



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**VIDEO GÖRÜNTÜLERİNDEN 3 BOYUTLU YAPININ
ELDE EDİLMESİ**

Elek.Müh. Evren BUĞDAYCIOĞLU
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Elektrik-Elektronik Mühendisliği Programı

Danışman
Prof.Dr. Aydın AKAN

Eylül, 2008

İSTANBUL



**İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

YÜKSEK LİSANS TEZİ

**VIDEO GÖRÜNTÜLERİNDEN 3 BOYUTLU YAPININ
ELDE EDİLMESİ**

Elek.Müh. Evren BUĞDAYCIOĞLU
Elektrik-Elektronik Mühendisliği Anabilim Dalı
Elektrik-Elektronik Mühendisliği Programı

Danışman
Prof.Dr. Aydın AKAN

Eylül, 2008

İSTANBUL

Bu çalışma 20/10/2008 tarihinde ařağıdaki jüri tarafından Elektrik-Elektronik Mühendisliğı Anabilim Dalı Elektrik-Elektronik Mühendisliğı programında Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Jürisi

Prof. Dr. Aydın AKAN (Danışman)
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof.Dr.Sıddık YARMAN
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof.Dr.Osman Nuri UÇAN
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof.Dr. Hakan Ali ÇIRPAN
İstanbul Üniversitesi
Mühendislik Fakültesi

Prof.Dr.Ahmet SERTBAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi

ÖNSÖZ

Lisans ve yüksek lisans öğrenimim sırasında ve tez çalışmalarım boyunca gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Prof.Dr.Aydın AKAN'a en içten dileklerle teşekkür ederim.

Bu çalışma boyunca yardımlarını esirgemeyen aileme teşekkürü borç bilirim.

Eylül, 2008

Evren BUĞDAYCIOĞLU

İÇİNDEKİLER

| | |
|--|------|
| ÖNSÖZ..... | I |
| İÇİNDEKİLER | II |
| ŞEKİL LİSTESİ..... | V |
| TABLO LİSTESİ | VIII |
| ÖZET..... | IX |
| SUMMARY | X |
| 1. GİRİŞ | 1 |
| 2. GENEL KISIMLAR | 2 |
| 2.1. BİR DEN FAZLA GÖRÜNTÜNÜN GEOMETRİSİ | 2 |
| 2.1.1. Kamera Modeli..... | 2 |
| 2.1.1.1 Sonlu Kamera Modeli | 2 |
| 2.1.2. Kalibrasyon..... | 6 |
| 2.1.2.1. 3-B Desenlerle Kalibrasyon: | 7 |
| 2.1.2.2. 2-B Desenlerle Kalibrasyon: | 7 |
| 2.1.2.3. 1-B Desenlerle Kalibrasyon: | 8 |
| 2.1.2.4. 0-B Desenlerle Kalibrasyon (Kendi Kendine Kalibrasyon): | 9 |
| 2.1.3. Epipolar Geometri | 10 |
| 2.1.3.1. Temel Matris: | 11 |
| 2.1.4. Üçgenleştirme ve Koordinatların Elde Edilmesi..... | 15 |
| 2.2. DEVİNİMDEN YAPI ELDE EDİLMESİ | 17 |
| 2.2.1. İlişkin Çiftlerin Bulunması..... | 18 |
| 2.2.1.1. Öznitelik Noktalarının Sezimi | 19 |
| 2.2.1.2. Nokta eşleşmelerinin bulunması | 19 |
| 2.2.2. Temel Matrisin Kestirilmesi | 20 |

| | |
|--|-----------|
| 2.2.2.1. Lineer Kestirim Yaklaşımı..... | 20 |
| 2.2.2.2. Döngülü Kestirim Yaklaşımı | 21 |
| 2.2.2.3. Gürbüz Kestirim Yaklaşımı | 21 |
| 2.2.3. Duruşların ve Kamera Matrislerinin Kestirilmesi | 21 |
| 2.2.3.1. R ve t 'yi Belirlemek İçin Lineer Algoritma | 23 |
| 2.2.3.2. R ve t 'yi Belirlemek İçin Gürbüz Algoritma..... | 24 |
| 2.2.3. 3B Noktaların Elde Edilmesi..... | 25 |
| 2.2.3.1. Orta-Nokta Yöntemi:..... | 26 |
| 2.2.3.2. Linner Üçgenleştirme Yöntemi: | 26 |
| 2.2.3.3. Lineer-Özdeğer Yöntemi:..... | 27 |
| 2.2.3.4. Lineer En Küçük Kareler Yöntemi:..... | 28 |
| 2.2.3.5. Döngülü Lineer Üçgenleştirme Yöntemi:..... | 28 |
| 2.2.3.6. Polinomsal Üçgenleştirme Yöntemi | 28 |
| 2.3. STEREO BAĞINTI YÖNTEMİ..... | 30 |
| 2.3.1. Aykırılık Haritası | 30 |
| 2.3.2. Eşleşme Maliyet Fonksiyonları | 32 |
| 2.3.3. Sahnedeki Nesnelerin Birbirinin Görüntülerini Kapatması..... | 34 |
| 2.3.3.1. Aykırılıkta Çift Dorukluluk (BMD) | 35 |
| 2.3.3.2. Eşleşme Kalite Atlamaları (MGJ)..... | 36 |
| 2.3.3.3. Sağ/Sol Kontrolü (LRC) | 36 |
| 2.3.3.4. Sıralama Kısıtı (ORD) | 37 |
| 2.3.3.5. Örtme Kısıtı (OCC)..... | 38 |
| 2.3.3.6. Yöntemlerin Karşılaştırılması | 38 |
| 2.3.4. Optimizasyon ve Aykırılık Hesabı..... | 40 |
| 2.3.4.1. Dinamik Programlama | 42 |
| 2.3.4.2. Grafik-Kesme Yöntemi(Graph Cut) | 43 |
| 2.3.4.3. İşbirliği Yaklaşımı | 43 |
| 2.3.4.4. İnanç Yayılımı(Belief Propagation) | 44 |
| 2.3.5. Daha Önce Yapılmış Çalışmalar | 45 |
| 3. YÖNTEM..... | 52 |
| 3.1. PİKSEL ALANINDA LOKAL EŞLEŞME | 52 |
| 3.1.1. Piksel Benzeşmezlik Ölçüsü | 53 |

| | |
|--|-----|
| 3.1.2. Temel Eşleşme Maliyet Fonksiyonu | 54 |
| 3.1.3. Çoklu Pencere Eşleşme Maliyet Fonksiyonu..... | 57 |
| 3.2. BÖLÜTLEME VE ENERJİ MİNİMİZASYONU | 59 |
| 3.2.1. Bölüt Bazlı İnanç Yayılımı | 59 |
| 3.3. ÖRTME HARİTASININ ÇIKARILMASI..... | 61 |
| 3.4. EĞİMLİ DÜZLEMLERLE AYKIRILIK KESTİRİMİ..... | 63 |
| 3.4.1. Piksellerin Sınıflandırılması..... | 63 |
| 3.4.2. Aykırı Piksellerin Ayıklanması..... | 64 |
| 3.4.3. Mevcut Bölütlerin Yeniden Yapılandırılması | 64 |
| 3.4.4. Düzlemlerin Kestirimi | 65 |
| 3.4.5. Enerji Minimizasyonu | 66 |
| 4. BULGULAR | 68 |
| 4.1. Eşleşme Maliyet Fonksiyonları Karşılaştırması..... | 68 |
| 4.1.1. “Tsukaba” İmgelerine ait Bulgular | 70 |
| 4.1.2. “Teddy” İmgelerine ait Bulgular | 75 |
| 4.1.1. “Cones” İmgelerine ait Bulgular | 81 |
| 4.2. ÖRTME HARİTASI ÇIKARMA BULGULARI..... | 86 |
| 4.3. HATALI VERİ AYIKLAMA VE ÖNERİLEN ALGORİTMA BULGULARI | 89 |
| 5. TARTIŞMA VE SONUÇ..... | 92 |
| KAYNAKLAR..... | 95 |
| EK | 102 |
| DÖNGÜSEL İNANÇ YAYILIMI | 102 |
| ÖZGEÇMİŞ..... | 106 |

ŞEKİL LİSTESİ

| | | |
|------------|---|----|
| Şekil 2.1 | : (a) iğne deliği kamera modeli. (b) iğne deliği kamera modelinin yandan görünüşü..... | 3 |
| Şekil 2.2 | : (a) Kamera tarafından algılanan kesik koni biçimindeki hacim. (b) Hacim arakesitinin 2B düzlemsel örneği | 3 |
| Şekil 2.3 | : 3B kalibrasyon deseni örneği [10] | 7 |
| Şekil 2.4 | : 2B kalibrasyon deseni [6]..... | 8 |
| Şekil 2.5 | : 1B kalibrasyon deseni [8]..... | 8 |
| Şekil 2.6 | : Epipolar Geometri | 9 |
| Şekil 2.7 | : Geri yansıtılan noktanın belirsizliği. | 10 |
| Şekil 2.9 | : Paralel kamera durumu [1] | 11 |
| Şekil 2.10 | : Yakınsayan kamera durumu [1] | 11 |
| Şekil 2.11 | : Düzlem yardımıyla dönüşüm. | 12 |
| Şekil 2.12 | : Normalize edilmiş imge koordinat sistemi..... | 14 |
| Şekil 2.13 | : Geriçatım yönteminin adımları | 18 |
| Şekil 2.14 | : Bağlı kamera konumları | 22 |
| Şekil 2.15 | : Orta-nokta yöntemi | 26 |
| Şekil 2.16 | : Polinomsal Üçgenleştirme (PÜ)..... | 29 |
| Şekil 2.17 | : İdeal stereo geometri | 31 |
| Şekil 2.18 | : Örtülmüş bölgelerin grafiksel gösterimi..... | 34 |
| Şekil 2.19 | : Aykırılıkta çift dorukluluk yöntemi | 36 |
| Şekil 2.20 | : Sıralama Kısıtı..... | 37 |
| Şekil 2.21 | : Örtülmüş bölge belirleme yöntemleri karşılaştırması | 39 |
| Şekil 2.22 | : Dinamik programlama kullanarak stereo eşleme..... | 42 |
| Şekil 2.23 | : Klaus ve diğ. [45] sunduğu algoritmanın temel yapısı..... | 46 |
| Şekil 2.24 | : [68] algoritmasının giriş modülü..... | 48 |
| Şekil 2.25 | : Piksel sınıflandırma modülü [68]..... | 49 |
| Şekil 2.26 | : Döngülü iyileştirme modülü [68]..... | 49 |
| Şekil 2.27 | : Zitnick [70] algoritmasının “Tsukaba” imgesi üzerinde oluşturduğu bölütleme..... | 50 |

| | | |
|-------------------|---|-----------|
| Şekil 2.28 | : Hem bölüt şekillerini hemde derinliği güncelleyen algoritma [70]. | 51 |
| Şekil 2.29 | : Enerji minimizasyonunu düzlem parametrelerini güncellemede kullanan algoritma [83]..... | 51 |
| Şekil 3.1 | : $\bar{d}(x_R, x_L, I_R, I_L)$ tanım ve hesaplaması | 54 |
| Şekil 3.2 | : “Teddy” imgesi için varyans sınıflandırması sonucu..... | 56 |
| Şekil 3.3 | : Standart pencerelere ek olarak kullanılan destek penceresi: kırmızı nokta merkez piksel, gri pikseller ortalamaya katılan diğer pikseller.. | 57 |
| Şekil 3.4 | : Çoklu Pencereli Eşleşme Maliyet Fonksiyonu Algoritma Adımları.... | 58 |
| Şekil 3.5 | : Stereo ve düzlük kısıtları MRA kullanarak modellenmiştir. Her düğüm bir bölütü temsil eder ve komşu bölütler kenarlarla birleştirilmiştir. .. | 60 |
| Şekil 3.6 | : Hatalı örtülmüş bölgelerin ayıklanması | 62 |
| Şekil 3.7 | : Piksel sınıflandırması | 64 |
| Şekil 3.8 | : Önerilen Stereo İlişkilendirme Algoritmasının Temel Yapısı | 67 |
| Şekil 4.1 | : Karşılaştırmalarda kullanılan imgeler ve derinlik haritalarının bölgelere ayrılmış hali..... | 69 |
| Şekil 4.2 | : KHA algoritmasıyla elde edilen derinlik haritaları | 71 |
| Şekil 4.3 | : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları | 72 |
| Şekil 4.4 | : Piksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 73 |
| Şekil 4.5 | : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 74 |
| Şekil 4.6 | : KHA algoritmasıyla elde edilen derinlik haritaları | 77 |
| Şekil 4.7 | : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları | 78 |
| Şekil 4.8 | : Piksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 79 |
| Şekil 4.9 | : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 80 |
| Şekil 4.10 | : KHA algoritmasıyla elde edilen derinlik haritaları | 82 |
| Şekil 4.11 | : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları | 83 |

| | | |
|-------------------|--|-----------|
| Şekil 4.12 | : Piksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 84 |
| Şekil 4.13 | : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları | 85 |
| Şekil 4.14 | : (a) LRC sonucu (b) Önerilen Yöntem (c) 2.Önerilen Yöntem..... | 87 |
| Şekil 4.15 | : Kestirilen örtülmüş bölge haritaları ve orjinal haritalar | 88 |
| Şekil 4.16 | : “Teddy” imgelerine ait kestirim sonuçları | 90 |
| Şekil 4.17 | : “Cones” imgelerine ait kestirim sonuçları..... | 91 |

TABLO LİSTESİ

| | | |
|-------------------|---|-----------|
| Tablo 2.1 | : Temel matrisin özellikleri..... | 13 |
| Tablo 4.1 | : İnançYayılmıAlgoritması Parametreleri..... | 69 |
| Tablo 4.2 | : KHA algoritmasına göre hata oranları..... | 70 |
| Tablo 4.3 | : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri | 70 |
| Tablo 4.4 | : Piksel Bazlı İnanç Yayılmı algoritmasına göre hata oranları | 75 |
| Tablo 4.5 | : Bölüt Bazlı İnanç Yayılmı algoritmasına göre hata oranları | 75 |
| Tablo 4.6 | : KHA algoritmasına göre hata oranları..... | 75 |
| Tablo 4.7 | : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri | 76 |
| Tablo 4.8 | : Piksel Bazlı İnanç Yayılmı algoritmasına göre hata oranları | 76 |
| Tablo 4.9 | : Bölüt Bazlı İnanç Yayılmı algoritmasına göre hata oranları | 76 |
| Tablo 4.10 | : KHA algoritmasına göre hata oranları..... | 81 |
| Tablo 4.11 | : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri | 81 |
| Tablo 4.12 | : Piksel Bazlı İnanç Yayılmı algoritmasına göre hata oranları..... | 81 |
| Tablo 4.13 | : Bölüt Bazlı İnanç Yayılmı algoritmasına göre hata oranları | 86 |
| Tablo 4.14 | : Örtme haritası çıkarmada önerilen yöntemlerin başarı yüzdeleri..... | 86 |
| Tablo 4.15 | : Aykırı verileri ayıklama algoritması uygulandıktan sonra kalan kararlı pikseller ve hata oranları | 89 |
| Tablo 4.16 | : “Teddy” imgesi için önerilen yöntemin “Middlebury” test ortamının ilk sıralarındaki algoritmalarla karşılaştırması..... | 89 |
| Tablo 4.17 | : “Cones” imgesi için önerilen yöntemin “Middlebury” test ortamının ilk sıralarındaki algoritmalarla karşılaştırması..... | 89 |

ÖZET

VIDEO GÖRÜNTÜLERİNDEN 3 BOYUTLU YAPININ ELDE EDİLMESİ

Bu çalışmada video görüntülerinden 3 boyutlu yapı elde etme yöntemleri araştırılmıştır. Çalışmayı iki bölüme ayırmak mümkündür. Birinci bölümde 3B yapı elde etme algoritmalarının temelini oluşturan geometri incelenmiştir. Temel epipolar geometri sunulmuş ve 3B yapı elde etmede kullanılan yöntemler iki ana kategoride incelenmiştir.

