

# LAB 01 RAPORU

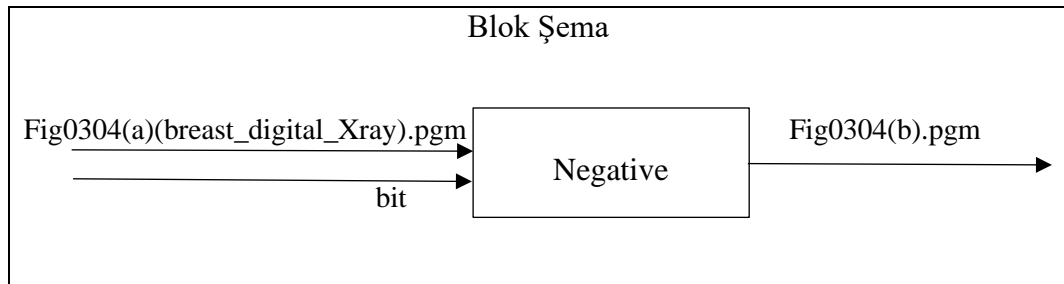
1. Birinci adımda, verilen 6 görüntüden istenilen görüntülerin elde edilmesi için gerekli iyileştirme fonksiyonları oluşturuldu.

- 1.1. Şekil 2.a'daki görüntünün negatifi alınarak Şekil 2.b'deki görüntü elde edildi.

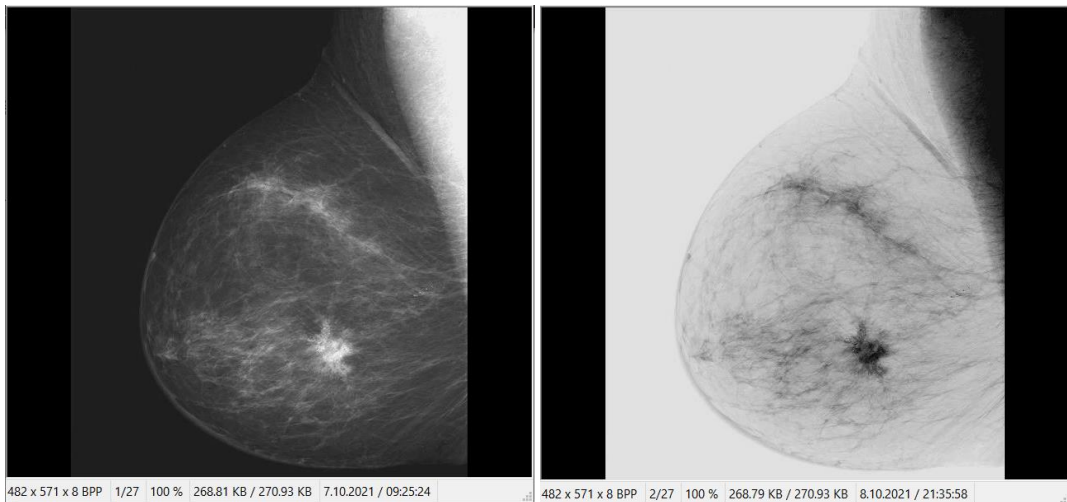
$$s = L - 1 - r \quad (\text{eşitlik- 1})$$

Görüntünün negatifi almak için `Image Negative(int)` fonksiyonu oluşturuldu. Fonksiyon, giriş parametresi olarak görüntünün bit bilgisini (n) almaktadır. Bu bit bilgisi  $L = 2^n$  değerini hesaplamak için kullanılmaktadır.

Yeni piksel değerlerini hesaplamak için eşitlik- 1'de verilen işlem yapıldı. Eşitlikte r giriş görüntüsünün piksel değeri, s oluşturulan görüntünün piksel değeridir.



