenmesi için kullanılacak Image HistogramEqualization(int) fonksiyonu yazıldı. Daha sonra bu fonksiyon verilen 4 görüntüye teker teker uygulanarak histogramı iyileştirilmiş görüntüler elde edildi.



Şekil 1. Histogram eşitleme blok şeması.

Şekil 2’de giriş görüntüleri, Şekil 3’te ise histogram eşitleme fonksiyonundan geçtikten sonra elde edilen çıkış görüntüleri görülmektedir.

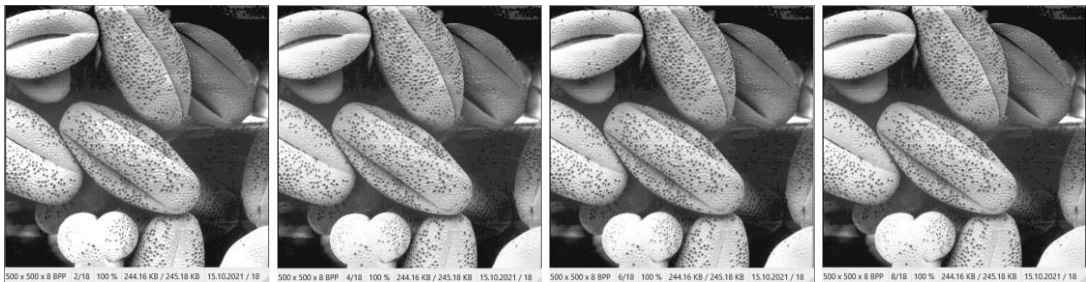


Şekil 2.a. Açık

b. Düşük kontrast

c. Yüksek kontrast

d. Koyu



Şekil 3.a

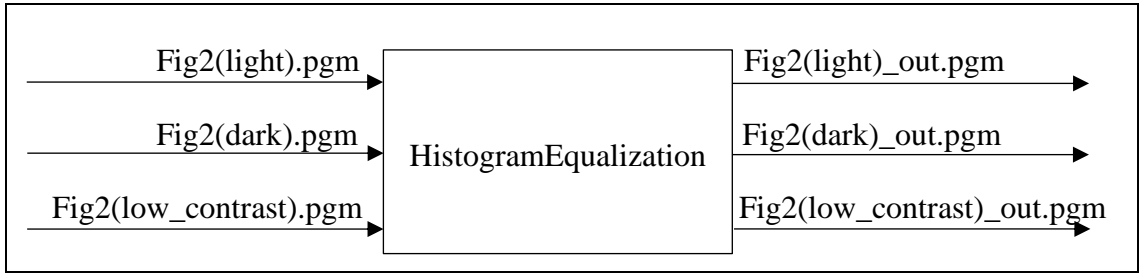
b.

c.

d.

Şekil 2.c ve Şekil 3.c hariç görüntüler giriş ve çıkış görüntüleri arasındaki fark belirgin bir şekilde görülmektedir. Şekil 2.c’deki görüntünün histogramı zaten uniform dağılıma yakın olduğu için çıkış görüntüsüyle aralarında çok azdır. Fakat Şekil 2.a’da açık, 2.b’de orta, 2.d’de ise koyu pikseller yoğunlukta olduğu için, histogram eşitlendikten sonra elde edilen görüntülerin, piksellerin ton dağılımı homojenleştirildiği için, belirgin şekilde iyileşmiş olduğu görülmektedir. Giriş görüntüleri farklı olsa da histogramları eşitlendiği için Şekil 3’teki 4 görüntünün aslında aynı olduğu görülmektedir.

2. Seçilen görüntü; çok parlak, çok karanlık ve düşük kontrastlı olmak üzere 3 farklı hale getirildi. Daha sonra bu görüntüler, birinci adımda üretilen histogram eşitleme fonksiyonu kullanılarak iyileştirildi.



Şekil 4. İkinci adımın blok şeması.

Şekil 4'te bu adımda yapılan işlemlerin blok şeması bulunmaktadır. "HistogramEqualization" blokunda yapılan işlemler Şekil 1'de detaylı şekilde görüldüğü için burada sadece blok halinde verilmiştir.

Şekil 5'te çok parlak, çok karanlık ve düşük kontrastlı 3 giriş görüntüsü, Şekil 6'da ise bu görüntülerin histogramı eşitlendikten sonra elde edilen çıkış görüntüleri görülmektedir.



