

1. GİRİŞ

Geçmişin anımsanması, insanların ve ülkelerin geleceği açısından büyük önem taşır. Çünkü toplumlar geçmişe bakarak yollarını çizerler. Geleceği ve yeni nesilleri bu sayede şekillendirirler. Geçmişin anımsanmasında ise tarihi eserlerin ve tarihi eser görüntülerinin büyük önemi vardır. Türkiye, tarihi ve kültürel mirası ile çevre değerleri bakımından dünyanın en zengin ülkelerindedir. Bu zenginliğin korunması, yaşatılması ve geleceğe aktarılması için kullanılabilir kaynaklar son derece sınırlıdır. Kültürel miras olarak adlandırdığımız taşınmaz kültür varlıklarının korunması, bakımı, yaşamsal mekânlar haline getirilmesi ve gelecek kuşaklara aktarılması yönünde çalışmalar yapmak, teknolojiyi ve ürünlerini kullanarak yeni metotlar geliştirmek hem bizim için hem de bizden sonraki nesillerin yapacağı çalışmalar için faydalı olacaktır.

Tarihi eserlerimizi yaşatmak ve yapılan restorasyon çalışmalarına katkıda bulunmak amacıyla yapılan bu tez çalışması sekiz bölümden oluşmaktadır. Birinci bölümde bu tezin oluşturulma amacından bahsedilerek tarih öncesi resimlerin ve sanat eserlerinin görüntü kalitelerinin artırılması ve eski resimlerde yılların bırakmış olduğu bulanıklıkların, bozuklukların, gürültülerin, pasların, lekelerin vb. sorunların giderilmesinin tarihimizi korumamız açısından ne gibi faydalar sağlayacağı açıklanmaktadır.

İkinci bölümde; insanın bir görüntüyü ve görüntüdeki gürültüyü nasıl algıladığı anlatılarak, beynin görmedeki rolü, çağrışım alanı ve gürültünün matematiğinden kısaca bahsedilmektedir. Ayrıca, görüntüde meydana gelen bulanıklığın ve kontrastın gürültü üzerinde nasıl etkiler meydana getirdiği ve görüntülerde yapay olarak oluşturulabilecek uzaysal gürültü modelleri anlatılmaktadır.

Üçüncü bölümde; görüntüdeki gürültünün etkisini azaltan görüntü işleme algoritmalarından bahsedilmekte ve kullanılan algoritmalar hakkında teorik ve

matematiksel bilgiler verilmektedir. Görüntüdeki gürültünün etkisini azaltmak için kullanılan algoritmalarından Median Filtreler'in, Merkez Ağırlıklı Median Filtreler'in, Adaptif Median Filtreler'in ve kenarları koruyan Hibrit Median Filtre'lerin çalışma prensipleri matematiksel modellerle açıklanmaktadır. Yine görüntüdeki gürültünün etkisini azaltmak için kullanılan Mean Filtre, Ortalama Filtre ve Şablon Eşleme teknikleri açıklanarak Şablon Eşleme tekniğinde ve diğer görüntü işleme tekniklerinde de oldukça sık kullanılan Konvolüsyon ve Korelasyon yöntemlerinden bahsedilmektedir.

Dördüncü bölümde, görüntüyü netleştirmek (deblurring) için kullanılan algoritmaların ve kenar bulma (edge detection) algoritmalarının dijital görüntü işleme içinde kullanım amaçlarından bahsedilmekte ve algoritmaların teorik ve matematiksel özellikleri anlatılmaktadır. Görüntüyü netleştirmek için kullanılan algoritmalarından; Wiener Algoritması, Düzenlenmiş (Regularized) Filtre, Lucy-Richardson Algoritması ve Blind-Dekonvolüsyon Algoritması anlatılarak bu algoritmaların matematiksel modelleri hakkında bilgi verilmekte ve çalışma prensipleri açıklanmaktadır. Kenar bulma algoritmalarından; Gradyent Kenar Bulma Algoritması, LoG (Laplacian of Gaussian) Kenar Bulma Algoritması ve Canny Kenar Bulma Algoritması hakkında ayrıntılı bilgi verilmekte ve Gradyent Kenar Bulma Algoritmaları'ndan Prewitt, Sobel ve Roberts Algoritmalar'ın filtre kalıplarından kısaca bahsedilmektedir.

Beşinci bölümde, üçüncü ve dördüncü bölümde anlatılan görüntü iyileştirme algoritmaları bazı tarihi eser görüntülerine uygulanmakta ve özellikle en iyi sonuç veren algoritmaların kullanılmasıyla elde edilen görüntü çıktıları karşılaştırmalı olarak verilmektedir. Ayrıca görüntülerde oluşan bozukluk çeşitlerine göre tarihi eser görüntüleri sınıflandırılmakta ve uygulamalar görüntüdeki gürültüyü azaltmak için yapılan uygulamalar, görüntüdeki çizikleri kaldırmak için yapılan uygulamalar, görüntüdeki parlaklık ayarını düzenlemek için yapılan uygulamalar, görüntüyü netleştirmek için yapılan uygulamalar ve görüntüdeki kenarları bulmak için yapılan uygulamalar şeklinde gruplara ayrılmıştır. Uygulamalar için kullanılan tarihi eser görüntüleri Hititler'e ait tarihi eser görüntüleri, Ankara Anadolu Medeniyetleri

Müze'si, Konya Mevlana Müzesi'si, İstanbul Türk ve İslami Eserler Müze'si ve Türkiye'deki daha bir çok müze ve sergiye ait tarihi eser görüntüleri arasından seçilmiştir. Seçilen görüntülere uygun olacak şekilde kullanılacak olan algoritmalar belirlenmekte ve görüntüdeki bozukluğu düzeltmek için seçilen uygun algoritmalar ile bunların çıkışında elde edilen işlenmiş görüntüler görsel olarak sergilenmektedir.

Altıncı bölümde, en iyi sonucu veren algoritmalar arasından altı adet algoritma seçilerek, altı farklı gürültü yoğunluğunda, beş farklı tarihi eser görüntüsüne uygulanmakta ve elde edilen sonuçlar çeşitli karşılaştırma kriterlerine göre istatistiksel olarak karşılaştırılmaktadır. Bütün tekniklerin avantajları ve dezavantajları karşılaştırılarak, hangi tekniklerin tarihi eser görüntülerinde ne gibi iyileştirmeler yaptığı tartışılmaktadır. Ayrıca yine bu bölümde kullanılan algoritmaların işlem süreleri birbirleriyle kıyaslanmakta ve tüm karşılaştırma kriterleri de dikkate alınarak genel bir değerlendirme yapılmaktadır.

Yedinci bölümde, yapılan bu tez çalışmasının hem tarihimizi ve tarihi eserlerimizi korumamız açısından hem de dijital görüntü işleme tekniklerini daha geniş uygulama alanlarına taşımak ve geliştirmek açısından ne gibi faydaları olduğu açıklanmakta ve yapılan çalışmaların avantajları ve dezavantajları özetlenmektedir. Sekizinci bölümde ise, tezde kullanılan kaynaklar verilmiştir.

1.1. Konunun Tanıtılması

Görüntü işleme yöntemleri, görüntülerin bilgisayar tarafından algılanmasını ve insan tarafından yorumlanmasını kolaylaştırmak amacıyla kullanılmakta ve son yıllarda hemen hemen her alanda kendine kullanım alanı bulmaktadır. Görüntü işleme konusunda önemli bir yere sahip olan alanlardan birisi de görüntülerin kalitelerinin artırılmasını sağlayan görüntü iyileştirme teknikleridir. Çeşitli nedenlerle üzerine gürültü eklenmiş, bulanıklaştırılmış ve kontrast ayarı bozulmuş olan görüntüler görsel açıdan fark edilemezler ve insanın görsel algısı için tercih edilmezler. Bu bozucu etkilerin ortadan kaldırılması veya etkilerinin azaltılmasını

sağlamak açısından görüntü işleme sistemlerinin temelini oluşturan görüntü iyileştirme teknikleri çok önemlidir.

Görüntü iyileştirme teknikleri, görüntüdeki gürültünün etkisini azaltma, bulanıklaşma sonucu kaybolan ayrıntıları önemli oranda elde etme ve görüntünün parlaklık ve kontrast ayarlarını düzenlemede kullanılan tekniklerdir. Bu tez çalışmasında da zamanın verdiği etkiyle üzerinde bulanıklıklar, tozlanmalar, çizikler veya paslanmalar meydana gelmiş, kontrast ayarı düzgün olmayan tarihi eser görüntüleri üzerine çeşitli hibrit gürültü temizleme, görüntü netleştirme ve kenar bulma algoritmaları uygulanmakta ve bozuk tarihi eser görüntülerinin insanın görsel algısı için optimum seviyeye getirilmesi sağlanmaktadır.

1.2. Çalışmanın Amacı ve Önemi

Teknolojik gelişmeler görüntü kalitelerini giderek artırmaktadır. Ancak eski görüntülerin güncellenmesi, eski görüntülerin yeni hale getirilmesi ya da objelerin, tarihi eserlerin vb. önemli kültürel varlıkların görüntülerinin geleceğe aktarılması da gerekmektedir. Son on yıla baktığımızda dijital ölçme tekniklerinin sayısında hızlı bir artışın olduğu görülmektedir. Bilgisayarlar son yıllarda daha hızlı işlemciler ile daha büyük hafızalara ve ayrıca yüksek kaliteli giriş ve çıkış birimlerine sahiptirler. Bu süregelen gelişme hemen hemen yaşamın bütün alanlarında açıkça görülmektedir. Ayrıca sanatın ve tarihi eserlerin düzgün bir şekilde korunması gerekliliği bu gelişmenin önemini daha da artırmaktadır. Ulusların nüfusunu, ilim ve fende gelişmelerini, askeri anlayışını, tarihi yaşatan bu fotoğraflar sayesinde kestirmek mümkün olmaktadır.

Şimdiye kadar kullanılan dijital görüntü işleme tekniklerinin, tarihi görüntü arşivlerinin düzenlenmesinde ve kalitelerinin artırılmasında kullanılmak üzere nasıl geliştirilebileceği konusu ise yapılacak çalışmaların başında akla gelen ilk soru olmaktadır. Her gün, her yerde yüzlerce binlerce fotoğraf gözümüze çarpmaktadır. Gazeteler, dergiler, kitap kapakları, billboardlar, ambalajlar, kutular, televizyonlar,

sinemalar... Ancak bu fotoğraflardan çok büyük bir bölümü işlenmiş görüntülerdir, pek azı çekildikleri halleriyle kullanılmaktadırlar.

Bir görüntü, kamera veya başka bir görüntüleme sistemi ile elde edildiğinde, genellikle doğrudan kullanılabilir durumda değildir. Görüntü, parlaklık, aydınlanma veya kontrast farklılıklarından kaynaklanan bozulmalara maruz kalmış olabilir. Ayrıca dijital görüntüler, görüntünün elde edilmesi veya aktarılması sırasında görüntü sensörlerindeki, kameralardaki, tarayıcılardaki veya iletişim kanallarındaki elverişsiz durumlardan dolayı sıklıkla dürtü (impulsive) gürültülerinden etkilenirler. Analog kameralarla çekilen eski fotoğraflar düşünüldüğünde bu tür bozuklukların daha fazla olacağını tahmin etmek çok zor değildir. Günümüzde, medikal görüntüler, uydu görüntüleri, jeolojik görüntüler, kriminolojik görüntüler vb. birçok görüntüde çeşitli görüntü iyileştirme çalışmaları yapılmakta ve görüntüler gürültülerden, tozlanmalardan, paslanmalardan, lekelerden, bulanıklıklardan ve daha birçok olumsuz koşulların meydana getirdiği etkilerden arındırılmaktadır.

Bu tez çalışmasında çeşitli nedenlerle bozulmuş tarihi eser görüntüleri üzerinde çeşitli hibrit algoritmalar ve filtreleme metotları kullanılarak onarım çalışmaları yapılmaktadır.

1.3. Bu Konuda Yapılan Çalışmaların Gelişimi

Görüntüdeki gürültünün etkisinin azaltılması konusunda yapılan en eski çalışmalardan biri Yang ve ark. (1991) tarafından yapılan bir çalışmadır. Bu çalışmada merkez ağırlıklı bir median filtrenin görüntüdeki gürültüleri azaltma konusunda nasıl bir etkiye sahip olduğu araştırılmıştır. Yapılan çalışmanın sonucu FIR (sonlu darbe cevabına sahip) filtrenin sonucuyla karşılaştırılmış ve merkez ağırlıklı median filtrenin FIR filtreye göre daha iyi sonuç verdiği görülmüştür. Ancak elde edilen merkez ağırlıklı median filtre adaptif median filtreler kadar iyi sonuç vermemektedir.

Pitas ve ark. (1991): dijital görüntü işlemede kullanılan ve sıralama istatistiği olarak bilinen görüntü işleme algoritmaları üzerinde çalışmalar yapmışlar ve lineer ile non-lineer filtreler arasındaki farkları belirtmişlerdir. Non-lineer filtrelerden özellikle median filtreler gürültünün etkisinin azaltılmasında önemli bir etkiye sahiptirler.

Ko ve Lee (1991); görüntü zenginleştirme ve iyileştirme amacıyla merkez ağırlıklı median filtreler ile adaptif merkez ağırlıklı median filtrelerin karşılaştırmasını yapmışlar ve elde ettikleri algoritmaları çeşitli görüntülere uygulayarak elde edilen iyileştirme sonuçlarını incelemişlerdir. Elde edilen sonuçlara göre adaptif median filtrenin merkez ağırlıklı filtreye göre daha iyi sonuç verdiği görülmektedir.

Daha sonra yine görüntüdeki gürültünün etkisini azaltmak için Chen ve ark. (1999) tri-state median filtre olarak adlandırdıkları üç aşamalı bir median filtre gerçekleştirmişler ve bu filtrenin sonuçları ile klasik median filtrenin sonuçlarını karşılaştırmışlardır. Kullanılan üç aşamalı filtre ile görüntüdeki gürültüler azaltılırken aynı zamanda görüntüdeki ayrıntılarda nispeten korunmakta ancak çok yüksek seviyeli gürültülerde istenilen sonuçlar elde edilememektedir.

Xiao ve Li (2004); çeşitli boyutlarda konvolüsyon maskesine sahip olan median filtreler ile görüntü iyileştirme çalışmaları yapmış fakat sadece %10, %20 ve %30 gürültü yoğunluklarında başarılı sonuçlar elde edebilmişlerdir.

Chan ve ark. (2005); kenar koruma yaklaşımını kullanarak görüntüdeki tuz-biber gürültüsünün etkisini median tipi filtrelerle azaltmak için bir çalışma yapmışlardır. Yapılan çalışma tuz ve biber tipi gürültülerde ve çok yoğun olmayan gürültülerde iyi sonuç vermekle birlikte, Gaussian tipi gürültülerde iyi sonuç vermemiştir.

Rabie (2004); ISO (International Organization for Standardization) standartlarında çalışan dijital fotoğraf makineleri ile elde edilen görüntülerdeki sensör gürültülerini bastırmak için hibrit mean ve median filtre yaklaşımlarını içeren

bir çalışma yapmıştır. Dijital görüntünün elde edilmesi esnasında meydana gelen üç adet gürültü çeşitini kullanarak elde ettiği sonuçları karşılaştırmış ve median filtrenin mean filtreye oranla daha iyi sonuç verdiğini belirtmiştir.

Hillery ve ark. (1990); bulanık görüntülerin onarımı için yinelemeli bir Wiener filtre algoritması kullanmışlardır. Wiener filtrelerde optimum sonucu sağlayabilmek için ideal görüntü hakkında bilgi sahibi olunması gerekmektedir. Ancak görüntü netleştirmede ortalama karesel hata (MSE) değeri bakımından en iyi sonucu veren algoritmalarından biri Wiener filtredir.

Jiang ve Wang (2003); görüntü netleştirme algoritmalarından Blind Dekonvolüsyon Algoritmasını incelemişler ve Blind Dekonvolüsyon algoritmasının diğer algoritmalara göre avantajlarını ve dezavantajlarını belirtmişlerdir. Kullanılan pek çok görüntü netleştirme algoritması görüntünün nokta yayılım fonksiyonu (PSF) bilgisine ihtiyaç duymaktadır. Ancak Blind Dekonvolüsyon algoritması bozuk görüntünün PSF'si hakkında herhangi bir bilgi sahibi olunmadan (bulanıklık veya gürültü) onarılmasını sağlayan oldukça kullanışlı bir görüntü netleştirme algoritmasıdır. Jiang ve Wang'ın çalışmaları, Kundur ve Hatzinakos'un 1996 yılında yaptıkları çalışmalara çok benzemekle beraber, görüntü kalitesi açısından daha iyi sonuçlar ortaya koymaktadır.

Khashman (1991); DNA bantlarının radyografik görüntülerindeki kenarları bulmak için görüntülere çeşitli kenar bulma algoritmaları uygulamış ve otomatik kenar bulma algoritmasıyla başarılı sonuçlar elde etmiştir. Tıpta ve radyografik görüntülerde olduğu gibi daha pek çok alanda çeşitli kenar bulma algoritmaları yaygın olarak kullanılmaktadır.

2. GÖRÜNTÜ OLUŞUMU VE GÜRÜLTÜ KAVRAMI

2.1 Görüntü Oluşumu

Göz sadece beynin dış dünyaya açılmasını sağlayan bir penceredir. Görme duyusunun oluşumunda göz yalnızca bir aracıdır. Görmenin gerçekleştiği yer ise çok daha derinde, beynin içinde gizlidir. Göze gelen ışık ışınları kornea, gözbebeği ardından da mercekten geçer. Saydam tabakanın bükümlü üst yüzeyi ve mercek, ışınları kırar ve nesnenin (resmin) görüntüsü ters çevrildikten sonra retinaya ulaşır. Işığa duyarlı hücreler (reseptörler; koni ve çubuklar) ışığı elektrik sinyallerine çevirirler ve sinir uçlarına sinyal yollarlar. Retinadan gelen görüntü orijinaline göre baş aşağı durumda ve ters taraftadır. Ancak beyin yeniden yorum yaparak görüntünün düz olmasını sağlar. Bu impulslar beyine nesnenin çeşidi, büyüklüğü, rengi, uzaklığı hakkında haber götürürler ve tüm bu dizi işlemler saniyenin binde biri kadarlık bir sürede gerçekleşir (Mayo Clinic Ans.). Görme gerçekleşirken bir saniyede meydana gelen işlem sayısı şu an mevcut hiçbir bilgisayarın yapamayacağı kadar yüksektir. Bu kadar hızlı olmasının yanı sıra görmenin en şaşırtıcı ve mucizevi yanı ağ tabakaya düşen ters görüntünün beynin optik merkezinde düzeltilmesidir. Elektrik sinyallerinden oluşan bir mesajı beynin nasıl olup da tam tersi bir şekilde yorumladığı henüz cevaplanamamış bir sorudur. (Guyton 1986).

2.1.1. Beynin görmedeki rolü

Lens tarafından retinada odaklanan görüntü elektrik sinyallerine dönüştürüldükten sonra saniyenin binde birinde optik sinirler aracılığıyla beyne ulaştırılır. Her iki gözden ayrı ayrı elde edilen sinyaller, bakılan cisme ait bütün özellikleri içerir. Beyin de iki gözden gelen görüntüleri tek bir görüntü halinde birleştirir. Nesnenin biçimini ve rengini ayırt eder, ne kadar uzakta olduğunu saptar. Kısacası nesnelere gören göz değil beyindir (Temel Britanicca).

Gözlerden gelen elektrik sinyalleri beynin arka kabuğunda yer alan primer görme alanına ulaşır. Bu merkez altı tabaka halinde yüz milyon sinir hücresi içerir. Uyarı önce dördüncü tabakaya gelir, burada analiz edildikten sonra diğer tabakalara dağılır. Bu merkezde her nöron bin kadar nörondan uyarı alır ve bin kadar nörona uyarı gönderir.

Son derece gelişmiş bir bilgisayar gibi çalışan beyin aslında tıpkı diğer organlar gibi milyonlarca küçük canlıdan oluşmuş bir canlılar topluluğudur. İnsan beyninin yüzeyinde her milimetre karede 100.000 dolayında sinir hücresi bulunur. Beyinde toplam olarak yaklaşık 10.000.000.000 (on milyar) sinir hücresi vardır. Yani beyin 10 milyar küçük canlının oluşturduğu bir organdır. Bu canlılardan bir kısmı gözden gelen mesajları yorumlayarak, birbirleri ile koordinasyon halinde görme olayını gerçekleştirirler.

2.1.2. Çağrışım alanı (Ne gördüğünü bilmek)

İnsan hafızası gördüğü görüntülerin bir kısmını depolar. Depolardaki dosyalar daha sonra kullanılmak üzere sık sık açılır. Örneğin, bir bebek ilk defa kalem gördüğünde hafızasında kaleme ait bir dosya açılır. Bebek bir süre sonra tekrar kalem gördüğünde daha önce açılan kaleme ait dosyadan çıkarılan görüntü, otomatik olarak yeni görüntü ile kıyaslanır. Bu sayede bebek gördüğü şeyin kalem olduğunu anlar.

Aslında bu sistem sadece bebekler için geçerli değildir. Bütün insanların beyinleri günlük hayatta bu işlemleri otomatik olarak yapar. Bir görüntü ile karşılaşıldığında, bu görüntü hafızadaki arşiv görüntülerle karşılaştırılır. Arşivdeki bilgilerle yapılan kıyas sonucunda yeni görüntünün ne olduğuna karar verilir. Eğer çağrışım alanındaki bu işlemler yapılmıyorsa kendi kalemimizi bile tanıyamazdık.

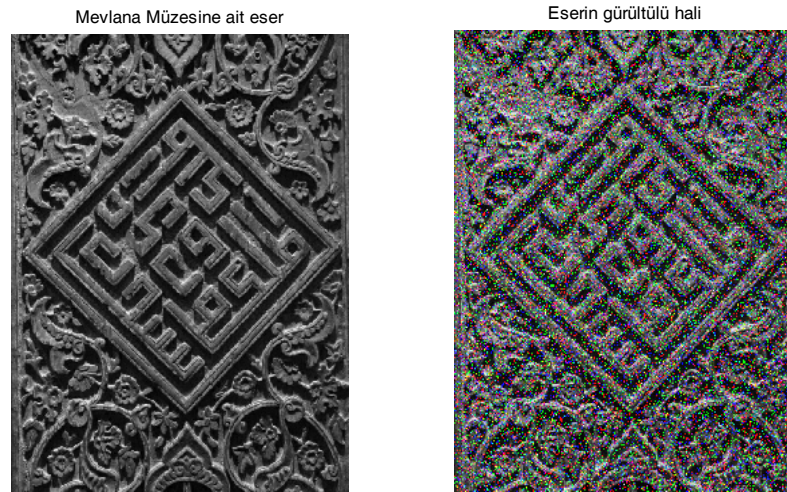
Çağrışım alanı hareket kavramının algılanmasını da sağlar. Hareket halinde bir cisim gördüğümüzde, hafıza işlemi devreye girerek o hareketi hafızaya alır ve bir sonraki hareketle karşılaştırır (Nilsson ve Little 1974). Tıpkı bir film şeridi gibi,

hareketler ardarda kaydedilir ve bir fotoğraf serisi oluşur. Nesnenin bulunduğu yer bir önceki bulunduğu yere göre kıyaslanarak hareket olgusu zihinde oluşturulur.

2.2. Gürültünün Görüntü ve İnsan Görüsü (Human Vision) Üzerindeki Etkisi

Görüntü parlaklığının üniform olması yani görüntünün her tarafında algılanması istenilen bir durumdur. Ancak zaman zaman bazı nedenlere bağlı olarak bu açık ve parlak dijital görüntünün belli bölgelerinde değişimler meydana gelir. Bu değişimler genellikle rastgeledirler ve belirli bir kalıp teşkil etmezler. Bir çok durumda, bu değişimler görüntü kalitesini azaltırlar ve özellikle görüntüdeki elemanların daha düşük kontrastlı olmaları durumunda görsel olarak daha çok algılanırlar. Görüntü parlaklığındaki bu rasgele değişimi insan gürültü olarak algılar.

Genellikle bütün görüntüler biraz da olsa görsel gürültü içerirler. Bir görüntüde gürültünün varlığı çeşitli şekillerde ortaya çıkabilir. Bunlar benekli, kumlu, dokulu veya karlı şekilde insan görsel algısında var olurlar. Şekil 2.1’de aynı görüntüye ait orijinal ve gürültülü görüntüler yer almaktadır.



Şekil 2.1 (a) Orijinal Görüntü (b) Gürültülü Görüntü

Görüntü gürültüleri çok çeşitli gürültü kaynakları nedeniyle meydana gelirler. Hiç bir görüntü elde etme metodu gürültüden bağımsız değildir. Bazı görüntü elde etme yöntemlerinde meydana gelen gürültü miktarı daha fazla olabilmektedir. Şekil 2.1 (a)'da Mevlana Müzesi'ne ait bir eserin çok fazla detay ve kenar içeren bir görüntüsü mevcuttur. Görüldüğü üzere bu görüntünün gürültü eklenmiş Şekil 2.1 (b)'deki halinde ise ayrıntılar büyük ölçüde kaybolmuş, detaylar ve kenarlar seçilemeyecek hale gelmiştir.

Görüntüdeki gürültü genellikle insanın görsel algısı için istenmeyen bir durumdur. Gürültü, görüntüdeki önemli özelliklerin ve ayrıntıların görülme miktarlarını azaltmakta hatta bazen tamamen kapatarak bunların tamamen görülemez hale gelmesine neden olmaktadır. Prensip, görüntüdeki gürültüyü azalttığımız zaman, görüntünün bütünündeki düşük kontrastlı objelerin birçoğu görülebilir hale gelir.

Maksimum görüş netliği sağlamak için gürültü seviyesini mümkün olan en düşük seviyeye getirmek yani gürültüyü azaltmak için görüntüleme faktörlerinde değişiklikler yapmak her zaman istenilen sonucu vermeyebilir. Bir görüntü elde edilirken gürültüyü optimum olarak azaltmanın yanında görüntünün kontrast ve bulanıklık (blur) özellikleri de göz önünde bulundurulmalıdır. Her bir görüntüleme prosedürü için gürültünün kabul edilebilir seviyesi belirlenmeli, olumsuz etkenleri minimuma düşürecek, görüntüleme süresini optimum düzeyde tutacak ve aynı zamanda diğer görüntü kalite karakteristiklerini (kontrast, blur vb.) de göz önünde bulunduracak görüntüleme faktörleri seçilmelidir.

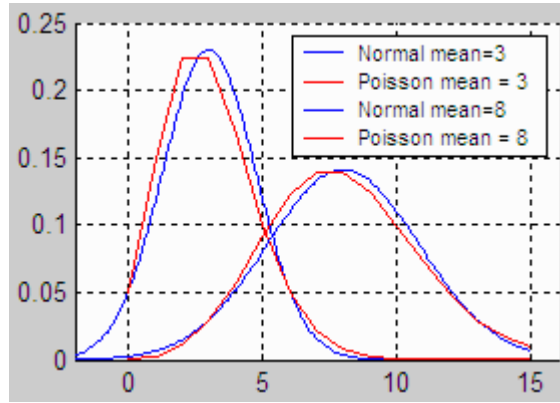
2.3. Gürültünün Matematiği

2.3.1. Genlik dağılımı

Bir çok durumda gürültü içeren piksel veya yoğunluk değişimleri *normal dağılım* ile modellenebilir. Bu durum Şekil 2.2'deki mavi eğrinin gösterdiği Gaussian veya çan eğrisi olarak da ifade edilebilir ve olasılık yoğunluk fonksiyonu,

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-a)^2}{2\sigma^2}} \quad (2.1)$$

ile ifade edilir. a , x 'in *ortalama* değeri ve σ ise *standart sapma*'dır. σ , dağılımın genişliği ile orantılıdır. Normal bir dağılım için, örneklerin %68'i $a \pm \sigma$ arasında, %95.5'i $a \pm 2\sigma$ arasında, %99.7'si de $a \pm 3\sigma$ arasındadır.



Şekil 2.2 Gürültünün Gaussian Dağılımı ve Poisson Dağılımı

x_m : x 'in ortalama değeri ve $x_i = 1$ 'den N 'e kadar örnek sayısını belirtmek üzere, standart sapma değerini eşitlik (2.2)'deki gibi ifade etmek mümkündür:

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - x_m)^2} \quad (2.2)$$

Gürültü genellikle RMS (root mean square) gerilimi veya piksel seviyesi olarak ölçülür. RMS, standart sapma ile eş değerdedir ve RMS gürültü gerilimi, gücün kare köküne eşittir. Kare kök içindeki değer aynı zamanda *varyans* olarak da adlandırılır. Gürültü piksel seviyesine göre değişiklik gösterir. Örneğin, farklı değerli piksel bölgelerinde gürültü de farklıdır.

Normal eğri, merkezi kısıtlama teoremi olarak da adlandırılan dikkate değer bir matematiksel teorem sonucunda elde edilir. Değişken bir gerilim veya piksel seviyesi büyük sayıda gürültüye maruz kaldığı zaman, gürültünün dağılımına

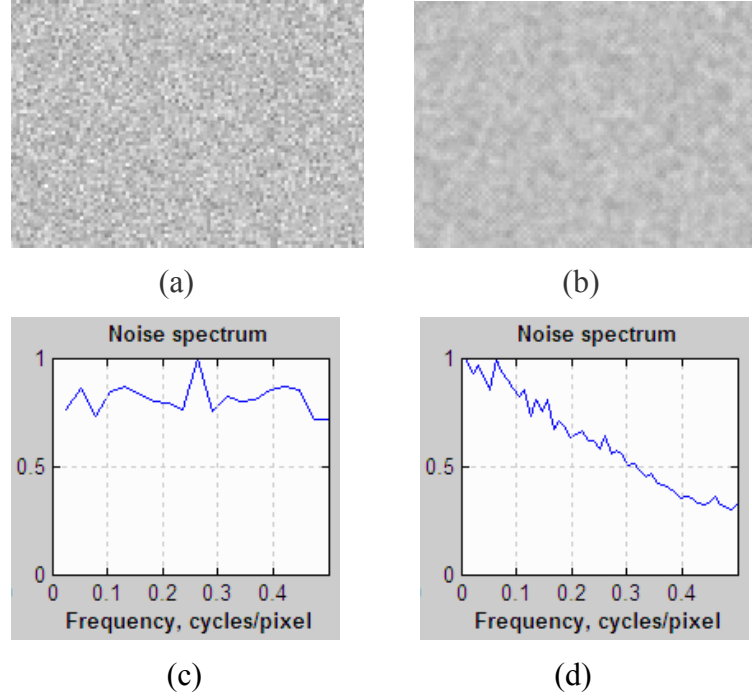
bakılmaksızın tüm yoğunluk fonksiyonu normal dalgaya yaklaşır. Bu, normal dalganın birçok ortak olasılık dağılımından daha üstün olmasının bir nedenidir. Fakat normal dağılım (Gauss Dağılımı), bütün durumlara uygulanamaz. Normal dağılımın negatif sayılarla sonuçlandığı düşük ışık seviyeleri için Poisson dağılımı (grafikte kırmızı ile gösterilen eğri) doğru sonucu verir.

$$f(s) = \frac{e^{-m} \cdot m^s}{s!} \quad (2.3)$$

Burada m:ortalama, $s \geq 0$ olan bir tam sayı ve $s! = (s)(s-1)(s-2)\dots(1)$ 'dir. Standart sapma $\sigma = \sqrt{m}$ 'dir. m'in büyük değerleri için, Poisson dağılımı normal dağılıma yaklaşır (www.imatest.com).

2.3.2. Gürültü frekans spektrumu

Gürültü genlik dağılımına ilave olarak, *frekans spektrumu* ile de karakterize edilebilir. Bu işlem uzaysal görüntünün Fourier dönüşümünün alınmasıyla yapılır. Aşağıda iki ayrı spektra görülmektedir. Birinci görüntü beyaz gürültü, ikinci görüntü ise bulanık gürültü içermektedir. Bu görüntüler, piksel dağılımını vurgulamak için en yakın komşuya göre yeniden boyutlandırma algoritması kullanılmak suretiyle 2X kadar büyütülmüştür. Bunlar spektrayı hesaplamak için gerçek görüntülere yaklaşık olarak benzemekle beraber birebir orijinal görüntülerle aynı değillerdir (www.imatest.com).