**Şekil 1.** Negatif görüntü blok şeması.



**Şekil 2.a.** Giriş görüntüsü (figure 3.4.a) **b.** Çıkış görüntüsü (figure 3.4.b)

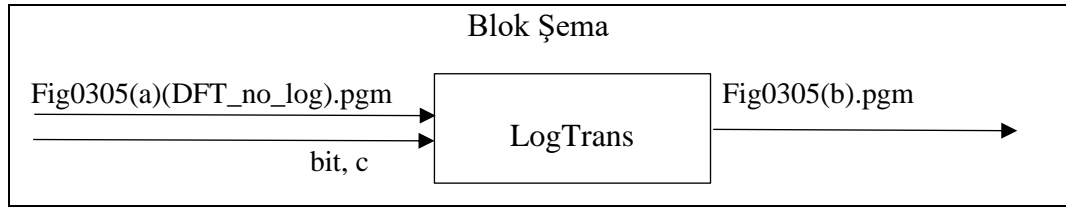
Görüntünün negatifi alındığında, o piksel değeri gri ton seviyesine göre tam tersi değeri alır. Şekil 2.b'deki görüntü incelendiğinde, Şekil 2.a'daki açık noktaların koyu, koyu noktaların açık olduğu görülmektedir. Fakat, seçilen iki piksel arasındaki değer farkı her iki görüntüde de aynıdır.

1.2. Şekil 4.a'daki görüntünün logaritmik dönüşümü alınarak Şekil 4.b'deki görüntü elde edildi.

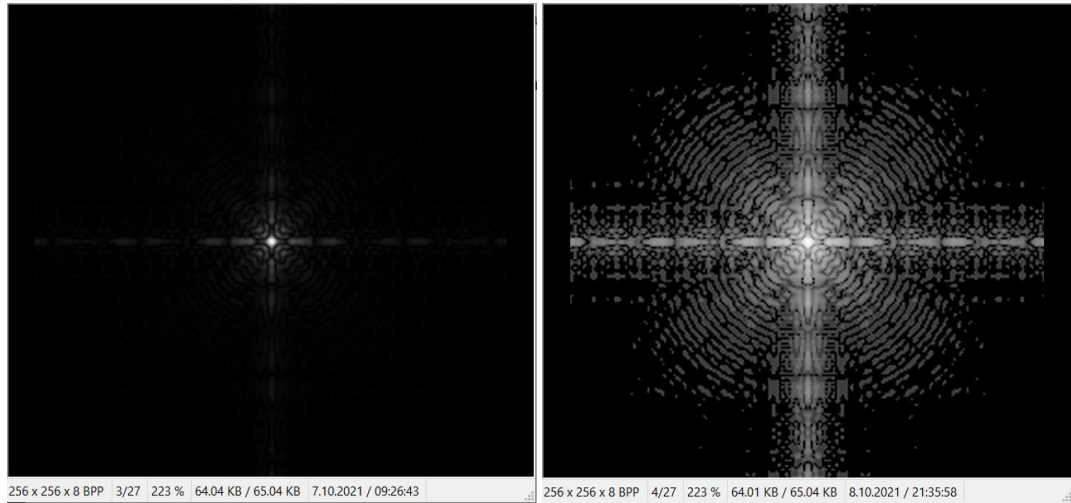
$$s = c \log(1 + r) \quad (\text{eşitlik- 2})$$

Görüntünün logaritmik dönüşümünü almak için Image logTrans(int, float) fonksiyonu oluşturuldu. Fonksiyon, giriş parametresi olarak görüntünün bit bilgisini (n) ve sabit c değerini almaktadır. Bit bilgisi gri ton seviyesini hesaplamak için kullanılmaktadır.

Yeni piksel değerlerini hesaplamak için eşitlik- 2'de verilen işlem yapıldı. Daha sonra bu piksel değerleri, hesaplanan gri seviyesi kullanılarak normalize edilerek Şekil 4.b'deki görüntü elde edildi.