Birinci kategoride “Devinimden Yapı Elde Etme” algoritmalarının genel hatları irdelenmiş ve bu yöntemi oluşturan tüm adımlar detaylı bir şekilde sunulmuştur.

İkinci kategoride “Stereo İlişkilendirme” algoritmalarının temel yapı taşları ve bu algoritmaları dizayn ederken ortaya çıkabilecek problemler irdelenmiş ve güncel başarılı algoritmaların yapıları kısaca sunulmuştur.

Çalışmanın ikinci bölümünde stereo ilişkilendirme algoritmalarının yapı taşları olan eşleşme maliyet fonksiyonu ve örtülmüş bölgelerin belirlenmesi bölümlerine yeni yaklaşımlar önerilmiştir. Ayrıca, başarıyı arttırmak amacıyla bölütlere düzlemlerin uydurulduğu algoritmalara uygulanabilecek yeni bir “aykırı veri ayıklama” yöntemi önerilmiştir.

Önerilen 3 farklı eşleşme maliyet fonksiyonu, tüm stereo algoritmaların ortak olarak kullandığı veri setleri üzerinde test edilmiş ve sonuçları diğer önde gelen algoritmalarla karşılaştırılmıştır. Aynı şekilde örtülmüş bölgelerin bulunmasını sağlayan yeni yöntemin sonuçları da diğer klasik yöntemlerle karşılaştırılmıştır. Aykırı veri ayıklama yöntemi elde edilen derinlik haritalarına uygulanmış ve eleme sonucu kalan noktaların doğruluk oranı sunulmuştur. En son bölümde karşılaştırma sonuçlarıyla ilgili bir değerlendirme ortaya konmuştur.

SUMMARY

3D STRUCTURE RECONSTRUCTION FROM VIDEO SEQUENCES

This thesis deals with the problem of three dimensional scene reconstruction from multiple camera images. This is a well established problem in computer vision and has been significantly researched. In recent years some excellent results have been achieved, however existing algorithms often fall short of many biological systems in terms of robustness and generality. The aim of this research was to develop improved algorithms for reconstructing 3D scenes, with a focus on accurate system modelling and correctly dealing with occlusions.

Thesis structure contains two parts. In the first part, foundations of 3D reconstruction algorithms are studied. After a brief description of epipolar geometry, methods for 3D reconstructions are explained in two categories.

In the first category, after a summary, every step of *structure from motion* algorithms is detailly explained.

In the second category, fundamental parts of *stereo correspondence* alghoritms like matching score functions, energy minimization techniques and occlusion handling methods are studied.

In the second part of this study, novel methods for matching score function and occlusion handling are proposed. Also, a novel method for outlier rejection which can be increase the effectiveness of plane fitting step is proposed.

Finally, all this proposed methods are tested with image sets from “Middlebury” test bed and final disparity maps are compared.

1. GİRİŞ

Görme duyusunun nasıl işlediği, etraftaki 3 boyutlu dünyanın nasıl algılanabildiği, antik çağlardan beri insanoğlunun merakını uyandıran sorulardır. Yaklaşık ikibin yıl boyunca kabul gören, Euclid (MÖ. 325-365) ve Ptolemy (MS. 85-165) gibi birçok araştırmacı tarafından dile getirilen teori, gözlerin dışarıya doğru, dünyayı “hissedebilen” ışınlar gönderdiği şeklindeydi. Bu yaklaşım 17. yüzyıl başlarına kadar geçerliliğini korumuştur. 1604 yılında Kepler, gözün optiğinin teorik açıklaması hakkındaki ilk makaleyi yayınlamıştır. Ardından Scheiner, 1625 yılında, gözün arkasında oluşan imgenin varlığını deneysel olarak kanıtlamıştır. Bu buluşlar, *derinlik* algısının nasıl işlediğini, retinada oluşan 2B imgelerden üç boyutlu dünyayı nasıl algıladığımız gibi daha başka soruları da beraberinde getirmiştir.

20. yüzyılın ortalarında bilgisayarların geliştirilmesiyle, artık sadece insan fizyolojisiyle değil başka yöntemlerle de derinliğin nasıl algılanabileceği konusu, araştırmacıların kafasını meşgul etmeye başlamıştır. İnsanın derinlik algılama duygusunun anlaşılmasına yardımcı olmanın yanısıra, başarılı bir makine derinlik algısı yönteminin, birçok pratik uygulama alanı vardır. Örnek olarak; arazi haritalama, robot navigasyonu, sanal gerçeklik, endüstriyel otomasyon, otomatik navigasyon, eş-zamanlı bilgisayar-insan etkileşimi uygulamaları verilebilir. Bu noktada Euclid ve Ptolemy'nin doğru bir fikir ortaya koyduğu düşünülebilir: Sonar, radar ve lazer yönbulucu gibi aktif ışımaya dayalı yöntemler yardımıyla yüksek doğrulukta derinlik bilgisi elde edilebilir. Ancak, bu yöntemlerin menzili sınırlıdır ve kısıtlı kullanım alanına sahiptirler; tamamen pasif yöntemlerin daha geniş kullanım alanı vardır.

Pasif yöntem olarak akla gelen ilk yöntem, 2B imgeler yardımıyla derinlik bilgisinin elde edilmesidir. Bu amaçla ortaya atılan yöntemler, devinime ve stereo ilişkilendirmeye dayalı olarak iki ana kategoride incelenebilir. Bu tez kapsamında, stereo ilişkilendirmeye dayalı yöntemler üzerine araştırma yapılmıştır. Bu yöntemlerin ana parçası olan eşleştirme maliyet fonksiyonlarına yeni bir alternatif sunulmuş ve

başarısı diğer yöntemlerle birkaç açıdan karşılaştırılmıştır. Ayrıca stereo sistemlerin bir başka problemi olan örtülmüş bölgelerin belirlenmesi konusunda yeni bir yaklaşım sunulmuştur. Son olarak, düzlem kestirimine dayalı yöntemlerde kullanılacak, aykırı verileri ayıklayan bir algoritma önerilmiştir.

2. GENEL KISIMLAR

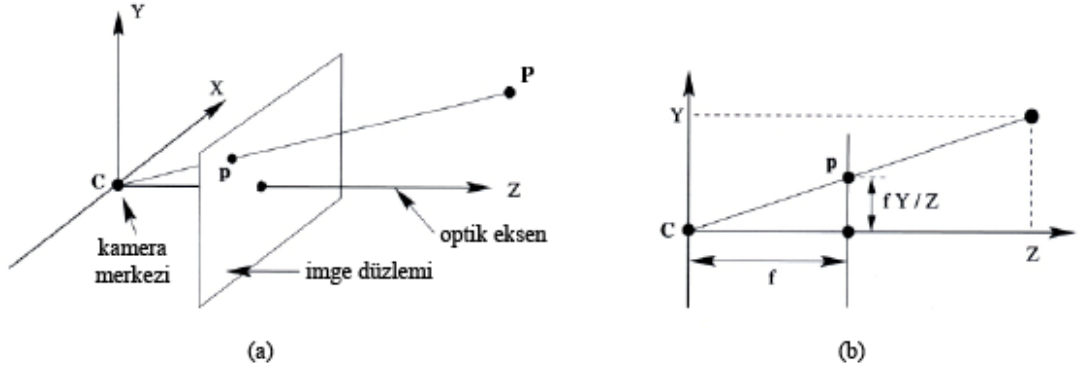
2.1. BİRDEN FAZLA GÖRÜNTÜNÜN GEOMETRİSİ

2.1.1. Kamera Modeli

Kamera modeli, optik kameranın görüntüleme sürecinin benzetimini yapmak amacıyla, 3 boyutlu genel koordinat sistemini 2 boyutlu imge düzlemine ilişkilendiren dönüşümdür. Bu dönüşüm genellikle matris şeklinde gösterilir ve noktaların izdüşümleri göz önüne alındığında 3×4 boyutlarında “İzdüşüm Matrisi (P)” ile tanımlanır. İzdüşüm matrisi, kameranın odak uzaklığı gibi dahili parametrelerini ve dönme, öteleme gibi harici parametrelerini içinde barındırır.

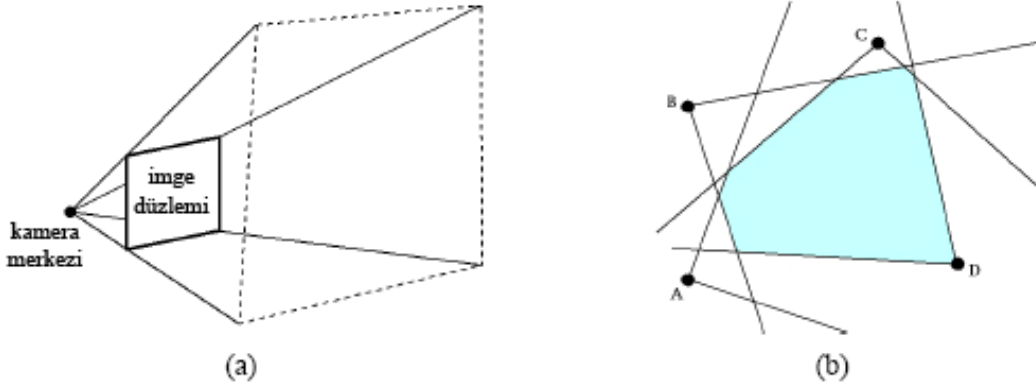
2.1.1.1 Sonlu Kamera Modeli

Genel uzaydaki 3 boyutlu bir noktanın imge düzlemine izdüşümü göz önüne alındığında, İğne Deliği Kamera Modelinde, tüm izdüşüm ışınlarının, kamera merkezi olarak adlandırılan bir noktada kesiştiği öngörülmektedir. Kamera merkezini (C) öklid koordinat sisteminin merkezi olarak alıp, görüntü düzlemini x ve y eksenlerine paralel şekilde z yönünde merkezden odak uzaklığı f kadarlık mesafeye yerleştirdiğimizde elde edilen yapı, *iğne deliği kamera modeli* olarak adlandırılmaktadır ve Şekil 2.1(a) da gösterilmiştir.



Şekil 2.1 : (a) iğne deliği kamera modeli. (b) iğne deliği kamera modelinin yandan görünüşü

Bu modelde, bir görüntünün içerdiği hacim, izdüşüm merkezinden geçen ve görüntünün sınırları içinde kalan tüm ışınların kümesidir. Bu hacim, tepesi izdüşüm merkezi olan kesik koni biçimindedir. Geri çatım uzayının sınırları, görüntülere ait kesik konilerin ikişerli kesişimlerinin, birleşiminden oluşur. Kesik koni ve kesik konilerin kesişimi kavramı Şekil 2.2 de gösterilmiştir.



Şekil 2.2 : (a) Kamera tarafından algılanan kesik koni biçimindeki hacim. (b) Hacim arakesitinin 2B düzlemsel örneği (Tüm kamera konilerinin kesişimi, sistemin algılayabildiği hacmi oluşturur.).

Şekil 2.1(b) de kamera modelinin yandan görünüşü tasvir edilmiştir. Üçgenler arasındaki benzerlikler kullanılarak 3B nokta $P = (X, Y, Z)^T$, imge düzlemindeki p koordinatına aşağıdaki şekilde dönüştürülebilmektedir:

$$(X, Y, Z)^T \rightarrow (fX/Z, fY/Z)^T \quad (2.1.1)$$

Homojen koordinatlar kullanılarak,

$$P = (X, Y, Z, W_p)^T \quad (2.1.2)$$

$$p = (fX / Z, fY / Z, w_p) \quad (2.1.3)$$

olarak ifade edilir. Bu formüller ile izdüşüm geometrisi modeline eklenen w_p parametresi ölçekleme çarpanıdır. Eğer W_p ve w_p çarpanlarının bire eşit olduğunu farzederseniz, p noktasının imge düzlemindeki izdüşümünün koordinatları,

$$p = (fX, fY, Z) \quad (2.1.4)$$

şeklini alır. Homojen vektörlerle gösterildiğinde, 3 boyutlu öklid uzayı R^3 ten 2 boyutlu öklid uzayı R^2 'ye dönüşüm aşağıdaki matris çarpımı ile tanımlanır.

$$p = \begin{bmatrix} f & & 0 \\ & f & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.1.5)$$

Yukarıdaki dönüşümde, imge düzleminin koordinat orijininin ile *ana noktanın* üst üste geldiği farzedilir. Ana nokta, kameranın optik eksenine imge düzleminin kesiştiği noktadır. Kamera orijinindeki kaymalar göz önüne alındığında ana nokta $(x_0, y_0)^T$ koordinatlarında konumlanır ve dönüşüm aşağıdaki şekli alır.

$$(X, Y, Z)^T \rightarrow (fX / Z + x_0, fY / Z + y_0)^T \quad (2.1.6)$$

(2.1.5) teki matris formuna çevirdiğimizde,

$$p = \begin{bmatrix} f & & x_0 & 0 \\ & f & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.1.7)$$

olur. Kamera izdüşüm matrisi bu şekilde tanımlandığında, x ve y yönlerinde oluşabilecek yönbağımsız olmayan ölçeklemeler gözardı edilmiş olur. Bu düzensizlik CCD kameralarda, piksellerin üretim sonuçlarının kare olmadığı durumlarda ortaya çıkar. Eşit olmayan ölçekleme katsayılarından dolayı oluşacak hatayı düzeltmek için izdüşüm matrisi,

$$\text{diag}(m_x, m_y, 1) \quad (2.1.8)$$

ifadesiyle çarpılır. Burada m_x ve m_y , x ve y yönlerinde birim uzaklık başına düşen piksel sayısıdır. Bu çarpımdan sonra izdüşüm matrisi aşağıdaki şekli alır.

$$p = \begin{bmatrix} \alpha_x & x_0 & 0 \\ & \alpha_y & y_0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.1.9)$$

Burada α_x ve α_y , x ve y yönlerindeki odak uzaklığıdır.