Şekil 2.3 (a) İki kat büyütülmüş beyaz gürültü görüntüsü (b) İki kat büyütülmüş bulanık gürültü görüntüsü (c) Beyaz gürültünün spektrumu (d) Bulanık gürültünün spektrumu

Beyaz gürültü iki anahtar özelliğe sahiptir. Birincisi, komşu piksellerin değerleri birbirinden bağımsızdır. İkincisi, spektrumu oldukça düzdür. Şekil 2.3 (c)'deki beyaz gürültü spektrumu, istatistiksel değişimi göstermektedir. Grafikte görüldüğü gibi eğri 0.25 cyles/pixel değerinde muhtemelen görüntü yeniden boyutlandırıldığı için küçük bir zirve yapmaktadır. Bulanık görüntü için, komşu pikseller arasında bir ilişki vardır ve spektrum düz değildir. Şekil 2.3 (b)'de alçak geçiren filtreleme olarak adlandırılan işlemde sonra elde edilen bulanıklaştırılmış görüntü ve Şekil 2.3 (d)'de de onun spektrumu görülmektedir.

Gürültünün görünürlüğü her ne kadar aralarında karmaşık bir ilişki de olsa gürültü spektrumuna bağlıdır. Yüksek uzaysal frekanslardaki gürültü küçük baskılarda (az büyütülmüş) daha az göze çarparken, büyük baskılarda (çok büyütülmüş) daha fazla göze çarpar. Bu ilişkinin doğasındaki karmaşıklık nedeniyle, gürültünün özel olarak ölçülmesi için bir takım teknikler geliştirilmiştir.

Keskinleştiren (sharpening) filtreler, bulanıklaştıran (blurring) filtrelerin tersi işlem yaparlar. Unsharp Masking (USM) filtreler spektrum parçalarını desteklerler ve komşu piksellerin birbirleriyle negatif ilişkili olmasını sağlarlar yani pikseller arasındaki farkları artırır. Unsharp filtreleme genellikle bir eşik değeri ile birlikte uygulanır. Bu eşik değeri kontrast sınırları civarlarındaki keskinleştirmeleri sınırlandırır. Bu da görüntülerdeki örneğin gökyüzü gibi açık alanların görünümündeki bozulmaları engeller (www.imatest.com).

Düşük kaliteli lensler kullanıldığında, görüntü yanlış odaklandığında veya çekim esnasında sarsıntı meydana geldiğinde, görüntü alçak geçiren filtreden geçirilmiş gibi yani bulanık şekilde elde edilir, gürültü yoktur. Bazı keskinlik kayıpları, keskinleştirici filtreler yardımıyla iyileştirilebilir. Bu işlem esnasında gürültülü bölgeler de desteklenmiş olur. Uygulamalarda dijital keskinleştirme mümkün olsa bile kameralarda iyi lens kullanılmasının önemi büyüktür. Bazı uygulamalar görüntülerde gürültü azaltım işlemini yaparken aynı zamanda keskinleştirme işlemini de yapabilmektedir.

2.4. Görüntü Kalitesi

Herhangi bir görüntünün kalitesi görüntüyü elde etmede kullanılan görüntüleme metoduyla doğrudan ilişkilidir. Kullanılan cihazların karakteristikleri ve çekim yapan kişinin seçtiği ayarlar görüntü kalitesini etkiler. Kontrast, bulanıklık, gürültü, insan kaynaklı hatalar ve bozukluklar görüntü kalitesini etkileyen başlıca faktörlerdir (Sprawls 1993).

2.5. Kontrastın ve Bulanıklığın Gürültü Üzerindeki Etkisi

Eğer bir görüntüleme sisteminin bütün kontrast transferi artmış ise görüntüdeki gürültü daha görülebilir hale gelir. Bu durum kontrastı ayarlanabilir görüntü göstergeleri kullanıldığı zaman düşünülebilir. Yüksek kontrastlı bir film gürültünün görünürlüğünü artırır (Sprawls 1993).

Görüntüdeki gürültünün görünürlüğü genellikle bulanıklıkla beraber azalır. Çünkü gürültü oldukça detaylı bir yapıya sahiptir. Bir görüntüye uygulanan bulanıklaştırma işlemi her bir görüntü noktası ve onun etrafındaki alanı birbirine uyumlu hale getirme eğilimi gösterir. Bu etki, gürültünün rastgele olan yapısını yumuşatır ve görüntüdeki gürültüyü daha az fark edilebilir hale getirir.

Gürültünün görünürlüğünü azaltmak için görüntü bulanıklaştırırken aynı zamanda yararlı görüntü detaylarının görünürlüğü de azalabilir. Yüksek duyarlılık (hızlı) yoğunlaştırılmış bazı ekranlar, detaylı ekranlara göre daha az kuantum gürültüsü içeren görüntüler üretirler. Çünkü bu ekranlar aynı zamanda bulanıklık da üretirler. Görüntü filtreleme ve iyileştirme algoritmalarındaki en büyük sorun da buradan kaynaklanmaktadır. Maksimum gürültü bastırımı sağlarken aynı zamanda kenarları ve detayları da korumak karşılaşılan en önemli sorunlardan biridir.

Bulanıklaştırma işlemi dijital görüntü işlemede gürültüyü azaltmak için kullanıldığı gibi, çeşitli web sitelerinin tasarımında kullanılan görüntü efektlerinin hazırlanmasında, sinema filmlerinde, reklamlarda, billboardlarda ve daha bir çok uygulamada kullanılan fotoğrafların ve video görüntülerinin hazırlanmasında kullanılan bir görüntü işleme tekniğidir.

2.6. Görüntülerde Oluşan Gürültü Çeşitleri

2.6.1. Görüntülerde doğal yollarla oluşan gürültü çeşitleri

Modern sensörlerce elde edilen görüntüler çeşitli gürültü kaynakları nedeniyle gürültüye maruz kalabilirler. Gölgeleme veya odak kaybı gibi rastgele olmayan bozuklukların dışında rastgele gürültüler de mevcuttur. Modern elektro-optik teknikler kullanılarak elde edilen görüntülerdeki özellikle fotonların ürettiği elektronları kullanan CCD¹ kameralar kullanılarak elde edilen görüntülerdeki gürültüler doğal yollarla oluşan gürültülerdir. Bununla beraber, doğal olarak ortaya çıkan gürültü çeşitlerinin bir çoğu yapay olarak elde edilen gürültü çeşitleri ile aynı özelliklere sahiptir.

Geliştirilen modern elektro-optik cihazlar sayesinde görüntüdeki gürültü seviyesini hemen hemen ihmal edilebilecek seviyeye azaltmak mümkün olduğu halde birçok gürültü kaynağının etkilediği bir görüntüde diğer gürültü kaynakları bastırılrsa bile kalan tek bir gürültü kaynağı görüntünün bozulmasına neden olabilmektedir. Bu sebeple tüm gürültülerin belirlenerek bertaraf edilmesi gerekmektedir.

Doğal yollarla ortaya çıkan gürültüler aşağıdaki gibi sınıflandırılmaktadır:

a) Foton gürültüsü: Işık tabanlı fiziksel sinyali elde etmede ışığın kuantum özelliği önemli rol oynamaktadır. $\lambda=500$ nm dalga boyundaki tek bir foton, $E = hv = hc / \lambda = 3.97 \times 10^{-19}$ Joule miktarında enerji taşır. Modern CCD kameralar tek tek fotonları sayabilecek hassasiyete sahiptir. Gürültü problemi foton üretiminin temel istatistiksel doğasından kaynaklanmaktadır.

¹ CCD (charge coupled device) görüntü sensörleri ışık desenini (görüntü), bir elektrik yükü desenine (bir elektronik görüntü) çevirmeye yarayan elektronik cihazlardır. CCD'ler verileri toplama, depolama ve elektrik yükünü bir elemandan diğerine taşımaya yarayan pek çok ek unsurdan meydana gelirler.

Aralarındaki mesafe T uzunluğundan bağımsız olan ardışık iki pikselin foton sayılarının aynı olacağı varsayılmaz. Foton üretimi, kuantum fiziği kanunlarına göre yapılır ve buna bağlı olarak verilen penceredeki ortalama foton sayısı hakkında bilgi sahibi olmamız pek olası değildir. Verilen penceredeki p fotonlarının muhtemel dağılımı Poisson olarak bilinmektedir.

$$P(p | \rho, T) = \frac{(\rho T)^p e^{-\rho T}}{p!} \quad (2.4)$$

Burada ρ , saniyedeki foton miktarı olarak ölçülen yoğunluk (intensity) parametresidir. Ortamda başka bir gürültü kaynağı olmasa bile T mesafesinde ve sonlu bir zamanda foton sayımı yapılması esnasında meydana gelen istatistiksel dalgalanmalar sonlu bir sinyal gürültü oranına (SNR) neden olacaktır. Ortalama değerin, $ortalama = \rho T$ olduğu ve standart sapmanın da $Poisson \text{ İşlemi} \Rightarrow \sigma = \sqrt{\rho T}$ ile ifade edildiği varsayılırsa, SNR değerini eşitlik (2.5)'deki gibi ifade etmek mümkündür:

$$Foton \text{ Gürültüsü} \Rightarrow SNR = 10 \log_{10}(\rho T) \text{ dB} \quad (2.5)$$

Sinyal ve gürültü arasındaki ilişki hakkında üç geleneksel yaklaşım foton gürültüsü için aynı değildir:

- Foton gürültüsü sinyalden bağımsız değildir.
- Foton gürültüsü Gaussian değildir.
- Foton gürültüsü toplamsal değildir.

ρT değerinin 10^5 'i geçtiği gürültüsüz sinyaller için, foton istatistiklerine bağlı olarak meydana gelen gürültü dalgalanmaları eğer sensör yeterince yüksek bir saturasyon seviyesine sahipse ihmal edilebilir (www.ph.tn.tudelft.nl).

b) Termal gürültü: Bir CCD'de elektronların kaynağı termal enerjidir. Elektronlar CCD maddesinden kendi termal vibrasyonları sayesinde ayrılırlar. Fakat doğru foto-

elektronların hangileri olduğunu ayırt etmek mümkün değildir. CCD çipini soğutarak termal gürültüye veya *karanlık akıma* neden olan termal elektronların sayısını büyük ölçüde azaltmak mümkündür. Entegrasyon süresi T arttıkça, termal elektronların sayısı da artar. Termal elektronların muhtemel dağılımı, yoğunluk parametresinin sıcaklığın artan bir fonksiyonu olduğu bir Poisson işlemi olarak düşünülebilir. CCD'yi soğutmak için yani karanlık akımı bastırmak için bazı alternatif teknikler mevcuttur. Bunlar genellikle verilen entegrasyon süresi için ortalama karanlık akımını yaklaşık olarak belirlemek ve daha sonra da bu değeri analog/dijital dönüşümden önce CCD piksel değerlerinden çıkarmak için kullanılan tekniklerdir. Bu olay karanlık akım ortalamasını azaltmasına rağmen, karanlık akım standart sapmasını azaltmaz (www.ph.tn.tudelft.nl).

c) Yonga üzerindeki elektronik gürültü: Sinyalin sensörden okunması sırasında meydana gelen bir gürültü çeşitidir. Bir CCD yongası içindeki alan etkili transistör (FET) içinden geçen sinyal üzerinde etkilidir. Okunan gürültünün güç spektral yoğunluğunun genel yapısı aşağıdaki gibi ifade edilebilir:

$$S_{j|Y|}(\omega) \propto \begin{cases} \omega^{-\beta} & \omega < \omega_{\min} & \beta > 0 \\ k & \omega_{\min} < \omega < \omega_{\max} \\ \omega^{\alpha} & \omega > \omega_{\max} & \alpha > 0 \end{cases} \quad (2.6)$$

Burada α ve β sabit sayılar, ω ise sinyalin CCD yongasından dış dünyaya yayılan vektörel (radyal) frekanstır. Çok düşük gürültü okuma oranlarında ($\omega < \omega_{\min}$) gürültü $1/f$ karakterine sahiptir. Okunan gürültü, uygun okuma oranları ve elektronik cihazlarla makul bir seviyeye düşürülebilir. Bununla birlikte çok düşük sinyal seviyelerinde, okuma gürültüsü yine de SNR'ı etkileyen önemli etkenlerden biri olmaktadır (www.ph.tn.tudelft.nl).

d) KTC gürültüsü: Bir FET'in gate bacağına bağlı kapasitördeki gürültü KTC (K: Boltzman sabiti, T: Sıcaklık, C: Kapasite) gürültüsü olarak adlandırılır ve bu gürültü ihmal edilemeyen bir gürültüdür. Bu gürültü voltajının RMS çıkış değeri aşağıdaki gibi verilmektedir:

$$KTC \text{ gürültüsü(gerilim)} \Rightarrow \sigma_{KTC} = \sqrt{\frac{kT}{C}} \quad (2.7)$$

Burada, C: FET'in gate bacağındaki kapasitans,

k: Boltzman sabiti,

T: CCD yongasının mutlak sıcaklık değeridir. (°K)

$$Q = CV = N_{e^-} \cdot e^- \quad (2.8)$$

eşitliğini kullanarak, foto-elektronların sayısına (N_{e^-}) göre ifade edilen KTC gürültüsünün RMS çıkış değeri aşağıdaki gibi verilmektedir:

$$KTC \text{ gürültüsü(elektronlar)} \Rightarrow \sigma_{N_{e^-}} = \frac{\sqrt{kTC}}{e^-} \quad (2.9)$$

Burada e^- , elektron yüküdür. $C = 0.5$ pF ve $T = 233$ K için elektron sayısı $N_{e^-} = 252$ elektron olmaktadır. Bu değer sinyal okuması sırasında meydana gelen piksel başına düşen birim zamanlık gürültü miktarını ifade eder ve entegrasyon zamanından bağımsızdır. Çift örnekleme ve çift eğim ilişkisi sağlanarak gerçekleştirilen entegrasyon ile KTC gürültüsünü hemen hemen ortadan kaldırmak mümkündür (www.ph.tn.tudelft.nl).

e) Yükseltici gürültüsü: Bu gürültü tipi için standart model toplamsal Gaussian modelidir ve sinyalden bağımsızdır. İyi tasarlanmış modern elektronik sistemlerde yükseltici gürültüsü genellikle ihmal edilebilir. Mavi renk kanalında kırmızı veya yeşil kanala göre daha fazla yükseltmenin yapıldığı renkli kameralarda mavi renk kanalının yükseltici gürültüsü ihmal edilemez (www.ph.tn.tudelft.nl).

f) Kuantalama gürültüsü: Kuantalama gürültüsü, genlik kuantalama işleminin doğasından kaynaklanan bir gürültü biçimidir ve analog-dijital dönüştürücülerde

(ADC) meydana gelir. Gürültü seviyesi $L \geq 16$ olduğu durumlarda, gürültü toplamsaldır ve sinyalden bağımsızdır. Bu değer $B \geq 4$ bit için elde edilen durumla aynıdır. Elektriksel forma dönüştürülmüş bir sinyal için, maksimum ve minimum elektriksel değerler SNR'ın belirlenmesinde kullanılırlar. Minimum elektriksel değer 0 ve maksimum elektriksel değer $2^B - 1$ olduğu durum için ADC ayarlanırsa, elde edilen SNR değeri aşağıdaki gibi olur:

$$\text{Kuantalama Gürültüsü} \Rightarrow \text{SNR} = 6B + 11 \text{ dB} \quad (2.10)$$

Bu durum $B \geq 8$ bit için geçerlidir ve aynı zamanda $\text{SNR} \geq 59 \text{ dB}$ olduğu anlamına gelmektedir. Komple bir sistemin toplam SNR değeri en küçük SNR'a göre baskın olduğundan kuantalama gürültüsü genellikle ihmal edilebilir. CCD kameralarda bu gürültü foton gürültüsü olarak adlandırılır (www.ph.tn.tudelft.nl).

2.6.2. Görüntülerde oluşturulan uzaysal gürültü modelleri

Görüntülerde kamera hataları, çekim hataları vb. nedenlerle meydana gelen doğal gürültülerin yanında bunlardan yola çıkıp yapay olarak oluşturulan gürültüler de mevcuttur. Bu uzaysal gürültü çeşitleri aşağıdaki gibi özetlenebilir:

a) Tuz ve biber gürültüsü (Salt and pepper noise): Tuz ve biber gürültüsü, görüntünün kamera ile çekimi esnasında kamera hataları nedeniyle görüntü örneklerinde meydana gelen bozukluklar olarak düşünülebilir. Tuz ve biber gürültüsünde pikseller rastgele ve uniform (tek biçim) olarak seçilmişlerdir. Bu piksellerin her biri ya tam ya da sıfır değerlidir. Görüntüye bakıldığında oluşan gürültüler tuz ve biberin serpilişini andığından bu tür gürültülerin ismi tuz ve biber gürültüsü olarak belirlenmiştir.

b) Gaussian gürültüsü : Gaussian eğrisi ile verilen rastgele değişkenin olası yoğunluğunun gerçek durumlarda da meydana geldiği iyi bir gürültü yaklaşımıdır. 1D Gaussian gürültüsünde μ , random değişkenin standart sapmasını ifade eder.

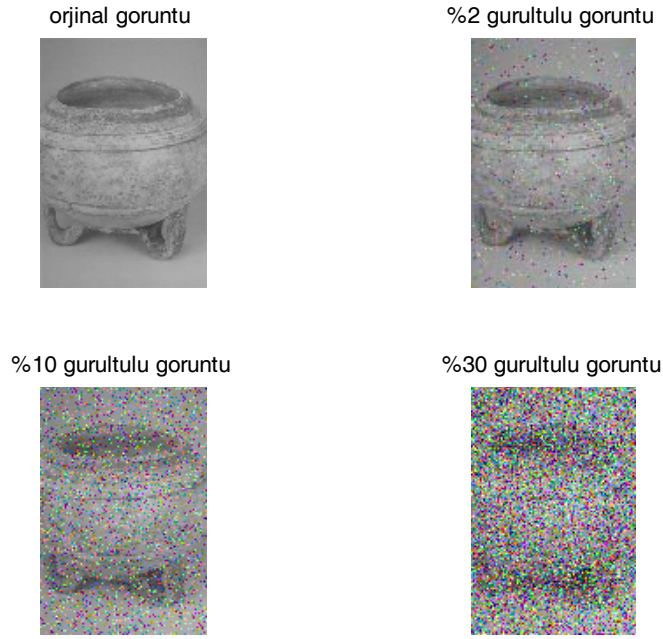
$$P(x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (2.11)$$

Durağan gaussian gürültüsü ile tuz-biber gürültüsü arasında temel bir kavramsal fark vardır. İlk durumda her bir piksele rastgele bir miktar gürültü eklenir. İkinci durumda ise öncelikle pikselleri seçmek için rastgele bir mekanizma kullanılır ve bu seçilen pikseller üzerine gürültü uygulanır. Pikselleri seçmek için kullanılan rastgele mekanizmalar genellikle *nokta işlemleri* olarak adlandırılırlar. Görüntü işlemede kullanılan pek çok nokta işlemi mevcuttur.

c) Poisson gürültüsü: Poisson nokta işlemi olarak adlandırılan nokta işleminde, görüntü düzlemindeki noktalar herhangi bir alt kümedeki noktaların tahmini sayısı alt küme alanıyla doğru orantılı olacak şekilde rastgele seçilirler. Bu orantı sabiti de *işlem yoğunluğu* olarak bilinir.

Bazı piksellerin beyaza ve bazı piksellerin de siyaha çevrilmesi gerektiği durumlarda *işaretli nokta işlemlerine* gerek duyulmaktadır. Bu modelde, noktaları seçmek için bir nokta işlemi kullanılır ve her seçilen nokta uygun bir dağılımla işaret olarak atanır. Yani o noktanın siyah mı, beyaz mı yoksa başka bir değer mi olacağı belirlenir. Uyumsuz veya doymuş piksellerin meydana geldiği bir kamera görüntüsü için bir noktanın siyah işaret taşıma olasılığı ile beyaz işaret taşıma olasılığı aynıdır. Eğer görüntüdeki uyumsuz piksellerin sayısı uyumlu piksellerin sayısından azsa, işaretlerdeki dağılım daha kolay görülebilir.

Şekil 2.4'de görüntüye çeşitli oranlarda eklenmiş tuz-biber gürültüleri görünmektedir.

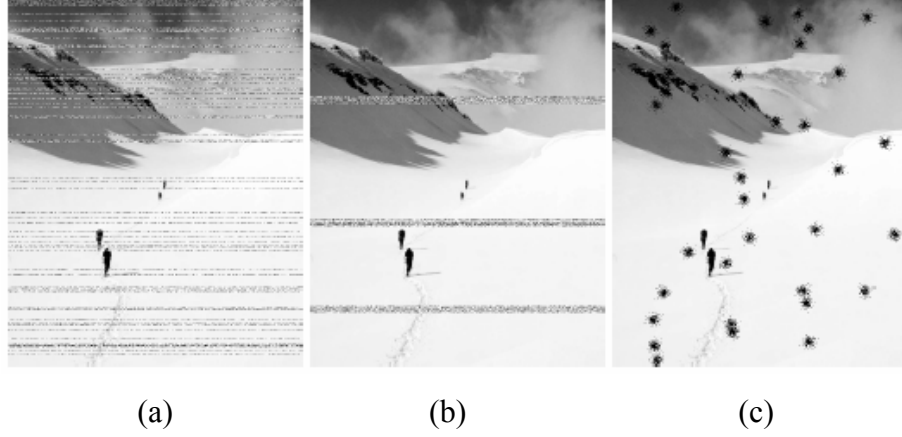


Şekil 2.4 Çeşitli oranlarda tuz ve biber gürültüsü eklenmiş görüntüler

Böyle bir gürültü işlemi, tamamen düzgün olarak dağıtılmış gürültüler için kullanılabilir. Büyük bir alana dağılmış olan ve gürültülü pikselleri içeren piksel kümesi istenmeyen bir durumdur. Bu modelde, Poisson işlemiyle seçilen noktalar alınır ve daha sonra bunların etrafındaki piksellerin kümesi işaretlenir. Bu kümenin şekli rastgele olarak seçilir. Diğer bir gürültü şekli ise genellikle video-teyp sistemlerinde meydana gelen ve görüntünün paralel taranan hatlarından bazılarının gürültü yüklenmesiyle oluşan gürültü çeşididir. Bozuk olan bu çizgiler Poisson nokta işlemi ile seçilebilir (Şekil 2.5). Bu gürültü formunun çeşitli gürültü modelleri mevcuttur.

Bu formun uzaysal gürültü modellerinde filtrenin cevabını genellikle analitik olarak hesaplamak imkansızdır. Bunun yerine simülasyonlar tercih edilmektedir. Muhtemel bir gürültü modelini oluşturmak için kullanılan esas fikir o modelden çok sayıda örnek çıkarmak ve söz konusu filtreyi örneklerle uygulamaktır. Elde edilen sonuçtan gürültü cevabının değişimi ve ortalaması gibi uygun istatistikler hesaplanır. Normalde bilgisayarla görüş (computer vision) literatüründe bu yolla çalışan filtrelerle pek karşılaşılmasa da prensipte bu yolla çalışan filtreler mevcuttur.

Şekil 2.5 (b)'da gösterilen gürültüler VCR (Video Cassette Recorder)'lerin hızlı ileri ve geri sarımları esnasında meydana gelen gürültülere örnek olarak verilebilir. Şekil 2.5 (c)'de meydana gelen gürültüler de CCD'lerdeki veya kamera lenslerindeki hatalar nedeniyle oluşan gürültülere örnek olarak verilebilir.



Şekil 2.5 (a) Yatay hatların rastgele gürültülerle yüklenmesi sonucu oluşan bir gürültülü görüntü (b) Birden fazla veya blok şeklinde birçok hattın gürültüyle yüklenmesi sonucu meydana gelen gürültülü bir görüntü (c) Poisson işleminin, her gürültülü noktanın komşuluğundaki pikselleri “gürültü noktaları” olarak seçmesi ve bu piksellerin gürültü noktasına olan uzaklığının azalan negatif exponansiyel değerde olduğunu varsayarak bu pikselleri siyah olarak işaretlemesi sonucu oluşan gürültülü görüntü

d) Beyaz gürültü: Sabit güç spektrumuna sahip (yoğunluğu frekans arttıkça azalmayan) görüntü gürültüsünün işlenmemiş hali olarak ifade edilebilir.

e) Toplamsal gürültü: Eşitlik (2.12)'de gösterildiği gibi bu tür gürültülerde, toplanır gürültü v ve görüntü sinyali g birbirinden bağımsızdır.

$$f(x, y) = g(x, y) + v(x, y) \quad (2.12)$$

f) Çarpımsal gürültü: Çarpımsal gürültü sinyal büyüklüğünün bir fonksiyonudur. (Eşitlik 2.13)

$$f = g + vg = g(1 + v) \approx gv \quad (2.13)$$

Benek (speckle) gürültüsü de çarpımsal bir gürültü çeşididir.

3. GÜRÜLTÜNÜN ETKİSİNİ AZALTAN GÖRÜNTÜ İŞLEME ALGORİTMALARI

Dijital görüntüler, görüntünün elde edilmesi veya aktarılması sırasında görüntü sensörlerindeki, kameralardaki, tarayıcılardaki veya iletişim kanallarındaki elverişsiz durumlardan dolayı sıklıkla dürtü gürültülerinden etkilenirler. Bu gürültüler görüntülere genellikle asıl görüntü elemanlarının yoğunluk değerleri yerine olası dinamik yoğunluk sahasının minimum veya maksimum değerlerini yerleştirerek etkir. Gürültü azaltımından sonra gerçekleştirilen kenar belirleme, görüntü bölütleme, nesne tanıma vb. gibi görüntü işleme tekniklerinin başarımı gürültünün giderilmesi işleminin başarımına doğrudan bağlıdır (Umbaugh 1998).

Görüntü işlemede eskiden kullanılan teknikler görüntüdeki yüksek frekanslı bileşenleri kaldıran fakat aynı zamanda iyi detayları da belli ölçülerde yok eden alçak geçiren lineer filtrelerden oluşmaktaydı. Günümüzde ise, Fourier ve Dalgacık dönüşümlerine dayalı modern metotlar (Donoho 1995) ve adaptif non-lineer filtreler (Smith ve Brady 1997) kullanılarak kenarları ve iyi detayları koruyarak gürültüyü maksimum oranda ortadan kaldırabilmek mümkün olmaktadır.

3.1. Mean Filtre ve Ortalama (Averaging) Filtre

Mean filtre, görüntüdeki gürültüyü azaltmak için yaygın olarak kullanılan, basit ve uygulaması kolay bir filtre çeşitidir. Mean filtreleme işleminde adından da anlaşılacağı üzere, gürültülü görüntü üzerindeki her bir pikselin değeri o pikselin komşuluğundaki piksel değerlerinin ortalaması ile değiştirilir. Mean filtreyi, konvolüsyon filtre olarak da düşünmek mümkündür. Diğer konvolüsyon işlemlerinde olduğu gibi bunda da boyutu ve şekli bizim tarafımızdan belirlenecek bir maske ile pikselin komşuluğundaki diğer pikseller belirlenir ve bu piksellerin ortalamaları alınır. Bu işlem sırasıyla görüntüdeki her bir piksel için tekrarlanır.

Gürültü cinsine ve miktarına göre kullanılacak maske tipi seçilmelidir. Genellikle 3X3'lük maskeler kullanılmakla birlikte, daha şiddetli gürültü temizleme

işlemleri için daha büyük boyutta maskeler kullanılmalıdır. (5X5, 7X7 vb. gibi) Küçük maskeleri görüntüye birden fazla arka arkaya uygulamak koşuluyla da büyük maskenin sağladığı etkiye yaklaşmak mümkündür.

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Şekil 3.1 Genellikle mean filtrelemede kullanılan 3X3'lük mean filtre maskesi

Şekil 3.1'deki maskeyle gürültülü görüntünün konvolüsyonu sonucu gürültüden arındırılmış görüntü elde edilmekte ve bu işlem mean filtreleme işlemi olarak adlandırılmaktadır.

Ortalama (averaging) filtrenin çalışma prensibi de mean filtrenin çalışma prensibine benzemektedir. Ortalama filtrede , seçilen farklı kalıplara göre elde edilen ortalama piksel değerleri ile gürültü piksellerin yerleri değiştirilerek görüntüdeki gürültülerin temizlenmesi sağlanır.

Mean filtrenin iki dezavantajı vardır:

- 1) Görüntüdeki oldukça düşük değerli bir piksel, komşuluğundaki bütün piksellerin ortalamasını büyük ölçüde etkiler.
- 2) Filtre kerneli belirgin kenarlar üzerinde işlem yaparken, gürültü pikseller için gösterdiği etkiyi kenarlarda da gösterecek ve böylece görüntüdeki iyi detaylar ve kenarlar da bulanıklaşacaktır. Eğer çıkış görüntüsünde keskin kenarlar önem teşkil ediyorsa, bu durum problem yaratacaktır.

Bu iki dezavantajın da üstesinden gelebilmek için daha sonraki bölümlerde anlatılacak olan median filtreler kullanılmaktadır. Median filtreler gürültü azaltımında da mean filtreden oldukça iyi sonuçlar vermektedir. Ancak buna bağlı olarak işlem süresi de uzamaktadır. Genel olarak mean filtre alçak geçiren frekans

filtresi olarak görev yapar. Yani yüksek seviyeli pikselleri filtrelerken alçak seviyeli pikselleri geçirir.

3.2. Şablon Eşleme (Template Matching) Tekniği

Şablon Eşleme, bir görüntüdeki özel bir şekil veya şablonu bulmak, veya görüntüdeki gürültüyü azaltmak, bulanıklığı gidermek için kullanılan bir görüntü işleme tekniğidir. Çeşitli filtre kalıplarının bozuk görüntülerle konvolüsyonu sonucunda onarılmış görüntünün elde edilmesi esasına dayanır. Şablon eşleme yönteminin esas kullanım alanı ise görüntünün küçük bir parçasının belirlenen bir şablonla karşılaştırılması esasına dayanan özellik bulma algoritmaları veya sınıflandırma algoritmalarıdır. Örneğin içinde H harflerinin bulunduğu bir yazıda H harflerini seçmek gibi.

Şablon eşleme, bir çok sinyal ve görüntü işleme uygulamasında kullanılan önemli yöntemlerden biridir. Örneğin video kodlamada görüntülerin karşılaştırılması için örüntü tanıma (Duda ve Hart 1973) ve blok eşleme yöntemleri kullanılmaktadır (ISO/IEC DIS 1992). Görüntü eşleme için kullanılan en iyi yaklaşımlar uyumlu (matched) filtrelerdir ancak bu filtreler bazı dezavantajlara sahiptirler (Ben-Arie ve Rao 1992). Matched filtrenin doğasında bulunan kısıtlamaları en aza indirmek için araştırmacılar gün geçtikçe çeşitli yöntemler geliştirmekte ve görüntü restorasyon işlemlerine uyarlamaya çalışmaktadırlar. Şablon eşleme tekniğinde kullanılan standart yöntemler konvolüsyon ve korelasyondur. Hızlı fourier dönüşümü kullanıldığı zaman korelasyon işlemleri daha kısa sürmektedir. Şablon eşleme tekniğini anlamak için korelasyon ve konvolüsyon tekniklerine değinmekte fayda vardır.