**Şekil 3.** Logaritmik dönüşüm blok şeması.



**Şekil 4.a.** Giriş görüntüsü (figure 3.5.a) **b.** Çıkış görüntüsü (figure 3.5.b)

Logaritmik dönüşüm, koyu noktaların parlaklığını artırmak için kullanılmaktadır. Şekil 4.a'da orta noktadan uzaklaştıkça görülmesi zorlaşan koyu detaylar, Şekil 4.b'de parlaklığı artırıldığı için artık görülmektedir.

Başlangıçta, piksel değerleri normalize edilmeden görüntü oluşturuldu. Dolayısıyla oluşması beklenen görüntü yerine oldukça karanlık, görünmesi zor bir görüntü oluştu. Bu durum, piksel değerlerinin normalize edilmemesinden kaynaklanmaktadır.

Şekil 4.a'daki görüntünün piksel değerlerinin logaritması alındığında elde edilen değerler çok küçüktür dolayısıyla normalize edilmediğinde değerlerin siyaha yakın çıkması normal bir durumdur. Normalize etmek için yeni piksel değerleri gri ton seviyesi ile çarpılıp, eski piksellerin en büyük değerinin logaritmasına bölündü.

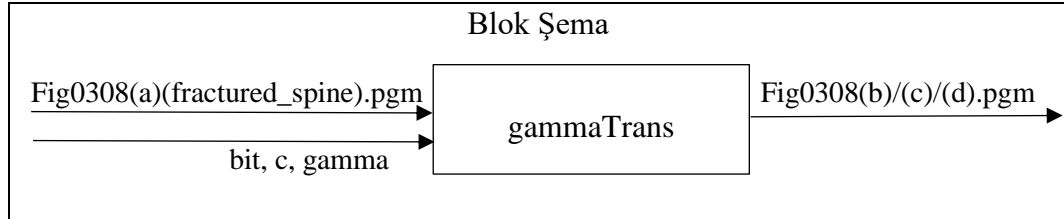
$$s_{normalized} = s 2^n \log(1 + r_{max}) = c \log(1 + r) 2^n \log(1 + r_{max}) \quad (\text{eşitlik- 3})$$

**1.3.** Şekil 6.a'daki görüntünün farklı gamma değerleri için gamma dönüşümü alınarak Şekil 6.b/c/d'deki görüntüler elde edildi.

$$s = cr^\gamma \quad (\text{eşitlik- 4})$$

Görüntünün gamma dönüşümünü almak için Image logTrans(int, float, float) fonksiyonu oluşturuldu. Fonksiyonun giriş parametreleri görüntünün bit sayısı ile c ve  $\gamma$  sabitleridir.

Yeni piksel değerlerini hesaplamak için eşitlik- 4'te verilen işlem yapıldı. Elde edilen s değerleri, yine s değerleri kullanılarak normalize edildi.



**Şekil 5.** Gamma dönüşümü blok şeması.



**Şekil 6.a.**Giriş görüntüsü

**b.** c = 1,  $\gamma = 0.6$

**c.** c = 1,  $\gamma = 0.4$

**d.** c = 1,  $\gamma = 0.3$

Gamma dönüşümünde,  $\gamma < 1$  için,  $\gamma$  küçüldükçe Şekil 6'da da görüldüğü gibi koyu pikseller açığa çıkar. Bu durumda logaritmik dönüşüme benzer etkisi vardır fakat gamma dönüşümünün dönüşüm eğrisi daha esnektir.