Göz önüne alınması gereken bir diğer parametre de piksellerin eğiminden kaynaklanan çarpıklık, s parametresidir. Günümüzdeki kameralarda değeri sıfır olarak alınabilmesine rağmen eski kameralar düşünülerek denkleme katılmaktadır. Çarpıklık parametresinide eklediğimizde izdüşüm matrisi,

$$p = \begin{bmatrix} \alpha_x & s & x_0 & 0 \\ & \alpha_y & y_0 & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (2.1.10)$$

şeklini alır. Bu matrisi aşağıdaki şekilde tekrar düzenleyebiliriz.

$$p = K[I \mid 0]P \quad (2.1.11)$$

Burada K matrisi, kameranın dahili parametrelerini oluşturur. (2.1.11) denkleminle tanımladığımız kamera modeli, kameranın perspektifinden imge düzlemine izdüşümleri belirler. P noktası kamera koordinat sistemine göre tanımlanmıştır. Ancak kamera koordinat sistemi, genel koordinat sisteminin içine yerleştirilmiştir ve kamera koordinat sisteminin orijini genel koordinat sistemininkiyle çakışmaz. 3B nokta P ile genel koordinat sistemi aşağıdaki şekilde ilişkilendirilebilir.

$$P = R(P_w - C) \quad (2.2.12)$$

Bu denklemde kullanılan R, kamera koordinat sisteminin yönelimini belirleyen dönme matrisi ve P_w , genel koordinat sistemindeki 3B noktadır. Matris şeklinde ifade edildildiğinde (2.2.12) aşağıdaki şekli alır.

$$P = \begin{bmatrix} R & -RC \\ 0 & 1 \end{bmatrix} P_w \quad (2.1.13)$$

Dahili kamera matrisiyle birleştirildiğinde,

$$p = KR[I \mid -C] P_w \quad (2.1.14)$$

elde edilir.

$KR[I \mid -C]$ terimi kamera izdüşüm matrisidir. Öte yandan, kamera merkezini açıkça belirtmek yerine koordinat sistemleri arasındaki dönüşümü, dönme ve sonrasında bir öteleme ile tanımlamak daha uygundur. Bu şekilde kamera izdüşüm matrisi son şeklini alır.

$$P_k = K[R \mid t] \quad (2.1.15)$$

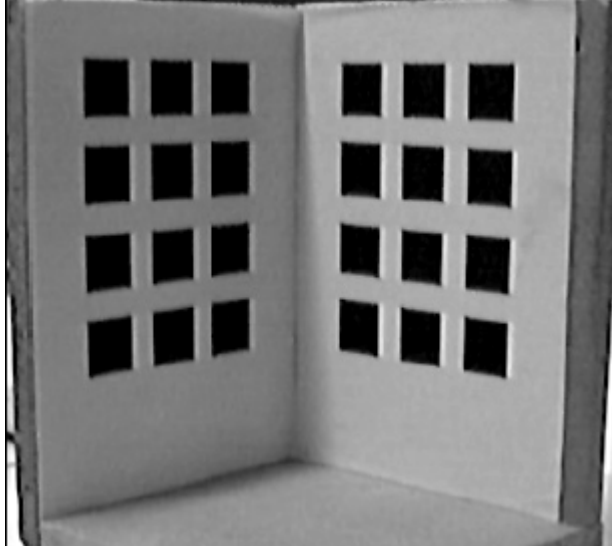
2.1.2. Kalibrasyon

Kalibrasyon, kamera dahili parametrelerini elde etme sürecine verilen isimdir [1,2]. 3B geriçatım işleminde, sahneden bilginin elde edilebilmesi için gerekli önemli adımlarından biridir. Örneğin, stereo ilişkilendirme algoritmalarında görüntülerin düzeltilmiş olduğu varsayılır ve bu işleminin yapılabilmesi için kamera kalibrasyonun bilinmesi gerekmektedir. Devrimden 3B yapı elde edilmesi problemlerinde de 3B geriçatım probleminin doğrusal olmamasından dolayı kamera matrisinin yüksek doğrulukla elde edilmemiş olması şarttır. Aksi takdirde, algoritmaların çoğu yakınsamıyacaktır veya yakınsasa bile fiziksel olarak anlamsız bir sonuç bulunacaktır. Bu önemli problem geçtiğimiz onyıllar boyunca yaygın olarak araştırılmıştır [2-9]. Kullanılan kalibrasyon deseninin [8] boyutlarına göre yöntemlerin sınıflandırması 4 kategoride yapılabilir :

- 3-B desenlerle kalibrasyon
- 2-B desenlerle kalibrasyon
- 1-B desenlerle kalibrasyon
- 0-B desenlerle kalibrasyon (kendi kendine kalibrasyon)

2.1.2.1. 3-B Desenlerle Kalibrasyon:

Bu yaklaşımda kamera kalibrasyonu, şekil 2.3 te bir örneği verilmiş olan, yapısı yüksek hassaslıkla bilinen 3B desen kullanılarak gerçekleştirilir.



Şekil 2. 3 : 3B kalibrasyon deseni örneği [10]

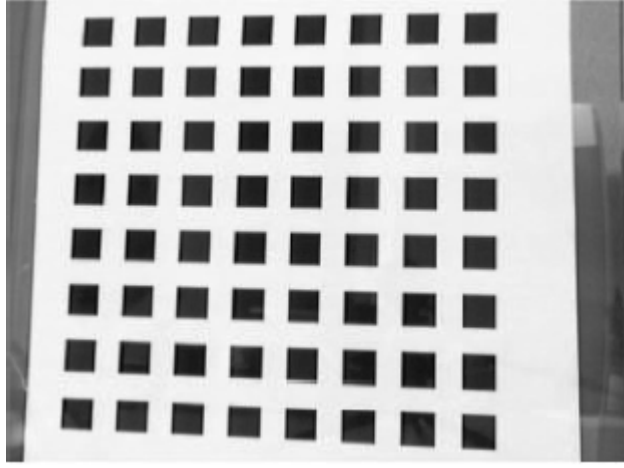
Örnek olarak Faugeras'ın makalesindeki [2] 3B deseninin sağladığı verimli kalibrasyon verilebilir. Bu yönetime bir başka yaklaşım, Tsai'nin ünlü makalesidir [4]. Bu yöntemde ötelemesi kesin bir şekilde bilinen 2B bir düzlem kullanılarak, üçüncü boyut hakkındaki bilgiler elde edilmektedir. Ancak pratikte Tsai'nin ulaştığı sonuçlara erişilememektedir.

2.1.2.2. 2-B Desenlerle Kalibrasyon:

Bu kategorideki yöntemler, sınırlı sayıda görünüşten şekil 2.4 teki gibi düzlemsel desenlerin gözlemine dayanır [6, 9]. Tsai'nin tekniğinden [4] farklı olarak düzlemin devinimi belirtilmemiştir, gerekli kalibrasyon deseni herkes tarafından kolayca hazırlanabilir ve sonuçların doğruluğu kabul edilebilir düzeydedir.

Zhnag'in yönteminde [6], düzlemde kalibrasyon deseni, kamerayı veya model düzlemi hareket ettirerek, birkaç kez farklı yönlerden kaydedilir. Genel koordinat sisteminin model düzelemine hizalı olduğu varsayılır. Bu varsayım kalibrasyon deseninin $z = 0$ düzleminde ve x- ve y- eksenlerine paralel olması anlamına gelir. Öznitelik noktaları otomatik olarak algılanır ve Tsai'nin makalesinde [4] açıklandığı gibi, kameranın dahili, harici ve çarpıklık parametrelerinin hesabında sadece bu bilgiler kullanılır. Bilinmeyen

kamera kalibrasyon parametrelerinin kestirilmesi Tsai'nin yöntemine çok benzemektedir. En belirgin fark, bazı derinlik bilgilerini elde etmek için kullanılan kesin devinim gereksiminin bu yöntemde olmamasıdır.



Şekil 2. 4 : 2B kalibrasyon deseni [6]

Kalibrasyon desenini $z = 0$ düzleminde olduğu varsayımı işlemlerin formülasyonunu basitleştiren en önemli etkidir.

2.1.2.3. 1-B Desenlerle Kalibrasyon:

1-B nesnelere kullanılarak yapılan kalibrasyon (Şekil 2.5) diğer kalibrasyon sınıflarına nazaran yaygın olarak araştırılmamıştır. Bu yöntemde [8], sabit bir nokta çevresinde hareket ettirilen doğrusal desen gözlemlenir. Kalibrasyon nesnesinin eş zamanlı olarak gözlemlenmesini gerektiren, birden fazla kameranın kalibrasyonu probleminde, bu yöntem önemli rol oynamaktadır [8].

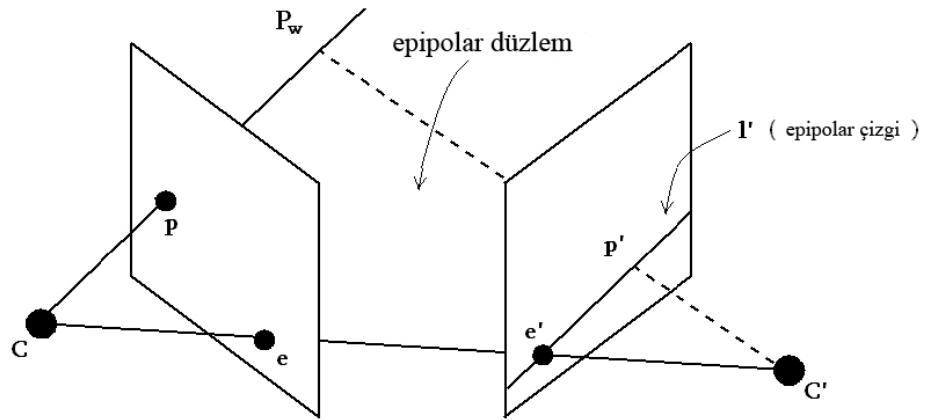


Şekil 2. 5 : 1B kalibrasyon deseni [8]

2.1.2.4. 0-B Desenlerle Kalibrasyon (Kendi Kendine Kalibrasyon):

Kendi kendine kalibrasyonda, herhangi bir desen kullanılmadığını ve sadece farklı görüntüler arasındaki nokta eşleşmelerinden [1, 2, 3, 5, 6] yararlanıldığından dolayı bu yöntemler 0-B yaklaşımı olarak adlandırılmaktadır. Eğer görüntüler sabit dahili parametrelili kamerayla elde edilmişse, üç imge, sahnenin 3-B yapısının geriçatımı için gerekli, dahili ve harici kamera parametrelerini elde etmeye yetecektir.

Kendi kendine kalibrasyon probleminde, eldeki veri, farklı konumlardan çekilmiş görüntülerden ibarettir. Bu problemin çözümü için birçok yöntem geliştirilmiştir. Bu yöntemin öncüleri Maybank ve Faugeras, *Kruppa* olarak adlandırılan, doğrusal olmayan karesel denklemleri kullanarak, temel matris ve bilinmeyen kamera matrislerini elde etmiştir. Bu öncü çalışmadan sonra, Kruppa denklemleri farklı yöntemlerle çözülmeye çalışılmıştır [11, 12, 13, 14, 15]. Kendi kendine kalibrasyonun bir diğer tip yönteminde [16, 17], parametreler, sanal konik ile kamera dahili parametreleri arasındaki ilişki kullanılarak elde edilir. Bu yöntemlerin ileriki adımlarda izdüşümsel geriçatım, metrik geriçatıma dönüştürülür. Bu yaklaşımlardan farklı olarak Pollefeys [18], kalibrasyonu çok aşamalı bir yöntemle çözmüştür. İlk olarak, sahnenin izdüşümsel geriçatılımı oluşturulup, daha sonra bu geriçatım, bir dizi kısıtlamaların [19] çözümüyle tanımlanmış sanal koniğine ait düzlemin konumu kullanılarak, ilgin hale dönüştürülür. Son olarak ilgin geriçatım, kendi kendine kalibrasyon denklemlerinin çözümüyle tanımlanmış kamera dahili parametreleri kullanılarak, metrik hale güncellenir.

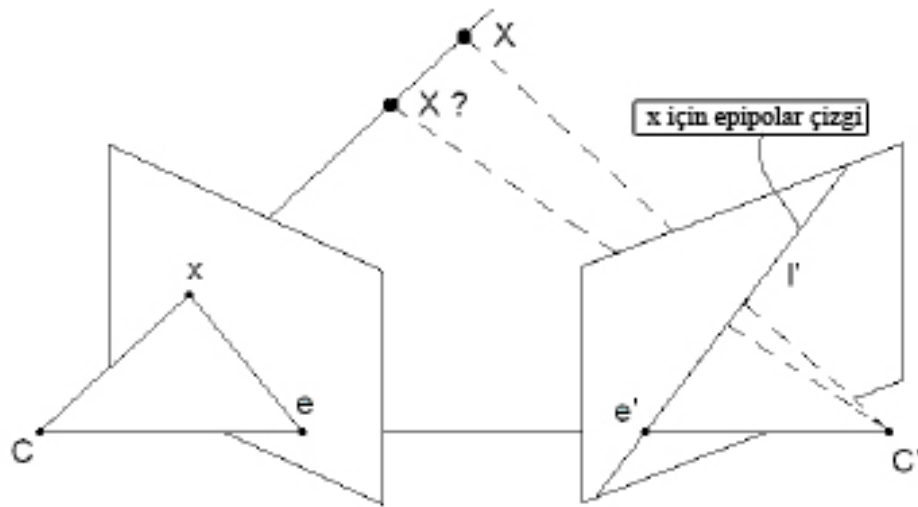


Şekil 2.6: Epipolar Geometri: C ve C' kamera merkezleri. P_w herhangi bir 3B nokta ve p, p' bu noktanın farklı kameralardaki izdüşümleri

2.1.3. Epipolar Geometri

Epipolar geometri, izdüşüm geometrisinden yararlanarak, iki perspektif imge arasındaki geometriyi irdeleyen çalışmadır. Kamera yönelim ve konumlarıyla oluşan yapı içindeki geometrik elemanlar arasındaki ilişkileri ve kısıtlamaları inceler. Şekil 2.6'da görülebileceği üzere, iki kamera merkezi ve 3B nokta arasında oluşan düzleme, *epipolar düzlem* denir. Farklı 3B noktalar için birçok epipolar düzlem mevcuttur. Ancak hepsi kamera merkezleri C ve C' arasında oluşan çizgiden geçer. Bu çizgiye *taban çizgisi* denir. Taban çizgisinin imge düzlemleriyle kesişimleri *epipolar noktalar* (veya *epipoller*) olarak adlandırılır. Bu noktalar kamera merkezlerinin imge düzlemlerine izdüşümleridir.