3.2.1. Konvolüsyon ve korelasyon teknikleri

a) Konvolüsyon tekniği: Bir görüntü sinyalinin ideal bir filtreden geçirmek için yapılan işlem temel olarak konvolüsyon tekniğine dayanmaktadır. Konvolüsyon işlemi, çıkış görüntüsündeki her bir pikselin, giriş görüntüsündeki aynı pikselin komşuluğundaki piksellerin ağırlıklı toplamına eşit olduğu bir komşuluk operasyonudur. Filtrelenecek görüntüyü konvolüsyona tabi tutacağımız fonksiyona *konvolüsyon kerneli* veya *konvolüsyon kalıbı* adı verilmektedir. Örneğin giriş görüntüsü,

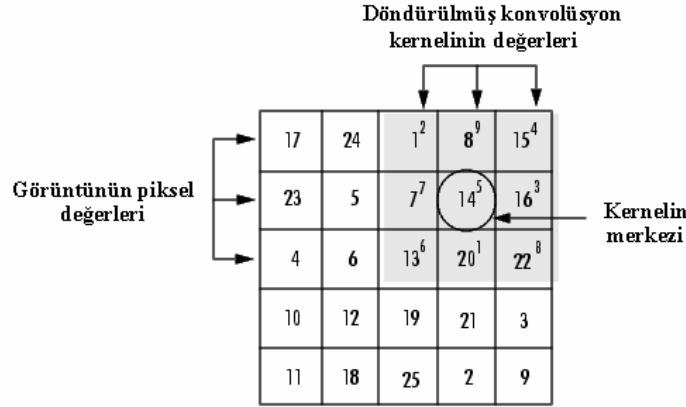
$$A = \begin{bmatrix} 17 & 24 & 1 & 8 & 15 \\ 23 & 5 & 7 & 14 & 16 \\ 4 & 6 & 13 & 20 & 22 \\ 10 & 12 & 19 & 21 & 3 \\ 11 & 18 & 25 & 2 & 9 \end{bmatrix}$$

şeklinde bir fonksiyon için konvolüsyon kerneli:

$$h = \begin{bmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{bmatrix}$$

şeklinde olsun. Buna göre A görüntüsündeki 2. satır 4. sütundaki pikselin konvolüsyondan sonraki değerinin ne olacağı Şekil 3.2'de gösterilmektedir. Yapılan işlem adımlarını aşağıdaki şekilde sıralamak mümkündür:

- Konvolüsyon kerneli, merkezdeki elemanın etrafında 180° döndürülür.
- A görüntüsünün 2. satır ve 4. sütunundaki pikseli ile kernelin merkezindeki piksel üst üste gelecek şekilde kernel görüntü üzerine yerleştirilir.
- Döndürülmüş konvolüsyon kernelindeki her bir değer ile bu değerlerle eşleşen A görüntüsündeki pikseller birer birer çarpılır.
- Çarpım sonuçları toplanır.



Şekil 3.2 Konvolüsyon kernelinin görüntü üzerine yerleştirilmesi

Böylece 2. satır ve 4. sütundaki çıkış pikselinin değeri,

$$1.2 + 8.9 + 15.4 + 7.7 + 14.5 + 16.3 + 13.6 + 20.1 + 22.8 = 575 \quad (3.1)$$

olarak elde edilir. Çıkış sinyalini oluşturmak üzere iki (çok boyutlu) sinyalin konvolüsyonunu göstermek için çok çeşitli notasyonlar mevcuttur. Bunlardan en çok kullanılanı:

$$c = a \otimes b = a * b \quad (3.2)$$

şeklinindedir. Konvolüsyonu sürekli zamanlı ve ayrık zamanlı; bir boyutlu (1-D) ve iki boyutlu (2-D) sinyaller için aşağıdaki biçimlerde ifade etmek mümkündür:

Sürekli Zamanlı Sinyaller için:

$$1-D : m * f(x) = \int f(u)m(x-u)du \quad (3.3)$$

$$2-D : m * f(x, y) = \int \int f(u, v)m(x-u, y-v)dudv \quad (3.4)$$

Ayrık Zamanlı Sinyaller için:

$$1-D : m * f(x) = \sum_i f(x-i)m(i) \quad (3.5)$$

$$2-D : m * f(x, y) = \sum_i \sum_j f(x-i, y-j)m(i, j) \quad (3.6)$$

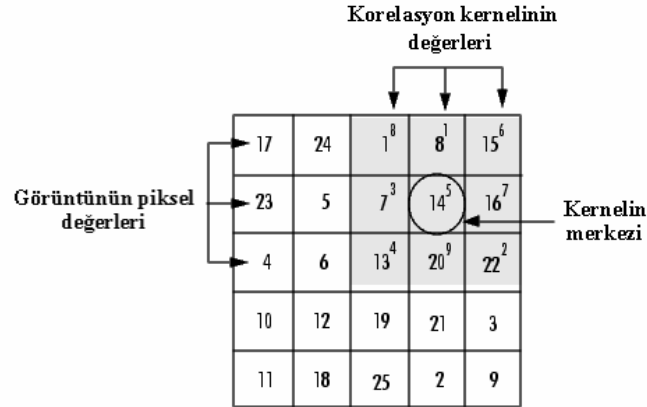
b) Korelasyon tekniği: Korelasyon işlemi de konvolüsyona çok benzeyen bir işlemdir. Burada da yine çıkış görüntüsündeki her bir piksel, giriş görüntüsündeki aynı pikselin komşuluğundaki piksellerin ağırlıklı toplamına eşittir. Buradaki tek fark ise korelasyon kernelinin döndürülmüyor olmasıdır. Giriş görüntüsü:

$$A = \begin{bmatrix} 17 & 24 & 1 & 8 & 15 \\ 23 & 5 & 7 & 14 & 16 \\ 4 & 6 & 13 & 20 & 22 \\ 10 & 12 & 19 & 21 & 3 \\ 11 & 18 & 25 & 2 & 9 \end{bmatrix}$$

ve kullanılacak korelasyon kerneli:

$$h = \begin{bmatrix} 8 & 1 & 6 \\ 3 & 5 & 7 \\ 4 & 9 & 2 \end{bmatrix}$$

olmak üzere, elde edilen çıkış görüntüsündeki pikselin değeri Şekil 3.3'de gösterildiği gibi hesaplanmaktadır.



Şekil 3.3 Korelasyon kernelinin görüntü üzerine yerleştirilmesi

Buna göre 2. satır ve 4. sütundaki çıkış pikselinin korelasyon sonucu elde edilen değeri:

$$1.8 + 8.1 + 15.6 + 7.3 + 14.5 + 16.7 + 13.4 + 20.9 + 22.2 = 585 \quad (3.7)$$

olmaktadır. Ayrık zamanlı sinyallerin korelasyonu matematiksel olarak aşağıdaki biçimde ifade edilebilir:

$$1-D : m \bullet f(x) = \sum_i f(x+i)m(i) \quad (3.8)$$

$$2-D : m \bullet f(x, y) = \sum_i \sum_j f(x+i, y+j)m(i, j) \quad (3.9)$$

3.2.2. Görüntü işlemede şablon eşleme

a) Kernel ile şablon eşleme: Bu tür şablon eşleme tekniğinde bir görüntü her hangi bir kernel ile konvolüsyona tabi tutulur. Uygun maskenin seçilmesi durumunda algoritma bir kenar bulma dedektörü olarak işlem yapabilir. Bu algoritmalar genel olarak özellik bulucu algoritmalar olarak anılırlar.

b) Bir model ile şablon eşleme: Bu tür şablon eşleme tekniğinde de bir görüntüden alınan herhangi bir model ile görüntünün korelasyonu yapılarak, büyük görüntü içerisinde küçük model bulunur. Bu tür işlemler de esnek özellik bulucu işlemler olarak adlandırılırlar.

3.2.3. Görüntü onarımında şablon eşleme

Şablon eşleme yöntemini görüntü işlemeye uyarlamak için aşağıdaki eşitliklerden faydalanmak mümkündür (Choi ve ark 1996). Bozulmuş görüntünün $g(n_1, n_2)$ ile, merkezi (m_1, m_2) olan şablonun ise $f(n_1, n_2)$ ile ifade edildiği varsayılırsa bozulmuş görüntünün matematiksel formu eşitlik (3.10)'daki gibi olmaktadır:

$$g(n_1, n_2) = \Phi[f(n_1 - m_1, n_2 - m_2)] \quad (3.10)$$

Burada $\Phi[\bullet]$, bozulma (degradation) operatörü olarak adlandırılmaktadır. Şablon eşleme ile görüntü iyileştirmenin amacı $\Psi[\bullet]$ ve $\bar{\Psi}[\bullet]$ operatörlerinin

değerlerini bulmaktır. Bu değerler sırasıyla eşitlik (3.11) ve (3.12)'deki gibi ifade edilebilir:

$$w(n_1, n_2) = \Psi[g(n_1, n_2)] \approx f(n_1 - m_1, n_2 - m_2) \quad (3.11)$$

$$\bar{w}(n_1, n_2) = \bar{\Psi}[g(n_1, n_2)] \approx \delta(n_1 - m_1, n_2 - m_2) \quad (3.12)$$

Şablon eşleme ile görüntü işlemeyi (3.13) ve (3.14)'deki denklemlerle kısaca ifade etmek mümkündür:

$$\Psi[g(n_1, n_2)] = \bar{\Psi}[g(n_1, n_2)] * f(n_1, n_2) \quad (3.13)$$

$$\bar{\Psi}[g(n_1, n_2)] = \Psi[g(n_1, n_2)] * f'(n_1, n_2) \quad (3.14)$$

Burada “*” lineer konvolüsyonu ifade etmektedir ve $f'(n_1, n_2)$ ise $f(n_1, n_2)$ 'nin tersidir. Böylece eşitlik (3.15)'deki denklem geçerlidir:

$$f'(n_1, n_2) * f(n_1, n_2) = \delta(n_1, n_2) \quad (3.15)$$

Var olan görüntü iyileştirme filtrelerinden şablon eşleme filtrelerinin elde edilmesi veya şablon eşleme filtrelerinden görüntü iyileştirme filtrelerinin elde edilmesi için çok basit metotlar geliştirmek mümkündür. Örneğin, bozulmuş görüntünün $g(n_1, n_2)$ ile, merkezi (m_1, m_2) olan şablonun $f(n_1, n_2)$ ile, toplanır gürültünün de $v(n_1, n_2)$ ile ifade edildiği düşünülürse,

$$g(n_1, n_2) = f(n_1 - m_1, n_2 - m_2) + v(n_1, n_2) \quad (3.16)$$

eşitliği geçerli olmaktadır. Burada amaç, şablon eşleme ile görüntü iyileştirme için kullanılacak lineer filtreleri bulmaktır ve filtreler $\psi(n_1, n_2)$ ve $\bar{\psi}(n_1, n_2)$ ile ifade edilirse:

$$w(n_1, n_2) = \psi(n_1, n_2) * g(n_1, n_2) \approx f(n_1 - m_1, n_2 - m_2) \quad (3.17)$$

$$\bar{w}(n_1, n_2) = \bar{\psi}(n_1, n_2) * g(n_1, n_2) \approx \delta(n_1 - m_1, n_2 - m_2) \quad (3.18)$$

eşitlikleri geçerli olmaktadır. Daha önceden de belirtildiği gibi, şablon eşleme ile görüntü iyileştirme için kullanılacak konvolüsyon denklemlerini eşitlik (3.19) ve eşitlik (3.20)'deki gibi vermek mümkündür:

$$\psi(n_1, n_2) = \bar{\psi}(n_1, n_2) * f(n_1, n_2) \quad (3.19)$$

$$\bar{\psi}(n_1, n_2) = \psi(n_1, n_2) * f'(n_1, n_2) \quad (3.20)$$

Burada, $f'(n_1, n_2)$, $f(n_1, n_2)$ 'nin tersidir ve eşitlik (3.21) geçerlidir.

$$f'(n_1, n_2) * f(n_1, n_2) = \delta(n_1, n_2) \quad (3.21)$$

Şablon eşleme tekniğini gürültü görüntülerin onarımında kullanmak mümkün olduğu gibi, bulanık görüntülerin netleştirilmesinde de kullanmak mümkündür. Seçilen filtre kalıplarının katsayılarında değişiklik yapmak suretiyle şablon eşleme kalıbını gürültü temizleme filtre kalıbı veya görüntü netleştirme filtre kalıbı olarak tasarlamak mümkündür. Bulanık görüntüyü netleştirmek için kullanılan algoritmalar 4.Bölüm'de anlatılacaktır.

3.3. Median Filtre

Düzgünleştirme metodunun en temel problemlerinden birisi, görüntü içerisindeki kenarların ve ayrıntıların zayıflamasıdır. Bir çok alçak geçiren filtre uygulaması sonucunda gürültü büyük oranda süzülmemekte fakat bulanıklaşmanın olması sonucu kenarlar ve ayrıntılar keskinliğini kaybetmektedir. Daha az bulanıklaşmaya karşılık gürültünün zayıflatılması için kullanılan alternatif bir yaklaşım median filtreleme metodudur. Buna göre median filtreleme işlemi, görüntüdeki her bir pikselin gri seviyesinin bu piksellerin komşuluğundaki gri seviyelerin toplamının ortalaması ile değil, medianı ile yer değiştirmesi mantığına göre çalışmaktadır. Eğer gürültünün etkisi çok büyük ise, o zaman bu filtreleme

işlemi etkili olmaktadır (Gonzales ve ark. 2004). Median filtre, piksellerdeki tuz-biber (salt-pepper) gürültüsünü, benek (speckle) ve dürtü (impulsive) gürültüsünü azaltmak için kullanılan doğrusal olmayan bir alçak geçiren filtre çeşitidir (Lim 1990, Low 1991, Lim 1984, Schalkoff 1989).

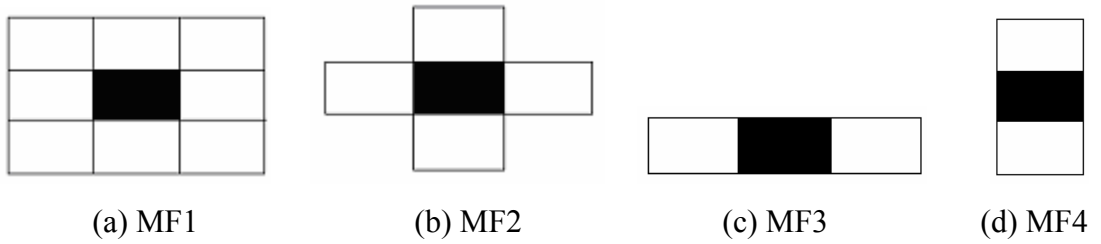
Median filtreleme için çeşitli kalıplar mevcuttur. (Şekil 3.4) Filtreleme için kullanılacak olan kalıp içerisindeki hücre sayısı N olmak üzere, maskenin görüntü üzerine yerleştirilmesi durumunda o maske içerisindeki median pozisyonu:

$$m = \frac{(N-1)}{2} \quad (3.22)$$

eşitliği ile verilir. Burada m değeri median parlaklığıdır. En çok kullanılan median filtre, N 'nin tek sayı durumu için türetilir. Seçilen filtre maskesinin tam ortasındaki hücre (Şekil 3.4'deki koyu hücreler), görüntü içerisindeki her bir piksel üzerine gelecek şekilde median filtre kalıbı, iki boyutlu diziye aktarılmış olan sayısal görüntünün pikselleri üzerinde kaydırılır. Her bir kaydırma sonucunda kalıp içerisinde bulunan gri seviye değerleri küçükten büyüğe sıralanır. Bu değerlerin $(N-1)/2$ adedinden büyük ama kalıp içerisinde kalan diğer piksel değerlerinde küçük olan ilk değer, median değeridir. Örneğin, Şekil 3.1 (a)'daki median filtre maskesi için $N=9$ 'dur. Filtreleme sonucunda elde edilen görüntüdeki yeni piksel değerleri, bu maske içerisindeki 9 değerın küçükten büyüğe sıralanması sonucunda elde edilen dizideki $(9-1)/2=4$ değerden büyük olan ilk değer, yani 5'inci değerdir. Bu işlem, median filtre maskesinin, görüntüdeki bütün pikseller üzerine uygulanmasıyla tüm bir görüntü için gerçekleştirilir (Kızılkaya 1997, Lim 1990, Low 1991, Lim 1984, Schalkoff 1989).

Kullanılan bir median filtrede çerçevenin büyüklüğü önemli bir parametredir. Çerçeve büyüklüğünün seçimi görüntünün içeriğine bağlıdır. Maskenin görüntü üzerine yerleştirilmesi durumunda, eğer maske içinde kalan gürültülü piksel sayısı kalıbın toplam piksel sayısının yarısına eşit veya daha fazla ise filtrenin performansı düşük olur. Bu nedenle optimal filtre boyutunu bulmak için, değişik boyutlarda

median filtreler kullanarak elde edilen sonuçlara göre çerçeve boyutunu seçmek daha iyidir. Görüldüğü üzere optimum çerçeve büyüklüğünü seçmek zordur. Daha önceden (3.22) eşitliği ile verilen median değerini bulabilmek için $3x(N^2-1)/8$ tane karşılaştırma işlemi yapmak gerekir. Bu işlem sayısı 3x3 boyutlu bir filtre kalıbı için 30 (N=9 olacağından dolayı) 5x5 boyutlu bir filtre kalıbı için ise 234 (N=25 olacağından dolayı) olur. İşlem sayısını azaltmak ve 2-boyutlu median filtrelerin basamak fonksiyonuna etkisini ortadan kaldırmak için ayrılabilir median filtreler kullanılmaktadır (Şekil 3.4 (c)-(d)). Ayrılabilir median filtreler, 1-boyutlu median filtrelerin (yatay ve düşey yönde) görüntüdeki her bir satır ve sütuna ayrı ayrı uygulanmasıyla elde edilmektedir. Diğer bir pratik median filtre tipi ise çapraz (+) median filtre tipidir (Şekil 3.4 (b)) (Ataman ve ark. 1980, Bednar ve Watt 1984, Kızılkaya 1997, Lim 1990, Low 1991, Schalkoff 1989).



Şekil 3.4 Median Filtre Kalıpları, (a) 2-boyutlu median filtre kalıbı, (b) Çapraz (+ tipi) median filtre kalıbı, (c) Yatay median filtre kalıbı, (d) Düşey median filtre kalıbı.

Median filtreler darbe ve tuz-biber gürültüleri için iyi sonuç verirler ama Gaussian gürültüleri için performansları, bunlara nazaran biraz kötüdür. Median filtrenin bulanıklaştırma etkisi, Gaussian gürültülü görüntülerde biraz daha fazladır (Kızılkaya 1997). Median filtrelerin sahip olduğu özellikler şu şekilde sıralanabilir (Schalkoff 1989):

- 1) Median filtre, görüntü içerisindeki parlaklıkların değişimini azaltır. Bundan dolayı median filtre, belirli bir şekilde görüntü yapısını değiştirme kabiliyetine sahiptir.

- 2) Eğer görüntü içerisindeki bölgesel gürültü dağılımı median filtreyi çevreleyen çerçeve içerisinde simetrik değilse median filtreler görüntü parlaklığı ortalama değerini değiştirir.
- 3) Median filtreler belirli olan kenar çizgilerini korur. Bu, çok geçişli median filtre gerçeklemede önemlidir. Çünkü median filtrenin sabit noktaları öncelikle kenarlar ve değişmez eğilimli bölgelerdir.
- 4) Simetrik olarak verilen bir median filtre, kenarların yerini korur.
- 5) Median filtrenin görüntü üzerine uygulanması sonucunda yeni gri seviye değerleri üretilmez. Sadece ilgilenilen pikselin gri seviyesinin yerini aynı komşuluk içerisinde bulunan başka bir gri seviye değeri alır. İkili görüntüler ikili olarak kalır ve median filtrelenmiş görüntünün dinamik aralığı giriş görüntüsünün dinamik aralığını aşamaz.
- 6) Median filtre için seçilen çerçeve şekli görüntüyü işleme sonuçlarını etkileyebilir.

3.3.1. Merkez ağırlıklı median filtre (CWMF)

Sayısal görüntülerden dürtü gürültüsünü gidermeye yönelik oldukça fazla sayıda yöntem geliştirilmiştir. Standart median filtre (Umbaugh 1998), filtre kalıbının merkez elemanının değerini kalıp içerisindeki elemanların orta değeri ile değiştirerek dürtü gürültüsünü gidermeye çalışır. Bu yaklaşım, her ne kadar düşük gürültü yoğunluklarında bile ince hatları zayıflatsa ve görüntüyü bulanıklaştırsa da kabul edilebilir bir gürültü giderme başarımına sahiptir. Merkez görüntü elemanına daha çok ağırlık verecek şekilde değiştirilmiş merkez ağırlıklı median filtre (Center Weighted Median Filter, CWMF) (Ko ve Lee 1991), standart median filtrenin doğasındaki yetersizliklerin üstesinden gelebilmek için geliştirilmiş bir median filtre çeşididir. Filtre gürültü giderme oranı çok az bozulsun da standart median filtreye göre daha iyi detay ve doku koruma başarımı göstermektedir.

Standart median filtre ve merkez ağırlıklı median filtre, gürültülü ve gürültüsüz görüntü elemanları arasında herhangi bir ayırım yapmadan görüntüleri

işleyen yani konumsal olarak değişkenlik göstermeyen operatörlerdir. Bu nedenle, bu filtreler görüntünün gürültülü bölgelerindeki gürültüyü giderirken gürültüsüz bölgelerini de bu sürece dahil ederek bu bölgeleri bulanıklaştırırlar. Bu probleme yönelik, gürültü sezme mekanizması içeren adaptif median filtre gerçekleştirilebilir. Adaptif median filtre ile ilgili ayrıntılı bilgi ilerleyen bölümlerde verilecektir.

Merkez Ağırlıklı Median filtrelere girmeden önce median filtrelerin matematiği hakkında bilgi vermek faydalı olacaktır.

$\{X(.,.)\}$ ve $\{Y(.,.)\}$ sırasıyla median filtrenin giriş ve çıkışını ifade etmek üzere, (3.23) eşitliği geçerlidir:

$$Y(i, j) = \text{median}\{X(i-s, j-t) \mid (s, t) \in W\} \quad (3.23)$$

Burada W , merkezin komşuluğundaki görüntü koordinatlarına göre tanımlanan maskeyi ifade etmektedir. Örneğin, $W = \{(s, t) \mid -N \leq s \leq N, -N \leq t \leq N\}$ olmak üzere $(2N+1) \times (2N+1)$ 'lik kare bir maske verilebilir. Maskedeki noktaların toplam sayısı ise maske boyutu olarak adlandırılmaktadır. W maskesinin merkezinin $(0,0)$ olduğunu ve orijine göre simetrik olduğunu varsayarak $(s, t) \in W$ ifadesinin $(-s, t) \in W$, $(s, -t) \in W$ ve $(-s, -t) \in W$ 'yi kapsadığını söyleyebiliriz. Her bir maske içindeki piksel değerlerinin medianını hesaplamak mümkündür. Bütün maske ise $2L+1$ olarak ifade edilmektedir. c 'nin maske boyutuna eşit veya maske boyutundan daha büyük tek bir tam sayı olduğu hesaba katılarak bir ağırlıklı median filtrenin ağırlıkları eşitlik (3.24)'deki gibi ifade edilebilir:

$$\{h(s, t) \mid (s, t) \in W, \sum_{(s, t) \in W} h(s, t) \equiv c\} \quad (3.24)$$

$Y(i, j)$ çıkışını elde etmek için ağırlıklı median filtre her $(s, t) \in W$ için $X(i-s, j-t)$ 'nin $h(s, t)$ kopyalarını üretir. Daha sonra da c örneklerinin median değeri hesaplanır. Böylece, $Y(i, j)$ değeri eşitlik (3.25)'deki gibi ifade edilebilir:

$$Y(i, j) = \text{median}\{X(i-s, j-t) \text{ 'nin } h(s, t) \text{ deęerleri} \mid (s, t) \in W\} \quad (3.25)$$

Örneęin aęırlıklı median filtre için, $W = \{(-1,0)(0,0)(1,0)\}$ ve aęırlıkların da $\{h(-1,0), h(0,0), h(1,0)\} = \{2,3,2\}$ olduęu varsayılırsa, $Y(i,j)$ çıkışı eřitlik (3.26)'daki gibi olmaktadır:

$$Y(i,j) = \text{median} \{X(i-1,j), X(i-1,j), X(i,j), X(i,j), X(i,j), X(i+1,j), X(i+1,j)\} \quad (3.26)$$

Merkez aęırlığı $h(0,0) = 2K+1$ ve her $(s, t) \neq (0, 0)$ için $h(s,t) = 1$ olan aęırlıklı median filtreye merkez aęırlıklı median filtre (CWMF) adı verilir. (K, negatif olmayan bir tamsayıdır) CWM filtrenin $Y(i,j)$ çıkışı aşıęıdaki gibi verilmektedir:

$$Y(i, j) = \text{median}\{X(i-s, j-t), X(i, j) \text{ 'nin } 2K \text{ deęerleri} \mid (s, t) \in W\} \quad (3.27)$$

$K=0$ olduęu zaman CWM filtre normal median filtre gibi davranır ve $2K+1$ deęeri maske boyutu $2L+1$ 'e eřit veya ondan daha büyük olduęu zaman filtreleme yapılmamıř olur. Açııkça görölmektedir ki, daha büyük merkez deęerli bir CWM filtre detayları korumada daha küçük merkez deęerli bir filtreden daha iyidir ancak gürültü azaltmada daha kötüdür (Ko ve Lee 1991).

3.3.2. Adaptif median filtre (AMF)

MXN boyutlu bir görüntünün (i,j) koordinatlarındaki pikselinin gri seviye deęerinin $x_{i,j}$ ve x 'in dinamik aralıęının da $[S_{\min}, S_{\max}]$ olduęunu varsayalım. ($(i, j) \in A \equiv \{1, \dots, M\} \times \{1, \dots, N\}$ olmak üzere) Yani, $S_{\min} \leq x_{i,j} \leq S_{\max}$, $(i, j) \in A$ ve y , gürültülü görüntü olmak üzere, klasik tuz-biber darbe gürültü modellerinde (i,j) piksel koordinatlarındaki gri seviye deęeri:

$$y_{i,j} = \begin{cases} S_{\min}, & p \text{ olasılıđıyla} \\ S_{\max}, & q \text{ olasılıđıyla} \\ x_{i,j}, & 1-p-q \text{ olasılıđıyla} \end{cases} \quad (3.28)$$

eşitliđiyle verilmektedir. Burada, $r = p + q$, deđeri gürültü seviyesini belirtmektedir. Merkezi (i,j) koordinatlarında bulunan $w \times w$ boyutlarındaki filtre kalıbının $S_{i,j}^w$ ve maksimum filtre maskesi boyutu da $w_{\max} \times w_{\max}$ olduđu varsayılırsa, eşitlik (3.29) geçerlidir:

$$S_{i,j}^w = \{(k,l) : |k-i| \leq w \text{ ve } |j-l| \leq w\} \quad (3.29)$$

Buna göre aşıđıdaki adımlar çerçevesinde, $y_{i,j}$ ile ifade edilen olası gürültü noktaları bulunmakta ve her $y_{i,j}$ deđeri, $S_{i,j}^w$ filtre kalıbında bulunan piksellerin median deđeri ile yani bu piksellerin bir çeşit orta deđeri ile deđiştirilmektedir. Her bir piksel deđeri için aşıđıdaki adımlar uygulanmaktadır (Chan ve ark. 2005):

- 1) $w = 3$, filtre boyutu 3'tür.
- 2) $S_{i,j}^w$ 'deki yani kalıptaki piksel deđerlerinin sırasıyla minimum, median (orta deđer) ve maksimumlarını ifade eden, $s_{i,j}^{\min,w}$, $s_{i,j}^{\text{med},w}$, $s_{i,j}^{\max,w}$ deđerleri hesaplanır.
- 3) Eđer, $s_{i,j}^{\min,w} < s_{i,j}^{\text{med},w} < s_{i,j}^{\max,w}$ ise 5. adıma geçilir. Deđilse, $w = w + 2$ yapılır.
- 4) Eđer, $w \leq w_{\max}$ ise 2.adıma geçilir. Deđilse, $y(i,j)$, $s_{i,j}^{\text{med},w_{\max}}$ ile deđiştirilir.
- 5) Eđer, $s_{i,j}^{\min,w} < y_{i,j} < s_{i,j}^{\max,w}$ ise $y_{i,j}$ gürültü bir piksel deđildir. Aksi taktirde $y(i,j)$ $s_{i,j}^{\text{med},w}$ ile deđiştirilir.

Bu adaptif filtre yapısına göre yüksek seviyeli darbe gürültülerinin birçođu filtre kalıbının boyutunun yeterince geniş seçilmesi şartıyla kolaylıkla tespit edilebilmektedir. Burada gözden kaçırılmaması gereken nokta, gürültülü piksellerin

değerlerinin kalıptaki piksellerin median değeri olan $s_{i,j}^{med,w}$ ile değiştirilmesine rağmen, diğer piksellerin değiştirilmeden kalmasıdır (Chan ve ark. 2005)

3.3.3. Adaptif merkez ağırlıklı median filtre (ACWMF)

Non-linear filtre tasarımında Gaussian ve dürtü gürültüsünü eşzamanlı kaldırmak için Mallows aşağıdaki iki adımlı yaklaşımı önermiştir (Mallows 1980): İlk olarak, orijinal sinyal üzerindeki Gaussian gürültüsünü azaltmak için uygun bir lineer filtre seçilir. İkinci adımda, lineer parametresi lineer filtreninkine yakın olan ve gürültüyü iyi bir şekilde azaltan bir non-linear filtre seçilir. Adaptif bir median filtre kenar koruma ve gürültü bastırma özelliklerinin daha iyi kombinasyonlarını merkez ağırlıklı bir median filtreden daha iyi elde etmekte ve eş zamanlı olarak hem dürtü gürültülerini, hem katkılı beyaz gürültüleri hem de sinyal-bağımlı gürültüleri bastırmaktadır (Ko ve Lee 1991).

Bir merkez ağırlıklı filtrenin lineer bileşeninin maskenin merkezindeki değeri en ağırlıklı değer olarak seçtikten sonra her maske içindeki verinin ortalamasını alan bir FIR filtre olduğu yaklaşımı kullanılmaktadır. Giriş bağımsız özdeş dağılımlı rastgele bir işlem olduğu zaman her maske içindeki veriden birini çıkış olarak seçen bir non-linear filtrenin lineer bileşeni, darbe cevabı eşitlik (3.30)'daki gibi olan bir FIR filtredir:

$$h(s,t) = \Pr\{Y(i,j) = X(i-s, j-t)\} \quad (3.30)$$

Burada $X(i-s,j-t)$ 'nin (i,j) noktasındaki çıkışını belirtmektedir (Mallows 1980). Bu sonuca göre, $(s,t) \in \mathcal{W}$ olmak üzere, merkez ağırlıklı filtrenin lineer bileşeninin aşağıdaki gibi olduğu kolaylıkla belirtilebilir:

$$h(s,t) = \begin{cases} (2K+1)/(2L+1), & \text{eğer } (s,t) = (0,0) \\ (1-K/L)/(2L+1), & \text{eğer } (s,t) \neq (0,0) \end{cases} \quad (3.31)$$

$\sum_{(s,t) \in W} h(s,t) = 1$ olduğu göz ardı edilmemelidir. $(s,t) \neq (0,0)$ olduğu zaman $h(0,0) \geq h(s,t)$ ve $h(s,t)$ tamamen eşit olduğu için daha büyük ağırlık sadece merkez değerine yüklenmiştir. Eşitlik (3.31)'deki bu filtre merkez ağırlıklı ortalama filtre görevi görmektedir (Ko ve Lee 1991).

Lee ve Kuan'ın uyarlamalı smoothing filtre daha çok ağırlığı her maskenin sadece merkezdeki değerine veren uyarlamalı bir filtredir (Lee 1980, Kuan ve ark. 1985). Katkılı beyaz gürültülü bir giriş için elde edilen adaptif filtre çıkışı aşağıdaki gibidir;

$$Y(i, j) = A(i, j) + R(i, j)[X(i, j) - A(i, j)] \quad (3.32)$$

Burada $A(i,j)$, (i,j) merkezli bir pencerenin içindeki değerlerin ortalamasını ifade etmektedir.

$$R(i, j) = \begin{cases} (\hat{\sigma}_x^2(i, j) - \sigma_n^2) / (\hat{\sigma}_x^2(i, j)), & \text{eğer } \hat{\sigma}_x^2(i, j) \geq \sigma_n^2 \\ 0, & \text{o.w.} \end{cases} \quad (3.33)$$

Burada, $\hat{\sigma}_x^2(i, j)$, pencere içindeki verinin örnek değişimini; σ_n^2 , katkılı beyaz gürültünün değişimini ifade etmektedir. Bu filtre ortalama filtre ve lokal istatistiklere bağlı özdeş filtre arasında bir filtre olarak tanımlanabilir. Kısaca ifade edersek (3.33) için eğer pencere içinde hem dürtü gürültüsü bileşenleri hem de kenarlar mevcutsa $\hat{\sigma}_x^2(i, j) \geq \sigma_n^2$ ve $R(i, j) \leq 1$ geçerlidir. Böyle bir durumda, $Y(i,j)=X(i,j)$ 'dir ve bu filtre kenarları korur ancak gürültüleri bastıramaz. Diğer taraftan, girişin derece derece değişen parçalarında, $\hat{\sigma}_x^2(i, j) \leq \sigma_n^2$ ve böylece $Y(i, j) \leq A(i, j)$ olmaktadır (Ko ve Lee 1991).