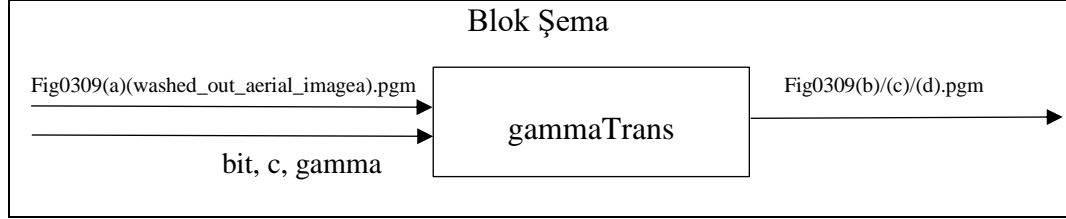
Eşitlik- 4 incelendiğinde, 1'den küçük gama değerleri için, s değerinin r değerine göre küçük çıkacağı görülmektedir. Bu sebeple piksel değerleri normalize edilmeden oluşturulan görselin açığa çıkması yerine koyulaşacağı, formül incelendiğinde görülmektedir. Bu durumu test etmek için piksel değerleri normalize edilmeden gama dönüşümü yapıp, görüntünün koyulaşmış şekilde elde edildiği gözlemlendi.

Piksel değerlerini normalize etmek için, hesaplanan s değerlerinin minimum ve maksimum değerleri arasındaki fark (gri ton seviyesi) hesaplandı. Daha sonra bu s değerleri giriş görüntüsünün gri ton seviyesi ile çarpılıp kendi gri ton seviyesine bölündü.

$$s_{normalized} = \frac{(2^n - 1)s}{s_{max} - s_{min}} \quad (\text{eşitlik- 5})$$

**1.4.** Şekil 8.a'daki görüntünün farklı gamma değerleri için gamma dönüşümü alınarak Şekil 8.b/c/d'deki görüntüler elde edildi.

Görüntünün gamma dönüşümünü almak için oluşturulan Image logTrans(int, float, float) fonksiyonu kullanıldı.



**Şekil 7.** Gamma dönüşümü blok şeması.



**Şekil 8.a.** Giriş görüntüsü



**b.**  $c = 1, \gamma = 3$



**c.**  $c = 1, \gamma = 4$



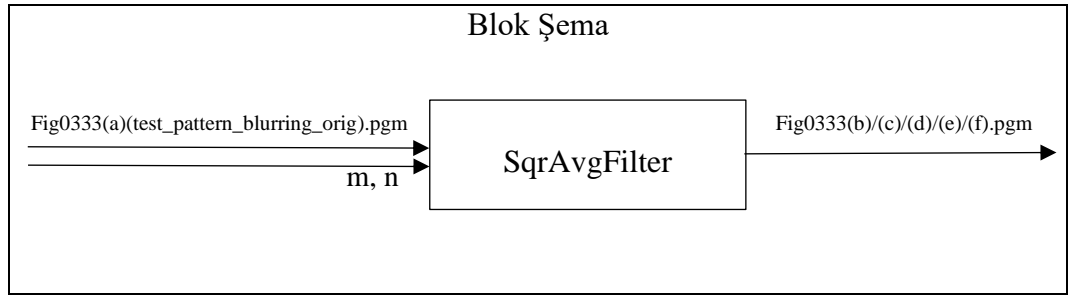
**d.**  $c = 1, \gamma = 5$

Gamma dönüşümünde,  $\gamma > 1$  için,  $\gamma$  büyüdükçe Şekil 8'de de görüldüğü gibi açık pikseller koyulaşır. Dönüşüm eğrisi  $\gamma < 1$ 'in dönüşüm eğrisinin tam tersi şeklindedir.  $\gamma = 1$  eğrisine göre iki durumun eğrileri birbirine simetriktr.  $\gamma = 1$  olduğunda görüntü aynı kalır.