Epipolün konumu kameranın dahili ve harici parametrelerinin her ikisinin de bağlıdır. Ayrıca, imge düzleminin konum ve yöneliminin değiştirilmesi de epipolün yeniden konumlanmasına neden olur. Epipolar çizgi, imge düzlemiyle epipolar düzlemin kesişimidir. Tüm epipolar düzlemler taban çizgisini içerdiğine göre tüm epipolar çizgiler epipolden geçer. Şekil 2.7'de gözlemlenebileceği üzere, 3B noktanın kesin konumunun sadece bir imge düzlemindeki izdüşümü ve kamera merkezlerini kullanarak belirlemek mümkün değildir. Noktanın derinliği hakkında hiçbir bilgiye sahip olmadığımızdan sadece bu noktayı barındıran çizgi elde edilebilir. Kalibre edilmiş kameralar için, bu noktanın ikinci imgedeki izdüşümünü barındıran çizgi, epipolar geometri yardımıyla bulunabilir.

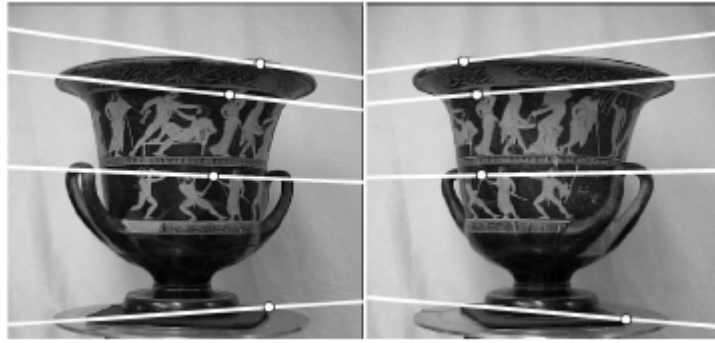


Şekil 2.7 : Geri yansıtılan noktanın belirsizliği: Kalibre edilmiş bir çift kamera için (C ve C' bilinen), 3B nokta X 'i bulmak için sadece x 'i bilmek yeterli değildir.

Bütün epipolar çizgiler epipolden geçtiği için, iki epipolar çizgi verildiğinde, epipolün konumu vektör çarpımıyla kolayca bulunabilir. Farklı kamera yapılarından örnekler şekil 2.8 ve şekil 2.9'da görülmektedir.



Şekil 2.9: Paralel kamera durumu [1]



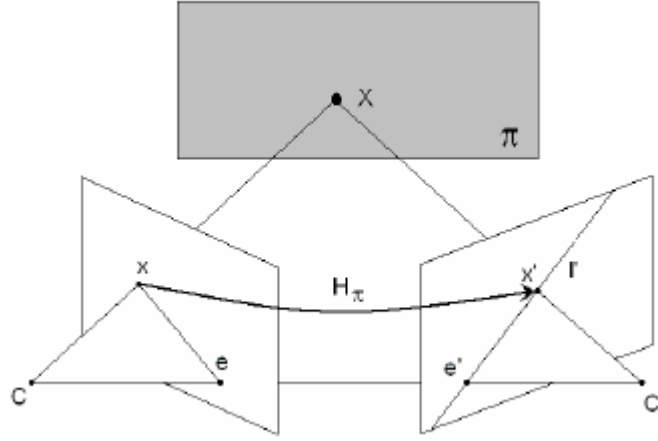
Şekil 2.10: Yakınsayan kamera durumu [1]

2.1.3.1. Temel Matris:

Temel matris, epipolar geometrinin cebirsel bağıntısıdır. Stereo imge çiftinde, birinci imgedeki bir noktayla diğerindeki çizgi arasındaki geometrik dönüşümü tanımlamak için kullanılır. Ayrıca kameranın dahili ve harici parametrelerini içinde barındırır. Önceki bölümde incelediğimiz üzere, birinci imgede verilen bir nokta için ikinci imgede bu noktanın eşini içeren bir çizgi, l' , mevcuttur. Bu çizgi 3-B uzayda kamera merkezinden seçili nokta x 'e doğru saçılan ışının izdüşümüdür.

$$x \rightarrow l' \quad (2.2.1)$$

Bu dönüşüm, temel matris olarak adlandırılan 3x3 bir matrisle gösterilir. Bu matrisin nasıl türetildiğini açıklamak için birinci resimdeki x noktasını bir düzlem aracılığıyla diğerine dönüştürdüğümüzü farzedelim.



Şekil 2.11: Düzlem yardımıyla dönüşüm.

Bu dönüşüm herhangi bir düzlem kullanılarak yapılabilir ve *homografik dönüşüm*, H olarak adlandırılır [1]. Buna bağlı olarak, x noktasının diğer resimdeki homografik tekabülü \tilde{x}' aşağıdaki şekilde elde edilebilir.

$$\tilde{x}' = H_x x \quad (2.2.2)$$

Kamera merkezinden x noktasına doğru olan ışın bozulmadığından, \tilde{x}' , x noktasının doğru eşleşmesini barındıran epipolar çizgi üzerindedir. Buradan epipolar çizgi,

$$l' = e' \times \tilde{x}' = [e']_x H_x x = Fx \text{ burada } F = [e']_x H_x \quad (2.2.3)$$

bağıntısıyla elde edilebilir. F temel matrisdir ve $[]_x$ terimi, matris formunda vektörel çarpımdır. Bu çıkarımdan anlaşılacağı üzere, aynı kamera ayarları için denk temel matrisler sınıfı mevcuttur. Temel matris (2B) noktadan (1B) çizgiye gönderimi ifade etmesi nedeniyle $[]_x$ teriminin ve temel matrisin rankı 2'dir.

Temel matris, izdüşüm matrisleri P ve P' kullanarak ifade edilebilmektedir [20]. Bu amaçla, geriye yansıtılan ışının denklemi aşağıdaki şekilde verilebilir:

$$X(\lambda) = P^\dagger x + \lambda C \quad (2.2.4)$$

Burada, λ herhangi bir pozitif reel sayı, C ilk kameranın merkezi ve P^\dagger ilk izdüşüm matrisinin $P^\dagger P = I$ bağıntısını sağlayan sözde tersidir. $PC = 0$ olduğuna göre,

$$PX(\lambda) = x \quad (2.2.5)$$

bağıntısı elde edilir. Herhangi bir λ değeri için, $X(\lambda)$ bu ışın üzerindeki 3B noktayı belirtir. Bu bilgilere dayanarak noktanın ikinci görüntüye izdüşümü aşağıdaki şekilde elde edilir.

$$x' = PX(\lambda) = P' P^\dagger x + \lambda P' C = P' P^\dagger x + \lambda e' \quad (2.2.6)$$

Bu noktanın epipol ile vektörel çarpımından epipolar çizgi denklemi elde edilir.

$$e' \times x' = e' \times (P' P^\perp x + \lambda e') = [e']_x P' P^\perp x + \lambda e' \quad (2.2.7)$$

Son olarak, F için aşağıdaki bağıntı elde edilir:

$$F = [e']_x P' P^\perp \quad (2.2.8)$$

Temel matris, birinci resimdeki bir noktayı ikinci resimdeki bir çizgi ile ilişkilendirdiğine göre, bu noktaya ait doğru eş, bu çizgi üzerinde olmalıdır. Bu bağıntı aşağıdaki şekilde gösterilir.

$$l' = Fx \quad (2.2.9)$$

$$l'^T x' = x'^T l' \rightarrow x'^T Fx = 0 \quad (2.2.10)$$

Bu bağıntıya *epipolar kısıt* denir ve kameranın dahili ve harici parametreleri bilinmesine gerek kalmadan temel matrisin kestirilmesine olanak sağladığı için çok önemlidir. Sadece nokta benzerliklerini kullanarak temel matrisin kestirilmesini sağlayan birçok algoritma [1, 2] mevcuttur. Temel matris kestirildikten sonra, kamera kalibrasyon matrislerini ve harici parametreleri hesaplamak mümkündür. Temel matrislerin özellikleri kısaca tablo 2.1'de özetlenmiştir.

Tablo 2.1: Temel matrisin özellikleri

- | |
|--|
| <ul style="list-style-type: none"> • F, 7 serbestlik derecesine sahip, rank-2 homojen matristir. • Epipolar kısıt: Eğer x ve x' ilişkili imge noktalarıysa $x'^T Fx = 0$ • Epipolar çizgiler: <ul style="list-style-type: none"> $l' = Fx$ x ile alakalı epipolar çizgi $l = F^T x'$ x' ile alakalı epipolar çizgi • Epipoller: <ul style="list-style-type: none"> $Fe = 0$ |
|--|

$$F^T e' = 0$$

- **F formülasyonları:**

Izdüşüm matrisleriyle: $F = [e']_x P' P^\perp$ burada P^\perp , P' 'nin sözde tersi

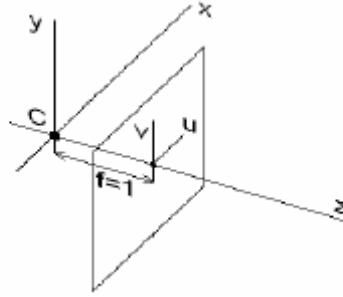
Düzlem ile dönüşüm: $F = [e']_x H_x$ burada H_x homografik dönüşüm

Temel matrisin, literatüre Longuet-Higgins [21] tarafında kazandırılan, normalize edilmiş haline *öz matris* denir. Bazen normalize edilmiş temel matris olarakta adlandırılan bu matris, sadece dönme ve öteleme hakkında bilgileri içerir. Kamera parametrelerinden bağımsız olmasından dolayı normalize edilmiş olarak adlandırılır. 3B X noktasının iki izdüşümünü $x = PX$ ve $x' = P' X$ olarak ele alalım. Normalize edilmiş imge koordinatları kolayca aşağıdaki şekilde bulunur.

$$x_{cap} = K^{-1}x \quad \text{ve} \quad x'_{cap} = K'^{-1}x' \quad (2.2.11)$$

x_{cap} ve x'_{cap} ilgili kalibrasyon matrislerinden bağımsızdır ve yeni izdüşüm matrisleri $K^{-1}P$ ve $K'^{-1}P'$ *normalize edilmiş izdüşüm matrisleri* olarak isimlendirilir. Normalize edilmiş koordinatlar arasındaki matris öz matristir ve aşağıdaki şekilde gösterilebilir[1] :

$$E = t \times R \quad (2.2.12)$$



Şekil 2.12: Normalize edilmiş imge koordinat sistemi

Temel matris ve imge koordinatları arasındaki epipolar kısıt, öz matris ve normalize edilmiş resim koordinatları arasında da mevcuttur:

$$x_{cap}^T E x_{cap} = 0 \quad (2.2.13)$$

Öz matris ve temel matris arasındaki bağıntı aşağıdaki şekilde gösterilir.

$$E = K'^T F K \quad (2.2.14)$$

Tüm kalibrasyon matrisleri hesaplamadan çıkarıldığından dolayı öz matrisin serbestlik derecesi beştir. Dönme ve ötelemenin serbestlik dereceleri üç olmasına rağmen genel ölçekteki belirsizlikten dolayı toplam serbestlik 1 derece düşer.

2.1.4. Üçgenleştirme ve Koordinatların Elde Edilmesi

Önceki bölümlerde, 3B noktanın imge üzerine izdüşümünün, izdüşüm matrisi yardımıyla elde edilmesi açıklanmıştır. Ancak bu problemin tersi, yani resimde verilen nokta kümesinden 3B noktanın konumunun bulunması da, geriçatım probleminin önemli bir parçasıdır. Bu bölümde sadece 2 gözlem veya 2 farklı imgedeki ilişkili koordinatlardan 3B koordinatların elde edildiği, temel üçgenleştirme teorisi kısaca anlatılmaktadır. Bu yöntemin doğruluğu, resim düzlemindeki 2 noktanın düzgün olarak eşleştirilmesine bağlıdır. Nokta eşleşmeleri genel olarak hatalar barındırdığından, 2 nokta yerine bir nokta kümesi kullanılır ve oluşan sistem çözümlenir.

Kullanılan kalibrasyon yöntemine bağlı olarak izdüşüm matrisinin yapısı değişiklik gösterir. Örneğin, Tsai'nin [4] yöntemi kullanıldığında ortaya çıkan izdüşüm matrisi 4 x 4'tür. Bu şekildeki izdüşüm matrisini T olarak ele alalım. Bu durumda 3B noktanın ($X_w, Y_w, Z_w, 1$) izdüşümü aşağıdaki şekli alır.

$$\begin{pmatrix} x_i \\ y_i \\ z_i \\ w_i \end{pmatrix} = T \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \quad (2.2.15)$$

Homojen imge koordinatlarının, w_i ve z_i olarak adlandırılan 2 tane ek ölçekleme katsayısına sahip olduğu görülmektedir. Bununla beraber piksel koordinatlarının elde edilmesinde aşağıdaki homejenleştirme fonksiyonu bağıntısı geçerlidir:

$$H \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix} = \begin{pmatrix} x_i / z_i \\ y_i / z_i \end{pmatrix} \quad (2.2.16)$$

Buradan (x, y) koordinatları,

$$(x, y) = \left(\frac{x_i}{z_i}, \frac{y_i}{z_i} \right) \quad (2.2.17)$$

bağıntısıyla elde edilir. Bu bağıntıda ölçekleme katsayısı w_i kullanılmamıştır. Aynı formülasyon 3x4 kamera izdüşüm matrisinde uygulanabilir.

Bazı durumlarda noktaların kesin konumlarını hesaplanması gerekmektedir. Bu işlem seçili 3B noktaların konumlarının, görünen tüm izdüşümleri kullanılarak çözümlenir.