(3.32)'deki $Y(i,j)$ çıkışı eşitlik (3.34)'deki gibi verilebilir:

$$Y(i, j) = \sum_{(s,t) \in W} h_{ij}(s, t) X(i-s, j-t) \quad (3.34)$$

Burada, $h_{ij}(s,t)$ ifadesi eşitlik (3.35)'de verildiği gibi olmaktadır:

$$h_{ij}(s,t) = \begin{cases} [2LR(i,j)+1]/(2L+1), & \text{eğer } (s,t) = (0,0) \\ [1-R(i,j)]/(2L+1), & \text{eğer } (s,t) \neq (0,0) \end{cases} \quad (3.35)$$

$\sum_{(s,t) \in W} h_{ij}(s,t) = 1$ olduğuna dikkat edilmelidir. Bu filtrenin en ağırlıklı değerinin sadece merkezde olduğu açıkça görülmektedir. Böylece (3.35)'deki bu özel filtrenin *adaptif merkez ağırlıklı ortalama* (ACWA) filtre olarak görev yaptığını söylemek mümkündür.

Merkez ağırlıklı median filtre ile merkez ağırlıklı ortalama filtre arasındaki ilişki göz önünde bulundurarak merkez değerini giriş datasına bağlı olarak ayarlayabilen bir ACWM filtrenin geliştirilmesi mümkündür. Merkez değeri $2K(i,j)+1$ olan bir ACWM filtre varsayılarak eşitlik (3.31)'den ACWM filtrenin lineer bileşeni eşitlik (3.36)'daki gibi yazılabilir:

$$h_{ij}(s,t) = \begin{cases} [2K(i,j)+1]/(2L+1), & \text{eğer } (s,t) = (0,0) \\ [1-K(i,j)/L]/(2L+1), & \text{eğer } (s,t) \neq (0,0) \end{cases} \quad (3.36)$$

(3.35) ve (3.36) eşitlikleri karşılaştırılırsa, eşitlik (3.37)'nin sağlanması durumunda:

$$K(i,j) = LR(i,j) \quad (3.37)$$

ACWM filtrenin lineer bileşenin ACWA filtrenin lineer bileşeni ile aynı olacağını görmek mümkündür. Ancak şu söylenebilir ki, (3.37)'deki eşitliği sağlayan ACWM, filtre gürültü bastırımında pek etkili değildir. Çünkü, $K(i,j)$ dürtülere duyarlı olan bir örnek değişimine bağlıdır ve bu filtrenin bir uç değeri özdeş bir filtredir. ACWA filtre gibi, ACWM filtre de eğer pencere içinde dürtü mevcutsa bir özdeş filtre olma eğilimi gösterir. Bunun kolay bir çözümü merkez ağırlık değerinin, ACWM filtrenin özdeş bir filtre olmasını sağlayan değeri almasını engelleyerek sınırlandırmaktır. Bu

adaptif merkez ağırlıklı median filtrenin $K(i,j)$ parametresini eşitlik (3.38)'deki gibi ifade etmek mümkündür (Ko ve Lee 1991):

$$K(i, j) = [(L - T)R(i, j)] \quad (3.38)$$

Burada T ; $0 \leq T \leq L$ olan bir tam sayıdır ve $[x]$, x 'in yuvarlanmış değerini ifade etmektedir. $0 \leq R(i, j) \leq 1$ olduğundan, $0 \leq K(i, j) \leq L - T$ elde edilir. Böylece ACWM filtre median filtre ve $2K+1=2(L-T)+1$ olan bir CWM filtre arasında değişir. Genellikle, ACWM filtre kenar, hat ve dürtülerin komşuluklarında $2(L-T)+1$ merkez ağırlıklı bir CWM filtre gibi davranırken, görüntünün aşamalı olarak değişen kısımlarında median filtre gibi davranır. ACWM filtreler, CWM filtrelerde olduğu gibi daha az dürtü gürültüsü olan görüntü parçalarında detayları korurlar. Ayrıca non-impulsive gürültülerin bastırımında CWM filtrelerden çok daha etkili olmaktadır. Uygun bir CWM filtrenin merkez değerine uygun olarak T parametresini belirleyebilmek için şu yöntem izlenebilir: Eğer merkez ağırlığı $2K_1+1$ olan bir CWM filtre kenarları korurken aynı zamanda dürtü gürültülerini de bastırabiliyorsa, T parametresini $L-K_1$ 'de belirleyebiliriz. Yani, $T=L-K_1$ olan bir ACWM filtre, hatların ve dürtülerin yakınında $2K_1+1$ 'lik bir CWM filtre gibi davranır. (Ko ve Lee 1991).

3.3.4. Kenarları ve iyi detayları koruyan hibrit median filtre yapısı

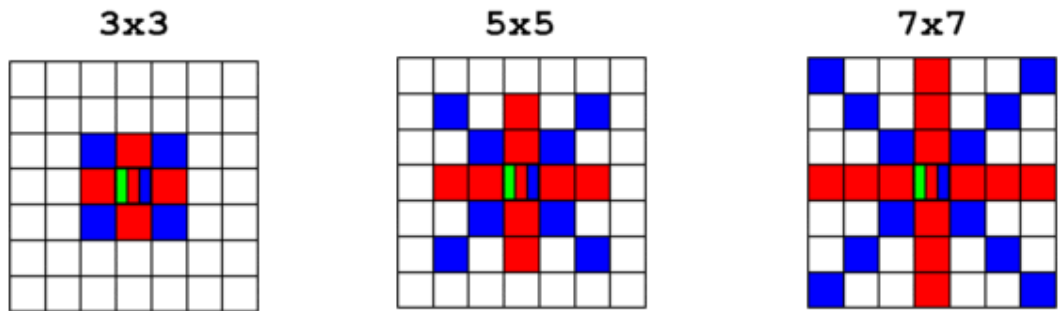
Daha önce de belirtildiği gibi median filtre, sıra istatistiği filtre grubuna dahil olan yani filtre istatistiklerinin filtre kalıbındaki elemanların sıralanması sonucunda belirlendiği bir filtre tipidir. Median filtre, hesaplama sonucunun bir ağırlıklı toplama eşit olmadığı konvolüsyon işlemine benzeyen non-lineer bir komşuluk işlemidir. Bunun yerine, komşuluk dahilindeki pikseller gri seviye değerlerine göre sıralanırlar ve bu grup içindeki orta değer çıkış pikseli olarak elde edilir.

Median filtreleme normalde konvolüsyondan daha yavaş bir işlemdir çünkü her bir komşuluktaki bütün piksellerin sıralanması gerekmektedir. Gün geçtikçe

geliştirilen algoritmalar sayesinde bu süreçler hızlandırılmaktadır (Huang ve ark. 1979). Median filtre, görüntüdeki kenarları çok bulanıklaştırmadan dürtü gürültülerini lineer alçak geçiren filtrelere göre daha iyi azalttığından daha çok tercih edilmektedir. Ancak Gaussian gürültüsü gibi bazı impulsive olmayan gürültülerde aynı etkiyi gösterememektedir (Khriji ve Gabbouj 1999, Taguchi ve Murata 1991). Hem gürültünün etkili bir biçimde azaltılması hem de kenarların ve iyi detayların korunabilmesi için genellikle lineer ve non-lineer işlemleri de içeren hibrid yapılar kullanmak gerekmektedir.

Bu tezde kullanılan hibrit filtre ile görüntünün gürültülü pikselleri yumuşatılırken, kenarlar ve iyi detaylar ise en uygun şekilde korunmaktadır. Temel median filtrenin en önemli dezavantajlarından biri konumunun değişmez (location-invariant) olmasıdır. Yani görüntünün gürültülü olmayan bölümlerine de etki eder. Bu yüzden gürültüsüz bölümlerde aşırı bulanıklaşmalar meydana gelebilmektedir.

Hibrit median filtreler görüntüdeki gürültüyü ortadan kaldırırken aynı zamanda kenarları daha iyi koruyan filtrelerdir. Buna göre normal median filtrelere göre daha optimal sonuçlar vermektedir. Hibrit median filtreler aşağıda da görüldüğü gibi kompleks bir yapıya sahiptirler. Bu nedenle sayısal işlemlerin yapılması dahi oldukça zor ve karmaşık olmaktadır. Şekil 3.5’de en çok kullanılan hibrit median filtre kalıpları görülmektedir.



Şekil 3.5 Hibrit Median Filtre Kalıbı

Bu tezde 5x5 boyutunda ve 7x7 boyutunda hibrit median filtre kalıpları kullanılmaktadır. Kullanılan 7X7 boyutundaki hibrit median filtre kalıbı Şekil 3.6’de

verildiği gibidir. Kalıbın merkezi (m,n) ile ifade edildiğinde kalıbın diğer katsayılarının koordinatları Şekil 3.7’de verildiği gibi olmaktadır:

$$\begin{array}{ccccc}
 e_3 & & d_3 & & c_3 \\
 & & e_2 & & d_2 & & c_2 \\
 & & & & e_1 & & d_1 & & c_1 \\
 f_3 & f_2 & f_1 & a & b_1 & b_2 & b_3 \\
 & & & & g_1 & & h_1 & & i_1 \\
 & & & & g_2 & & h_2 & & i_2 \\
 g_3 & & & & h_3 & & & & i_3
 \end{array}$$

Şekil 3.6 7x7’lik bir median filtre maskesi

$$\begin{array}{ccccccc}
 (m-3, n-3) & & & & (m-3, n) & & & & (m-3, n+3) \\
 & & & & (m-2, n) & & & & (m-2, n+2) \\
 & & & & (m-1, n-1) & (m-1, n) & (m-1, n+1) & & \\
 (m, n-3) & (m, n-2) & (m, n-1) & (m, n) & (m, n+1) & (m, n+2) & (m, n+3) & & \\
 & & & & (m+1, n-1) & (m+1, n) & (m+1, n+1) & & \\
 & & & & (m+2, n) & & & & (m+2, n+2) \\
 (m+3, n-3) & & & & (m+3, n) & & & & (m+3, n+3)
 \end{array}$$

Şekil 3.7 Kullanılan 7x7’lik hibrit median filtre kalıbının koordinatlarının gösterimi

Merkezdeki (m,n) koordinatındaki ilk değer a ile gösterilirse, medianı hesaplanmış değer a' ile gösterilmektedir. Buna göre,

$$a' = \text{median}(a, \text{median}(a, b, d, f, h), \text{median}(a, c, e, g, i)) \quad (3.39)$$

eşitliği geçerli olmaktadır.

4. GÖRÜNTÜ NETLEŞTİRME VE KENAR BULMA ALGORİTMALARI

4.1. Görüntüyü Netleştirmek İçin Kullanılan Algoritmalar

Görüntülerde netleştirme işleminin temel amacı, görüntü içerisindeki iyi detayları kuvvetlendirmek ve görüntünün elde edilmesi sırasında kullanılan metodun doğal etkisi ya da hatası sonucu oluşan bulanıklaşmanın giderilmesini sağlayarak görüntünün detaylarını daha görünebilir hale getirmektir. Çünkü bulanık bir görüntüde kenarların keskinliği az, çizgiler ve ayrıntılar tam olarak belirgin değil yani nesnelerin seçimi zordur.

Görüntü içerisindeki kenarlar ve çizgiler görüntünün yüksek frekanslı bileşenlerini içermektedir. Bulanıklaşmaya ise daha önce de belirtildiği üzere alçak frekanslı bileşenler sebebiyet vermektedir. Genel olarak yüksek frekans geçiren filtreler kullanılarak, görüntü içerisindeki yüksek frekanslı bileşenler öne çıkarılırken alçak frekanslı bileşenler süzülerek zayıflatılır. Görüntüdeki yüksek frekanslı bileşenler; kenarlarda bulunan, ayrıntıları izleyen, ani parlaklık/gri seviye değişimleridir. Yüksek geçiren filtreleme sonucu görüntüdeki ufak değişiklikler kolaylıkla sezilebilir. Filtreleme ile görüntü içerisindeki alçak frekanslı bileşenler azaldığından ve de yüksek frekanslı bileşenler görüntüye hakim olduğundan bu görüntüde arka plan gürültüsü artmaktadır. Çünkü görüntünün arka plan gürültüsü tipik olarak yüksek frekanslı bileşenlere sahiptir (Lim 1990, Lim 1984).

Yüksek geçiren filtrelemedeki mantık, ilgilenilen pikselin gri seviyesinin öne çıkarılması ve buna komşu olan diğer piksellerdeki gri seviye değerlerinin zayıflatılması olduğundan bunu sağlamak için kullanılan 3x3'lük bir yüksek geçiren filtre kalıbı şu şekilde gerçekleştirilmektedir: Filtre kalıbının merkezi pozitif bir sayı ile, buna komşu olan pikseller ise negatif sayılarla ifade edilmektedir. Bu filtre kalıbı içerisindeki katsayıların toplamı sıfır olduğundan değişik katsayılarla çarpılarak bu filtre kalıbı kontrol edilebilmektedir. Yani bu şekilde birçok filtre kalıbı elde edilebilmektedir (Kızılkaya 1997). Bu tezde kullanılacak olan netleştirme

algoritmaları Wiener filtre, Düzenlenmiş (Regularized) filtre, Blind-Dekonvolüsyon filtresi ve Lucy-Richardson filtresidir.

4.1.1. Wiener filtre

Wiener filtre, hem görüntüdeki bulanıklığın giderilmesinde hem de görüntüdeki gürültülerin etkilerinin azaltılmasında kullanılabilen bir filtre modelidir. Görüntülerde bulanıklıklar çok çeşitli nedenlerle meydana gelmektedirler. Örneğin fotoğraf makinesindeki bozukluk veya çekim esnasındaki odak kayması nedeniyle meydana gelmiş bulanık bir görüntüde optimum netleştirme sağlamak için kullanılacak en uygun yöntemlerden biri Wiener Filtre'dir.

Bulanık veya bozuk bir görüntüyü yaklaşık olarak aşağıdaki denklemle ifade etmek mümkündür;

$$g = Hf + n \quad (4.1)$$

Burada, g : bulanık görüntü, H : PSF (nokta yayılım fonksiyonu) olarak adlandırılan bozma operatörü, f : orijinal görüntü, n : görüntü data esnasında meydana gelen ve görüntüde ek bozulmalara neden olan gürültülerdir

Uzay domeninde PSF ışık yayan optik bir sistemin derecelerini tanımlayan bir fonksiyon olarak belirtilmektedir. PSF, optik transfer fonksiyonu (OTF)'nin ters fourier dönüşümüne, OTF de PSF'nin fourier dönüşümüne eşittir. Frekans domeninde ise optik transfer fonksiyonu lineer sistemin impuls cevabını tanımlar. Bir görüntü üzerinde yapay olarak bulanıklık oluşturmak için PSF kullanılmaktadır. Bunun dışında görüntü üzerinde yapay bulanıklık oluşturmak için kullanılan pek çok fonksiyon mevcuttur.

Eşitlik (4.1)'deki g görüntüsüne Wiener yaklaşımı uygulanarak, elde edilecek görüntü f' ile ifade edilmektedir:

$$f' = Lg \quad (4.2)$$

Wiener filtreyi de eşitlik (4.3)'deki gibi ifade edersek,

$$L_{Wiener} = \phi f (\phi f + \phi n)^{-1} \quad (4.3)$$

Elde edilen filtrelenmiş f' görüntüsü eşitlik (4.4)'deki gibi olmaktadır.

$$f' = \phi f (\phi f + \phi n)^{-1} g \quad (4.4)$$

Bir görüntüde bulanıklığın düzeltilmesi istenen görüntünün kalitesi, birincil olarak PSF bilgisine bağlıdır. Yani en iyi düzelme işlemi sağlamak için bozulma parametreleri hakkında bilgi sahibi olmak gerekmektedir.

4.1.2. Düzenlenmiş (Regularized) filtre

Görüntüdeki bulanıklığın yanında görüntüye eklenmiş olan gürültünün nedenleri hakkında her hangi bir fikrimizin olmadığı durumlarda uygulanabilecek en iyi görüntü netleştirme tekniklerinden biridir. Ancak bu algorithmada da girişte bozukluğa neden olan PSF fonksiyonu hakkında bilgi sahibi olmak daha iyi sonuçlar alınmasını sağlayacaktır.

4.1.3. Lucy-Richardson algoritması

Lucy-Richardson Algoritması, bulanık görüntüyü yinelemeli yöntemle onaran bir netleştirme algoritmasıdır. İterasyonun özü eşitlik (4.5)'deki gibidir: (www.numis.northwestern.edu)

$$görüntü_{n+1} = görüntü_n \frac{orjinaldata}{görüntü_n * PSF} * reflect(PSF) \quad (4.5)$$

Burada, * operatörü konvolüsyonu, $reflect(PSF)$ ifadesi ise PSF'nin yansımasını yani tersini ifade etmektedir. Ayrıca,

$$reflect((PSF)(x, y)) = PSF(-x, -y) \quad (4.6)$$

eşitliği geçerlidir, eğer konvolüsyon işlemi hızlı fourier dönüşümü (FFT) kullanılarak yapılacaksa,

$$FFT(\text{reflect}(PSF)) = \text{conj}(FFT(PSF)) \quad (4.7)$$

eşitliği geçerlidir. Burada *conj* ifadesi karmaşık-eşlenik sayıyı ifade etmektedir. Eğer görüntü hakkında bir ön bilgi mevcutsa kullanılacak olan PSF fonksiyonunu isteğe göre özelleştirmek mümkündür. Eğer görüntü hakkında bir ön bilgi yoksa dekonvolüsyon için kullanılacak olan PSF fonksiyonu, giriş görüntüsüne yani orijinal görüntüye uyan sabit değerlerle belirlenir.

4.1.4. Blind-Dekonvolüsyon algoritması

Blind dekonvolüsyon/restorasyon/netleştirme algoritması, bozuk görüntünün PSF'si hakkında herhangi bir bilgi sahibi olmadan (bulanıklık veya gürültü) görüntünün onarılmasını sağlayan oldukça kullanışlı bir görüntü netleştirme algoritmasıdır. Aynı zamanda Blind Dekonvolüsyon Algoritması görüntü kalitesini karmaşık kalibrasyon metotlarına gerek duymadan artırmak için kullanılan alternatif yöntemlerden biridir. Genellikle hiç bir görüntü netleştirme modelinde görüntüdeki bulanıklığı meydana getiren bütün etkileri tam anlamıyla belirlemek mümkün olmamaktadır. Bununla birlikte, bu bağımsız rastgele faktörler arasındaki karmaşık etkileşimin sonucunu, standart sapması σ olan bir Gaussian nokta yayılım fonksiyonu (PSF) ile yaklaşık olarak hesaplamak mümkündür: (Jiang ve Wang 2003)

$$G_{\sigma}(x) = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{|x|^2}{2\sigma^2}\right] \quad (4.8)$$

Blind dekonvolüsyon terimi ilk olarak Stockham ve arkadaşları tarafından incelenmiştir (Stockham ve ark. 1975). Kavram olarak ortaya çıkışı ise Oppenheim ve arkadaşlarının 1968'deki çalışmalarına kadar uzanmaktadır (Oppenheim ve ark. 1968). Blind dekonvolüsyon kavramının ortaya çıkışını takip eden yıllar boyunca aynı amaçla bir çok metot geliştirilmiş ancak algoritmalarından bir çoğu, malesef

yöntemlerin kararlılığı, verimliliği ve eşsizliği konusunda pek memnun edici sonuçlar ortaya koyamamıştır. Fakat uygulamalarda geniş kullanım alanı bulması ve önemli bir algoritma olmasından dolayı Blind Dekonvolüsyon algoritması gelişimini oldukça hızlı bir biçimde sürdürmüştür (Jiang ve Wang 2003).

4.2. Kenar Bulma Algoritmaları

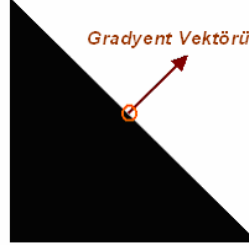
Bir görüntüdeki kenarları bulmanın temel amacı , görüntü içerisinde istenilen detayları ortaya çıkarmak, istenmeyen detayları ise ortadan kaldırmaktır. Ayrıca görüntüye kenar bulma algoritmalarını uygulamak suretiyle görüntüdeki renk geçişlerini keskinleştirmek ve böylece resim içindeki objeleri ya da farklılıkları elde etmek mümkün olmaktadır. Görüntü keskinleştirme; elektronik baskı, medikal görüntüleme, endüstriyel muayene, güçlü silahların otomatik hedef tayini gibi pek çok alanda kullanılan bir görüntü işleme tekniğidir (Gonzales ve ark. 2004). Başlıca kenar bulma algoritmaları aşağıdaki şekilde sıralanmaktadır:

- 1) Gradyent kenar bulma algoritması
 - a) Roberts metodu
 - b) Sobel metodu
 - c) Prewitt metodu
- 2) LoG (Laplacian of Gradient) kenar bulma algoritması
- 3) Canny kenar bulma algoritması

4.2.1. Gradyent kenar bulma algoritması

Bir bölge boyunca piksellerin ortalamasının alınması görüntü içerisindeki detayları bulanıklaştırmaktadır. Ortalama alma işlemi integral alma işlemine benzemektedir. Bunun gibi, fark işleminin ortalama alma (integral) işleminin tam tersi olması beklenebilir. Böylece bu fark alma işlemi görüntüyü keskinleştirir. Görüntü işleme uygulamalarında fark alma işleminin en genel metodu *gradyent* işlemidir.

Bir görüntüde yüksek gradyente sahip olan pikseller kenar (edge) pikselleri olarak ifade edilebilir. Yani Gradyent vektörü yönündeki parlaklık şiddetindeki değişim oranı kenar pikseller üzerinde gözlenmektedir.



Şekil 4.1 Gradyent vektörü ve bir kenar pikseli

Şekil 4.1’de ideal bir kenar pikseli ve onun gradyent vektörü görülmektedir. İşaretli olan nokta, kenar pikselin yerini göstermektedir. Piksel üzerinde parlaklık şiddeti, 0’dan 255’e gradyent vektörü doğrultusunda değişmektedir. Gradyentin büyüklüğü (magnitüde) kenarın şiddetini göstermektedir. Eğer gradyent uniform bir bölgede hesaplanırsa sıfır elde edilir ki bu da o bölgede hiç kenar piksel olmadığını gösterir.

Doğal görüntülerde bu şekilde olduğu gibi ideal süreksiz bölgeler veya ideal uniform bölgeler yoktur. Bu nedenle kenar noktaları bulmak için gerekli olan gradyentin büyüklüğünü hesaplamak bu kadar kolay değildir. Bu durumda yapılacak en basit işlem bir eşik değeri uygulamaktır. Eğer gradyentin büyüklüğü bu eşik değerinden büyükse o noktadaki pikselin bir kenar piksel olduğuna karar verilir. Bir kenar piksel iki önemli özelliklerle ifade edilebilir:

- 1) Gradyentin büyüklüğü olarak ifade edilen kenar şiddeti,
- 2) Gradyentin açısı olarak ifade edilen kenar yönü,

Bir görüntü fonksiyonu $I(x,y)$ 'nin (x,y) koordinatlarındaki gradyenti $\nabla I(x,y)$,

$$\nabla I(x, y) = \frac{\partial I(x, y)}{\partial x} \cos\theta + \frac{\partial I(x, y)}{\partial y} \sin\theta \quad (4.9)$$

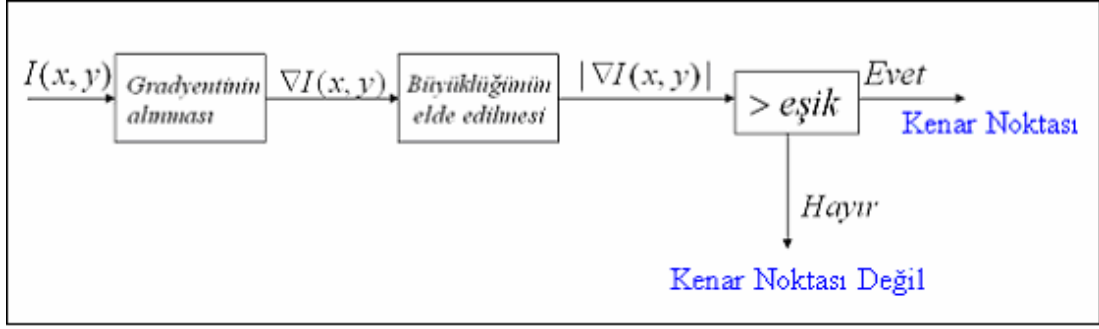
eşitliği ile verilir. Burada θ , gri seviyelerin değişim açısıdır. Gradyent kenar bölgelerinde yöresel maksimum yaptığı için (4.9) eşitliğinin sıfıra eşitlenmesi kenar büyüklüğü ve yönünü verir. Gradyentin genliği,

$$|\nabla I(x, y)| = \sqrt{\left(\frac{\partial I(x, y)}{\partial x}\right)^2 + \left(\frac{\partial I(x, y)}{\partial y}\right)^2} \quad (4.10)$$

ve yönü ise,

$$\theta_G = \tan^{-1}\left(\frac{\partial I_y}{\partial I_x}\right) \quad (4.11)$$

ifadesi ile bulunur. (4.10) eşitliği ile hesaplanan gradyentin büyüklüğü belli bir eşik değerinden büyükse bu piksel değeri kenar noktası olarak tanımlanır. Buradaki eşik değeri, kenar kabul edilebilecek olan gri seviyesi değişiminin gradyent değeridir. Kenar görüntüsünün performansı seçilen eşik değeri ile yakından ilgilidir. Eşik değeri, gradyent değerleri büyük olanları küçük olanlardan ayıracak şekilde seçilmelidir. Uygun eşik değeri gradyentin histogramına bakılarak sağlıklı bir biçimde belirlenebilir. Genellikle eşik değeri maksimum gradyent değerinin % 5 ile %10 'u olarak seçilebilir. Kenar görüntüsünde eşik değerinden büyük bütün noktalar kenar noktası olarak tanımlanırsa, oluşan kenar görüntüsü ince çizgiler halinde değil kalın şeritler halinde görünecektir. İnce çizgiler elde etmek için kenar görüntüsüne kenar inceltme yöntemi uygulanır ve böylece kenar kalınlıkları, istenilen inceliğe getirilebilir (Gonzales ve Woods 1993). Şekil 4.2'de bir kenar tanıma sisteminin blok diyagramı verilmektedir.



Şekil 4.2 Gradyent Metodu Blok Diyagramı

Gradyent metoduna dayalı kenar tanıma sistemleri iki değişik şekilde uygulanabilir; bunlar *yönlü* ve *yönsüz kenar operatörleri* şeklindedir. $|\nabla I(x, y)|$ büyüklüğünü kullanan sistemlere yönsüz kenar operatörü adı verilmektedir. Çünkü bu sistem herhangi bir doğrultu için ayarlı değildir ve her yön için eşit ağırlıklı sonuçlar verir. Eğer sistem her hangi bir yön için ayarlanmışsa bu sistemlere *yönlü kenar dedektörü* adı verilmektedir. Örnek olarak, $|\partial I(x, y)/\partial x|$ şeklindeki fonksiyonu kullanan bir sistem dikey doğrultudaki kenarları tanıırken yatay doğrultudaki kenarları tanımaz. Fonksiyonlara ilişkin türev hesaplamaları fark denklemleri ile ifade edilebilir. Bu durum iki-boyutlu (2-D) görüntü fonksiyonu için de geçerlidir. Bu durumda $\frac{\partial I(x, y)}{\partial x}$, $\frac{\partial I(x, y)}{\partial y}$ türev ifadeleri,

$$\frac{\partial I(x, y)}{\partial x} = [I(x+1, y+1) - I(x-1, y+1) + I(x+1, y) - I(x-1, y) + I(x+1, y-1) - I(x-1, y-1)] \quad (4.12)$$

$$\frac{\partial I(x, y)}{\partial y} = [I(x+1, y+1) - I(x+1, y-1) + I(x, y+1) - I(x, y-1) + I(x-1, y+1) - I(x-1, y-1)] \quad (4.13)$$

şeklinde belirtilebilir. (4.12) ve (4.13)' deki fark denklemleri; $I(x, y)$ görüntü fonksiyonu ile impuls cevabı $h(x, y)$ olan bir filtre kalıbının, konvolüsyona tabi tutulması sonucu elde edilen değerlerdir. Buradaki $h(x, y)$ impuls cevablı filtreler, kullanılan yönlü veya yönsüz kenar operatörleridir. Kenar keskinleştirme ve ortaya çıkarma için bu çalışmada kullanılan yönsüz kenar operatörlerinde, hem x hem de y doğrultusu olmak üzere iki yönde iki ayrı filtre kalıbı kullanılarak konvolüsyon

işlemi uygulanır. Ve her iki yönde elde edilen değerlerin kareleri alınarak sonuçlar toplanır ve daha sonra toplamın kare kökü alınır. Elde edilen sonuç gradyentin mutlak değerini verir. $I(x,y)$ 'nin x 'e ve y 'ye göre türevleri,

$$\frac{\partial I(x,y)}{\partial x} = I(x,y) * h_x(x,y) \quad (4.14)$$

$$\frac{\partial I(x,y)}{\partial y} = I(x,y) * h_y(x,y) \quad (4.15)$$

şeklinde ifade edilebilir. (4.14) ve (4.15) eşitliklerindeki h_x ve h_y yönsüz kenar tanıma operatörlerinin sırasıyla x ve y yönündeki maskeleridir. Yönsüz kenar tanıma maskeleri (4.12) ve (4.13) eşitliklerinde görüldüğü üzere Çizelge 4.1 'deki verilen bir yapı ile elde edilebilir.

Çizelge 4.1 3x3'lük bir görüntü bölgesi

$I(x-1,y-1)$	$I(x-1,y)$	$I(x-1,y+1)$
$I(x,y-1)$	$I(x,y)$	$I(x,y+1)$
$I(x+1,y-1)$	$I(x+1,y)$	$I(x+1,y+1)$

Görüntü fonksiyonunun x -yönündeki türevi için (4.12) eşitliğini, y -yönündeki türevi için de (4.13) eşitliğini kullanarak ve Çizelge 4.1'deki yapıdan faydalanarak sırasıyla x ve y yönünde kenar tanıma için, Şekil 4.3'deki gibi kalıplar elde edilir. Elde edilen bu kalıplara *Prewitt kenar tanıma* kalıpları adı verilir. Görüntü fonksiyonunun x ve y yönündeki türevlerinin farklı kombinasyonları için, Çizelge 4.1'den yararlanarak daha başka kenar tanıma kalıpları elde edilebilmektedir.

a) Prewitt kenar bulma metodu: Şekil 4.3'de Prewitt kenar tanıma kalıpları görülmektedir.

-1	-1	-1
0	0	0
1	1	1

(a)

-1	0	1
-1	0	1
-1	0	1

(b)

Şekil 4.3 (a) Yatay Prewitt Kalıbı (Horizontal Mask)

(b) Dikey Prewitt Kalıbı (Vertical Mask)

b) Sobel kenar bulma metodu: Sobel kalıbı da yine 3x3 boyutlu olup, Prewitt kalıbına benzemekle beraber tek farkı katsayılarıdır. Sobel kenar tanıma kalıpları da Şekil 4.4'da gösterildiği gibidir.

-1	-2	-1
0	0	0
1	2	1

(a)

-1	0	1
-2	0	2
-1	0	1

(b)

Şekil 4.4 (a) Yatay Sobel Kalıbı (Horizontal Mask)

(b) Dikey Sobel Kalıbı (Vertical Mask)

c) Roberts kenar bulma metodu: Roberts kenar bulma kalıbı 2x2'lik bir kalıptır ve Şekil 4.5'de gösterildiği gibidir.