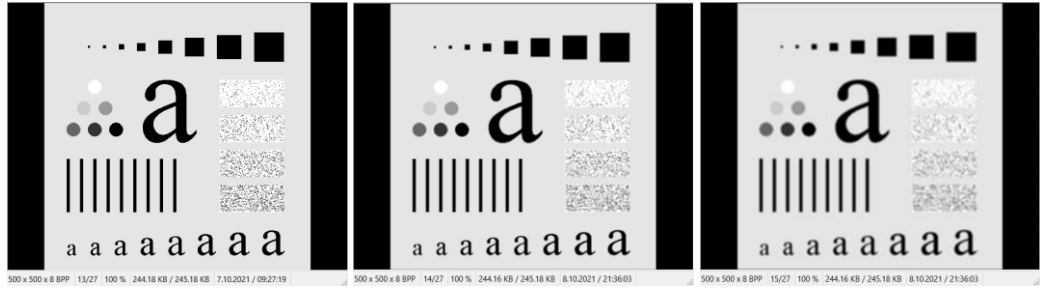
1.5. Şekil 10.a'daki görüntüye ortalama filtresi uygulanarak Şekil 10.b/c/d/e/f'deki yumuşatılmış/bulanıklaştırılmış görüntüler elde edildi.

$$g(x, y) = \sum_{s=-a}^a \sum_{t=-b}^b w(s, t) f(x + s, y + t) \quad (\text{eşitlik- 6})$$

Görüntüye ortalama filtresini uygulamak için Image SqrAvgFilter(int,int) fonksiyonu oluşturuldu. Fonksiyonun giriş parametreleri m ve n filtre maskesinin büyüklük (m x n) bilgisidir. Bu giriş parametrelerinden eşitlik- 6'daki a ve b değerleri hesaplanarak yeni görüntünün piksel değerleri (g(x, y)) ayarlandı. Kullanılan filtre ortalama filtresi olduğu için w(s, t) fonksiyonu sabit olup,  $\frac{1}{m \times n}$ 'e eşittir.

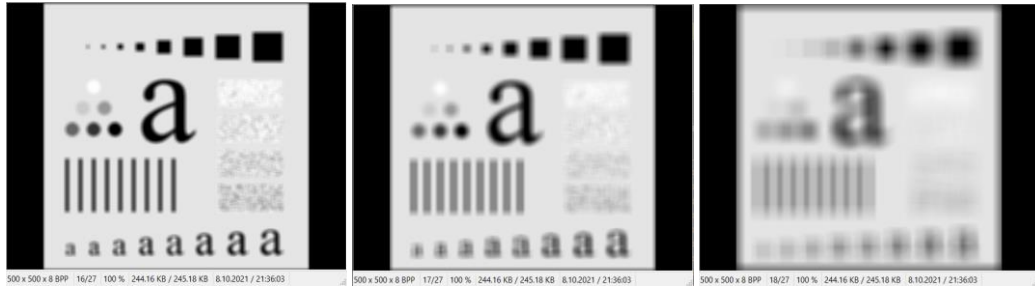


**Şekil 9.** Görüntü yumuşatma/bulanıklaştırma blok şeması.



**Şekil 10.a.**Giriş görüntüsü **b.** m = 3, n = 3

**c.** m = 5, n = 5



**d.** m = 9, n = 9

**e.** m = 15, n = 15

**f.** m = 35, n = 35

Filtrenin boyu büyüdükçe, değerlerinin ortalamasının alındığı piksel miktarı da arttığı için, görüntünün daha da bulanıklaştığı Şekil 10'da görülmektedir.