(2.3.1) denklemini ele alalım, homojenleştirme fonksiyonu içine katıldığında,

$$\begin{pmatrix} x_i \cdot z_i \\ y_i \cdot z_i \\ z_i \\ w_i \end{pmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & T_{14} \\ T_{21} & T_{22} & T_{23} & T_{24} \\ T_{31} & T_{32} & T_{33} & T_{34} \\ T_{41} & T_{42} & T_{43} & T_{44} \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \quad (2.2.18)$$

Yukarıdaki bağıntıyı açtığımızda 4 denklem elde ederiz:

$$0 = T_{11}X_w + T_{12}Y_w + T_{13}Z_w + T_{14} - x \cdot z_i$$

$$0 = T_{21}X_w + T_{22}Y_w + T_{23}Z_w + T_{24} - y \cdot z_i$$

$$0 = T_{31}X_w + T_{32}Y_w + T_{33}Z_w + T_{34} - z_i$$

$$0 = T_{41}X_w + T_{42}Y_w + T_{43}Z_w + T_{44} - w_i$$

Bu denklemler aşağıdaki şekilde tekrar düzenlenebilir.

$$\begin{pmatrix} -T_{14} \\ -T_{24} \\ -T_{34} \\ -T_{44} \end{pmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & -x & 0 \\ T_{21} & T_{22} & T_{23} & -y & 0 \\ T_{31} & T_{32} & T_{33} & -1 & 0 \\ T_{41} & T_{42} & T_{43} & 0 & -1 \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ z_i \\ w_i \end{pmatrix} \quad (2.2.19)$$

Problemde (x, y) noktalarının girdiler olduğu ve 3B noktanın hesaplanması gerektiğini düşünüldüğünde, 5 bilinmeyen olmasına rağmen 4 denklem bulunduğundan tekil çözüm yoktur. Ancak hesaplamaya, elle veya eşleşen pikseller aracılığıyla seçtiğimiz, aynı 3B noktanın 2 farklı imgedeki izdüşümleri olan (x_1, y_1) ve (x_2, y_2) noktaları eklendiğinde, konumu hesaplanmaya çalışılan 3B noktanın koordinatları $(X_w, Y_w, Z_w, 1)$ ortak

olduğundan, sistemde 8 denklem ama 7 bilinmeyen olacaktır. Birinci imge için kamera izdüşüm matrisinin T ve ikinci imge için izdüşüm matrisinin U olsun. Bu durumda aşağıdaki sistem elde edilir.

$$\begin{pmatrix} -T_{14} \\ -T_{24} \\ -T_{34} \\ -T_{44} \\ -U_{14} \\ -U_{24} \\ -U_{34} \\ -U_{44} \end{pmatrix} = \begin{bmatrix} T_{11} & T_{12} & T_{13} & -x^1 & 0 & 0 & 0 \\ T_{21} & T_{22} & T_{23} & -y^1 & 0 & 0 & 0 \\ T_{31} & T_{32} & T_{33} & -1 & 0 & 0 & 0 \\ T_{41} & T_{42} & T_{43} & 0 & -1 & 0 & 0 \\ U_{11} & U_{12} & U_{13} & 0 & 0 & -x^2 & 0 \\ U_{21} & U_{22} & U_{23} & 0 & 0 & -y^2 & 0 \\ U_{31} & U_{32} & U_{33} & 0 & 0 & -1 & 0 \\ U_{41} & U_{42} & U_{43} & 0 & 0 & 0 & -1 \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ z_i^1 \\ w_i^1 \\ z_i^2 \\ w_i^2 \end{pmatrix} \quad (2.2.20)$$

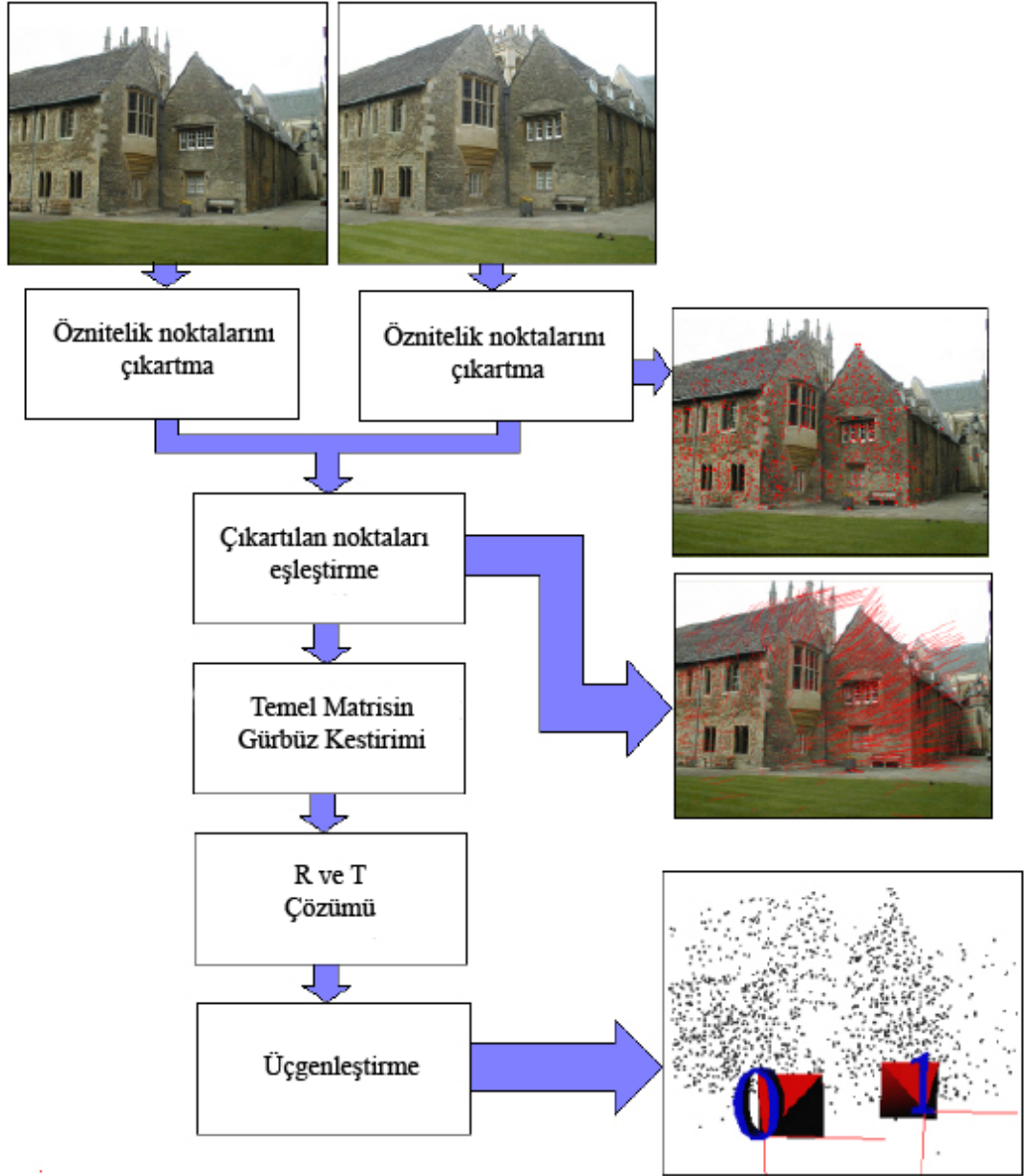
Bu sistem en küçük kareler yöntemiyle çözümlenebilir. Hatalar eşleşme işleminde oluştuğundan, sistemin doğruluğu daha çok gözlem eklenerek artırılabilir.

2.2. DEVİNİMDEN YAPI ELDE EDİLMESİ

Bu bölümde, kalibre edilmiş bir sistemde iki imgedeki seyrek noktaları kullanarak sahne geriçatımını amaçlayan, *devinimden yapı elde etme* algoritmalarının genel yapısı incelenecektir. Seyreklik terimi, geriçatılan sahnenin imgedeki tüm noktaların derinlik bilgilerini içermemesinden ve sadece küçük bir alt kümenin kestirebilmesinden dolayı kullanılmaktadır. Öte yandan, kalibre edilmiş sistem terimiyle, kameranın dahili parametrelerinin önceden bilindiği kastedilmektedir.

Devinimden yapı geriçatım algoritmalarını 4 ana adımda özetleyebiliriz (Şekil 2.13):

- Uygun çiftlerin bulunması
- Görünüşler arasındaki temel matrisin kestirilmesi
- Görünüşlerin birbirlerine göre duruşlarının ve kamera matrislerinin hesaplanması
- Her ilintili nokta çifti için izdüşümü oldukları noktanın 3-B uzayda belirlenmesi



Şekil 2.13: Geriçatım yönteminin adımları

2.2.1. İlişkin Çiftlerin Bulunması

İki imge arasındaki bağıl geometriyi kestirebilmek için bu iki imge arasındaki ilişkin nokta çiftlerini bulmak gerekir. Sahneye ait her imge, diğer karelerle arasındaki bağıl geometriyi kestirmesine olanak sağlayacak birçok bilgi içerir. Ancak, bu bilgileri kestirim problemine yardımcı olacak şekilde elemek gerekir.

Bu amaçla imgenin bariz temel öğeleri olan *öznitelikler* çıkarılır. Birçok farklı nokta seçilebilmesine rağmen, genel olarak “bariz temel öğe” olarak resimdeki köşeler kullanılır. Bu köşelerin 2 boyutlu konumuna, *öznitelik noktası*, 3-B konumuna, *nesne noktası* denir.

İlişkin çift, aynı nesne noktasının farklı imgelerdeki izdüşümleri olan, öznitelik noktalarından oluşan çifttir. İlişkin çiftleri bulma problemi, verilen pikselin eşinin diğer imgelerde kestirilmesidir. Çoğu durumda eldeki tek girdi imgelerin yeğlilik bilgileridir ve bu yeğlilik haritası kullanılarak aranan pikselin konumu bulunmalıdır. Ancak piksellerin geçirdiği dönüşümlerden dolayı bu işlemin apaçık bir çözümü yoktur. Bu dönüşümlerden bazıları; döndürme, öteleme, ölçek değişiklikleri, ilgin dönüşümler, aydınlık ve kamera değişimlerinden kaynaklanan yeğlilik değişiklikleridir. Bölüm 2.2’de incelediğimiz üzere ilişkin çift kümesinden temel matrisi, yani incelenen imge çifti arasındaki geometrik bağıntıları elde etmek mümkündür. Bu amaçla ilk olarak imgelerdeki öznitelik noktalarının bulunması gerekmektedir.

2.2.1.1. *Öznitelik Noktalarının Sezimi*

Öznitelik noktaları bulmak için birbirinden farklı birçok yöntem kullanılmaktadır [22, 23, 24, 25, 26, 27]. Örneğin, Harris ve Stephens [22] tarafından sunulan yöntem, imge gradyanlarından yararlanmaktadır. Bu yöntem, aydınlanmadaki değişimlere ve öteleme farklarına karşı duyarsız olduğundan yaygın olarak tercih edilmektedir ve resimler arası kamera hareketinin az olduğu durumlarda iyi sonuçlar vermektedir. Öte yandan, Mikolajczyk ve diğ. [23], daha komplike bir öznitelik sezici sunmuşlardır. Yöntemleri, ölçek değişimi gibi ilgin dönüşümlere duyarsızdır ve farklı ölçeklerde, gelişigüzel hareket koşullarında, bu koşullardan bağımsız sonuçlar verebilmektedir. Ancak, bu yöntem diğerlerine göre daha fazla hesaplama süresi gerektirdiğinden video dizisi gibi gelişigüzel olmayan girdilerle kullanılması mantıklı değildir. Yapılan araştırmaların gösterdiği üzere [28], Harris köşe sezicisinin, diğer öznitelik sezicilere göre daha tutarlı sonuçlar vermektedir.

2.2.1.2. *Nokta Eşleşmelerinin Bulunması*

İki imgedeki bariz öznitelikler çıkarıldıktan sonra, aralarındaki ilişkiyi bulmak için bir yordam gerekir. Bu problem literatürde eşleştirme (ilişkilendirme) olarak geçmektedir.

Problemin çözümü için birçok algoritma önerilmiştir. Bunlardan en basit olanı *ilinti-bazlı eşleştirme* [29] algoritmasıdır. Bu yöntemde öznitelikler, belli bir mesafedeki diğer pikseller ile aralarındaki ilinti derecesine göre eşleştirilirler. Ancak bu yöntem az sayıda aykırılık seviyesi barındıran resimlerde kullanılabilir ve genel görüntüler için pek uygun değildir. Yöntemi geliştirmek amacıyla eşleşme adaylarına bazı ek kısıtlamalar önerilmiştir [30, 31, 32]. Örneğin *Komşuluk kısıtı* [30], hatalı eşleşmeleri engellemek için kullanılan sınırlamalardan biridir. Bu tip eşleştirmelerde sonuca, eşleşmenin doğruluğunu arttıracak, komşu piksellerin eşleşme durumlarını hesaba katan, ek bir derece kullanarak ve devamında *gevşeme yordamıyla* [33] ulaşılır. Bu yöntemler küçük veya orta taban çizgisi mesafelerinde gerçekten başarılıdır [29].

2.2.2. Temel Matrisin Kestirilmesi

Daha önceki bölümlerde açıklandığı gibi, temel matris, iki perspektif görüntü arasındaki geometrik ilişkiyi tanımlayan cebirsel bağıntıdır. Stereo imge çiftinde, bir imgedeki noktanın, diğerindeki çizgiye olan eşleşmesini göstermek için kullanılır. Bu bağıntı, imge çiftindeki tüm ilişkili çiftler için sağlanmalıdır. Bu nedenle eşleşmelerin tutarlılık ölçüsü olarak da kullanılabilir.

Bilindiği üzere temel matris sahnedeki ilintili nokta çiftlerini kullanarak hesaplanabilir. Geçtiğimiz on yıllık sürede, birçok araştırmacı öznitelik noktalarının eşleşmelerinden temel matrisin çözümü sağlayacak, farklı algoritmalar önermişlerdir. Bu algoritmalar kabaca üç yaklaşımda gruplanabilir; *lineer kestirim yaklaşımı*, *döngülü kestirim yaklaşımı* ve *gürbüz kestirim yaklaşımı* [1, 2]. Bu yaklaşımlar kısaca bölümün devamında incelenmiştir.

2.2.2.1 Lineer Kestirim Yaklaşımı

Lineer kestirim yönteminde, temel matrisi kestirmek için, ilişkili çiftlere ait lineer denklemler alt alta yazılır ve oluşan homojen lineer sistem çözümlenir. Oluşan sistem *artık* belirtilmiştir. Bu nedenle, oluşan homojen sistemden temel matrisi hesaplayabilmek için yedi-nokta [2] ve sekiz-nokta yöntemleri [5, 6, 34] önerilmiştir. Lineer kestirim yönteminin avantajları uygulama açısından basit olması ve hesaplamadaki verimliliğidir. Ancak nokta eşleşmelerinden doğacak gürültüye karşı çok hassastır. Bu problem özellikle ilişkili çiftler arasında aykırı değer olduğu zaman çok ciddi bir hal almaktadır. Pratikte, otomatik öznitelik çıkarma ve eşleştirme işleminde

ortaya çıkan tutarsızlıklar engellenemez. Zira bu problemler, günümüzün uygulanabilen sistemlerinin ortak sorunudur.

2.2.2.2. Döngülü Kestirim Yaklaşımı

Döngülü kestirim yaklaşımı [2, 5], öznitelik noktalarının koordinatları ve temel matris yardımıyla, belirlenen epipolar çizgiler arasındaki geometrik mesafelerin toplamının asgari düzeye indirilmesini amaçlar. Mesafe fonksiyonun lineer olmayan yapısından dolayı, fonksiyonun asgari düzeye indirilmesi, temel matrisin döngü içinde güncellenmesiyle sağlanır. Mesafe fonksiyonuna, *temel nümerik yöntem ve kısıtlı temel nümerik yöntem* [2] gibi bazı varyantlar önerilmiştir. Bu yaklaşımın avantajı, anlamlı bir geometrik ölçümle direkt olarak bağlantısı olmasıdır. Öte yandan zayıflığı da, özellikle veride aykırı çiftler olduğunda, gürültüye duyarlılığıdır.