Şekil 5.a. Çok parlak

b. Çok karanlık

c. Düşük kontrast



Şekil 6.a

b.

c.

Çıkış görüntüleri birbirlerine oldukça yakın olup, aralarında dikkatli bir şekilde incelenmediği takdirde fark edilmeyen çok ufak farklılıklar bulunmaktadır. Bunun sebebi, orijinal görüntüde parlak olan yerlerdeki ufak ton farklılıklarının görüntü parlaklaştırıldıktan sonra, koyu yerlerdeki ufak ton farklılıklarının görüntü koyulaştırıldıktan sonra kaybolması vb. durumlardır.

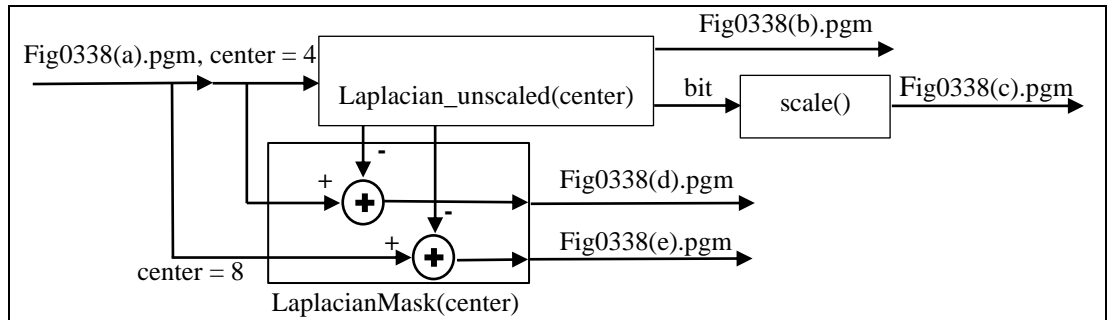
Birbirine yakın tonlarda çok fazla detayı olan görüntüler parlaklaştırıldıktan ya da koyulaştırıldıktan sonra detaylarda kaybolma olacağı için histogram eşitleme işlemi sonucunda bu noktalarda pikseli bir görüntü elde edilebilir.

3. Bu adımda, verilen iki görüntü farklı iki yöntemle keskinleştirildi.

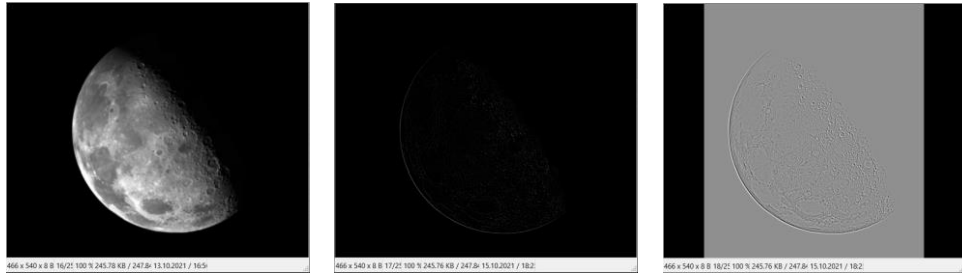
**3.1.** Laplace maskesi ile keskinleştirme işlemi, görüntünün Laplace'ının alınıp daha sonra orijinal görüntüye eklenmesi ile yapılmaktadır.

$$g(x, y) = f(x, y) + c[\nabla^2 f(x, y)] \quad (\text{eşitlik- 1})$$

İlgili işlemlerin yapılabilmesi için, Image Laplacian\_unscaled(int) ve Image LaplacianMask(int) fonksiyonları yazıldı. Her iki fonksiyonda da giriş parametreleri maskelerin merkezindeki pikselin pozitif değeridir. Bu şekilde elemanları farklı iki Laplace maskesi için tek fonksiyon yazılması yeterli oldu. Ölçekleme işlemi için giriş parametresi görüntünün bit bilgisi olan Image scale(int) fonksiyonu oluşturuldu.