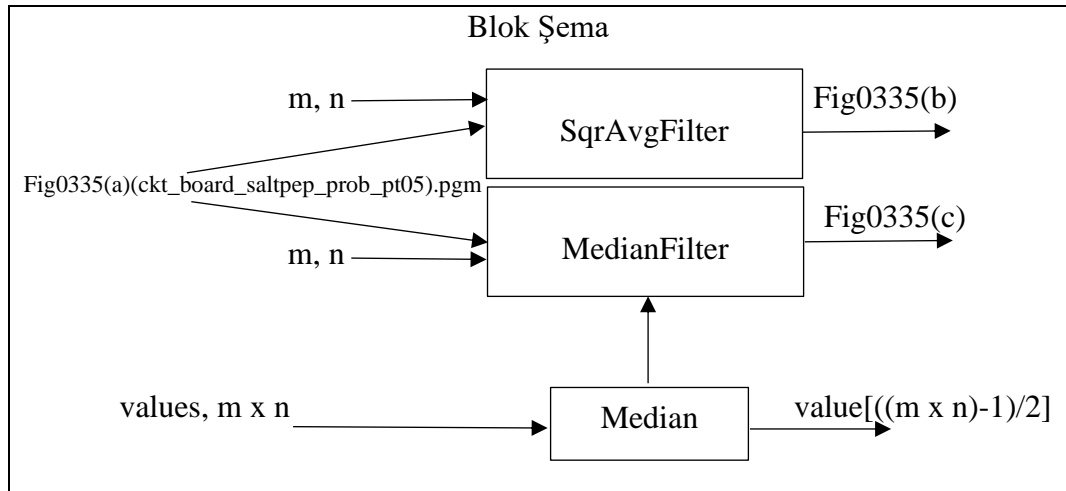
Kenardaki pikseller için hesaplama yapılırken, dışarıda kalan noktadaki değerler 0 olarak alındı. Bu sebeple maske boyu arttıkça ortalama hesabına giren 0 değerleri de artacağından, ortalama değer küçülmekte ve kenarlarda koyulaşmalar meydana gelmektedir.



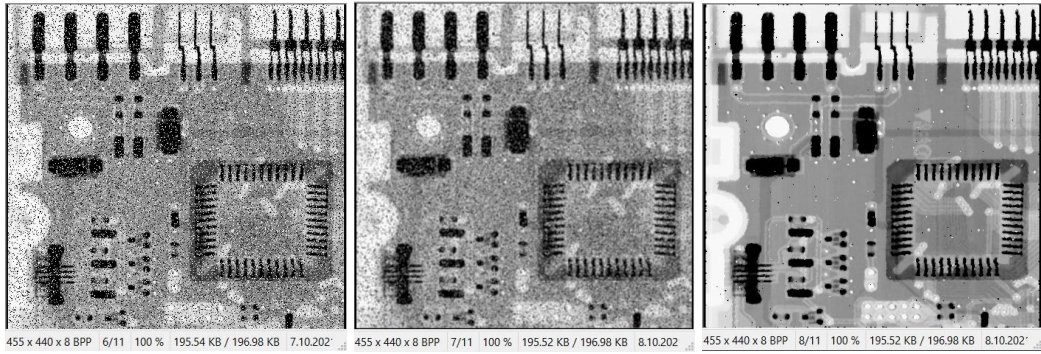
**1.6.** Şekil 12.a görüntüsündeki gürültüyü temizlemeyek için görüntüye ortalama filtresi uygulanarak Şekil 12.b, medyan filtresi uygulanarak Şekil 12.c görüntüleri oluşturuldu.

Medyan filtresini uygulamak için, `Image MedianFilter(int,int)` fonksiyonu oluşturuldu. Fonksiyonun giriş parametreleri ortalama filtre fonksiyonunda olduğu gibi filtre maskesinin büyüklük bilgisidir

İlgili piksellerin medyan değerini almak için, `float Median(float[],int)` fonksiyonu oluşturuldu. Fonksiyonun giriş parametreleri, piksel değerlerinden oluşan dizi ve dizinin eleman sayısıdır. Çıkışta dizinin medyanını döndürür.