1	0
0	-1

0	1
-1	0

Şekil 4.5 Roberts kenar bulma kalıpları

4.2.2. LoG (Laplacian of Gaussian) kenar bulma algoritması

Laplasyen filtreler görüntüdeki keskin geçişleri bulmak için kullanılan türev bazlı filtrelerdir. Türev bazlı filtreler gürültüye çok duyarlı oldukları için, görüntüye laplasyeni uygulamadan önce Gaussian filtresi gibi bir smoothing fitresinden geçirerek gürültüleri azaltmak daha iyi sonuç vermektedir. Bu iki aşama sonucu elde edilen filtreleme işlemine LoG filtreleme metodu adı verilmektedir (<http://academic.mu.edu>). Bir $f(x,y)$ fonksiyonunun laplasyen uygulanmış şekli aşağıda gösterildiği gibidir:

$$L(x, y) = \nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (4.16)$$

Bu değeri bir konvolüsyon filtresi kullanarak bulmak mümkündür. Bu değere yaklaşık olarak da olsa ulaşmak için kullanılacak olan konvolüsyon kalıplarından en çok kullanılanları Şekil 4.6'da verildiği gibidir:

0	1	0
1	-4	1
0	1	0

1	1	1
1	-8	1
1	1	1

-1	2	-1
2	-4	2
-1	2	-1

Şekil 4.6 En çok kullanılan Laplasyen kalıpları

Bu kalıplardan herhangi biri kullanılarak, standart konvolüsyon metodları sayesinde görüntü fonksiyonunun laplasyenini hesaplamak mümkündür. Bu kalıplar görüntünün ikinci türev değerine yaklaştığı için gürültüye oldukça duyarlıdır. Bunu azaltmak için genellikle görüntü laplasyen filtresinden önce Gaussian filtresi gibi bir smoothing filtreden geçirilir. Bu ön işlem sayesinde görüntüdeki yüksek frekanslı gürültü bileşenleri azaltılmış olur.

Normalde, konvolüsyon işlemi birleşme özelliğine sahip bir işlem olduğundan Gaussian yumuşatma filtresiyle Laplasyen filtresini kendi aralarında

konvolüsyona tabi tutarak, daha sonra bu hibrid filtre ile görüntünün konvolüsyonunu almak suretiyle istenilen sonucu elde etmek mümkündür. Bunu yapmanın iki türlü avantajı vardır:

- 1) Hem Gaussian hem de Laplasyen kalıpları genellikle görüntüden daha küçük boyutta olduğundan, bu metod sayesinde daha az aritmetik işleme gerek duyulmaktadır.
- 2) LoG kalıbı ön bir işlemle önceden oluşturulduğundan görüntü üzerinde tek bir konvolüsyon işlemiyle sonuca ulaşılmaktadır.

Sıfır merkezli olmak üzere iki boyutlu (2-D) bir LoG fonksiyonu;

$$LoG(x, y) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2 + y^2}{2\sigma^2}} \quad (4.17)$$

σ ; Gaussian standart sapması'dır. Şekil 4.7'de farklı LoG kalıpları görülmektedir.

0	-1	0
-1	5	-1
0	-1	0

0	-2	0
-2	5	-2
0	-2	0

Şekil 4.7 Farklı LoG kalıpları

4.2.3. Canny kenar bulma algoritması

Canny kenar bulma algoritması aşağıdaki dört adımla uygulanabilir:

- Görüntüye bir Gaussian filtresi yardımıyla bulanıklaştırma işlemi yapılır.
- Kısmi türevler için sonlu-fark yaklaşımları kullanılarak gradyentin büyüklüğü ve yönü hesaplanır.
- Görüntünün büyüklük değerinde maksimum olmayan noktaların bastırılması işlemi uygulanır.
- Kenar pikselleri bulmak için çift eşikleme algoritması kullanılır.

Canny kenar bulma algoritması sinyal gürültü oranını optimize edecek şekilde tasarlanmıştır (<http://ari.cankaya.edu.tr>).

a) Bulanıklaştırma (Smoothing): $I(i,j)$; Orijinal görüntü, $G(i,j;\sigma)$; Gaussian bulanıklaştırıcı filtre ve σ ; Gaussian filtresinin standart sapması (yumuşatma derecesi) olmak üzere, orijinal $I(i,j)$ görüntüsü ve $G(i,j;\sigma)$ filtresinin konvolüsyonu sonucu elde edilen yumuşatılmış görüntü $S[i,j]$ ile ifade edilmektedir.

$$S[i,j] = G [i,j, \sigma] * I(i,j) \quad (4.18)$$

b) Gradyentin hesaplanması: İlk olarak eşitlik (4.18)'de verilen $S[i,j]$ 'nin kısmi türevleri elde edilir. Bunlar:

$$P[i,j] \approx (S[i,j+1]-S[i,j]+S[i+1,j+1]-S[i+1,j])/2 \quad (4.19)$$

$$Q[i,j] \approx (S[i,j]-S[i+1,j]+S[i,j+1]-S[i+1,j+1])/2 \quad (4.20)$$

şeklinindedir. x ve y kısmi türevleri 2x2'lik kare matris üzerindeki sonlu farkların ortalaması alınarak hesaplanır. Buna göre gradyentin büyüklük değeri:

$$M(i, j) = \sqrt{P[i, j]^2 + Q[i, j]^2} \quad (4.21)$$

ve açısı:

$$Q[i, j] = \arctan(Q[i, j], P[i, j]) \quad (4.22)$$

olur.

c) Maksimum Olmayan Noktaların Bastırılması (Nonmaxima Suppression): Gradyent algoritmasında, giriş görüntüsünün büyüklük değerinin gradyenti alınarak kenar pikselleri belirlemek mümkündür. Fakat Canny metodu karmaşık bir yaklaşıma sahiptir. Bu yaklaşıma göre bir kenar noktası, şiddeti gradyent vektörü yönünde lokal olarak maksimum olan bir nokta olarak düşünülebilir. Bu durum oldukça kısıtlayıcı

bir durumdur ve eşikleme yöntemiyle bulunan kenar piksellerinden oluşan çizgileri inceltmek için kullanılan bu işleme *Maksimum Olmayan Noktaların Bastırılması* adı verilir. Bu işlemden sonra elde edilen $N[i,j]$ görüntüsü aşağıdaki gibidir:

$$N[i, j] = nms(M[i, j], \zeta[i, j]) \quad (4.23)$$

Yerel maksimum noktaları olarak kabul edilen noktalarda bu değer sıfırdır.

d) Eşikleme (Thresholding): İlk adımda smoothing işlemi gerçekleştirilmesine rağmen, maksimum olmayan noktaları bastırılmış olan $N[i,j]$ görüntüsünde, gürültüler nedeniyle bazı hatalı kenar noktalarının oluşması olasıdır. Bu hatalı kenar noktalarının etkisi çok fazla değildir. Maksimum olmayan noktaları bastırılmış bu görüntüdeki hatalı kenar noktaları azaltmak gerekmektedir. Bunları azaltmanın yöntemi $N[i,j]$ 'ye bir eşik değeri uygulayarak eşik değerinin altındaki bütün noktaları sıfır yapmaktır.

Maksimum olmayan noktaları bastırılmış görüntüye eşikleme işlemi uyguladıktan sonra elde edilen görüntü yani $E(i,j)$ kenarları belirginleşmiş bir görüntü haline gelmiştir. Ancak bu metotta uygun eşik değerini bulmak zordur, deneme yoluyla bulmak gerekmektedir. Eğer eşik değerini çok küçük seçersek, $E(i,j)$ görüntüsünde karşılaşılan yanlış kenarlar fazla olur, eşik değerini çok büyük seçersek de bazı kenarlar kaybolabilir. Bu nedenle daha etkili bir eşikleme işlemi için iki eşik değeri beraber kullanılmaktadır. τ_1 ve τ_2 olarak belirtilen bu eşik değerleri $N[i,j]$ 'ye uygulanarak istenilen görüntü elde edilir. Bu eşik değerleri için $\tau_2 = 2\tau_1$ bağıntısı geçerlidir. Bu eşik değerlerine göre $T_1[i,j]$ ve $T_2[i,j]$ olmak üzere iki yeni görüntü elde edilir. T_2 görüntüsünde, kenar piksellerden oluşan hat üzerinde açıklıklar vardır. Fakat dışarıda kalan yanlış kenarların sayısı o kadar fazla değildir. Bu nedenle T_1 görüntüsü kullanılarak, T_2 kenar hattındaki açıklıklar kapanarak optimum düzeltme sağlanmış olur (<http://ari.cankaya.edu.tr>).

5. GÖRÜNTÜ İŞLEME ALGORİTMALARININ TARİHİ ESER GÖRÜNTÜLERİNE UYGULANMASI

5.1. Görüntüdeki Gürültüyü Azaltmak İçin Yapılan Uygulamalar

Bölüm 2’de anlatıldığı gibi, görüntüdeki gürültü genellikle insanın görsel algısı için istenmeyen bir durumdur ve bu gürültü görüntüdeki önemli özelliklerin ve ayrıntıların görülme miktarlarını azaltmakta hatta bazen tamamen kapatarak bunların tamamen görülemez hale gelmesine neden olmaktadır. Gürültünün etkisinin azaltılmasından sonra gerçekleştirilecek olan kenar belirleme, görüntü bölütleme, nesne tanıma vb. gibi görüntü işleme tekniklerinin başarılı sonuçlar verebilmesi için gürültünün etkisinin optimum şekilde giderilmesi gerekmektedir. Bu bölümde çeşitli gürültülere sahip tarihi eser görüntülerine çeşitli gürültü azaltım algoritmaları uygulanacaktır.

5.1.1. Hitirlere ait askeri alay figürü

Şekil 5.1’de Ankara Anadolu Medeniyetleri Müzesi’ne ait, 19.Yüzyıl’dan kalma, bazalt taşı üzerine yapılmış, Hitirlere ait bir askeri alay figürü görülmektedir. Şekil 5.1’de de görüldüğü üzere bazalt taşının yapısı itibariyle ve zamanın da verdiği etkiyle görüntü üzerinde oldukça belirgin pürüzler ve görüntünün bütünü olumsuz biçimde etkileyerek insan görüşünü bozan benekler mevcuttur. Bu beneklerin etkisini azaltmak için görüntüye sırasıyla *normal median filtre* ve 7×7 boyutunda maskeye sahip olan *hibrit median filtre* uygulanmış ve elde edilen sonuçlar Şekil 5.2 ve Şekil 5.3’ de verilmiştir. Görüntülerde de görüldüğü gibi normal median filtre gürültülü görüntüdeki benekleri azaltmakla beraber, görüntüdeki iyi detayları da azaltmıştır. Ancak uygulanan hibrit median filtre algoritması görüntüdeki iyi detayları da koruyarak optimum sonuç elde etmemizi sağlamıştır.



Şekil 5.1 Hititlere ait askeri alay figürünün gürültülü görüntüsü



Şekil 5.2 Hititlere ait askeri alay figürünün gürültülü görüntüsünün normal median filtreden geçirilmiş hali



Şekil 5.3 Hititlere ait askeri alay figürünün gürültülü görüntüsünün 7X7 maske boyutuna sahip hibrit median filtreden geçirilmiş hali

5.1.2. İstanbul Türk ve İslami Eserler Müzesi'nden alınmış bir görüntü

Şekil 5.4'de Anadolu Selçukluları'ndan kalma, 13'üncü -14'üncü yüzyıllarda yapıldığı düşünülen ve alçı üzerine oyularak yapılmış atları üzerinde savaşan askerleri ifade eden bir görüntü görülmektedir. Bu görüntü üzerine yapay olarak tuz-biber gürültüsü ekleyerek uygulanan hibrit algoritmanın görüntü üzerinde ne tür bir iyileşme sağladığını görmek daha kolay olacaktır. Şekil 5.4'de görüntünün orijinal hali ve Şekil 5.5'de de görüntünün gürültülü hali verilmektedir.

Yıllar öncesine ait görüntülerde genellikle toz zerreciklerinin ve pasın etkisiyle bu tür gürültüler meydana gelmektedir. Uygulanan hibrit algoritmanın etkisini göstermek amacıyla görüntüye suni gürültü eklenmiştir. Görüntünün gürültülü hali Şekil 5.5'de verildiği gibidir. Gürültülü görüntüye hibrit median filtrenin uygulanması sonucu elde edilen görüntü ise Şekil 5.6'da verildiği gibidir.



Şekil 5.4 Anadolu Selçukluları'na ait alçı üzerine yapılmış bir tarihi eser görüntüsü



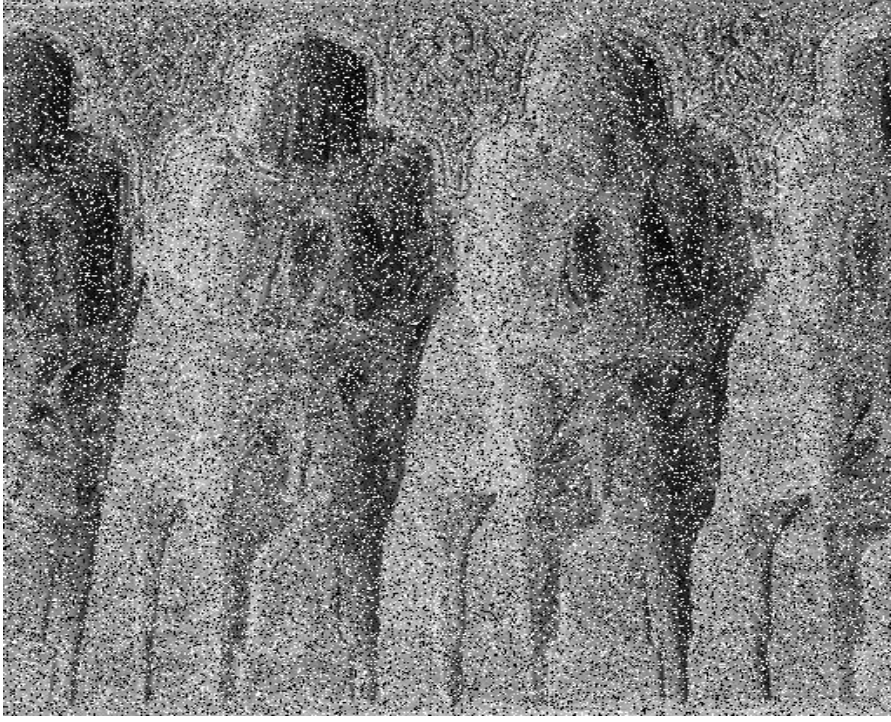
Şekil 5.5 Anadolu Selçukluları'na ait alçı üzerine yapılmış tarihi eser görüntüsünün yapay gürültü eklenmiş hali



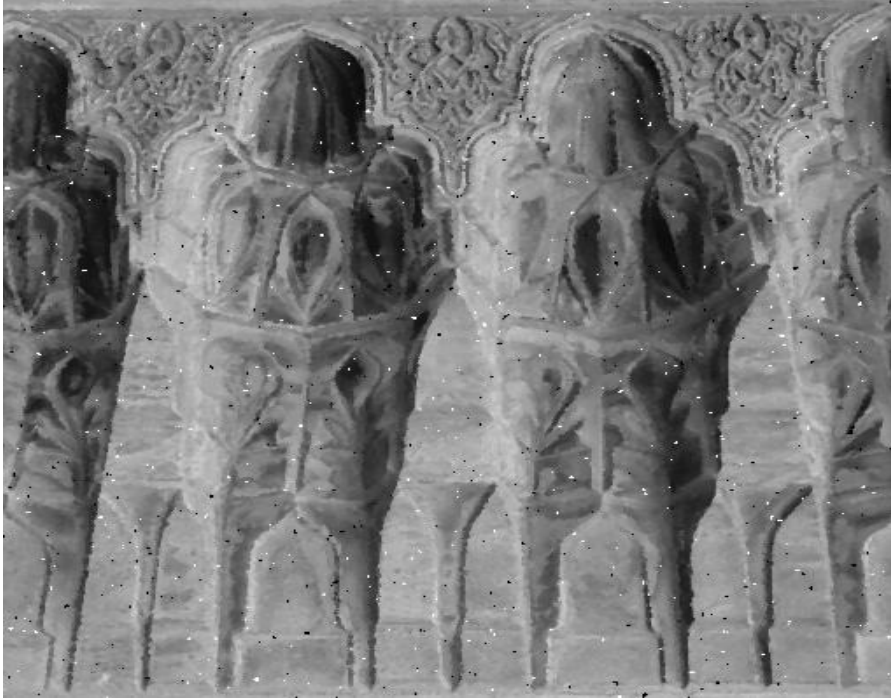
Şekil 5.6 Anadolu Selçukluları'na ait alçı üzerine yapılmış tarihi eser görüntüsünün hibrit median filtre ile onarılmış hali

5.1.3. İznik Yeşil Cami'ye ait bir süsleme görüntüsü

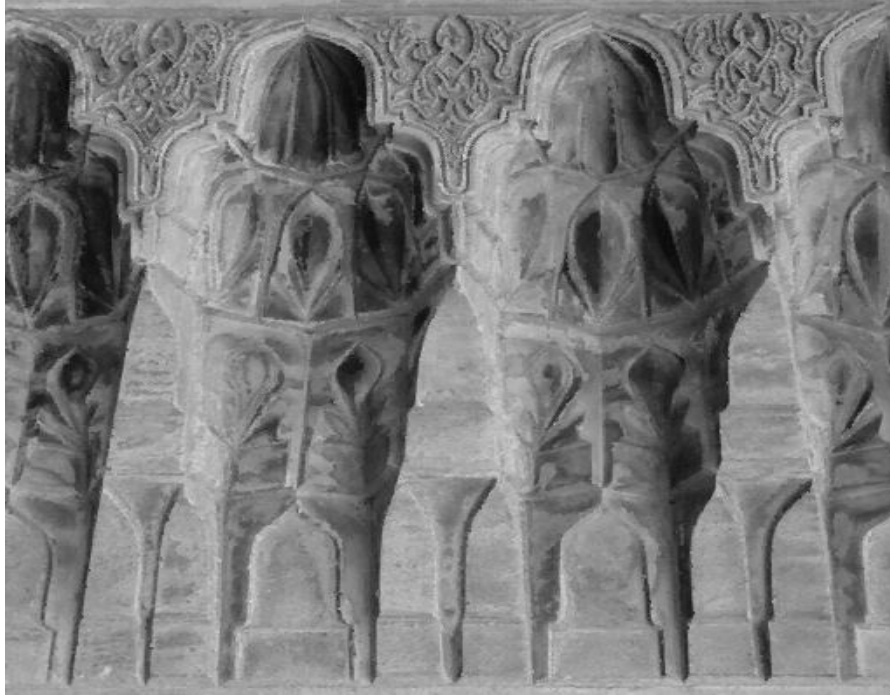
Şekil 5.7'de İznik'deki Yeşil Cami'nin ön duvarını süsleyen, klasik Selçuklu mimarisinin eski usul kübik formunu yansıtan bir süsleme görüntüsü görülmektedir. Bazı kaynaklara göre 1492 yılında yapıldığı söylenen bazı kaynaklara göre ise bir asır önce inşa edildiği belirtilen bu eser tarihimiz açısından oldukça önemli bir yere sahiptir. Şekil 5.7'deki gürültülü süsleme görüntüsüne adaptif median filtre uygulanması sonucu elde edilen görüntü Şekil 5.9'de verildiği gibidir. Karşılaştırılmanın yapılması açısından normal median filtre uygulanmış görüntü de Şekil 5.8'de verilmiştir. Şekil 5.8'de ve Şekil 5.9'da görüldüğü gibi gürültülü görüntüye adaptif median filtre uygulandığında elde edilen sonuç normal filtre uygulandığında elde edilen sonuca göre daha iyidir. Normal median filtreye tabi tutulmuş görüntüde gürültünün etkileri az da olsa devam ederken, adaptif median filtre uygulanmış görüntü gürültülerden büyük ölçüde arındırılmıştır.



Şekil 5.7. İznik Yeşil Cami'nin ön duvarındaki süsleme görüntüsü



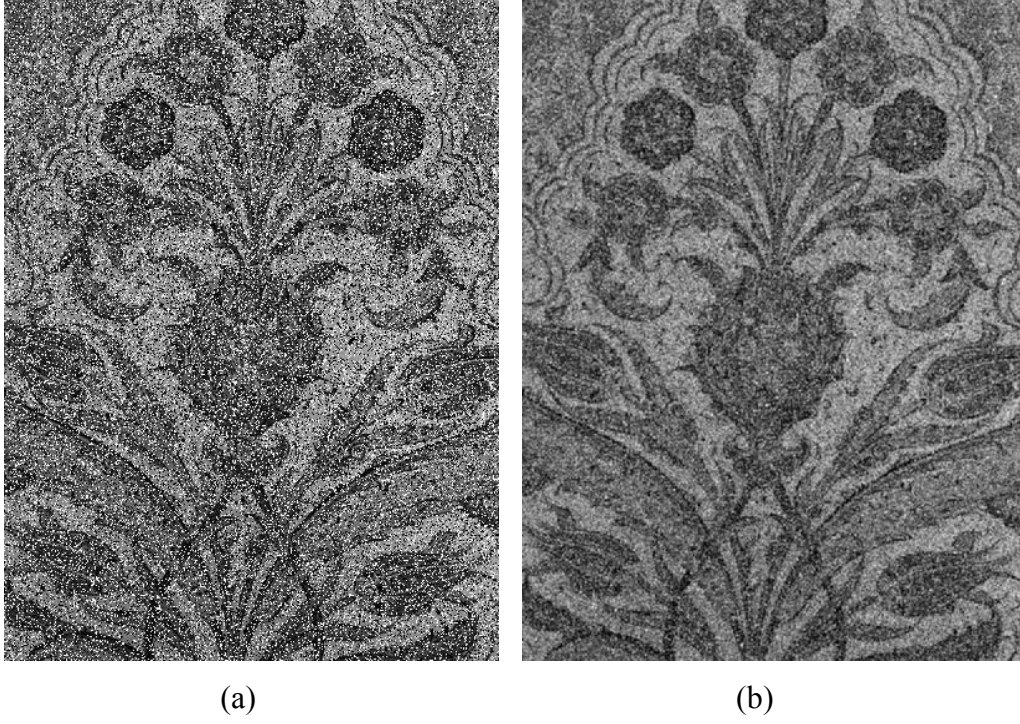
Şekil 5.8. İznik Yeşil Cami'nin ön duvarındaki süsleme görüntüsünün normal median filtre uygulanmış hali



Şekil 5.9. İznik Yeşil Cami'nin ön duvarındaki süsleme görüntüsünün adaptif median filtre (AMF) uygulanmış hali

5.1.4. Kütahya Seramik Müzesi'nden alınmış bir seramik eser görüntüsü

Şekil 5.10 (a)'da Kütahya Seramik Müzesi'nden alınmış bir seramik görüntüsünün gürültülü hali görülmektedir. Bu görüntüye mean filtre uyguladığımızda elde edilen görüntü Şekil 5.10 (b)'deki gibi, 5X5 boyutunda maskeye sahip bir hibrit median filtre uyguladığımızda elde edilen görüntü ise Şekil 5.10 (c)'deki gibi olmaktadır.



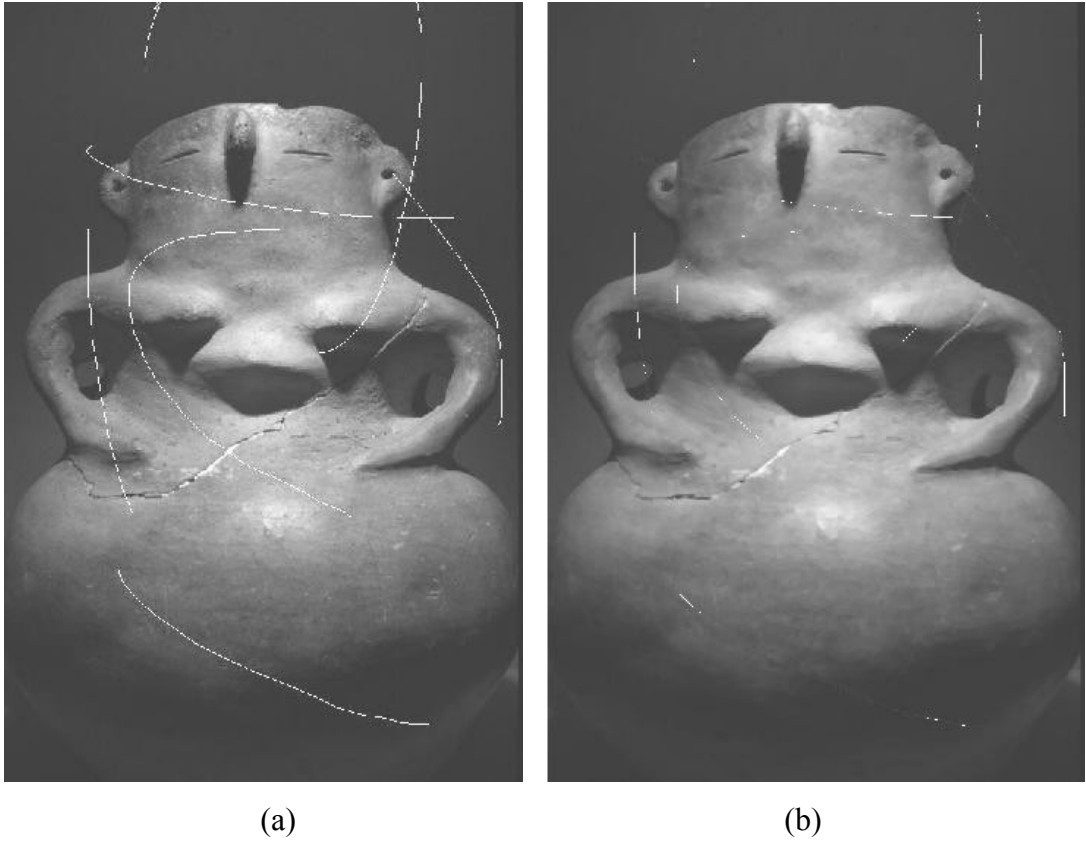
Şekil 5.10 (a) Kütahya Seramik Müzesi'nden alınmış seramik eser görüntüsünün gürültülü hali (b) Gürültülü Kütahya seramik görüntüsüne mean (ortalama) filtre uygulandıktan sonra elde edilen görüntü (c) Gürültülü Kütahya seramik görüntüsüne 5x5'lik hibrit median filtre uygulandıktan sonra elde edilen görüntü

5.2. Görüntüdeki Çizikleri veya Yırtıkları Kaldırmak İçin Yapılan Uygulamalar

Görüntüler üzerinde sonradan meydana gelen çiziklerin ve yırtıkların onarılması için seçilen örnek tarihi eser görüntüleri üzerinde uygulamalar yapılmıştır.

5.2.1. Antropomorfik (İnsan Formunda) vazo görüntüsü

Şekil 5.11 (a)'da Ankara Anadolu Medeniyetleri Müzesi'ne ait, yaklaşık 6000 yıllık bir antropomorfik vazo görüntüsü görülmektedir. Bu eser görüntüsü üzerine yapay olarak beyaz kalemle çizgiler çizerek görüntüler sırayla normal median filtre ve hibrit median filtreden geçirilmiştir. Hibrit median filtreleme sonucunda elde edilen görüntü Şekil 5.11 (b)'de görüldüğü gibi olurken, normal median filtreleme sonucunda elde edilen görüntü ise Şekil 5.12'deki gibi olmaktadır.



Şekil 5.11 (a) Ankara Anadolu Medeniyetleri Müzesi'ne ait Antropomorfik vazo görüntüsü
(b) Antropomorfik vazo görüntüsünün hibrit median filtreden geçirilmiş hali



Şekil 5.12 Ankara Anadolu Medeniyetleri Müzesi'ne ait Antropomorfik vazo görüntüsünün normal median filtreden geçirilmiş hali

Bu eserin elimizde kalan tek kanıtının kağıt üzerinde bir fotoğraf olduğu ve bu fotoğraf üzerinde de zamanla çizikler oluştuğu varsayılırsa, resmi dijital hale getirip optimum düzeyde gerekli onarımlar yaptıktan sonra tekrar kağıda basmak görüntünün daha uzun yıllar saklanması ve tarihimize ışık tutması açısından faydalı olacaktır.

Elde edilen sonuçlar karşılaştırıldığında, vazo görüntüsü için normal median filtrenin, hibrit median filtreye göre daha iyi sonuç verdiği görülmektedir. Çünkü genel olarak median filtre görüntüdeki gürültüyü azaltırken, hibrit median filtre ise aynı zamanda detayları da korumakta ve daha çok detayların ön plana çıkarılmasının gerektiği durumlarda etkin olarak kullanılmaktadır. Vazo görüntüsü ise, diğer görüntülerde olduğu gibi çok fazla detay ve kenar içermediğinden, görüntünün bulanıklaşmamasından ve detayların kaybolmamasından ziyade, çizgilerin yok edilmesi görüntünü onarımında bizim için daha önemli olmaktadır. Yani aynı algoritma farklı görüntüler üzerinde farklı sonuçlar verdiği için, bir görüntüyü

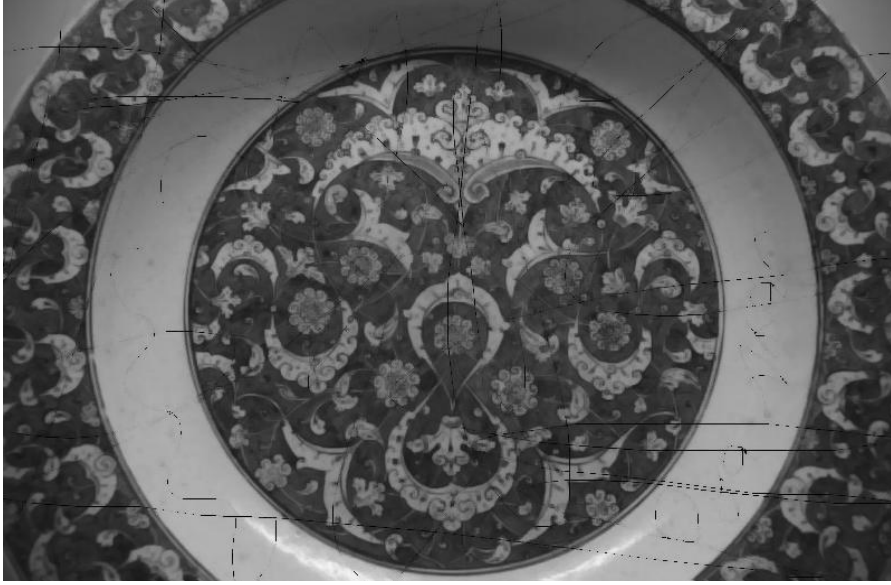
onarırken o görüntü için kullanılacak algoritmaları seçerek elde edilen sonuçlar arasında amacımıza en uygun görüntüyü kullanmak çalışmaların verimliliği açısından faydalı olacaktır.

5.2.2. İstanbul Çinili Köşk'den alınmış seramik bir tabak görüntüsü

İstanbul Çinili Köşk Seramik Müzesi'ne ait Şekil 5.13'deki görüntü üzerine çizgiler çizilmiş daha sonra bu görüntü 7X7'lik hibrit median filtreden ve normal median filtreden geçirilmiştir. Şekil 5.13'de görülen seramik tabak görüntüsüne hibrit median filtre uygulandıktan sonra elde edilen görüntü Şekil 5.14'de gösterildiği gibi, normal median filtre uygulandıktan sonra elde edilen görüntü ise Şekil 5.15'da gösterildiği gibi olmaktadır.



Şekil 5.13 İstanbul Çinili Köşk Seramik Müzesi'ne ait tabak görüntüsü



Şekil 5.14 İstanbul Çinili Köşk Seramik Müzesi'ne ait tabak görüntüsünün hibrit median filtre uygulandıktan sonraki hali



Şekil 5.15 İstanbul Çinili Köşk Seramik Müzesi'ne ait tabak görüntüsünün normal median filtre uygulandıktan sonraki hali

5.3. Görüntüdeki Parlaklık Ayarını Düzenlemek Yapılan Uygulamalar

Görüntülerdeki parlaklık ayarlarının yapılması için seçilen örnek tarihi eser görüntüleri üzerinde uygulamalar yapılmıştır.