**Şekil 7.** Laplace ile keskinleştirme blok şeması.



**Şekil 8.a.** Giriş görüntüsü. **b.** Ölçeklenmemiş Laplace görüntüsü (center = 4) **c.** Ölçeklenmiş Laplace görüntüsü (center = 4)



**d.** Maskeleme sonucu (center = 4)

**e.** Maskeleme sonucu (center = 8)

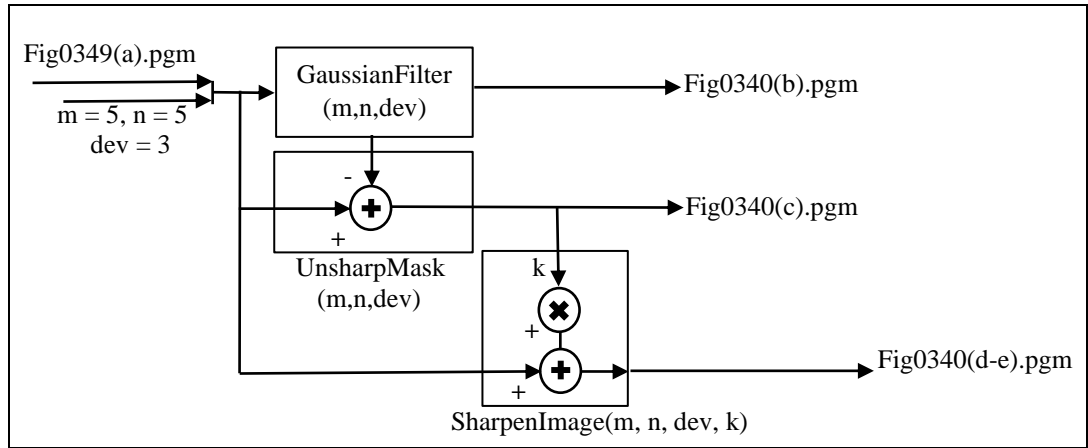
Başlangıçta hafif bulanık olan giriş görüntüsünün Laplace'ı alınarak Şekil 8.b görüntüsü elde edildi. Görüntünün değer aralığı  $[-255, 255]$  oldu ve sıfırdan küçük pikseller siyah olarak alındığı için siyah yoğunluklu görüntü elde edildi. Bu görüntü ölçeklenerek 8.c görüntüsü elde edildi. Şekil 8.d ve 8.e'deki keskinleştirilmiş görüntüler merkez piksel değerleri sırasıyla -4 ve -8 olan Laplace maskeleri ile orijinal görüntünün Laplace'ının alınıp sonra kendisinden çıkarılmasıyla (bkz. eşitlik- 1,  $c = -1$ ) elde edildi. Merkezindeki piksel değeri -8 olan maskenin daha fazla keskinleştirme yaptığı görülmektedir ve Laplace görüntüsüne bakılıyorsa 8.b'dekinden daha çok detay olurdu.

**3.2.** Orijinal görüntüden (şekil 10.a), bulanıklaştırılmış halinin (şekil 10.b) çıkarılmasıyla unsharp maskesi (şekil 10.c) elde edilir. Daha sonra bu maskenin sabitle çarpılıp orijinal görüntüye eklenmesiyle keskinleştirilmiş görüntü (şekil 10.d-e) elde edilir. Sabitin 1'e eşit olduğu durumlarda *unsharp maskeleme*, 1'den büyük olduğu durumlarda ise *highboost filtreleme* yapılmış olur.

$$g_{mask}(x, y) = f(x, y) - \bar{f}(x, y) \quad (\text{eşitlik- 2})$$

$$g(x, y) = f(x, y) + k * g_{mask}(x, y) \quad (\text{eşitlik- 3})$$

Görüntüyü bulanıklaştırma işlemi için giriş parametreleri olarak filtrenin boyunu ve standart sapma değerini alan `Image GaussianFilter(int, int, float)` fonksiyonu oluşturuldu.



**Şekil 9.** Unsharp maskesiyle keskinleştirme blok şeması.



**Şekil 10.a.** Orijinal görüntü. **b.** Gauss filtresi ile bulanıklaştırılmış görüntü. **c.** Görüntünün unsharp maskesi.



**d.** Unsharp maskeleme sonucu ( $k = 1$ ).

**e.** Highboost filtreleme sonucu ( $k = 4.5$ ).

Şekil 10.d ve 10.e incelendiğinde,  $k$  büyüdükçe keskinlik miktarının arttığı görülmektedir. Bunun sebebi, görüntüye eklenen Şekil 10.c'deki unsharp maskesinin etkisinin  $k$  büyüdükçe artmasıdır.

Keskinleştirilmiş görüntüde, eğimin değiştiği noktalardaki vurgu  $k$  değeriyle orantılı olarak değişmektedir. Vurgu ne kadar fazlaysa görüntü o kadar keskinleştirilmiştir.