2.2.2.3. Gürbüz Kestirim Yaklaşımı

Aykırı verilerin temel matris kestirmine etkilerini azaltmak için gürbüz kestirim yaklaşımı önerilmiştir. Temel matrisin kestirilmesi için birçok farklı gürbüz kestirim yöntemi kullanılmaktadır. Önde gelen üç gürbüz kestirimci; M-kestirimcisi [2, 6, 10, 35], LMedS (Least Median of Squares) [1, 6, 10] ve RANSAC (RANdom SAMple Consensus)[4, 5, 6, 22] kestirimcileridir. Gürbüz kestirim yaklaşımı, yanlış nokta eşleşmelerinden kaynaklanan problemlere ve verideki gürültüye karşı başarılı olabilmesine rağmen diğer iki yaklaşıma göre çok daha fazla hesaplama süresi gerektirmektedir. Sağladığı avantaja karşın gürbüz kestirimin bazı kısıtlamaları da vardır. Örneğin, M-kestiricisi [6], başlangıçtaki temel matris tahmininin başarılı şekilde yapıldığı veya düşük aykırı veri yüzdesiyle çalışıldığında verimli olur. RANSAC algoritmasında [4, 5], çoğu zaman elimizde olmamasına rağmen, aykırı nokta yüzdesinin bilinmesi gerekmektedir. LMedS [1] kestirimi bu şekilde bilgiler gerektirmez ama algoritma hesabı çok uzun zaman alır.

2.2.3. Duruşların ve Kamera Matrislerinin Kestirilmesi

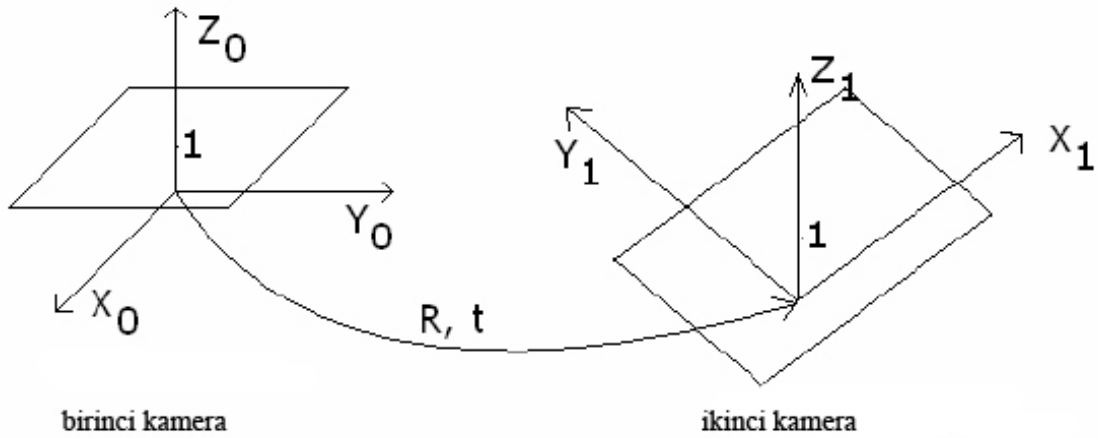
Aynı sahneye ait iki farklı görünüş arasındaki ilişki, 3x3 tekil matris ile tanımlanan epipolar geometriyle gösterilir. Eğer kamera dahili parametreleri önceden biliniyorsa resim koordinatları, normalize edilmiş resim koordinatlarına [1, 27] dönüştürülebilir ve aralarındaki ilişki öz matris [36, 27] ile tanımlanır. Aksi durumda ilişkiyi tanımlamak için temel matris [1] kullanılır. Temel matris ve öz matris arasındaki ilişkiyi [35]:

$$E = K^T F K \quad (2.3.1)$$

Burada K , kamera kalibrasyon matrisidir. Bu denkleme bağı olarak ilişki çiftler için normalizasyon işlemi aşağıdaki şekilde tanımlanabilir.

$$m'_E = K^{-1} m' \text{ ve } m_E = K^{-1} m \quad (2.3.2)$$

Burada m ve m' , birinci ve ikinci imgedeki gerçek koordinatlar, m_E ve m'_E ise, ikinci kamera matrisine göre izdüşümü alınmış normalize koordinatlardır. Eğer birinci kamera çerçevesi genel koordinat sistemi olarak seçilirse, döndürme matrisi R ve öteleme matrisi t , ikinci kamera çerçevesinin birinciye göre dönüşümüyle gösterilebilir. (Şekil 2.14)



Şekil 2.14: Bağlı kamera konumları

Bu suretle, birinci kamera koordinat çerçevesindeki herhangi bir nokta $M = [M_x, M_y, M_z]^T$, ikinci imge çerçevesindeki $M' = [M'_x, M'_y, M'_z]^T$ noktasına aşağıdaki bağıntı kullanılarak dönüştürülebilir:

$$M' = RM + T \quad (2.3.3)$$

Noktaların, normalize edilmiş kamera modeline göre, imge düzlemlerine izdüşümleri alındığında, koordinatlar aşağıdaki şekilde yazılır.

$$m_E = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} M_x/M_z \\ M_y/M_z \\ 1 \end{bmatrix}, m'_E = \begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} M'_x/M'_z \\ M'_y/M'_z \\ 1 \end{bmatrix} \quad (2.3.4)$$

(2.3.3) ve (2.3.4) birleştirdiğinde,

$$M'_z m'_E = M_z R m_E + t \quad (2.3.5)$$

elde edilir. Eğer $\|t\| \neq 0$ alınırsa denklem,

$$\frac{M'_z}{\|t\|} m'_E = \frac{M_z}{\|t\|} R m_E + t_0 \quad (2.3.6)$$

şeklini alır. Burada $t_0 = t / \|t\|$ olarak tanımlanmıştır.

n tane ilişkili nokta (m_i, m'_i) verildiğinde, döndürme matrisi R hesaplanabilir. Ek olarak, eğer öteleme vektörü t kaybolmuyorsa, t_0 ile gösterilen öteleme yönü kestirilebilir. Sadece öteleme vektörünün yönü belirlenebildiği için, noktanın 3B koordinatlarını kesin olarak hesaplamak mümkün değildir. Bu duruma “ölçekleme belirsizliği” denir ve R ile t_0 kestirildikten sonra sahnenin sadece ölçeklenmiş bir versiyonunun elde edilebileceği anlamına gelir. (2.3.3) denklemde, M' , RM ve t düzlemdeştir. Dolayısıyla, $t \times RM$, M' ile diktir.

$$M'(t \times RM) = 0 \quad \text{burada} \quad E \equiv t \times R \quad (2.3.7)$$

2.2.3.1. R ve t 'yi belirlemek için lineer algoritma

(2.3.7) denklemde gösterildiği üzere E, t ve R 'nin vektörel çarpımıdır. Bu denklem tekrar düzenlendiğinde aşağıdaki bağıntı elde edilir:

$$E = [e_1 \quad e_2 \quad e_3] = [k\hat{t} \times r_1 \quad k\hat{t} \times r_2 \quad k\hat{t} \times r_3] \quad (2.3.8)$$

burada \hat{t} birim vektördür ve t ile aynı yöndedir. k, t'nin bilinmeyen büyüklüğü ve r_i 'ler döndürme matrisi R'nin kolon vektörleridir. (2.3.8) denklemden aşağıdaki gösterim [52] elde edilir.

$$\hat{t} \perp e_1, \hat{t} \perp e_2 \quad \text{ve} \quad \hat{t} \perp e_3 \quad (2.3.9)$$

Bundan ,

$$\hat{t} = \pm \frac{e_i \times e_j}{\|e_i \times e_j\|} \quad i \neq j, i, j = 1, 2, 3 \quad \& \quad k^2 = 0.5 * (e_1^2 + e_2^2 + e_3^2) \quad (2.3.10)$$

Son olarak, biraz vektör cebirinden sonra, döndürme matrisi elde edilir [27] :

$$r_1 = \left[\frac{1}{k^2} \hat{t} (e_2 \times e_3) \right] \hat{t} + \frac{1}{k} (e_1 \times \hat{t}) \quad (2.3.11)$$

Benzer şekilde r_2 ve r_3 elde edilebilir. Ancak bu yaklaşım hatalara karşı aşırı şekilde hassastır ve bu özelliği pratikte çoğu zaman kullanışsız olmasına neden olur.

2.2.3.2. R ve t'yi belirlemek için gürbüz algoritma

E matrisinin t vektörüne dik olduğu (2.3.9) bilindiğine göre aşağıdaki ilişki geçerlidir.

$$E^T \hat{t} = 0 \quad (2.3.12)$$

Ancak E matrisinin kestirilmesindeki gürültünün varlığından dolayı, aşağıdaki denklem için sonuç aramak daha gerçekçidir.

$$\min_t \|E^T \hat{t}\| \quad \|\hat{t}\| = 1 \text{ olacak şekilde} \quad (2.3.13)$$

Bilindiği üzere $\min_x \|Ax\|$ sisteminin $\|x\| = 1$ 'e göre optimizasyonu, en küçük özdeğer ile ilişkili özvektördür [35]. Bu yüzden \hat{t} , $E^T E$ 'nin en küçük özdeğerine ait özvektör olarak tanımlanabilir.

Gürültü varlığında döndürme matrisi, $\min_R \|R^T [-t_s]_x - E^T\|$ 'nin "R'nin döndürme matrisi" olmasına göre optimize edilmesiyle bulunabilir. Minimizasyon uygulamak yerine "quaternion" gösterimi [27] kullanılarak çözüme ulaşmak mümkündür. Bu durumda A matrisi B,

$$B = \sum_1^3 B_i^T B_i \quad (2.3.14)$$

burada $B_i = \begin{bmatrix} 0 & (C_i - D_i)^T \\ (D_i - C_i) & [D_i + C_i]_x \end{bmatrix}$, $C = [-t_s]_x$ ve $D = E^T$

B matrisinin minimum özdeğeri ile ilişkili özvektör (q), optimal “quaternion” dur ve bu değer kullanılarak R aşağıdaki şekilde [27] bulunabilir.

$$R = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2*(q_1q_2 - q_0q_3) & 2*(q_1q_3 + q_0q_2) \\ 2*(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2*(q_2q_3 - q_0q_1) \\ 2*(q_1q_3 - q_0q_2) & 2*(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.3.15)$$

burada $q = [q_0 \quad q_1 \quad q_2 \quad q_3]^T$

Lineer algoritma teorik olarak doğru olmasına rağmen, E matrisinin kestirilmesindeki gürültü nedeniyle her zaman doğru sonuçlar vermemektedir. Bu yüzden sahne geriçatımları için gürbüz kestirim tercih edilmektedir.

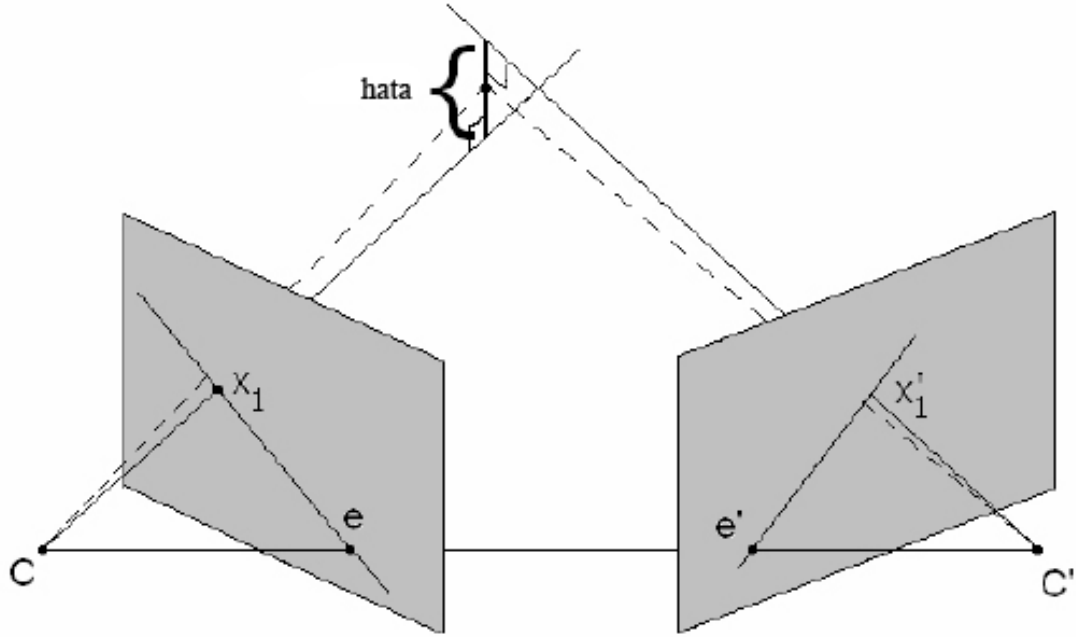
2.2.3. 3B Noktaların Elde Edilmesi

Yapıda geriçatımın en önemli adımlarından biri, ilişkili 2-B noktalardan, 3-B noktanın konumunun kestirildiği üçgenleştirme adımıdır. Bu bölümde, iki imgedeki izdüşümleri ve ilgili kamera izdüşüm matrisleri verilmiş noktaların, 3-B koordinatlarını hesaplamak için kullanılan yöntemler anlatılmaktadır. İşlemlerde, temel matrisin yüksek doğrulukla kestirildiği ve ilgili nokta çiftlerinin hatalar içerdiği varsayılmaktadır. Bu varsayımlar altında, geri yansıtılan ışınların genellikle tek bir 3-B noktada kesişmesi beklenemez.

Bu nedenle basit bir üçgenleştirme yöntemi doğru sonuçlar vermeyecektir ve nokta konumunu kestirmek için gürültüye dayanıklı yöntemler kullanılmalıdır. Gürültünün dışında kamera kalibrasyon parametreleri de her zaman bilinmeyebilir. Bu durumda otomatik kalibrasyon için gereken verileri elde etmek amacıyla, izdüşümsel (veya ilgin) değişimsiz derinlik değerleri gerekmektedir [1, 37]. Üçgenleştirme yönteminin bir diğer önemli özelliği, değişimsiz izdüşümsel (veya geriçatım yöntemine göre ilgin) olmasıdır. Üçgenleştirme yöntemleri 4 ana sınıfta incelenebilir; orta-nokta yöntemi[39], lineer yöntemler [2], dögüsel yöntemler [38] ve son olarak, polinomsal üçgenleştirme yöntemi [38].

2.2.3.1. Orta-Nokta Yöntemi:

Üçgenleştirme yöntemlerinde popüler bir yaklaşım olan orta-nokta yöntemi, kısaca, eşleşen noktaların geri yansıtılan ışınları arasındaki dikmenin orta noktasının seçimi olarak açıklanabilir (Şekil 2.15) [39]. Bu yöntem, diklik kavramının ilgin, orta nokta kavramının izdüşümsel olmaması nedeniyle izdüşümsel ve ilgin dönüşümlerde kötü sonuçlar vermektedir. Bu nedenle sadece Öklid geriçatım problemlerinde kullanılabilir.



Şekil 2.15 : Orta-nokta yöntemi

Noktaların ışınlarla geri yansıtılması, iki nokta kullanılarak hesaplanabilir: kamera merkezi C ve nokta P^+x . P^+ , P matrisinin sözde tersidir. Sözde ters $P^+ = P^T(PP^T)^{-1}$ denklemiyle hesaplanır. Ayrıca $P^+P = I$ eşitliğide geçerlidir. P^+x noktası, imge noktası x 'in izdüşümü ifade ettiğinden ışının üstünde olmalıdır. Bu iki noktanın birleşimi ışını oluşturur:

$$X(\lambda) = P^+x + \lambda C \quad (2.3.16)$$

İki ışında elde edildikten sonra, çizgilerin birbirine en yakın olduğu noktadaki dikmenin orta noktası, çözüm olarak alınır.