5.3.1. TBMM’de Açılan “Türkiye Mozaïği” adlı sergiden alınmış insan figürü

TBMM’de açılan “Türkiye Mozaïği” adlı sergiden alınan Şekil 5.16’deki insan figürlü tarihi eser görüntüsünde az miktarda da olsa gürültü mevcuttur ve parlaklık seviyesi görüntüdeki ayrıntıları algılamamızı engelleyecek şekilde yüksektir. Görüntüdeki bu bozukluklar, fotoğraf makinesinin flaşındaki sorunlar veya çekim açısının ayarlanmaması gibi nedenlerle meydana gelen bozukluklar olarak nitelendirilebilir. Bozuk görüntü Şekil 5.16’de gösterildiği gibi, gerekli işlemlerden sonra optimum hale getirilen görüntü de Şekil 5.17’de gösterildiği gibidir.



Şekil 5.16 TBMM’de açılan “Türkiye Mozaïği” adlı sergiden alınmış insan figürünün gürültülü ve çok aydınlık görüntüsü



Şekil 5.17 “Türkiye Mozaği” adlı sergiden alınmış insan figürünün onarılmış yeni görüntüsü

5.3.2. TBMM’de Açılan “Türkiye Mozaği” adlı sergiden alınmış tarihi para figürü

TBMM’de açılan “Türkiye Mozaği” adlı sergiden alınan Şekil 5.18’deki tarihi para görüntüsü detayların seçilmesini engelleyecek şekilde karanlık bir görüntüdür. Ayrıca karanlık olmasının yanında, çekim hatasından kaynaklanmış olabilecek ve görüntünün tamamına yayılmış bir bulanıklık (blur) mevcuttur. Görüntüdeki bu bozuklukları gidermek için görüntü öncelikle blind dekonvolüsyon algoritmasına tabi tutularak optimum seviyede netleştirilmiş ve daha sonra da kontrast ayarı yapılarak görüntüdeki iyi detay ve kenarların rahatça görülebilmesi sağlanmıştır. Bozuk görüntü Şekil 5.18’de gösterildiği gibi, kontrast ayarı yapılmadan önceki netleştirilmiş görüntü Şekil 5.19’de gösterildiği gibi, kontrast ayarı yapıldıktan sonraki görüntü ise Şekil 5.20’de gösterildiği gibidir. Şekil 5.18’deki seçilmeyen ayrıntıların, Şekil 5.20’de oldukça belirgin hale geldiği, görüntüyü oluşturan detayların ortaya çıktığı açıkça görülmektedir.



Şekil 5.18 TBMM’de açılan “Türkiye Mozaığı” adlı sergiden alınmış tarihi para görüntüsün orijinal hali



Şekil 5.19 TBMM’de açılan “Türkiye Mozaığı” adlı sergiden alınmış tarihi para görüntüsün blind-dekonvolüsyon algoritmasından sonraki hali



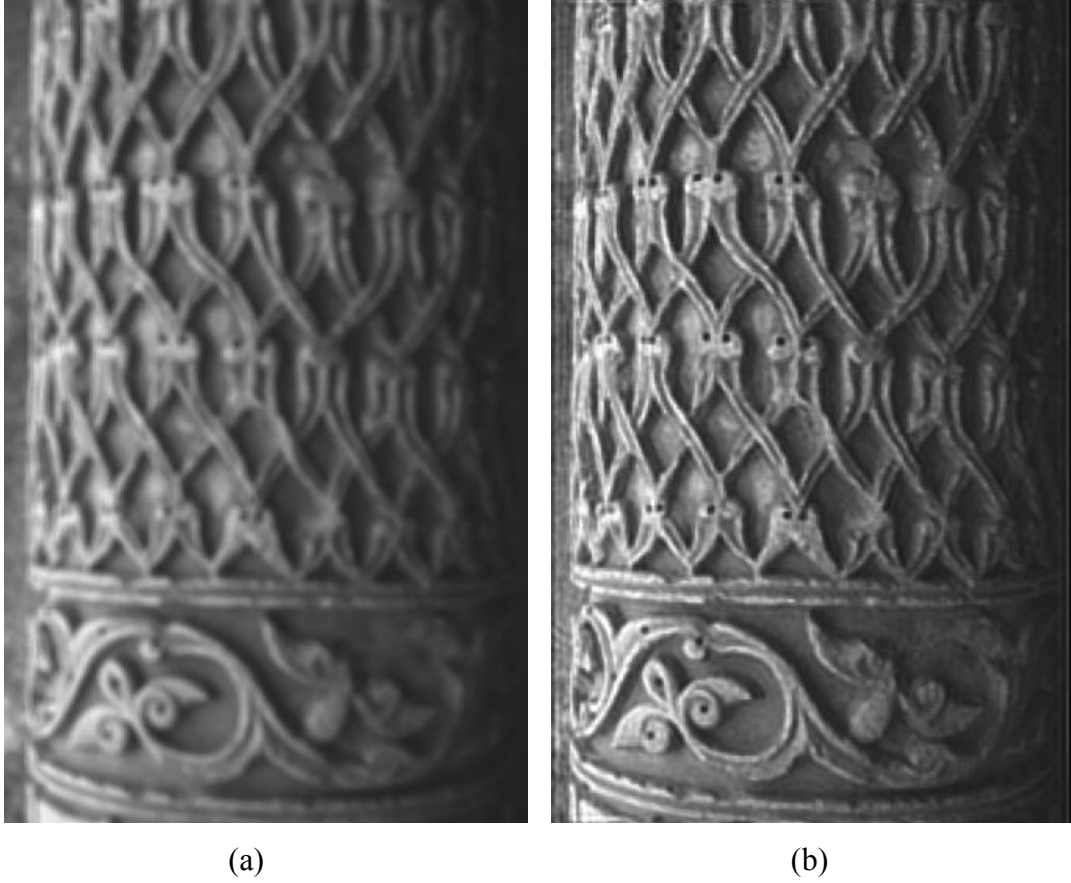
Şekil 5.20 Orijinal tarihi para görüntüsün blind-dekonvolüsyon algoritmasına tabi tutulduktan ve kontrast ayarı yapıldıktan sonraki hali

5.4. Görüntüyü Netleştirmek (Deblurring) İçin Yapılan Uygulamalar

Görüntülerin netleştirilmesi için seçilen örnek tarihi eser görüntüleri üzerinde uygulamalar yapılmıştır.

5.4.1. Sivas Divriği Ulu Cami'ye ait bir süsleme görüntüsü

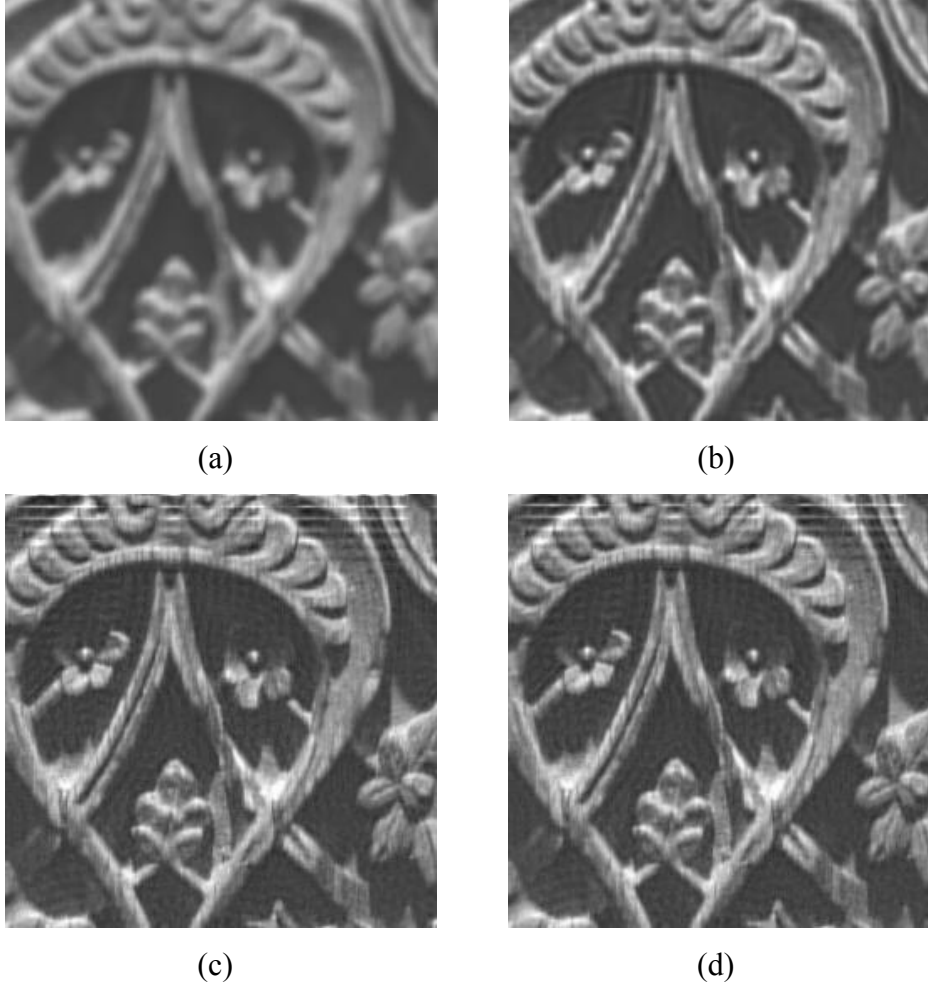
Sivas Divriği'deki Ulu Cami 1228'den kalma ve ender görülecek nitelikte bir eserdir. Ulu Cami'den alınmış Şekil 5.21 (a)'daki görüntüde fotoğraf makinesindeki bir problem veya çekim esnasında meydana gelen odak kayması nedeniyle bulanıklık meydana gelmiştir. Bulanıklık nedeniyle desenler arasındaki geçiş farklılıkları algılanamamakta ve bu da insan görüşü için olumsuzluk oluşturmaktadır. Bu bulanıklığı gidermek için görüntüye Şablon Eşleme (Template Matching) yöntemi uygulanmış ve elde edilen sonuç Şekil 5.21 (b)'deki gibi olmuştur.



řekil 5.21 (a) Sivas Divrięi Ulu Cami'ye ait bir sűsleme gűrűntűsű
 (b) Sűsleme gűrűntűsűnűnűn netleřtirilmiř hali

5.4.2. Konya Mevlana Műzesi'ne ait ahřap bir sűsleme gűrűntűsű

řekil 5.22 (a)'da Konya Mevlana Műzesi'ne ait eski bir ahřap sűsleme gűrűntűsű gűrűlmektedir. Tahta űzerine elle oyularak yapılmıř eserin sűslemelerinde ince detaylar mevcuttur. Ancak gűrűntűdeki bulanıklıktan dolayı detaylar yeterince seęilememektedir. Gűrűntűlerde netleřtirme iřleminin amacı, gűrűntű ięerisindeki detayları kuvvetlendirmek ve bulanıklıęın giderilmesini saęlamaktır. Bu eserdeki bulanıklıęı gidermek ięin gűrűntűye sırasıyla Lucy-Richardson, Wiener filtre ve Dűzenlenmiř (Regularized) filtre algoritmaları uygulanarak elde edilen sonuęlar sırasıyla řekil 5.22 (b), řekil 5.22 (c) ve řekil 5.22 (d)'de verilmiřtir.



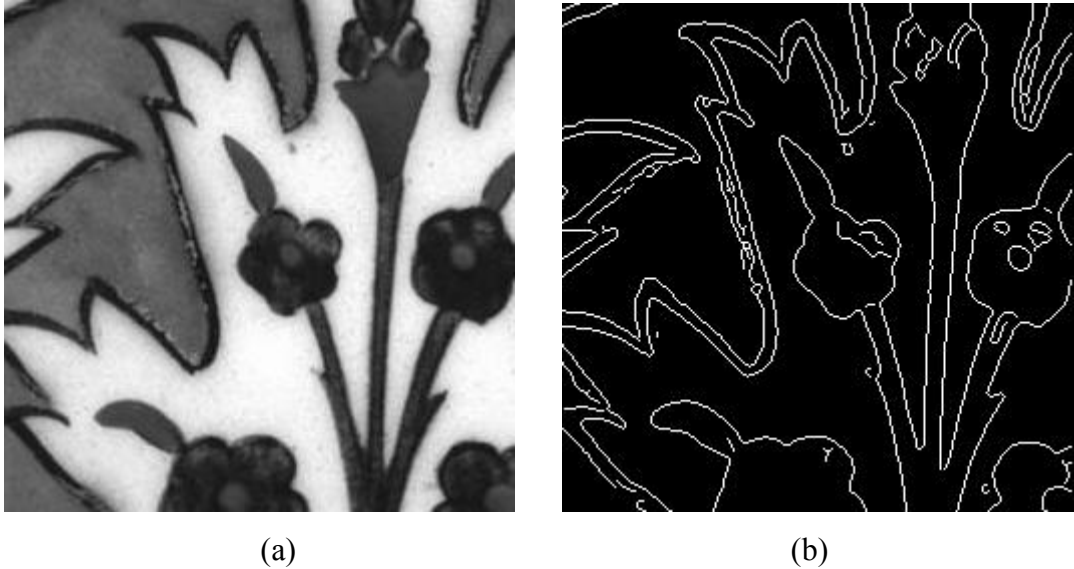
Şekil 5.22 (a) Konya Mevlana Müzesi'ne ait bir süsleme görüntüsü
(b) Süsleme görüntüsünün Lucy-Richardson algoritması ile netleştirilmiş hali
(c) Süsleme görüntüsünün Düzenlenmiş filtre algoritması ile netleştirilmiş hali
(d) Süsleme görüntüsünün Wiener filtre algoritması ile netleştirilmiş hali

5.5. Görüntüdeki Kenarları Bulmak İçin Yapılan Uygulamalar

Kenar bulma algoritmalarının seçilen örnek tarihi eser görüntüleri üzerinde uygulamaları yapılmıştır.

5.5.1. İstanbul Çinili Köşk Seramik Müzesi'ne ait seramik bir eser

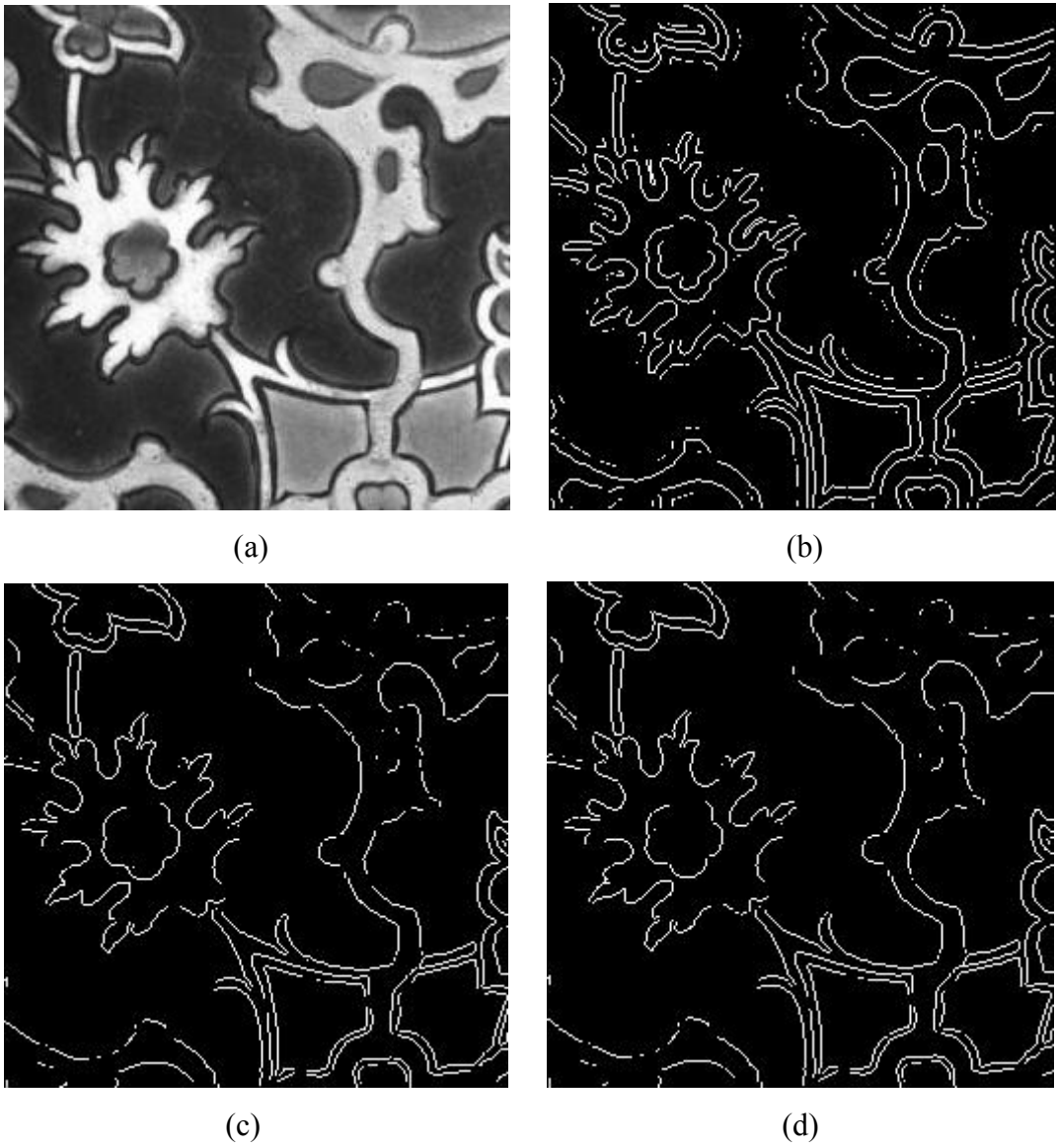
İstanbul Çinili Köşk Seramik Müzesi'ne ait Şekil 5.23 (a)'daki eser üzerindeki ayrıntılar özellikler koyu renkli bölgede seçilebilecek gibi değildir. Eser görüntüsünü uygun bir kenar bulma (edge finding) algoritmasından geçirerek karanlık bölgedeki ayrıntıları daha görülebilir hale getirmek mümkündür. Bu tür kenar bulma işlemleri, seramik üzerindeki desenlerin modelinin tekrar çıkarılmasının istendiği durumlarda veya herhangi kırık bir parçasının yeniden boyanarak onarılmasının istendiği durumlarda kullanılabilir. Ayrıca 4. Bölüm'de de belirtildiği gibi kenar bulma işlemi; medikal görüntüleme, güçlü silahların otomatik hedef tayinine kadar pek çok alanda kullanılan bir görüntü işleme tekniğidir. Şekil 5.23 (a)'da verilen seramik eser görüntüsündeki kenarlar Canny kenar bulma algoritması kullanılarak bulunmuş elde edilen yeni görüntü Şekil 5.23 (b)'deki gibi olmuştur.



Şekil 5.23 (a) İstanbul Çinili Köşk Seramik Müzesi'ne ait seramik eser görüntüsü
 (b) Seramik eser görüntüsünün Canny kenar bulma algoritması uygulandıktan sonraki hali

5.5.2. İstanbul Çinili Köşk Seramik Müzesi'ne ait bir çini görüntüsü

İstanbul Çinili Köşk Seramik Müzesi'ne ait Şekil 5.24 (a)'daki çini görüntüsüne kenar bulma algoritmalarından Log kenar bulma algoritması, Sobel kenar bulma algoritması ve Prewitt kenar bulma algoritması uygulanarak elde edilen görüntüler sırasıyla Şekil 5.24 (b), Şekil 5.24 (c) ve Şekil 5.24 (d)'deki gibi olmuştur. Bu üç algoritma arasından en yüksek başarıma sahip olan algoritmanın Log kenar bulma algoritması olduğu görülmektedir.

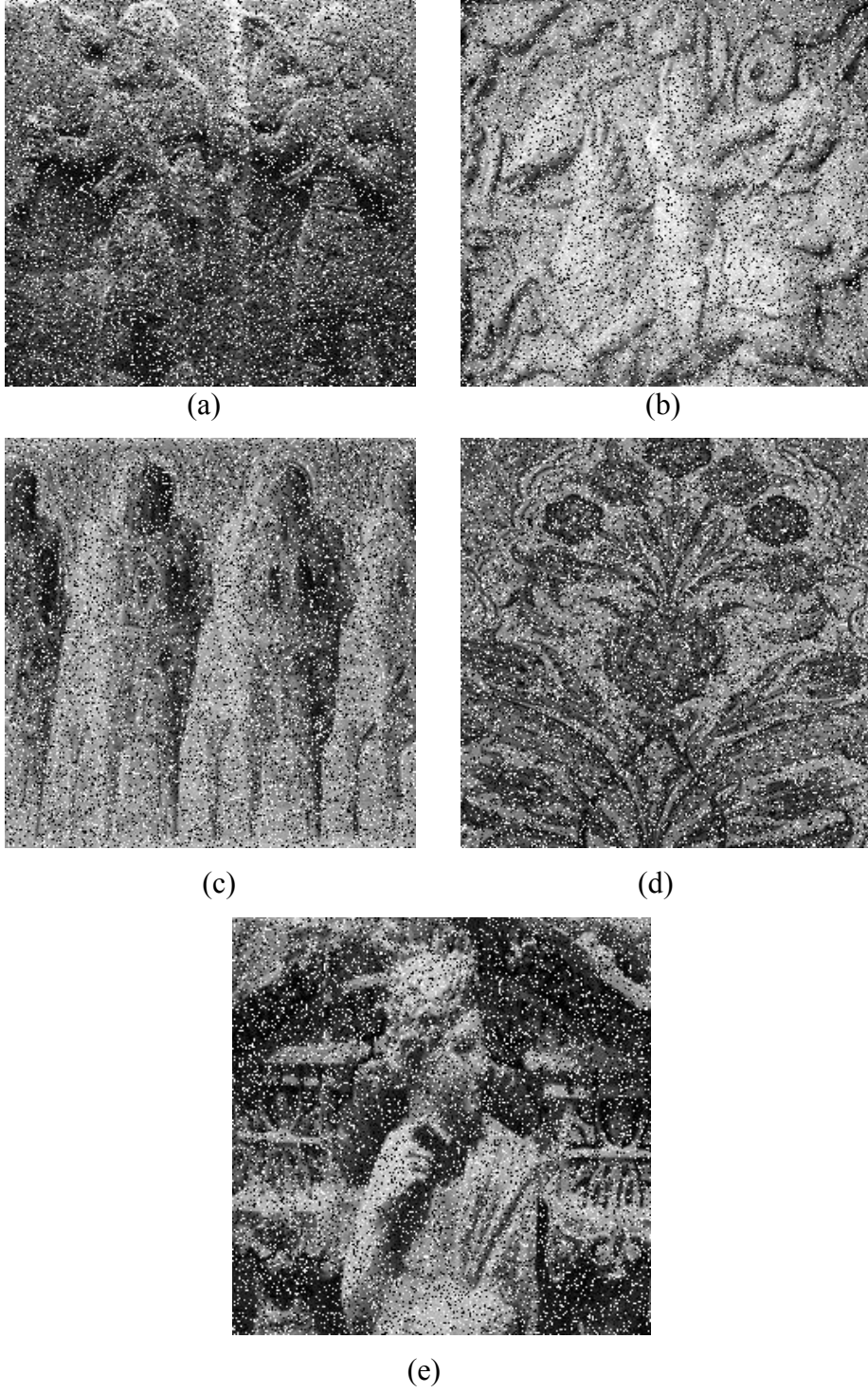


Şekil 5.24 (a) Çini görüntüsünün orijinal hali (b) Log kenar bulma algoritması
(c) Sobel kenar bulma algoritması (d) Prewitt kenar bulma algoritması

6. SONUÇLAR VE ÖNERİLER

Görüntüdeki gürültüyü azaltan görüntü işleme algoritmaları arasından en iyi sonuçları veren altı algoritma seçilerek altı farklı gürültü yoğunluğunda ve beş farklı tarihi eser görüntüsüne uygulanmıştır. Kullanılan bu algoritmalar; klasik median filtre (MED), adaptif median filtre (AMF), 5x5 boyutunda filtre kalıbına sahip olan kenarları koruyan hibrit median filtre (5x5 HM), 7x7 boyutunda filtre kalıbına sahip olan kenarları koruyan hibrit median filtre (7x7 HM), mean filtre (MEAN) ve ortalama (averaging) (AVR) filtre algoritmalarıdır. Kullanılan tarihi eser görüntüleri 8-bit gri seviyeli görüntüler olup, sonuçların sağlıklı olarak karşılaştırılabilmesi için hepsinin 256x256 boyutundaki görüntüleri oluşturularak test için bu görüntüler kullanılmıştır. Örnek teşkil etmesi açısından test için seçilen tarihi eser görüntülerinin %20 oranında gürültü eklenmiş halleri Şekil 6.1’de verilmektedir.

İstatistiksel sonuçları elde etmek için kullanılan gürültülü tarihi eser görüntüleri, her bir görüntünün dürtü gürültüsü ile değişken oranlarda bozulmasıyla elde edilmiştir. Her bir görüntüye %10, %20, %30, %50, %70 ve %90 oranlarında dürtü gürültüsü eklenerek, elde edilen görüntüler gürültü temizleme algoritmalarına giriş olarak verildikten sonra elde edilen çıkışlar karşılaştırılmıştır. Karşılaştırma ve test için kullanılan hemen hemen tüm algoritmalar %10 ve %20 gürültü oranındaki gürültü yoğunluklarında oldukça iyi sonuçlar vermekte ve bu yoğunluklarda başarımları birbirlerine oldukça yakın olmaktadır. Başarım farklılığı gürültü yoğunluğu yükseldikçe daha belirgin hale gelmektedir. %30 ve %50 gürültü yoğunluklarında algoritmaların sonuçları arasındaki farklar daha iyi ortaya çıkmakta ve karşılaştırmalar daha kolay yapılabilmektedir. %70 gürültü yoğunluğunda algoritmaların çıkışları zor seçilecek hale gelirken, %90 gürültü yoğunluğunda ise tüm algoritma çıkışları anlamsız veriler haline gelmektedir.



Şekil 6.1 %20 oranında dürtü gürültüsü eklenmiş tarihi eser görüntüleri (a) Ankara Anadolu Medeniyetleri Müzesi'ne ait Hitit Alay Figürü (b) Türk ve İslami Eserler Müzesi'ne ait alçı bir eser görüntüsü (c) İznik Yeşil Cami'ye ait süsleme görüntüsü (d) Kütahya Seramik Müzesi'ne ait bir seramik eser görüntüsü (e) Konya yöresine ait alçı bir insan figürü görüntüsü

Gürültülü tarihi eser görüntüleri görüntüdeki gürültünün etkisini azaltmak için kullanılan MED, AMF, 5x5 HM, 7x7 HM, MEAN ve AVR filtre algoritmalarına giriş olarak verildikten sonra elde edilen çıkışlar karşılaştırılmıştır. Algoritmaların başarımları, ortalama karesel hata (MSE) ve tepe işaret gürültü oranı (PSNR) kriterleri kullanılarak değerlendirilmiştir. Ancak algoritmaların elde ettiği çıkış görüntülerinin kalitesi kadar, algoritmaların filtreleme işlemini gerçekleştirme süreleri yani işlem yapma süreleri de karşılaştırma açısından önemlidir. İşlem yapma sürelerini değerlendirirken karşılaştırma açısından yeterli olduğu için her bir algoritma sadece %30 ve %70 gürültü yoğunluğundaki her bir tarihi eser görüntüsüne uygulanmıştır. Uygulamalar Intel Pentium 4, 2.40 GHz İşlemcili, 512 MB RAM'a sahip bir bilgisayarda ve MATLAB 7.0.4 programı kullanılarak yapılmıştır. Algoritmaların işlem yapma süreleri saniye cinsinden Çizelge 6.1'de verildiği gibi olmaktadır.

Çizelge 6.1 Gürültü temizleme algoritmalarının işlem yapma süreleri (sn.)

Kullanılan Algoritmalar	MED	AMF	5X5 HM	7X7 HM	MEAN	AVR
<i>Gürültü Yoğunluğu</i>	30%					
Hitit Askeri Alay Figürü	0,1784	11,2111	11,8099	13,667	15,3431	4,9368
Selçuklu Tarihi Eseri	0,1801	11,2671	12,0284	13,6088	15,3808	4,9399
İznik Yeşil Cami	0,1781	11,2959	12,0992	13,6732	15,3804	4,9354
Kütahya Seramik Eser	0,1792	11,4619	12,06	13,7458	15,4182	4,9384
Konya Tarihi Eser	0,1794	11,2622	11,8162	13,6449	15,3659	4,9343
<i>Gürültü Yoğunluğu</i>	70%					
Hitit Askeri Alay Figürü	0,1803	11,8389	12,0432	13,6567	15,3437	4,9282
Selçuklu Tarihi Eseri	0,179	11,832	12,1406	13,6231	15,3678	4,9475
İznik Yeşil Cami	0,1811	11,8125	12,0464	13,6432	15,3567	4,9524
Kütahya Seramik Eser	0,1804	11,7802	12,0433	13,6339	15,3893	4,9351
Konya Tarihi Eser	0,1787	11,9245	12,1596	13,6625	15,4165	4,9368

Algoritmaların başarımlarını değerlendirmek için kullanılan ortalama karesel hata (MSE) ve tepe işaret gürültü oranı (PSNR) kriterleri 8 bitlik görüntüler için sırasıyla eşitlik 6.1 ve 6.2'deki gibi tanımlanmaktadır:

$$MSE = \frac{1}{LC} \sum_{l=1}^L \sum_{c=1}^C (s[l,c] - y[l,c])^2 \quad (6.1)$$

$$PSNR = 10 \log_{10} \frac{\sum_{l=1}^L \sum_{c=1}^C 255^2}{\sum_{l=1}^L \sum_{c=1}^C (s[l,c] - y[l,c])^2} \quad (6.2)$$

Burada $s[l,c]$ ve $y[l,c]$ sırasıyla orijinal ve onarılmış görüntüleri temsil etmektedir. Her algoritma; %10, %20, %30, %50 ve %90 gürültü yoğunluğuna sahip her bir tarihi eser görüntüsü için tekrarlanmış ve elde edilen MSE (mean square error) değerleri Çizelge 6.2'de gösterildiği gibi kaydedilmiştir. Çizelge 6.2'de de görüldüğü gibi her bir tarihi eser görüntüsü için ayrı MSE değerleri elde edilmiştir. Ancak bu verileri grafik ortamına aktarmak ve algoritmaların farklı gürültü yoğunluklarında nasıl sonuç verdiğini tespit etmek amacıyla, her bir gürültü yoğunluğu için altı görüntünün MSE değerlerinin ortalaması alınmış ve ortalama MSE değerinin gürültü yoğunluğuna bağlı olarak değişimini gösteren grafik Şekil 6.2'deki gibi elde edilmiştir.

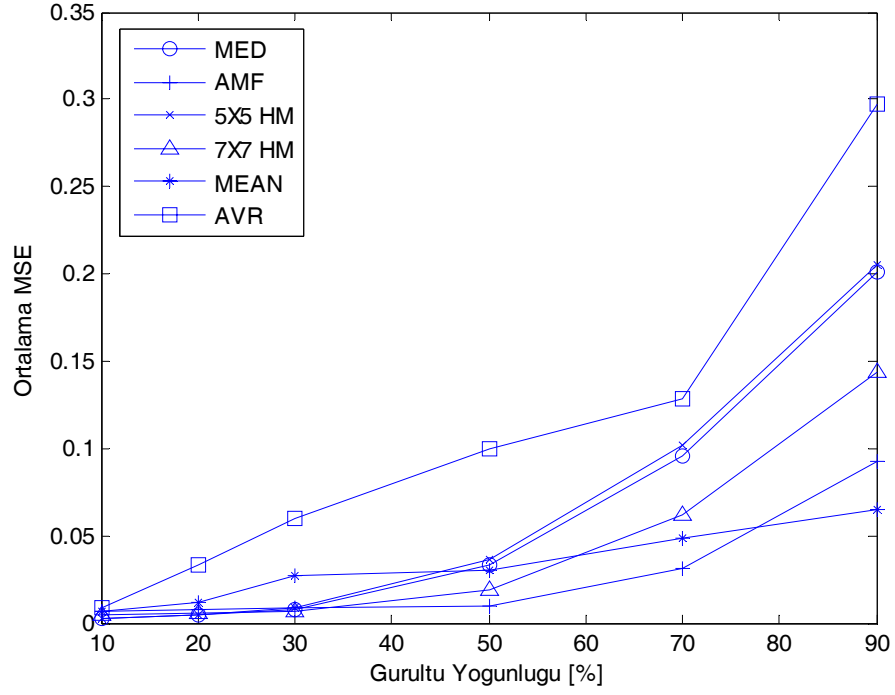
Aynı şekilde yine bütün gürültü yoğunlukları için, altı algoritmanın her bir tarihi eser görüntüsüne uygulanması sonucu elde edilen PSNR (peak signal-to-noise ratio) değerleri Çizelge 6.3'de gösterildiği gibi olmuştur. Her bir gürültü yoğunluğu için elde edilen tarihi eser görüntülerinin PSNR değerlerinin ortalaması alınarak, ortalama PSNR değerinin gürültü yoğunluğuna bağlı olarak değişimini gösteren grafik Şekil 6.3'deki gibi elde edilmiştir. Gürültünün rastgele bir süreç olması ve bilgisayarın performansındaki değişimler göz önünde bulundurularak aynı deneyin her gerçekleşmesi, şartlar aynı olsa bile farklı sonuçlar verebilmektedir. Bu nedenle her bir algoritma her bir görüntü ve gürültü yoğunluğu için 5'er kez tekrarlanmıştır.