**Şekil 11.** Gamma dönüşümü blok şeması.



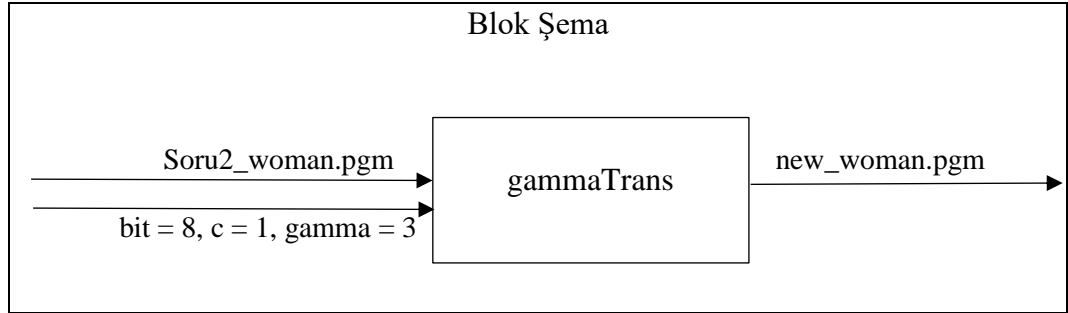
**Şekil 12.a.**Giriş görüntüsü **b.** Ortalama filtre (3x3) **c.** Medyan Filtre (3x3)

Şekil 12.b incelendiğinde, ortalama filtrenin giriş görüntüsündeki gürültüyü azalttığı fakat görüntünün hala gürültülü olduğu görülmektedir. Filtrenin yumuşatma oranı yeterli değil fakat maske boyutu artırıldığında da görüntü bulanıklaşacağı için istenilen sonuç yine elde edilmeyecektir.

Şekil 12.c incelendiğinde ise, gürültünün oldukça iyi temizlemesinin yanı sıra görüntüyü bulanıklaştırmadığı görülmektedir. Aynı maske boyutu kullanıldığında medyan filtre gürültüyü temizlemekte ortalama filtreden çok daha iyi sonuç vermiştir.

2. Bu adımda, kontrast sorunu olan 3 görüntü bulunup, bu görüntüler iyileştirildi.

2.1. İnternet üzerinden düşük kontrastlı Şekil 14.a'daki kadın görüntüsü bulundu. Görüntü düşük kontrastlı, açık tonlu bir görüntü olduğu için  $\gamma > 1$  olacak şekilde gamma dönüşümünün kullanılmasının, iyileştirme için uygun bir çözüm olacağı düşünüldü. Çeşitli  $\gamma$  değerleri denenerek,  $\gamma = 3$  alınmasına karar verildi.



**Şekil 13.** İyileştirilmiş görüntü için blok şema.



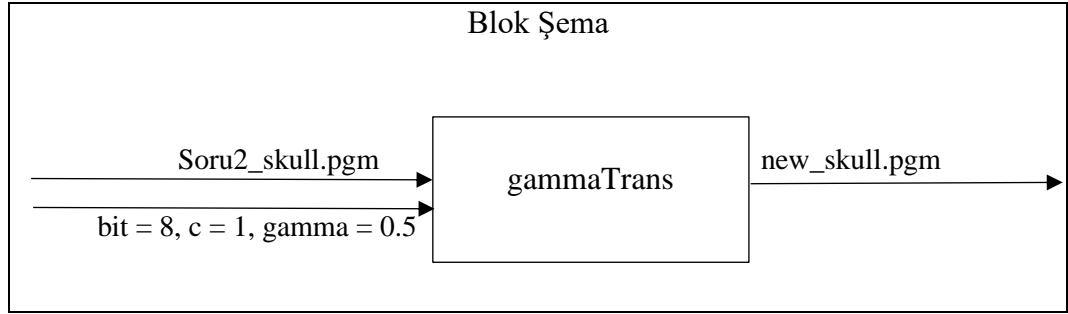
**Şekil 14.a.** Orijinal görüntü

**b.** İyileştirilmiş görüntü

Şekil 14.a ve b incelendiğinde, b görüntüsünün daha fazla tona sahip olduğu görülmektedir. Tonun fazlalığı sayesinde, özellikle cildin ve şapkanın tülünün detayları daha belirginleşmiştir.

Gürültü ve benzeri herhangi bir problem olmadığı için, ekstra bir fonksiyon kullanılmasına ihtiyaç duyulmadı. Gürültü olsaydı, medyan filtre ile bu problem giderilebilirdi.