2.2.3.2. Linner üçgenleştirme yöntemi:

Uygulama kolaylıklarından dolayı lineer üçgenleştirme yöntemi[1, 37] en çok kullanılan yöntemdir. $m = PM$ izdüşüm matrisini ele alalım. Burada $m = w(u \ v \ 1)^T$, (u

v) gözlenen noktalar ve w , bilinmeyen ölçekleme katsayısıdır. Eğer izdüşüm matrisinin i 'nci satırı p_i^T olarak belirtirsek, $m = PM$ aşağıdaki şekilde yazılabilir.

$$\begin{aligned} wu &= p_1^T M \\ wv &= p_2^T M \\ w &= p_3^T M \end{aligned} \quad (2.3.17)$$

Bu denklemler tekrar düzenlenirse,

$$\begin{aligned} (up_3^T - p_1^T)M &= 0 \\ (vp_3^T - p_2^T)M &= 0 \end{aligned} \quad (2.3.18)$$

Diğer imgede bulunan, m ile ilişkili pikselden (2.3.18) denklemine benzer bir denklem elde edilir. Oluşan denklem kümesi incelendiğinde problemin $AM = 0$ olacak şekilde M 'i bulmaya dönüştüğü görülür.

$$A = \begin{bmatrix} up_3^T - p_1^T \\ vp_3^T - p_2^T \\ u'p_3'^T - p_1'^T \\ v'p_3'^T - p_2'^T \end{bmatrix} \quad (2.3.19)$$

2.2.3.3. Lineer-Özdeğer Yöntemi:

$AM = 0$ probleminin çözümü, A matrisdeki gürültüden dolayı tam olarak bulunamaz. Bu nedenle bir maliyet fonksiyonu tanımlanmalıdır. Lineer-Özdeğer yönteminde, M , $AM = 0$ denkleminde $M = 1$ 'e göre optimizasyon yapılarak bulunur. Bu problemin çözümü $A^T A$ [35] matrisinin özdeğerine ait birim özvektördür. Bu yöntemin uygulanması kolay olmasına rağmen, izdüşümsel ve ilgin geriçatımlar için uygun değildir. Bu durumu gözlemlemek için, P ve P' kamera matrislerini, PH^{-1} ve $P'H^{-1}$ matrislerine dönüştürecek H dönüşümü yapılır. Bu durumda A , AH olur ve M noktası kendisiyle aynı hata oranına sahip HM noktasına tekabül eder ($AM = e$ ve $AH^{-1}HM = e$). Buna rağmen, $M = 1$ koşulu izdüşümsel veya ilgin dönüşümlerde değişimsiz değildir. Bu sorun, lineer-özdeğer yönteminin izdüşümsel veya ilgin dönüşümlerde değişimsiz olmasını engeller.

2.2.3.4. Lineer En Küçük Kareler Yöntemi:

Lineer en küçük kareler yönteminde, $AM = 0$ denklemi, M vektörünün dördüncü parametresi bire eşitlenerek çözülür. Bu yaklaşımla, $AM = 0$ bağıntısı, “4 denklem 3 bilinmeyen” problemine dönüştürülür ve sonuç, tekil değer ayrışımı (SVD) veya sözdeters gibi yöntemlerle elde edilebilir [40]. Bu yöntemde, çözümün sonsuzda olmadığı varsayıldığından dolayı, dördüncü parametre 1 olarak ayarlanmıştır. Bu yaklaşım, noktaların sonsuzdaki düzlemde olabileceği, izdüşümsel geriçatım için problem oluşturmaktadır. Sonsuzdaki noktaların dışında, $[x, y, z, 1]^T$ ‘in izdüşümsel dönüşümün H dönüşümü altında değişimsiz olmaması da, uygunluğunu engelleyen başka bir nedendir. Öte yandan ilgin dönüşüm, sonsuzdaki düzlemi değiştirmedeği için, bu dönüşüm altında $[x, y, z, 1]^T$ değişimsiz olacağından, lineer en küçük kareler yöntemi ilgin değişimsizdir.

2.2.3.5. Döngülü Lineer Üçgenleştirme Yöntemi:

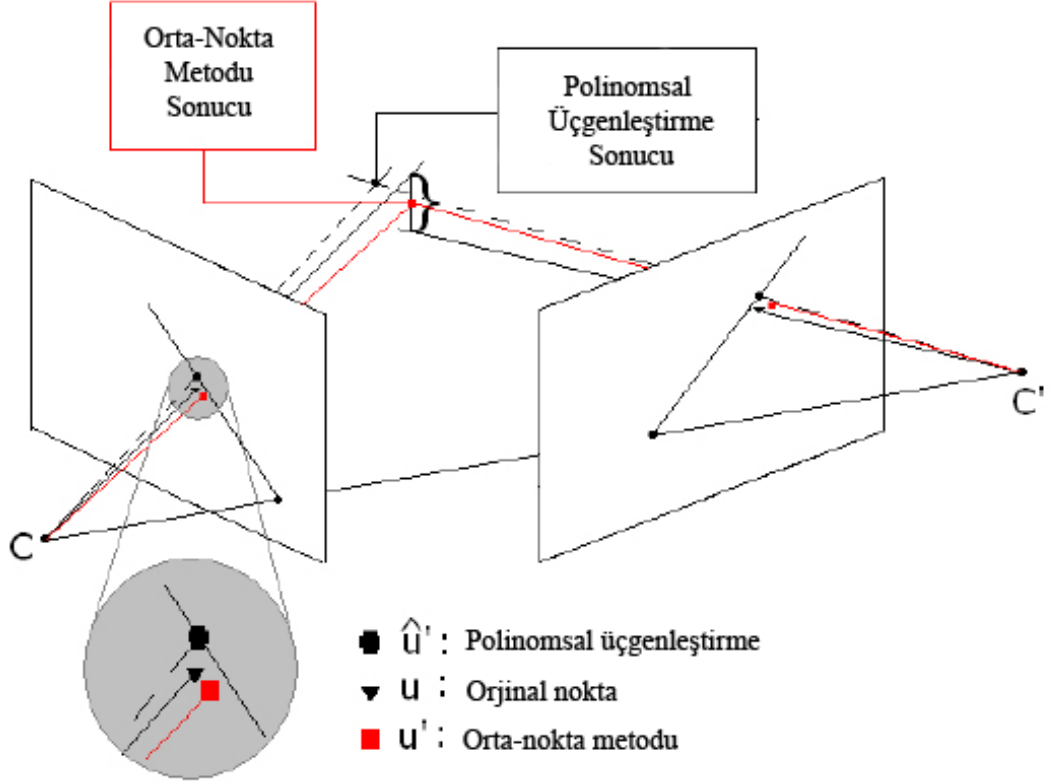
$\|AX\|$ optimizasyonuna dayalı lineer üçgenleştirme yöntemlerinin geometrik manası yoktur. Bu nedenle sonuçlarda bazı tutarsızlıklar oluşabilir. Ancak döngülü lineer üçgenleştirme yönteminde olduğu gibi, A matrisindeki (2.3.19) satırlar ağırlıklandırılarak, daha iyi bir sonuç elde edilebilir [38]. Bu şekilde uyarlanmış A matrisi geometrik hata fonksiyonunun ölçüsünü verir.

Döngüsel olarak, düzgün bir şekilde A matrisinin ağırlıklandırılmasının, maliyet fonksiyonunun optimizasyonuna denk geldiği gösterilebilir. $BAX = 0$ çözümü için lineer-özdeğer veya en küçük kareler yöntemlerinden biride kullanılabilir. Bu durumda yöntemler sırasıyla *döngülü özdeğer* ve *döngülü en küçük kareler* olarak adlandırılır. Bu yöntemlerle ilgili detaylar [38]’te açıklanmıştır. Levenberg-Marquardt [41] gibi diğer döngülü yöntemlerin aksine, ayrı bir ilklendirme algoritması gerektirmemesi ve basit bir durma kriterine sahip olması, bu yöntemlerin çok daha kolay uygulanabilir olmasına olanak sağlar. Ancak döngü içeren birçok algoritmada olduğu gibi yakınsamayı garanti etmez. Ayrıca [38]’te gösterildiği gibi izdüşümsel değişimsiz değildir.

2.2.3.6. Polinomsal üçgenleştirme yöntemi

Gürültülü nokta çiftleri, epipolar kısıtı genel olarak karşılamaz ve geri yansıtılan ışınlar 3-B nokta oluşturamaz. Ancak[38]’te gösterildiği üzere, geri yansıtma hatasını minimize edecek bir maliyet fonksiyonu belirlenerek en iyi çözüme ulaşılabilir. Ayrıca

Levenberg Marquardt gibi lineer olmayan bir optimizasyon fonksiyonu kullanarak sonuç elde etmek mümkündür. Problem tekrar formüle edilip, polinomsal üçgenleştirme yöntemi yardımıyla, döngüsel olmayan bir şekilde geri yansıtılma hatası minimize edilebilir.



Şekil 2.16 : Polinomsal Üçgenleştirme (PÜ): PÜ, epipolar çizgi demeti üzerindeki en yakın noktaları bulur ve 3B noktanın koordinatlarını bu noktalar aracılığıyla kestirir. Orta-nokta yöntemi (OM), geri yansıtılan çizgilerdeki en yakın noktaların orta noktasını seçerek 3B hatayı minimize eder.

Bu yöntemde problem, epipolar çizgi demetinin parametrize edilmesiyle birlikte, bir değişkenli 6. derece bir polinomun köklerinin bulunmasına indirgenmiştir. İlk olarak verilen nokta eşleşmelerine en yakın epiplar çizgiler bulunur. Daha sonra bu çizgiler üzerindeki eşleşen noktalara en yakın noktalar seçilir ve daha sonra seçilen bu noktalar yardımıyla 3B nokta hesaplanır. Bu noktalar epipolar kısıtı sağladığı için geri yansıtılan ışınlar tek bir noktada kesişir.

Bu yöntem izdüşümsel ve ilgin dönüşümlerde değişimsiz bir maliyet fonksiyonunu minimize ettiğinden, izdüşümsel ve ilgin değişimsizdir. Ayrıca, bu yöntemin gaus gürültü modeli varsayımında en iyi sonucu verdiği kanıtlanmıştır [38].

2.3. STEREO BAĞINTI YÖNTEMİ

İki gözümüzle algıladığımız görüntüleri kaynaştırıp, aralarındaki farkları (veya aykırılıkları) saptayabilmemiz, bize güçlü bir derinlik algısı kazandırır. Stereopsis olarak bilinen yöntemler, bu algının bilgisayar ortamında taklit edilmesini amaçlar ve günümüzün en çok araştırılan bilgisayarla görme konularından biridir. Bu araştırmaların sağlıklı ve hızlı bir şekilde ilerlemesinde, Scharstein ve Szeliski tarafından 2002 yılında sunulan sınıflandırma ve bu yayın sonrasında hayata geçirdikleri internet sitesi üzerinden, araştırmacılara gelişmeleri takip etme ve yöntemlerinin başarılarını ortak bir test zemininde karşılaştırma şansı vermelerinin payı büyüktür.

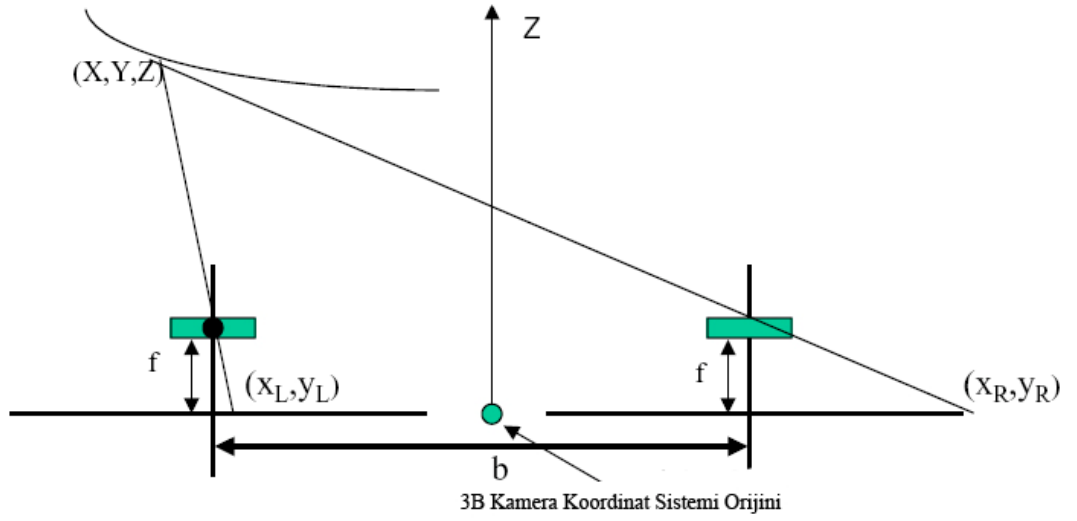
Günümüzdeki yöntemler birçok yöntemi içinde barındırmasına rağmen, Szeliski'nin sınıflandırmasındaki temel bölümler halen geçerliliğini korumaktadır. Çalışmanın yapıldığı dönemden farklı olarak, araştırmacılar önerdikleri yöntemleri, imge çiftlerindeki örtülmüş bölgelerden kaynaklanan hataları giderecek şekilde güçlendirmişlerdir. Ayrıca algoritmaların genel başarısına yaptığı olumlu katkıdan dolayı, bölütleme tekniği yaygın olarak kullanılmaya başlanmıştır.

Bu bölümde stereo bağıntı yöntemlerinin temel elemanları, örtülmüş bölge kavramı ve üzerlerinde yapılmış çalışmalar kısaca anlatılmaktadır.

2.3.1. Aykırılık Haritası

Stereo bağıntıların amacı aykırılık haritasının çıkarılmasıdır. Aykırılık, görüntülerde aynı nesneye ait konumlar arasındaki fark olarak tanımlanabilir. Geometrik olarak bir tanım çıkarmak amacıyla şekil 2.17'de görülen ideal stereo geometriyi ele alalım.

İdeal stereo geometride, kameraların optik eksenleri paralel ve lenslerin merkezlerini bağlayan çizgiler bu optik eksenlere diktir. Optik eksenler taban çizgisi, b ile ayrılmıştır ve 3B koordinat sistemi, kameralar arasına ortalanmış "Cyclopean" sistemdir.