Çizelge 6.2 Gürültü temizleme algoritmalarının başarımlarının MSE (ortalama karesel hata) kriterine göre karşılaştırılması

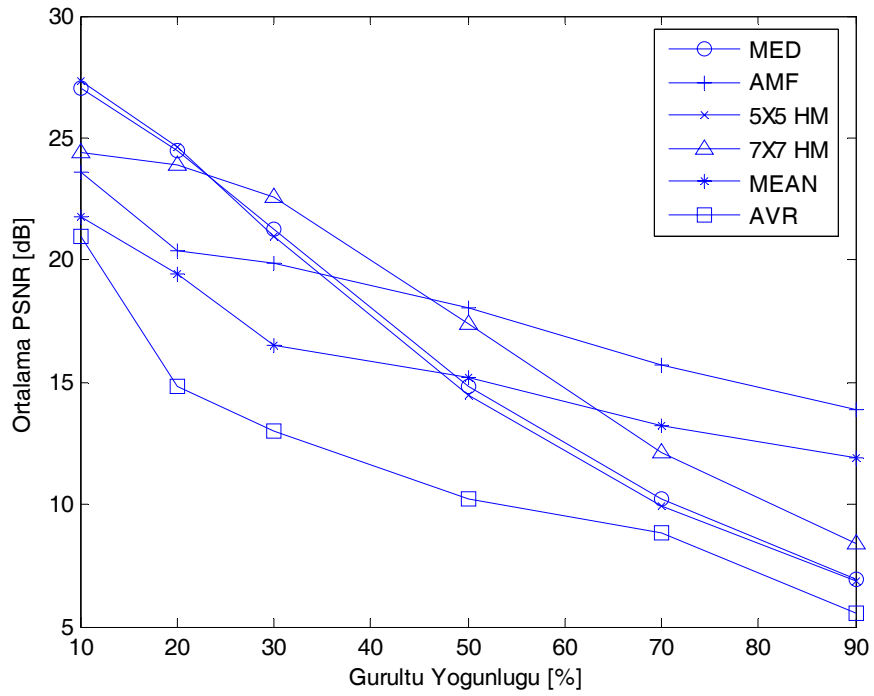
Kullanılan Algoritmalar	MED	AMF	5X5 HM	7X7 HM	MEAN	AVR
Gürültü Yoğunluğu	10%					
Hitit Askeri Alay Figürü	0,0081	0,0168	0,0074	0,0125	0,0127	0,0171
Selçuklu Tarihi Eseri	0,0006	0,0023	0,0005	0,0011	0,0049	0,0041
İznik Yeşil Cami	0,0013	0,0088	0,0013	0,0026	0,0051	0,0065
Kütahya Seramik Eser	0,0022	0,001	0,0023	0,0054	0,0066	0,0108
Konya Tarihi Eser	0,0019	0,0046	0,0017	0,0031	0,006	0,0065
Gürültü Yoğunluğu	20%					
Hitit Askeri Alay Figürü	0,0105	0,0129	0,01	0,0136	0,0177	0,0411
Selçuklu Tarihi Eseri	0,0019	0,0023	0,0017	0,0014	0,0097	0,0302
İznik Yeşil Cami	0,0026	0,008	0,0024	0,0029	0,0092	0,0308
Kütahya Seramik Eser	0,0038	0,0066	0,0043	0,0061	0,011	0,0344
Konya Tarihi Eser	0,0028	0,0041	0,0029	0,0034	0,0107	0,0286
Gürültü Yoğunluğu	30%					
Hitit Askeri Alay Figürü	0,0151	0,0175	0,0156	0,0157	0,0228	0,0485
Selçuklu Tarihi Eseri	0,005	0,0007	0,0048	0,0022	0,0157	0,0311
İznik Yeşil Cami	0,0057	0,0027	0,0061	0,004	0,0138	0,0376
Kütahya Seramik Eser	0,0078	0,0061	0,0092	0,008	0,0166	0,0442
Konya Tarihi Eser	0,0069	0,0056	0,0075	0,0049	0,0677	0,0381
Gürültü Yoğunluğu	50%					
Hitit Askeri Alay Figürü	0,0417	0,0166	0,0443	0,0297	0,0373	0,1056
Selçuklu Tarihi Eseri	0,0301	0,0033	0,0322	0,0136	0,0286	0,0596
İznik Yeşil Cami	0,0296	0,0024	0,0323	0,0151	0,0254	0,1006
Kütahya Seramik Eser	0,0317	0,0051	0,0357	0,0199	0,0285	0,1143
Konya Tarihi Eser	0,0322	0,0045	0,0363	0,0173	0,0315	0,1062
Gürültü Yoğunluğu	70%					
Hitit Askeri Alay Figürü	0,1043	0,0253	0,1098	0,0721	0,0512	0,1051
Selçuklu Tarihi Eseri	0,091	0,0032	0,0955	0,0538	0,0442	0,0598
İznik Yeşil Cami	0,0895	0,0054	0,0949	0,0557	0,038	0,1007
Kütahya Seramik Eser	0,095	0,0117	0,1025	0,063	0,0438	0,1138
Konya Tarihi Eser	0,0975	0,0089	0,1054	0,0629	0,0635	0,105
Gürültü Yoğunluğu	90%					
Hitit Askeri Alay Figürü	0,2067	0,0427	0,2106	0,1515	0,0702	0,3052
Selçuklu Tarihi Eseri	0,2058	0,0112	0,2081	0,145	0,0662	0,1268
İznik Yeşil Cami	0,1904	0,0131	0,1942	0,1325	0,0534	0,3204
Kütahya Seramik Eser	0,1975	0,0263	0,203	0,1433	0,0614	0,3792
Konya Tarihi Eser	0,2034	0,022	0,21	0,1458	0,0714	0,3726

Çizelge 6.3 Gürültü temizleme algoritmalarının başarımlarının **PSNR (dB)** kriterine göre karşılaştırılması

Kullanılan Algoritmalar	MED	AMF	5X5 HM	7X7 HM	MEAN	AVR
Gürültü Yoğunluğu	10%					
Hitit Askeri Alay Figürü	20,92	17,74	21,31	19,04	18,96	17,67
Selçuklu Tarihi Eseri	31,78	26,41	32,56	29,4	23,07	23,87
İznik Yeşil Cami	28,7	20,54	28,79	25,88	22,91	21,9
Kütahya Seramik Eser	26,5	29,89	26,32	22,66	21,82	19,68
Konya Tarihi Eser	27,25	23,28	27,63	25,11	22,2	21,89
Gürültü Yoğunluğu	20%					
Hitit Askeri Alay Figürü	19,78	18,91	20,02	18,67	17,52	13,86
Selçuklu Tarihi Eseri	27,17	26,3	27,74	28,54	20,15	15,19
İznik Yeşil Cami	25,8	20,98	26,11	25,41	20,38	15,11
Kütahya Seramik Eser	24,17	21,83	23,7	22,14	19,59	14,64
Konya Tarihi Eser	25,51	23,84	25,41	24,63	19,69	15,43
Gürültü Yoğunluğu	30%					
Hitit Askeri Alay Figürü	18,2	17,57	18,08	18,04	16,43	13,14
Selçuklu Tarihi Eseri	23,01	31,54	23,17	26,61	18,05	15,08
İznik Yeşil Cami	22,45	25,67	22,12	24,01	18,61	14,25
Kütahya Seramik Eser	21,06	22,17	20,35	20,97	17,79	13,54
Konya Tarihi Eser	21,59	22,54	21,26	23,09	11,69	14,19
Gürültü Yoğunluğu	50%					
Hitit Askeri Alay Figürü	13,8	17,79	13,54	15,28	14,28	9,76
Selçuklu Tarihi Eseri	15,21	24,8	14,92	18,65	15,44	12,25
İznik Yeşil Cami	15,28	26,2	14,9	18,21	15,95	9,97
Kütahya Seramik Eser	14,99	22,9	14,48	17,02	15,46	9,42
Konya Tarihi Eser	14,92	23,48	14,4	17,63	15,01	9,74
Gürültü Yoğunluğu	70%					
Hitit Askeri Alay Figürü	9,82	15,97	9,59	11,42	12,91	9,78
Selçuklu Tarihi Eseri	10,41	25	10,2	12,69	13,55	12,23
İznik Yeşil Cami	10,48	22,64	10,23	12,54	14,2	9,97
Kütahya Seramik Eser	10,22	19,33	9,89	12,01	13,59	9,44
Konya Tarihi Eser	10,11	20,52	9,77	12,01	11,97	9,79
Gürültü Yoğunluğu	90%					
Hitit Askeri Alay Figürü	6,83	13,69	6,77	8,2	11,54	5,15
Selçuklu Tarihi Eseri	6,87	19,51	6,82	8,39	11,79	8,97
İznik Yeşil Cami	7,2	18,83	7,12	8,78	12,72	4,94
Kütahya Seramik Eser	7,04	15,8	6,93	8,44	12,12	4,21
Konya Tarihi Eser	6,92	16,57	6,78	8,36	11,46	4,29



Şekil 6.2 Gürültü yoğunluğuna bağlı olarak ortalama MSE değerindeki değişim



Şekil 6.3 Gürültü yoğunluğuna bağlı olarak ortalama PSNR değerindeki değişim

Şekil 6.2 ve Şekil 6.3'de görüldüğü gibi, klasik median filtre (MED), 5x5 boyutunda filtre kalıbına sahip olan hibrit median filtrenin (5x5 HM) başarımları birbirine oldukça yakındır. 7x7 boyutunda filtre kalıbına sahip olan hibrit median filtre (7X7 HM), %10, %20 ve %30 gürültü yoğunluklarında daha iyi sonuç verirken %50'den sonraki gürültü yoğunluklarında daha kötü sonuçlar vermiştir. Maksimum gürültü seviyesinde yani %90 gürültü seviyesinde en iyi sonucu veren algoritma adaptif median filtre (AMF) ve en iyi sonucu veren ikinci algoritma ise mean filtre algoritması (MEAN) olmuştur. Gerek MSE sonuçlarına gerekse PSNR sonuçlarına bakılacak olursa en kötü başarımları averaging (AVR) algoritmasının gerçekleştirdiği görülmektedir. En düşük gürültü seviyesinde yani %10 gürültü seviyesinde en iyi sonucu veren algoritmalar MED filtre ve 5x5 HM olmaktadır. Burada önemli olan en yüksek gürültü oranlarında en iyi sonucu veren yani en bozuk görüntülerde en iyi onarımı sağlayan algoritmayı bulmak olduğundan, kullanılan algoritmalar arasında en iyi sonucu veren algoritmanın adaptif median filtre (AMF) algoritması, en iyi sonucu veren ikinci algoritmanın da 7x7'lik HM filtre algoritması ile MEAN filtre algoritması olduğunu söylemek mümkündür.

Bu tezde kullanılan algoritmalar işlem süresine göre karşılaştırılacak olursa, en kısa işlem süresine sahip algoritmanın MED filtre ve ikinci en kısa işlem süresine sahip algoritmanın da AVR filtre olduğu görülmektedir. İşlem süresi en uzun olan algoritma ise MEAN filtre ve işlem süresi en uzun olan ikinci algoritma da 7x7 HM filtredir. İşlem süresi algoritmanın kullanılabilirliği ve zaman kazanımı açısından oldukça önemlidir.

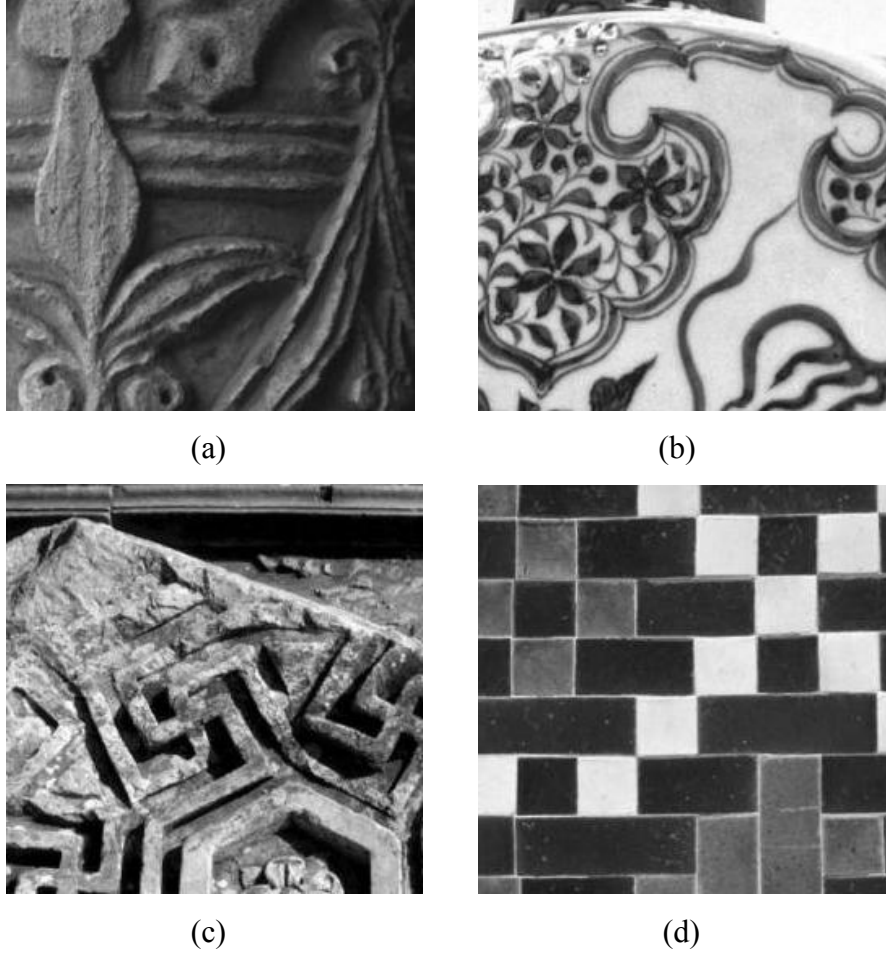
Gürültü temizleme algoritmalarının oluşturulması aşamasında dikkat edilmesi gereken en önemli nokta, görüntüdeki gürültüleri temizlerken aynı zamanda görüntüdeki kenarların ve iyi detayların da kaybolmasını engellemektir. Çünkü görüntüdeki iyi detaylar, görüntüyü ifade eden ve insan görsü için gerekli olan en önemli parçalardır. Görüntü işlemede eskiden kullanılan teknikler görüntüdeki yüksek uzay domeni frekanslarını kaldırarak fakat aynı zamanda iyi detayları da belli ölçülerde yok ederek gürültüyü azaltan alçak geçiren lineer filtrelerden oluşmaktaydı. Ancak günümüzde elde edilen non-lineer filtre yapıları ve modern

görüntü işleme metotları ile bu tür sorunların üstesinden gelmek daha kolay olmaktadır.

Bu tezde gerçekleştirilen klasik median filtre ve ortalama filtre yapılarında görüntüdeki gürültüler kaybolurken aynı zamanda iyi detaylar da kaybolmaktadır. Bu sorunları ortadan kaldırmak amacıyla kenarları ve iyi detayları koruyan bir çok hibrit median filtre uygulaması yapılmış ve en iyi sonuç veren algoritmalar çeşitli tarihi eser görüntüleri üzerinde denenmiştir. Elde edilen sonuçlara göre kenarları en iyi koruyan algoritmalar adaptif median filtre (AMF) ve kenarları koruyan hibrit median filtre (KKHMF) (özellikle 7x7 filtre kalıbına sahip HM filtre) olarak belirlenmiştir.

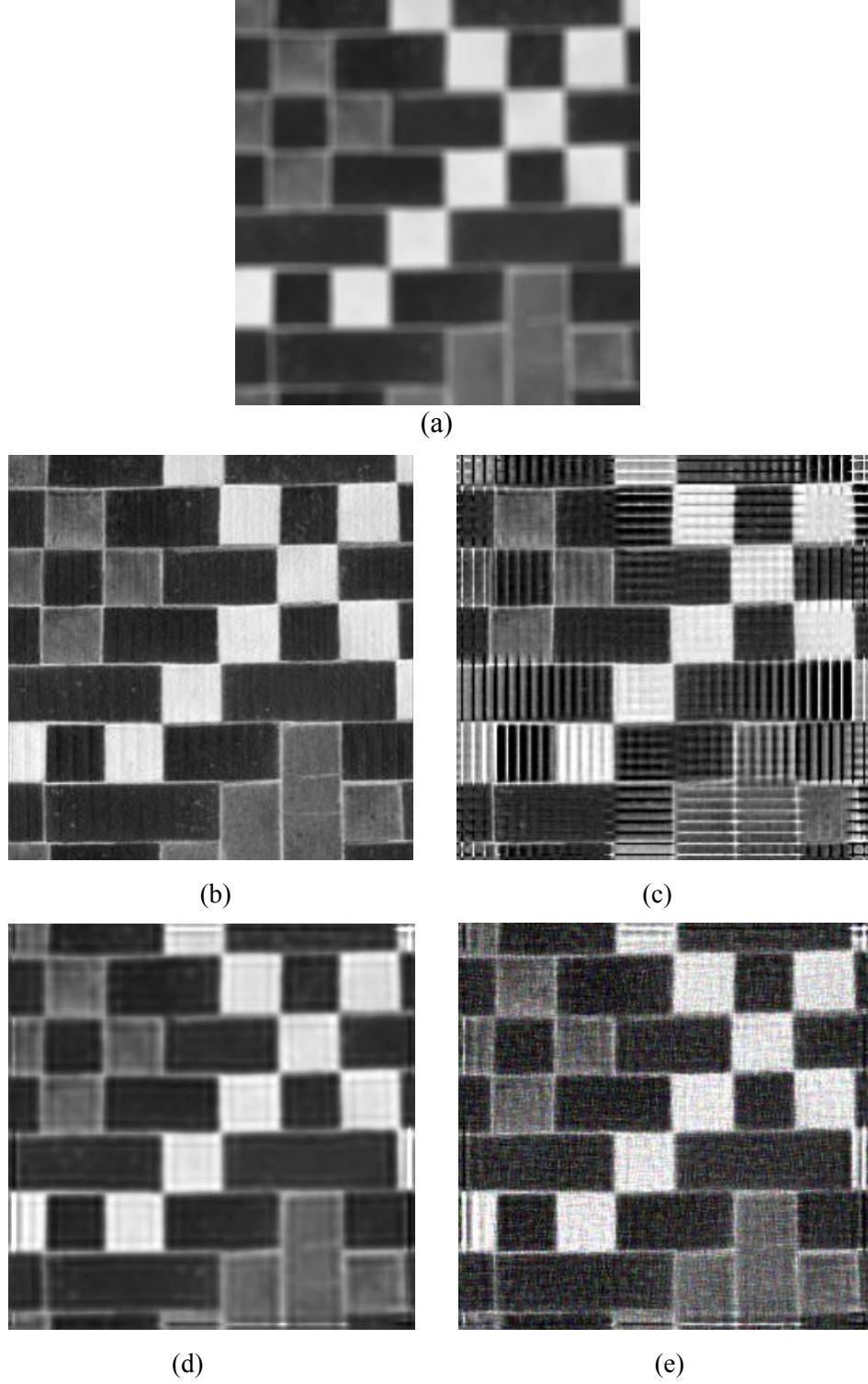
Ancak görülmektedir ki, iyi sonuç veren algoritmaların işlem süreleri de oldukça uzamaktadır. Bundan sonra yapılacak çalışmalarla daha iyi sonuçlar elde etmek yani daha kısa işlem süresine sahip, kenarları daha iyi koruyan ve en yüksek gürültülerde daha kaliteli sonuçlar veren, farklı görüntü tiplerine göre farklı özellikler taşıyan çalışmaların yapılması gerekliliği araştırmacılara öneri olarak sunulmakta ve yapılan bu çalışmaların ve testlerin bundan sonraki çalışmalara katkıda bulunacağı düşünülmektedir.

Bu tezde ayrıca görüntüyü netleştirme ve görüntüdeki kenarları bulma uygulamaları da yapılmıştır. Görüntüyü netleştirmek için kullanılan algoritmalar Wiener filtre algoritması, Düzenlenmiş (Regularized) filtre algoritması, Lucy-Richardson Algoritması ve Blind-Dekonvolüsyon algoritmasıdır. Görüntü netleştirme algoritmalarını karşılaştırmak için dört tane tarihi eser görüntüsüne yapay olarak bulanıklık eklenmekte ve her bir algoritmanın her görüntüye uygulanması suretiyle elde edilen MSE değerleri Çizelge 6.4'deki gibi olmaktadır. Görüntü netleştirme algoritmalarında kullanılan tarihi eser görüntüleri Şekil 6.4'de verilmektedir.



Şekil 6.4 (a) Sivas Divriği Ulu Cami'ye ait bir süsleme görüntüsü (b) İstanbul Topkapı Sarayı'na ait bir porselen tabak görüntüsü (c) Efes Antik Kent'e ait bir süsleme görüntüsü (d) İstanbul Çinili Köşk'e ait bir seramik süsleme görüntüsü

Karşılaştırılan algoritmaların başarımlarını görsel olarak da inceleyebilmek amacıyla İstanbul Çinili Köşk'e ait seramik süsleme görüntüsüne yapay olarak blur (bulanıklık) eklenmekte ve her bir algoritmanın bu giriş görüntüsü için ürettikleri çıkış görüntüleri Şekil 6.5'de gösterildiği gibi olmaktadır. Şekil 6.5'de de görüldüğü gibi en iyi sonucu veren algoritma Wiener filtre algoritması, en kötü sonucu veren algoritma da Düzenlenmiş (Regularized) filtre algoritmasıdır.



Şekil 6.5 (a) Bulanık görüntü (b) Wiener Algoritması uygulandıktan sonraki görüntü
(c) Düzenlenmiş (Regularized) filtre uygulandıktan sonraki görüntü
(d) Lucy- Richardson Algoritması uygulandıktan sonraki görüntü
(e) Blind-Dekonvolüsyon Algoritması uygulandıktan sonraki görüntü

Çizelge 6.4 Görüntü netleştirme (deblurring) algoritmalarının başarımlarının **MSE** (ortalama karesel hata) kriterine göre karşılaştırılması

Kullanılan Algoritmalar	Wiener	Reg.(Düz.)	Lucy-Rich.	Blind-Dekonv.
Divriği Ulu Cami	0,0007	0,0207	0,002	0,0031
İstanbul Topkapı Sarayı	0,0005	0,0387	0,0044	0,0054
Efes Antik Kent	0,0002	0,0055	0,0022	0,0022
İstanbul Çinili Köşk	0,0001	0,006	0,008	0,0022

Görüntüdeki kenarları bulmak (edge detection) için kullanılan algoritmalar 5.bölümde İstanbul Çinili Köşk Seramik Müzesi'ne ait bir çini görüntüsünde uygulandığı ve elde edilen görüntüler karşılaştırmalı olarak verildiğinden, tekrar bu bölümde uygulaması yapılmamaktadır. Kenar bulma algoritmaları arasında başarımları en yüksek olan algoritmalar Canny kenar bulma algoritması ve Log kenar bulma algoritması olmuştur.

Gerçekleştirilen bütün algoritmalar karşılaştırılacak olursa: görüntüdeki gürültünün etkisini azaltan algoritmalarından başarımları en yüksek olan algoritmalar Adaptif Median Filtre (AMF) algoritması ve Hibrit Median Filtre (HM) algoritması, görüntüyü netleştiren algoritmalarından başarımları en yüksek olan algoritma Wiener Filtre algoritması, kenar bulma algoritmalarından başarımları en yüksek olan algoritmalar ise Canny ve Log kenar bulma algoritmalarıdır.

7. TARTIŞMA

Tarihi eser görüntülerinin onarılması, kalitesinin artırılması tarihimizin yaşatılması ve gelecek nesillere aktarabilmesi açısından önem taşımaktadır. Eskiden günümüze kalan fotoğraflarda genellikle zamanın verdiği etkiyle sararmalar, bulanıklıklar, gürültüler ve gürültü nedeniyle meydana gelmiş detay kayıpları mevcuttur. Fotoğrafları çekilen bu tarihi eserlerin orijinalleri de aynı şekilde zamanın verdiği etkiyle ayrıca güneş, nem gibi dış etkenler nedeniyle, kırılmış, dökülmüş, üzerindeki önemli ayrıntıları kaybetmiş hatta yok olmaya yüz tutmuş olabilmektedir. Eserin geçmişten günümüze kazandırılması için yani restorasyon çalışmalarına katkıda bulunması için kullanılacak tek seçenek bazen bu görüntüler olabilmektedir. Bu durumda fotoğrafın optimum düzeyde dijital olarak onarılması, gürültülerin optimum seviyede azaltılması, karanlık resimlerin çeşitli kontrast algoritmalarıyla aydınlatılması, sararmış resimlerin aydınlık seviyelerinin ayarlanması, odak kayması nedeniyle bulanık çıkmış resimlerin netleştirilmesi yani görüntünün detaylarının en iyi görülebilecek seviyeye getirilmesi, restorasyon işleminin başarısı için çok önem taşıyacaktır.

Ayrıca bir tarihi bina üzerinde restorasyon çalışmaları yapılacağında binanın eskiden kalma fotoğrafları değil de yeni çekilmiş fotoğrafları kullanılarak da dijital görüntü işleme teknolojisinin nimetlerinden yararlanmak mümkün olmaktadır. Örneğin bir cami veya medrese duvarındaki yazının herhangi bir yerinde bir kararma, bir anıtın veya heykelin üzerinde paslanmalar, lekelenmeler veya tozlanma nedeniyle küçük benekler olabilir. Bu anıtın fotoğrafını çekip dijital görüntü işlemeye sunarak gürültülerin, bulanıklıkların, sorunlu bölgelerin düzeltilmesini, önemli detayları içeren kenarların belirginleştirilmesi sağlamak suretiyle görüntü restore edilebilir. Daha sonra görüntünün restore edilmiş son hali incelenerek gerçek eserdeki restorasyon çalışmalarına yön vermek mümkündür. Görüntüyü sanal olarak onarmak, değişiklikleri denemek bu şekilde mümkün olacaktır. Değişiklikler sanal olduğundan geri alması oldukça kolaydır. Bu değişikliklerin orijinal eser üzerinde denenmesi genellikle mümkün olmayabilir, ya da geri dönüşü olmayan sonuçlar elde edilebilir. Dijital görüntü işleme teknikleri sayesinde tarihi eserin restorasyon

aşamaları hakkında bir fikir sahibi olunabilmekte, bu dijital çalışmalar sayesinde restorasyon çalışmalarına yön vermek mümkün olabilmektedir.

Bunların dışında tarihi eser görüntülerinin dijital görüntü işleme teknikleri ile onarımının yararlı olacağı alanlardan biri de tarihi eser tanıtım dergileri, kitapçıklar, ilanlar, afişler ve broşürler için yapılan çalışmalardır. Her hangi bir tarihi eser sergisi, bir sergi için basılacak olan kitapçık, tanıtımı yapılacak olan tarihi bir bölgenin afişi, ilanı veya broşüründe kullanılacak olan resimlerin hem orijinal resimleri ifade etmesi hem de estetik normlarda olması gerekir. Bu açıdan kitapçığa, broşüre veya ilana basılacak olan tarihi eser görüntülerinin belli görüntü işleme süreçlerinden geçirilmiş olması ve optimum düzeyde iyi görünür olması gerekmektedir. Böyle bir durumda da görüntü işlemenin eski görüntüler üzerindeki onarıcı etkisi işimize yaramaktadır.

Bütün bunlar göz önünde bulundurulduğunda yapılan bu tez çalışmasının hem tarihi eserlerimizi korumamız açısından hem de görüntü işleme tekniklerinin restorasyon çalışmalarındaki uygulama alanlarını genişletmek açısından ne kadar faydalı olduğu görülmektedir. Tezde kullanılan algoritmalar genel olarak gürültü temizleme algoritmaları, görüntü netleştirme algoritmaları ve kenar bulma algoritmaları olarak sınıflandırılmıştır. Bu algoritmalar arasında en iyi sonuç veren algoritmalar seçilerek, çeşitli tarihi eser görüntülerine uygulanmış ve elde edilen sonuçlar 6.Bölüm’de ayrıntılı olarak tartışılmıştır. Kullanılan tekniklerin birbirlerine göre avantajları ve dezavantajları olmakla birlikte genel olarak elde edilen sonuçlar optimum düzeydedir.

Yapılan bu tez çalışmasında kullanılan algoritmalar MATLAB programıyla gerçekleştirilmiştir. İşlemlerin gerçekleştiği bilgisayarın donanımsal özelliği algoritmaların performansını etkilemekle birlikte, PSNR ve MSE çizelgelerine bakıldığında özellikle adaptif median filtre (AMF) ve hibrit median filtre (HM) yapılarında oldukça iyi sonuçların elde edildiği görülmektedir. Görüntü netleştirme algoritmaları arasında en iyi sonucu Wiener filtrenin verdiği ve kenar bulma algoritmaları arasında da en iyi sonucu Canny kenar bulma algoritması ile Log kenar bulma algoritmasının verdiği görülmektedir.

8. KAYNAKLAR

- Ataman E., Aatre V.K., Wong K.M., A Fast Method for Real Time Median Filtering, IEEE Trans. Acoust., Speech and Signal Process., Vol. 28, pp. 415-420, 1980.
- Bednar J.B., Watt T.L., Alpha Trimmed Means and Their Relationship to Median Filters, IEEE Trans. Acoust., Speech and Signal Process., Vol. 32, pp. 145-153, 1984.
- Ben-Arie J., Rao K. R, A Novel Approach to Template Matching by Nonorthogonal Image Expansion, IEEE Trans. on Circ. and Systems for Video Technology Vol. 3, No. 1, pp.71-84, 1993.
- Chan R.H., Ho C.W., Nikolova M., Salt-and-Pepper Noise Removal by Median-Type Noise Detectors and Detail-Preserving Regularization, IEEE Trans.on Img Processing, Vol. 14, No.10, October 2005.
- Chen T., Ma K.K., Chen L.H., Tri-State Median Filter for Image Denoising, IEEE Transactions on Image Processing, Vol. 8, No. 12, 1999.
- Choi M., Galatsanos N. P., Schonfeld D., On the Relation of Image Restoration and Template Matching: Application to Block-Matching Motion Estimation, IEEE Trans. Inform. Theory, Vol.3, pp. 95-99, 1996.
- Donoho D., Denoising by Soft-Thresholding , IEEE Trans. Inform. Theory, Vol. 41, pp. 613-627, 1995.
- Duda R.O., Hart P.E, Pattern Classification and Scene Analysis, New York, New York:John Wiley & Sons, 1973.
- Gonzales R.C., Woods R.E., Digital Image Processing, Addison-Wesley PublishingCompany, Inc., ABD, 1993.
- Gonzales R.C., Woods R.E., Eddins S.L., Digital Image Processing using MATLAB, Pearson Prentice Hall, 2004.
- Guyton A.C. , Tibbi Fizyoloji, 7.b., Merk Publishing, s.1012, 1986.
- Hillery A.D., Chin R.T., Iterative Wiener Filters for Image Restoration, IEEE, 1990.
- Huang T. S., Yang G. J., Tang G. Y., A Fast Two-Dimensional Median Filtering Algorithm, IEEE Trans. Acoust., Speech, Signal Process., Vol. 27, pp.13–18, 1979.
- ISO/IEC DIS 11172, Coding of Moving Pictures and Associated Audio for Digital Storage Media up to about 1.5 Mbits/s, 1992.