**2.2.** İnternet üzerinden yüksek kontrastlı Şekil 16.a'daki kuru kafa görüntüsü bulundu. Görüntü yüksek kontrastlı, koyu bir görüntü olduğu için  $\gamma < 1$  olacak şekilde gamma dönüşümünün kullanılmasının, iyileştirme için uygun bir çözüm olacağı düşünüldü.  $\gamma = 0.5$  alındı.



**Şekil 15.** İyileştirilmiş görüntü için blok şema.



**Şekil 16.a.** Orijinal görüntü

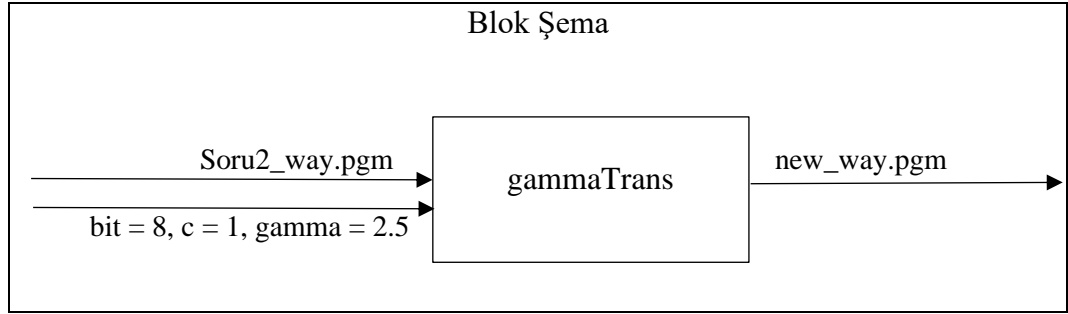
**b.** İyileştirilmiş görüntü

İyileştirilmiş Şekil 16.b görüntüsünde kuru kafanın daha fazla görünmektedir ve orijinal görüntüde görünmeyen gölgesi ortaya çıkmıştır. İyileştirilmiş görüntüde, orijinal görüntüden daha fazla detay vardır fakat  $\gamma$  değeri fazla küçültülürse, pikseller fazla açıklaşacağı için bu sefer de koyu renkli detaylar kaybedilebilir.

Gürültü ve benzeri herhangi bir problem olmadığı için, ekstra bir fonksiyon kullanılmadı.



**2.3.** İnternet üzerinden düşük kontrastlı Şekil 18.a'daki görüntü bulundu. Görüntü düşük kontrastlı, açık tonlu bir görüntü olduğu için  $\gamma > 1$  olacak şekilde gamma dönüşümünün kullanılmasının, iyileştirme için uygun bir çözüm olacağı düşünüldü.  $\gamma = 2.5$  alınmasına karar verildi.



**Şekil 17.** İyileştirilmiş görüntü için blok şema.



**Şekil 18.a.** Orijinal görüntü

**b.** İyileştirilmiş görüntü

2.1'deki gibi düşük kontrastlı görüntüye, gamma dönüşümü ile daha fazla ton ve detay kazandırıldı. Detayların arttığı, bulutlar ve kumdaki adım izlerinden açıkça belli olmaktadır.

Diğer iki görüntüde olduğu gibi, iyileştirme işlemi yapılırken gürültü vb. sorunlarla karşılaşmadığı için, görüntünün iyileştirilmesi için gamma dönüşümü kullanılması yeterli oldu.

Maskeleme filtreleri için oluşturulan fonksiyonlar dışında, ton bazındaki fonksiyonlarda gamma dönüşümünün oldukça kullanışlı ve esnek olduğu görüldü. Negatif fonksiyonu bu üç görselde iyileştirme için kullanılamadı fakat x-ray görüntülerinde bazı detayların daha iyi incelenmedi için oldukça kullanışlı bir yöntemdir.  $\gamma < 1$  için gamma dönüşümü yerine logaritmik dönüşüm de kullanılabilir fakat dönüşüm eğrisi sabit olduğundan gamma dönüşümü kadar kullanışlı değildir.