- Jiang M., Wang G., Development of Blind Image Deconvolution and Its Applications, *Journal of X-Ray Science and Technology*, Vol.11, pp.13–19, 2003.
- Khashman A., Automatic Edge Detection of DNA Bands in Autoradiograph Images, *IEEE*, 1999.
- Khriji L., Gabbouj M., Vector Median-Rational Hybrid Filters for Multichannel Image Processing, *IEEE Signal Process.*, Vol. 6, pp. 186–190, 1999.
- Kızılkaya A., Taranmış Bozuk Görüntülerin Kalitelerinin Yazılım Teknikleri ile Artırılması, Denizli, 1997.
- Kundur D., Hatzinakos D., Blind image deconvolution, *IEEE Signal Processing Magazine*, Vol. 13, pp. 43–64, 1996.
- Ko S. J., Lee Y. H., Center Weighted Median Filters and Their Applications to Image Enhancement, *IEEE Trans. Circuits and Systems*, Vol. 38, No. 9, pp. 984–993, 1991.
- Kuan D.T., Sawchuck A.A., Strand T.C., Chavel P., Adaptive Noise Smoothing Filter for Images with Signal-Dependent Noise, *IEEE Trans. Pattern Anal. Mach.Intell.*, Vol.PAMI-7, pp.165-177, 1985.
- Lee J.S., Digital Image Enhancement and Noise Filtering by use of Local Statistics, *IEEE Trans. Pattern Anal.Mach.Intell.*, Vol.PAMI-2, pp.165-168, 1980.
- Lim J.S., *Digital Image Processing Techniques*, edited by Michael P. Ekstrom, Academic Pres, New York, 1984.
- Lim J.S, *Two-Dimensional Signal and Image Processing*, Second Edition, Prentice Hall Signal Process. Series, PTR Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1990.
- Low A., *Introductory Computer Vision and Image Processing*, McGraw-Hill Book C, London, 1991.
- Mallows C.L., Some Theory of Non-Linear Smoothers, *The Annals of Statistics*, Vol. 8, pp. 695-715, 1980.
- Mayo Clinic Ansiklopedisi, Cilt 2, s. 436.
- Nilsson L, Little J.L., Man B., Boston: Brown and Company, s.190, 1974.
- Oppenheim A.V., Schafer R.W., Stockham T.G, Nonlinear Filtering of Multiplied and Convolved Signals, *Proc. IEEE*, Vol.56, pp. 1264–1291, 1968.

- Pitas I., Venetsanopoulos A. N., Order Statistics in Digital Image Processing, Proceedings of the IEEE, Vol. 80, No. 12, 1992.
- Rabie T., Adaptive Hybrid Mean and Median Filtering of High-ISO Long-Exposure Sensor Noise for Digital Photography, Journal of Electronic Imaging, Vol. 13, 2004.
- Schalkoff R.J., Digital Image Processing and Computer Vision, John Willey & Sons, Inc., Canada, 1989.
- Smith S.M., Brady J.M., Susan-a New Approach to Low Level Image Processing, International Journal of Computer Vision, Vol. 23(1), pp. 45-78, 1997.
- Sprawls P., Ph.D., The Physical Principles of Medical Imaging, 2nd Ed., Online Textbook, 1993.
- Stockham T.G., Cannon T.M., Ingebresten R.B., Blind Deconvolution Through Digital Signal Processing, Proc. IEEE, Vol. 63, pp. 678–692, 1975.
- Taguchi A., Murata Y., The Median and The Mean Hybrid Filters, in Proc., IEEE Conference on Circuits and Systems, pp. 93–96, 1991.
- Temel Britanicca, Cilt 7, s. 207
- Umbaugh E., Computer Vision and Image Processing, Upper Saddle River, NJ. Prentice- Hall International Inc., 1998.
- Yang R., Yin L., Gabbouj M., Astola J., Neuvo Y., Performance of Detail-Preserving Weighted Median Filters for Image Processing, 1991.
- Xiao X., Li S., Detail-Preserving Approach for Impulse Noise Removal from Images, Proceedings of the Fourth International Conference on Computer and Information Technology (CIT'04), 2004.

<http://academic.mu.edu>

<http://ari.cankaya.edu.tr>

<http://www.numis.northwestern.edu>

<http://www.imatest.com/docs/noise.html>

<http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-Referenc.html>

EKLER

- Ek 1 : Averaging (Ortalama) Filtre ile Görüntüdeki Gürültünün Azaltılması
- Ek 2 : Averaging-2 Filtresi ile Görüntüdeki Gürültünün Azaltılması
- Ek 3 : 5x5 Hibrit Median Filtre ile Görüntüdeki Gürültünün Azaltılması
- Ek 4 : 7x7 Hibrit Median Filtre ile Görüntüdeki Gürültünün Azaltılması ve Görüntü Parlaklığının Ayarlanması
- Ek 5 : Adaptif Median Filtre (AMF) ile Görüntüdeki Gürültünün Azaltılması
- Ek 6 : MEAN Filtre ile Görüntüdeki Gürültünün Azaltılması
- Ek 7 : Blind Deconvolution Algoritması ile Görüntünün Netleştirilmesi
- Ek 8 : Regularized Filtre İle Görüntünün Netleştirilmesi
- Ek 9 : Lucy-Richardson Algoritması ile Görüntünün Netleştirilmesi
- Ek 10 : Wiener Filtresi ile Görüntünün Netleştirilmesi
- Ek 11 : Canny Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması
- Ek 12 : Laplace of Gaussian (LoG) Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması
- Ek 13 : Sobel Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması
- Ek 14 : Prewitt Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması
- Ek 15 : PSNR ve MSE Değerlerini Bulmak İçin Kullanılan Algoritma

Ek 1 : Averaging (Ortalama) Filtre ile Görüntüdeki Gürültünün Azaltılması

```

clear all;
close all;
clc;
tic
SI = imread('C:/konya.jpg');
figure(1);
imshow(SI);
imagesc(SI);
title('Oriijinal Goruntu');
B =im2double(SI);
randimg= randsrc(256,256,[-0.1 0.1;0.7 0.3]);
noisy=B+ randimg;
figure(2);
imshow(randimg);
imagesc(randimg);
title('Rastgele gurultu');
figure(3);
imshow(noisy);
imagesc(noisy);
title('Gurultulu Goruntu');
A=zeros(256,256);

for i=1:256
    for j=1:246
A(i,j)=(noisy(i,j)+noisy(i,j+1)+noisy(i,j+2)+noisy(i,j+3)+noisy(i,j+4)+noisy(i,j+5)+noisy(i,j+6)+noisy(i,j+7)+noisy(i,j+8)+noisy(i,j+9))/10;
    end
end

figure(4);
AA1=A(1:256,1:246);
AA1=imresize(AA1,[256 256]);
imshow(AA1);
imagesc(AA1);
title('Birinci Filtre');

for i=1:246
    for j=1:246
A(i,j)=(noisy(i,j)+noisy(i+1,j)+noisy(i+2,j)+noisy(i+3,j)+noisy(i+4,j)+noisy(i+5,j)+noisy(i+6,j)+noisy(i+7,j)+noisy(i+8,j)+noisy(i,j+9))/10;
    end
end

AA2=A(1:246,1:246);
AA2=imresize(AA2,[256 256]);
figure(5);
imshow(AA2);

```

```

imagesc(AA2);
title('Ikinci Filtre');

for i=1:246
    for j=1:246
        A(i,j)=(noisy(i,j)+noisy(i+1,j+1)+noisy(i+2,j+2)+noisy(i+3,j+3)+noisy(i+4,j+4)+noisy(i+5,j+5)+noisy(i+6,j+6)+noisy(i+7,j+7)+noisy(i+8,j+8)+noisy(i+9,j+9))/10;
    end
end

AA3=A(1:246,1:246);
AA3=imresize(AA3,[256 256]);
figure(6);
imshow(AA3);
imagesc(AA3);
title('Ucuncu Filtre');
figure(7);

for i=1:253
    for j=1:253
        A(i,j)=(noisy(i,j)+noisy(i,j+1)+noisy(i,j+2)+noisy(i+1,j)+noisy(i+1,j+1)+noisy(i+1,j+2)+noisy(i+2,j)+noisy(i+2,j+1)+noisy(i+2,j+2))/9;
    end
end

AA4=A(1:253,1:253);
AA4=imresize(AA4,[256 256]);
imshow(AA4);
imagesc(AA4);
title('Dorduncu Filtre');

t=toc
imwrite(AA1,'C:/averaging3/avrgkonya1.jpg');
imwrite(AA2,'C:/averaging3/avrgkonya2.jpg');
imwrite(AA3,'C:/averaging3/avrgkonya3.jpg');
imwrite(AA4,'C:/averaging3/avrgkonya4.jpg');

```

Ek 2 : Averaging-2 Filtresi ile Görüntüdeki Gürültünün Azaltılması

```

clear all;
close all;
clc;
tic;
OG = imread('C:/hititalayi.jpg');
figure(1);
imshow(OG);
title('Orjinal Goruntu');

if ndims(OG) == 3
KYM = OG;
GG = rgb2gray(KYM); % Renkli görüntüyse gray görüntüye çeviriyor.
figure(2);
imshow(GG);
title('Gray Goruntu');
else
GG = OG;
end

KG = imresize(GG,[256 256],'bicubic'); % Görüntünün boyutu [256 256] olarak
değiştiriliyor.

figure(3);
imshow(KG);
title('[256 256] boyutlu kare goruntu');

KG = im2double(KG); % Görüntü normalize ediliyor. (0-1 arasında)
a = -.35; b = .35;
RG = a + (b-a) * rand(256); % [256 256] boyutlarında ve bu matris içindeki tüm
değerlerin a ve b arasında olacağı random
(rastgele) bir gürültü matrisi oluşturuluyor.

figure(4);
imshow(RG);
title('Rastgele Gurultulu Goruntu');
gurultulu = KG+RG; % Orjinal görüntüye random gürültü ekleniyor.
figure(5);
imshow(gurultulu);
title('Gurultulu goruntu');

for x=1:256; % Matrisi okumaya başlıyor.
for y=1:256;
if gurultulu(x,y)<0; %Matristeki herhangi bir değer 0'dan küçükse o değer
0 yapılıyor.

gurultulu(x,y)=0;
end
end
end
end

```

```

NE=gurultulu;
NE(258,258)=0;           % Matrise 0 değerli 2 satır ve 2 sütun ekleniyor.
NS = circshift(NE,[1 1]); % Matris satır ve sütunları sağa ve aşağıya öteleniyor.

for m=2:257;             % Döngü için matrisi okutuluyor.
for n=2:257;
DS(m,n)=(NS(m,n)+NS(m+1,n)+NS(m,n+1)+NS(m,n-1)+NS(m-1,n))/5;

% Matrisdeki 5 komşu elemanın ortalaması alınıyor ve ortalama sonucu ilk orta
% elemanın yerine yazılıyor ve böylece bütün resim taranıyor.

end
end

DI=DS;                   % Matris başka bir değişkene atanıyor..
DI(:,1) = [ ];           % İlk sütunu siliniyor.
DI(1,:) = [ ];           % İlk satır siliniyor.
figure(6);
imshow(DI);
title('Filtrelenmiş hititalayi görüntüsü 1');
DE=DI;
DE(258,258)=0;
DS = circshift(DE,[1 1]);

for i=2:257;             % Döngü için matrisi okutuluyor.
for j=2:257;
FS(i,j)=(DS(i,j)+DS(i+1,j)+DS(i,j+1)+DS(i+1,j+1)+DS(i-1,j-1)+DS(i,j-1)+DS(i-
1,j)+DS(i+1,j-1)+DS(i-1,j+1))/9;

% Filtre 9 piksel komşuluğu için gerçekleştiriliyor.

end
end

FI=FS;
FI(:,1) = [ ];
FI(1,:) = [ ];
figure(7);
imshow(FI);
title('Filtrelenmiş hititalayi görüntüsü 2');
t=toc

```

Ek 3 : 5x5 Hibrit Median Filtre ile Görüntüdeki Gürültünün Azaltılması

```

clear all;
close all;
clc;

I = imread('C:/konya.jpg');
figure (1);
imshow(I);
title ('gurultulu goruntu');

tic
dim = size(I);      % Görüntü renkli görüntüyse 2 boyutlu renksiz görüntüye
                   % çeviriliyor.
if length(dim)==3
    I = rgb2gray(I);
end

% Giriş görüntüsünün boyutu değiştiriliyor.

PI = padarray(I,[2,2],'replicate'); %Boyutu değiştirilirken kullanılan maske 2x2 liktir
                                     ve pad array (dizileri doldurma) işlemi replicate
                                     olarak yapılmaktadır.

dim = size(PI);    % Giriş görüntüsünün yeni boyutu

% Filtreleme işlemi için loopun başlatılması

for m = (1+2):(dim(1,1)-2)
    for n = (1+2):(dim(1,2)-2)

        % Filtreleme için kullanılacak maskenin katsayıları

        a = PI(m,n);
        b1 = PI(m,n+1);
        b2 = PI(m,n+2);
        c1 = PI(m-1,n+1);
        c2 = PI(m-2,n+2);
        d1 = PI(m-1,n);
        d2 = PI(m-2,n);
        e1 = PI(m-1,n-1);
        e2 = PI(m-2,n-2);
        f1 = PI(m,n-1);
        f2 = PI(m,n-2);
        g1 = PI(m+1,n-1);
        g2 = PI(m+2,n-2);
        h1 = PI(m+1,n);
        h2 = PI(m+2,n);
        i1 = PI(m+1,n+1);

```

```
i2 = PI(m+2,n+2);
para1 = [a,b1,b2,d1,d2,f1,f2,h1,h2];
para2 = [a,c1,c2,e1,e2,g1,g2,i1,i2];
para3 = [a,median(para1),median(para2)];

% Filtrelenmiş çıkış görüntüsü

FPI(m-2,n-2) = median(para3);
end
end

figure(2);
imshow(FPI);
title('5x5 Hybrid median filtreden geçirilmis goruntu');
t = toc

imwrite(FPI,'C:/konyaduz.jpg');
```

Ek 4 : 7x7 Hibrit Median Filtre ile Görüntüdeki Gürültünün Azaltılması ve Görüntü Parlaklığının Ayarlanması

*%I: Giriş Görüntüsü, Türkiye Mozaïği adlı sergiden alınan tarihi eserin
%gürültülü görüntüsü, IM: Çıkış Görüntüsü; gürültüden arındırılmış, parlaklığı
%optimumlaştırılmış görüntü*

```
function FPI = parmed1(I);
clear all;
close all;
```

```
I = imread('C:/anadoluselcuk.jpg');
figure (1);
imshow(I);
title ('gurultulu goruntu');
```

```
tic
dim = size(I); % Görüntü renkli görüntüyse 2 boyutlu renksiz görüntüye çeviriliyor.
if length(dim)==3
    I = rgb2gray(I);
end
```

% Giriş görüntüsünün boyutu değiştiriliyor.

```
PI = padarray(I,[2,2],'replicate'); %Boyutu değiştirilirken kullanılan maske 2x2 liktir
ve pad array (dizileri doldurma işlemi) replicate
olarak yapılmaktadır.
```

```
dim = size(PI); % Giriş görüntüsünün yeni boyutu
```

% Filtreleme işlemi için loopun başlatılması

```
for m = (1+3):(dim(1,1)-3)
    for n = (1+3):(dim(1,2)-3)
```

% Filtreleme için kullanılacak maskenin katsayıları

```
a = PI(m,n);
b1 = PI(m,n+1);
b2 = PI(m,n+2);
b3 = PI(m,n+3);
c1 = PI(m-1,n+1);
c2 = PI(m-2,n+2);
c3 = PI(m-3,n+3);
d1 = PI(m-1,n);
```

```

d2 = PI(m-2,n);
d3 = PI(m-3,n);
e1 = PI(m-1,n-1);
e2 = PI(m-2,n-2);
e3 = PI(m-3,n-3);
f1 = PI(m,n-1);
f2 = PI(m,n-2);
f3 = PI(m,n-3);
g1 = PI(m+1,n-1);
g2 = PI(m+2,n-2);
g3 = PI(m+3,n-3);
h1 = PI(m+1,n);
h2 = PI(m+2,n);
h3 = PI(m+3,n);
i1 = PI(m+1,n+1);
i2 = PI(m+2,n+2);
i3 = PI(m+3,n+3);

```

```

para1 = [a,b1,b2,b3,d1,d2,d3,f1,f2,f3,h1,h2,h3];
para2 = [a,c1,c2,c3,e1,e2,e3,g1,g2,g3,i1,i2,i3];
para3 = [a,median(para1),median(para2)];

```

% Filtrelenmiş çıkış görüntüsü

```

FPI(m-3,n-3) = median(para3);
end
end

```

% Karşılaştırma için normal median filtreleme işleminin yapılması

```

M = medfilt2(I,[5,5]);
figure (2);
imshow(M);
title ('Median filtreden geçirilmiş görüntü');

```

```

figure(3);
imshow(FPI);
title('Hybrid median filtreden geçirilmiş görüntü');

```

%Görüntü Parlaklık Ayarının Yapılması

```

[m1 n1 r1]=size(FPI);

if(r1==1)
    a=double(FPI)./255;
end

```

```
[m1 n1 r1]=size(a);
S=mean(mean(a(:,1)));
limit=0.5;
Fark=limit-S;

for(i=1:m1)
    for(j=1:n1)
        if(Fark<0)
            if(a(i,j,1)<=0.92)
                a(i,j,1)=a(i,j,1)+Fark;
            end
        end
        if(a(i,j,1)>1)
            a(i,j,1)=1;
        end
        if(a(i,j,1)<0)
            a(i,j,1)=0;
        end
    end
end

if(r1==1)
    IM=a;
end

IM=IM.*255;
IM=round(IM);
IM=uint8(IM);
figure(4);
imshow(IM);
title('Parlakligi optimumlastirilmis goruntu');

t = toc
```

Ek 5 : Adaptif Median Filtre (AMF) ile Görüntüdeki Gürültünün Azaltılması

```

clear all;
close all;
clc;

I = imread('C:/konya.jpg');
figure (1);
imshow(I);

tic
dim = size(I);    % Görüntü renkli görüntüyse 2 boyutlu renksiz görüntüye
                  çeviriliyor.
if length(dim)==3
    I = rgb2gray(I);
end

Gurultulu = imnoise (I,'salt & pepper',0.7);
figure(2);
imshow(Gurultulu);

Gurultulu_edge = zeros (276,276);

Gurultulu_convert = double (Gurultulu);

Gurultulu_edge (11:266, 11:266) = Gurultulu_convert (11:266, 11:266) +
Gurultulu_convert;

smax=9;

for i=11:266
    for j=11:266
        sx=3;
        sy=3;
        while ((sx<=smax) & (sy<=smax))
            Gurultulu_edge_min = Gurultulu_convert (i, j);
            Gurultulu_edge_max = Gurultulu_convert (i, j);
            Gurultulu_edge_median = median(median(Gurultulu_convert((i-
            floor(sx/2)):(i+floor(sx/2)),(j-floor(sy/2)):(j+floor(sy/2))))));
            for k= (i-floor (sx/2)) :( i+floor (sx/2))
                for l= (j-floor (sy/2)) :( j+floor (sy/2))
                    if Gurultulu_convert (k, l) < Gurultulu_edge_min
                        Gurultulu_edge_min = Gurultulu_convert (k, l);
                    end
                    if Gurultulu_convert (k, l) > Gurultulu_edge_max
                        Gurultulu_convert (k, l);
                    end
                end
            end
        end
    end
end

```

```
end
A = Gurultulu_edge_median - Gurultulu_edge_min;
B = Gurultulu_edge_median - Gurultulu_edge_max;
if (A>0) & (B<0)
    C = Gurultulu_edge(i, j) - Gurultulu_edge_min;
    D = Gurultulu_edge(i, j) - Gurultulu_edge_max;
    if (C>0) & (D<0)
        Gurultulu_edge(i, j) = Gurultulu_edge(i, j);
        break
    else
        Gurultulu_edge(i, j) = Gurultulu_edge_median;
        break
    end
else
    sx=sx+2;
    sy=sy+2;
    if (sx>smax) & (sy>smax)
        Gurultulu_edge(i,j) = Gurultulu_edge(i,j);
    end
end
end
end
end
end
figure(3), clf;
imshow(Gurultulu_edge);
t=toc
```

Ek 6 : MEAN Filtre ile Görüntüdeki Gürültünün Azaltılması

```

clear all;
close all;
clc;
I = imread('C:/kutahya.jpg');
figure (1);
imshow(I);
title ('gurultulu goruntu');

tic;
dim = size(I); % Görüntü renkli görüntüyse 2 boyutlu renksiz görüntüye çeviriliyor.

if length(dim)==3
    I = rgb2gray(I);
end

[x y z]=size(I);
N=zeros(x,y,z);
for i=1:x

    for j=1:y

        %İlk satırın birinci ve sonuncu pikseli check ediliyor.

        if (i==1 & j==1)
            mat(1)=0;
            mat(2)=0;
            mat(3)=0;
            mat(4)=0;
            mat(5)=I(i,j);
            mat(6)=I(i,j+1);
            mat(7)=0;
            mat(8)=I(i+1,j);
            mat(9)=I(i+1,j+1);
        end

        if (i==1 & j==y)
            mat(1)=0;
            mat(2)=0;
            mat(3)=0;
            mat(4)=I(i,j-1);
            mat(5)=I(i,j);
            mat(6)=0;
            mat(7)=I(i+1,j-1);
            mat(8)=I(i+1,j);
            mat(9)=0;
        end
    end
end

```

% Son satırın birinci ve sonuncu pikseli check ediliyor.

```
if (i==x & j==1)
    mat(1)=0;
    mat(2)=I(i-1,j);
    mat(3)=I(i-1,j+1);
    mat(4)=0;
    mat(5)=I(i,j);
    mat(6)=I(i,j+1);
    mat(7)=0;
    mat(8)=0;
    mat(9)=0;
end
```

```
if (i==x & j==y)
    mat(1)=I(i-1,j-1);
    mat(2)=I(i-1,j);
    mat(3)=0;
    mat(4)=I(i,j-1);
    mat(5)=I(i,j);
    mat(6)=0;
    mat(7)=0;
    mat(8)=0;
    mat(9)=0;
end
```

% Görüntünün geri kalanı check ediliyor.

```
if (i>1 & i<x & j>1 & j<y)
    mat(1)=I(i-1,j-1);
    mat(2)=I(i-1,j);
    mat(3)=I(i-1,j+1);
    mat(4)=I(i,j-1);
    mat(5)=I(i,j);
    mat(6)=I(i,j+1);
    mat(7)=I(i+1,j-1);
    mat(8)=I(i+1,j);
    mat(9)=I(i+1,j+1);
end
y1=I(i,j);
ybar=mean(mean(mat));
if ybar~=0
    ystad=std2(mat);
    ENL=(ybar/ystad)^2;
    sx2=((ENL*(ystad)^2)-(ybar^2)/(ENL+1));
    xcap=ybar+(sx2*(y1-ybar)/(sx2+(ybar^2/ENL)));
    N(i,j)=xcap;
else
```

```
        N(i,j)=y1;
    end
end
end
A=uint8(N);
figure(2);
imshow(A);
title ('Mean uygulanmis goruntu');
t=toc

imwrite(A,'C:/mekutahyaduz.jpg');
```

Ek 7 : Blind Deconvolution Algoritması ile Görüntünün Netleştirilmesi

```
clear all;
close all;
```

```
I=imread ('C:\divrigi256.jpg');
figure(1);
imshow(I);
title ('orjinal goruntu');
```

```
PSF = fspecial('gaussian',7,10);
Blurred = imfilter(I,PSF,'symmetric','conv');
figure(2);
imshow(Blurred);
title('Bulanik goruntu');
```

%Bozuk görüntüyü çeşitli boyutlardaki PSF(point-spread function)'lerin kullanılarak düzeltilmesi bu görüntü gerçek resmin boyutundan 4 piksel daha küçüktür

```
UNDERPSF = ones(size(PSF)-4);
[J1 P1] = deconvblind(Blurred,UNDERPSF);
figure(3);
imshow(J1);
title('Undersized yani boyut olarak 4 piksel daha küçük psf ile düzeltilmiş görüntü');
```

%İkinci aşamada ise birinci aşamada elde edilen görüntünün boyut olarak 4 piksel daha fazla yani oversized psf ile düzeltilmesi

```
OVERPSF = padarray(UNDERPSF,[4 4],'replicate','both');
[J2 P2] = deconvblind(Blurred,OVERPSF);
figure(4);
imshow(J2);
title('Oversized yani boyut olarak 4 piksel daha büyük psf ile düzeltilmiş görüntü');
```

%Üçüncü aşamada ise esas görüntüyle aynı boyuta sahip yani INITPSF ile görüntünün düzeltilmesi

```
INITPSF = padarray(UNDERPSF,[2 2],'replicate','both');
[J3 P3] = deconvblind(Blurred,INITPSF);
figure(5);
imshow(J3);
title('INITPSF ile puppyinin yani görüntünün düzeltilmesi');
imwrite(J3,'C:/divrigibland.jpg');
```

Ek 8 : Regularized Filtre İle Görüntünün Netleştirilmesi

```

clear all;
close all;
clc;
I = imread('C:/6.jpg');
I = I(10+[1:256],222+[1:256],:);
figure(1);
imshow(I);
title('orjinal resim');

PSF = fspecial('gaussian',11,5); %PSF'nin oluşturulması
Bulanik = imfilter(I,PSF,'conv'); % imfilter fonksiyonu kullanılarak görüntünün PSF
                                %fonksiyonuyla birleştirilmesi sonucu
                                %bulanıklaştırılması

figure(2);
imshow(Bulanik);
title('Bulanik görüntü');
V = .02;
Bozuk = imnoise(Bulanik,'gaussian',0,V); % imnoise fonksiyonu kullanılarak
                                         %bulanık görüntüye V değişkenli
                                         %Gaussian gürültüsü ekleniyor.

figure(3);
imshow(Bozuk);
title('Bulanık ve gürültülü görüntü');

NP = V*prod(size(I)); % gürültü gücü

[reg1 LAGRA] = deconvreg(Bozuk,PSF,NP);

% Bulanık ve gürültülü görüntüye noise power (NP) uygulanarak görüntü restore
%ediliyor. Bu algoritmanın NP değerine ne kadar duyarlı olup olmadığını anlamak
%için 3 deneme yapıp sonuçlar karşılaştırılıyor. İlk restorasyon sonucu yani reg1,
%gerçek NP değerini kullanır. Burada elde edilen çıkış değerinin iki parametresinin
%olduğu gözden kaçmamalıdır. Bunlardan biri reg1 değeri yani restore edilen
%görüntü, ikincisi ise sayısal bir ifade olan ve Lagrange çarpanını ifade eden LAGRA
%değeridir.

figure(4);
imshow(reg1);
title('NP ile onarılmış görüntü');

reg2 = deconvreg(Bozuk,PSF,NP*1.3);
figure(5);
imshow(reg2);
title('Daha büyük NP ile onarılmış görüntü');

```

```
reg3 = deconvreg(Bozuk,PSF,NP/1.3);  
figure(6);  
imshow(reg3);  
title('Daha küçük NP ile onarılmis goruntu');
```

```
Edged = edgetaper(Bozuk,PSF);  
reg4 = deconvreg(Edged,PSF,NP/1.3);  
figure(7);  
imshow(reg4);  
title('Edgetaper etkisi');
```

Ek 9 : Lucy-Richardson Algoritması ile Görüntünün Netleştirilmesi

```
close all;
clear all;
clc;

% Orijinal görüntü

I = imread('C:/cinili256.jpg');
figure(1);
imshow(I);
title('Orjinal goruntu');

% Gurultulu ve bulanik goruntu

PSF = fspecial('gaussian',7,10);
B = imfilter(I,PSF,'symmetric','conv');
BG = imnoise(B,'gaussian',0,0.002);
figure(2);
imshow(BG);
title('Bulanik ve gurultulu');

% Goruntunun duzeltilmesi

luc1 = deconvlucy(BG,PSF,6);
figure(3);
imshow(luc1);
title('Duzeltimis goruntu, NUMIT = 6');
imwrite (luc1,'C:/cinililucy.jpg');
```

Ek 10 : Wiener Filtresi ile Görüntünün Netleştirilmesi

```
clear all;
close all;
clc;

I = imread('C:/divrigi256.jpg');
figure(1);
imshow(I);
title('orjinal resim');

LEN = 10;
THETA = 7;
PSF = fspecial('motion',LEN,THETA); % PSF'nin yaratılması

Bulanik = imfilter(I,PSF,'circular','conv');
figure (2);
imshow(Bulanik);
title('Bulanik goruntu');

Yeni = deconvwnr(Bulanik,PSF);
figure(3);
imshow(Yeni);
title('wiener ile netlestirilmis goruntu');

imwrite(Yeni,'C:/divrigiwiener.jpg');
```

Ek 11 : Canny Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması

```

clear all;
close all;
clc;

I=imread('C:\cinili.jpg');
I = I(10+[1:256],222+[1:256],:);
figure(1);
imshow (I);
BW1=edge(I,'canny',[],1); % Eşik değerini otomatik atıyor ve sigma:1
figure(2);
imshow (BW1);
BW1=edge(I,'canny',[],2); % Eşik değerini otomatik atıyor ve sigma:2
figure(3);
imshow (BW1);
BW1=edge(I,'canny',[],3); % Eşik değerini otomatik atıyor ve sigma:3
figure(4)
imshow (BW1);

```

Ek 12 : Laplace of Gaussian (LoG) Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması

```

clear all;
close all;
clc;

I=imread('C:\cml.jpg');
I = I(10+[1:256],222+[1:256],:);
BW1=edge(I,'log');
figure(1);
imshow (BW1);
BW1=edge(I,'log',[],1); % Eşik değerini otomatik atıyor ve sigma:1
figure(2);
imshow (BW1);
BW1=edge(I,'log',[],2); % Eşik değerini otomatik atıyor ve sigma:2
figure(3)
imshow (BW1);

```

Ek 13 : Sobel Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması

```
I=imread('C:\cını.jpg');
I = I(10+[1:256],222+[1:256],:);
figure(1);imshow(I);
title('orjinal');
```

```
BW1=edge(I,'sobel');
figure(2);imshow(BW1);
title('sobel');
```

```
BW2=edge(I,'sobel','horizontal');
figure(3);imshow(BW2);
title('yatay sobel');
```

```
BW3=edge(I,'sobel','vertical');
figure(4);imshow(BW3);
title('dikey sobel');
```

Ek 14 : Prewitt Kenar Bulma Algoritması ile Görüntüdeki Kenarların Bulunması

```
I=imread('C:\cını.jpg');
I = I(10+[1:256],222+[1:256],:);
BW1=edge(I,'prewitt');
figure(1);
imshow(I);
title('orjinal');
```

```
figure(2);
imshow(BW1);
title('Prewitt');
```

```
figure(3);
BW1=edge(I,'prewitt','horizontal');
imshow(BW1);
title('Yatay Prewitt uygulanmis cını görüntüsü');
```

```
figure(4);
BW1=edge(I,'prewitt','vertical');
imshow(BW1);
title('Dikey Prewitt uygulanmis cını görüntüsü');
```

Ek 15 : PSNR ve MSE Değerlerini Bulmak İçin Kullanılan Algoritma

```

clear all;
close all;
clc;

A = imread('C:/divrigi.jpg');
B = imread('C:/divgurultulu.jpg');

A=im2double(A);
B=im2double(B);

if A == B
    error('Images are identical: PSNR has infinite value')
end

max2_A = max(max(A));
max2_B = max(max(B));
min2_A = min(min(A));
min2_B = min(min(B));

if max2_A > 1 | max2_B > 1 | min2_A < 0 | min2_B < 0
    error('input matrices must have values in the interval [0,1]')
end

error = A - B;
[m1 n1] = size(A);
MSE = (sum(sum(error.*error)))/(m1*n1)
decibels = 10*log10(1/(sqrt(mean(mean(error.^2)))));
disp(sprintf('PSNR = +%5.2f dB',decibels))

```