Şekil 2.17 : İdeal stereo geometri

P noktasının sol ve sağ kamera sistemindeki karşılıkları aşağıdaki gibidir.

$$(X_L, Y_L, Z_L) = (X - b/2, Y, Z) \quad (2.4.1)$$

$$(X_R, Y_R, Z_R) = (X + b/2, Y, Z)$$

Buradan P noktasının izdüşümlerinin x koordinatları,

$$x_L = (X + b/2)f / Z \quad (2.4.2)$$

$$x_R = (X - b/2)f / Z$$

olur ve ikinci denklemi biriciden çıkartıp, Z için çözdüğümüzde aşağıdaki denklemi elde ederiz.

$$Z = bf / (x_L - x_R) \quad (2.4.3)$$

Aynı şekilde X ve Y noktaları içinde çözümler bulunabilir.

$$X = b(x_L + x_R) / 2(x_L - x_R) \quad (2.4.4)$$

$$Y = by / (x_L - x_R)$$

Burada $x_L - x_R$ aykırılık, d, olarak adlandırılır ve her zaman negatiftir. X, Y ve Z için denklemleri tekrar düzenlediğimizde,

$$X = (b(x_L + x_R) / 2) / d, \quad Y = by / d, \quad Z = bf/d \quad (2.4.5)$$

elde edilir.

Bu denklemlerden anlaşılacağı üzere mesafe, aykırılığın mutlak değeriyle ters orantılıdır. Aykırılık değerinin sıfır olması, noktaların kameradan sonsuz uzaklıkta olduğu anlamına gelir. Ayrıca aykırılık değeri, taban çizgisi uzunluğu b ile doğru orantılıdır. Taban çizgisinin uzunluğu arttıkça hesaplanabilen mesafe artar ancak resmin ortak görüş alanı daralır.

Geometrik tanımından anlaşılacağı üzere, aykırılık hesabını doğru yapabilmek için geometrinin ideal olması şarttır. Ancak pratikte bu pek mümkün değildir. Bu nedenle, stereo bağıntı uygulanmadan önce doğrultma [42] işlemi yapılır. Diğer tüm stereo bağıntı algoritmalarında olduğu gibi bu tez kapsamında da görüntülerin doğrultulmuş olduğu varsayılacaktır.

2.3.2. Eşleşme Maliyet Fonksiyonları

Önceki bölümde sunulan geometriden anlaşılacağı gibi, aykırılığı hesaplamak için birinci görüntüdeki noktanın diğerindeki eşini bulmak yeterlidir. Eşleşmeyi bulmak için sıklıkla kullanılan yöntem, noktaların birbirine uyumunu sayısallaştıracak bir maliyet fonksiyonunun kullanılmasıdır. Bu amaçla birçok maliyet fonksiyonu önerilmiştir [43]. Bu yöntemlerden en klasikleri; sapma karelerinin toplamı (SSD), çapraz-ilinti (NCC) ve mutlak farkların toplamı (SAD) yöntemleridir.

$$SSD = \sum_{i=1}^n \sum_{j=1}^n [M(i, j) - N(i, j)]^2 \quad (2.4.6)$$

$$C = \frac{\sum_{i=1}^n \sum_{j=1}^n M(i, j)N(i, j)}{\left[\sum_{i=1}^n \sum_{j=1}^n M(i, j)^2 \sum_{i=1}^n \sum_{j=1}^n N(i, j)^2 \right]^{1/2}} \quad (2.4.7)$$

$$SAD = \sum_{i=1}^n \sum_{j=1}^n |M(i, j) - N(i, j)| \quad (2.4.8)$$

Bu klasik yöntemlerin çeşitli alternatifleri de önerilmiştir. Örneğin Birchfield ve Tomasi, piksellerin kendi yeşinlik değerlerinin yanısıra komşu piksellerin yeşinlik değerlerini de kullanarak, örneklemeye karşı duyarsız bir maliyet fonksiyonu geliştirmişlerdir [44]. Bu tezde önerilen eşleşme fonksiyonun temelini oluşturan bu yöntem hakkında 4.1.1 bölümünde detaylı bilgi verilmiştir.

Klasik yöntemlerin dışında, bunların kombinasyonlarıda birçok araştırmacı tarafından kullanılmıştır. Örneğin, Klaus ve diğ. [45], mutlak farkların toplamı ve gradyeni birleştirip tutarlı bir eşleşme fonksiyonu ortaya koymuştur.

Maliyet fonksiyonlarının başarısında, kullanılan pencerenin boyutunun önemli etkisi vardır. Çok geniş pencereler imgedeki detayların kaybolmasına neden olurken, dokunun az olduğu bölümlerde küçük pencereler hatalı sonuçlar üretecektir. Bu problemle başedebilmek için ortaya atılan *uyarlanıır pencere* yöntemlerinde [46, 47, 48, 49], her piksel için en iyi destek penceresi bulunmaya çalışılır. Kanade ve Okutomi [46], pencere içindeki lokal yeşinlik ve aykırılık değişimlerinden yola çıkarak sonuca giden bir sistem önermiştir. Ancak bu yöntem, başlangıç aykırılık kestirimine karşı çok hassastır ve hesaplama süresi uzundur. Ayrıca, destek penceresinin şeklinin sabit dikdörtgen olarak kısıtlanması, derinlik süreksizliklerinin olduğu noktalarda problem çıkarmaktadır. Bu nedenle, Boykov ve diğ.[47], olasılık hipotez testi yaparak pencere şeklini duruma bağılı olarak seçmeye çalışmıştır.

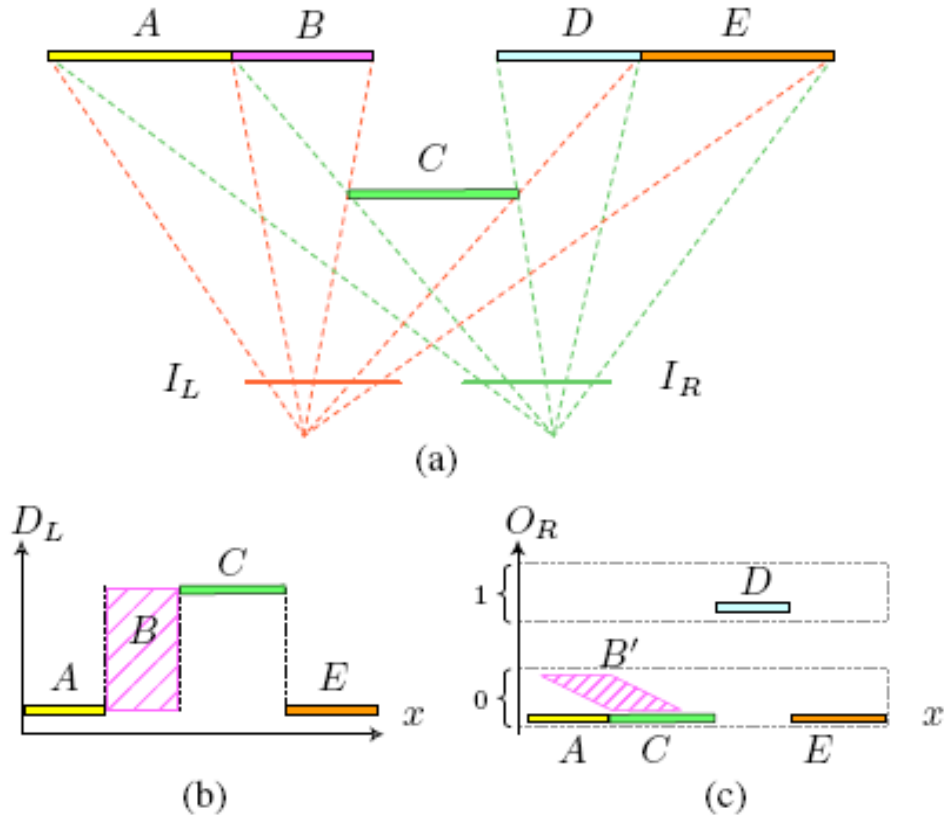
Bir diğeri grupta, çoklu-pencere yöntemleri [50, 51, 52] incelenebilir. Bu yöntemler önceden belirlenmiş destek pencereleri arasından uygun olanın seçilmesine dayanmaktadır. Örneğin, Fusiello ve diğ. [50], 9 farklı pencere ile ilinti hesaplamış ve en düşük eşleşme maliyetine sahip pencereyi seçmişlerdir.

Bazı yöntemlerde [53, 54, 55], destek penceresinin boyutlarını sabit tutulup içindeki ağırlıkların değiştirilmesi önerilmiştir. Örneğin Prazdny [53], komşu piksellere ait destek ağırlıklarının döngü içinde ayarlanmasını önermiştir. Bu yöntemde, komşu piksellerin aynı nesneye ait olduklarında, birbirlerine yakın aykırılık değerine sahip olacağı ve birbirini destekleyeceği öngörülmüştür. Xu ve diğ. [55], başlangıç aykırılık kestiriminin kesinkesliği yardımıyla, destek ağırlıklarını merkez açılımlı bir şekilde

hesaplamıştır. Ancak bu yöntemler başlangıçtaki aykırılık kestirimine karşı çok hassastır. Bu nedenle Yoon ve Kweon [56], piksellerin renk benzerliklerini ve birbirlerine uzaklıklarını kullanarak destek ağırlıklarının belirlenmesini önermişlerdir.

2.3.3. Sahnedeki Nesnelerin Birbirinin Görüntülerini Kapatması

Stereo sistemlerde, arařtırmacıların karşısına çıkan en önemli problemlerden biri, görüntünün birinde gözlemlenen noktanın diğesinde gözükmemesidir. Bu durum “örtme” (Occlusion) olarak adlandırılır. Örtülmüş bölgeler arařtırmacıların karşısına iki şekilde çıkar. Birinci durum, referans imgenin solundaki piksellerin diğere resimde eşinin bulunmasının imkansız olmasıdır. Bu bölgedeki bir pikselin x değerinden aykırılık değerini çıkardığımızda eksi bir değer elde ederiz ve doğal olarak diğere imgede karşılığı olamaz. İkinci durum sahnedeki nesnelerin birbirlerinin görüntüsü kapamasıyla ortaya çıkar. Şekil de bu durum grafiksel olarak anlatılmıştır.



Şekil 2.18 : Örtülmüş bölgelerin grafiksel gösterimi

Şekil 2.18(a)'da görüldüğü üzere, stereo imge çifti I_L ve I_R , içinde A'dan E'ye kadar sıralanmış nesnelerin bulunduğu sahneye aittir. Şekil 2.18(b)'deki derinlik değerleri incelendiğinde C nesnesinin I_R 'den bakıldığında önünü kapamasından dolayı taralı alan B'nin örtülmüş bölgede kaldığı ve o bölgedeki değerlerin belirsiz olduğu gözükmemektedir. Aynı şekilde, soldan bakıldığında D bölgesi C'nin arkasında kalmaktadır. Bu durumda oluşan örtülmüş bölge haritası Şekil 2.18(c)'de gözlemlenebilir.

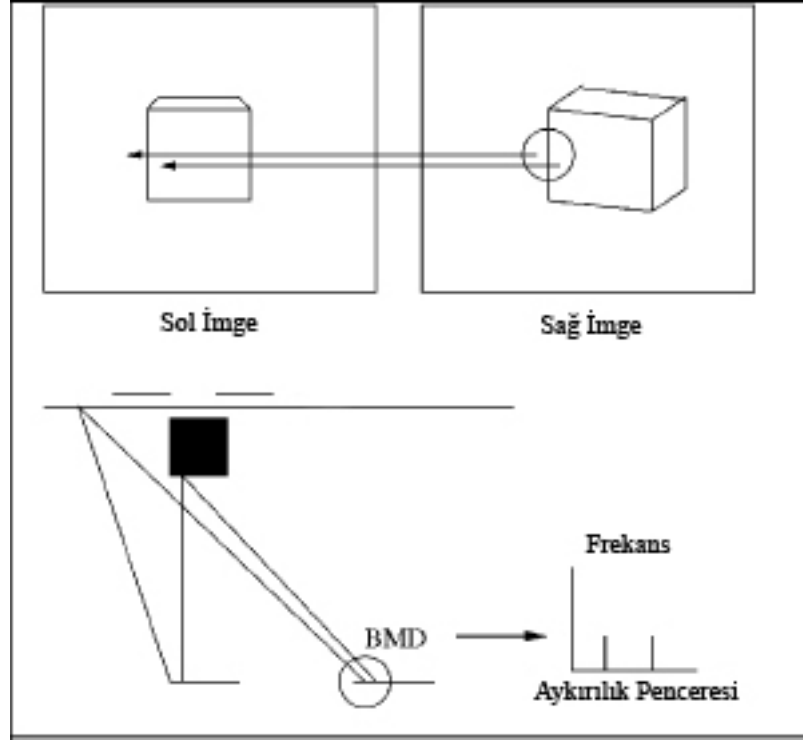
Örtülmüş bölgeler nesne sınırlarında ve sahne süreksizliklerinde oluştuğu için, tutarlı bir kestirim yapılabilmesi, bu noktaların doğru bir şekilde ele alınmasına bağlıdır. Bu nedenle birçok yaklaşım önerilmiştir[57]. Bunlar; Aykırılıkta Çift Dorukluluk (BMD: “Bimodalities in Disparity”), Eşleşme Kalite Atlamaları (MGJ: “Match Goodness Jumps”), Sağ/Sol Kontrolü (LRC: “Left/Right Checking”), Sıralama Kısıtı (ORD: “Ordering Constraint”) ve Örtme Kısıtı yöntemleridir.

2.3.3.1. Aykırılıkta Çift Dorukluluk (BMD)

Örtülmüş bölgenin sınırlarında ve çevresinde hem üzeri kapanan nesneye ait aykırılık değerlerinin hemde kaplayan nesneye ait değerlerin olduğu varsayılır (Şekil 2.19). Bu durumda aykırılık histogramında 2 tane doruk bulunmalıdır. Bu varsayımdan yola çıkarak örtülmüş bölgelerin sınırları bulunabilir [58]. Aykırılık histogramındaki en iyi 2 değer alınıp birbirlerine oranlandığında tepe oranı ortaya çıkar. Bu oranın bire yakın olması çift dorukluluğu belirtir.

$$BMD = \frac{\max(M_2)}{\max(M_1)} \quad (2.4.9)$$

Burada M_1 ve M_2 , aykırılık alanındaki en büyük bir ve ikinci tepe değerleridir.



Şekil 2.19 : Aykırılıkta çift dorukluluk yöntemi

2.3.3.2. Eşleşme Kalite Atlamaları (MGJ)

Bu yöntemde, yanlış şekilde eşleşen noktaların örtülmüş olduğu ve eşleşme algoritmasının kendisinin hataları algılayabileceği varsayılır [59, 60]. Eşleştirici, iki resimde de görünen bir yüzey üzerinde olduğunda, eşleşme kalitesinin göreceli olarak yüksek olması beklenir. Aynı şekilde eğer eşleştirici örtülmüş bölgede olursa, bu noktada doğru eşleşmeyi bulmak imkansız olduğu için, eşleşme kalitesi düşük olmalıdır. Bu yöntem analitik olarak aşağıdaki ifadeyle açıklanabilir.

$$Hata = \max(\bar{C}_x - \bar{C}_{x+w}, \bar{C}_x - \bar{C}_{x-w}), \quad (2.4.10)$$

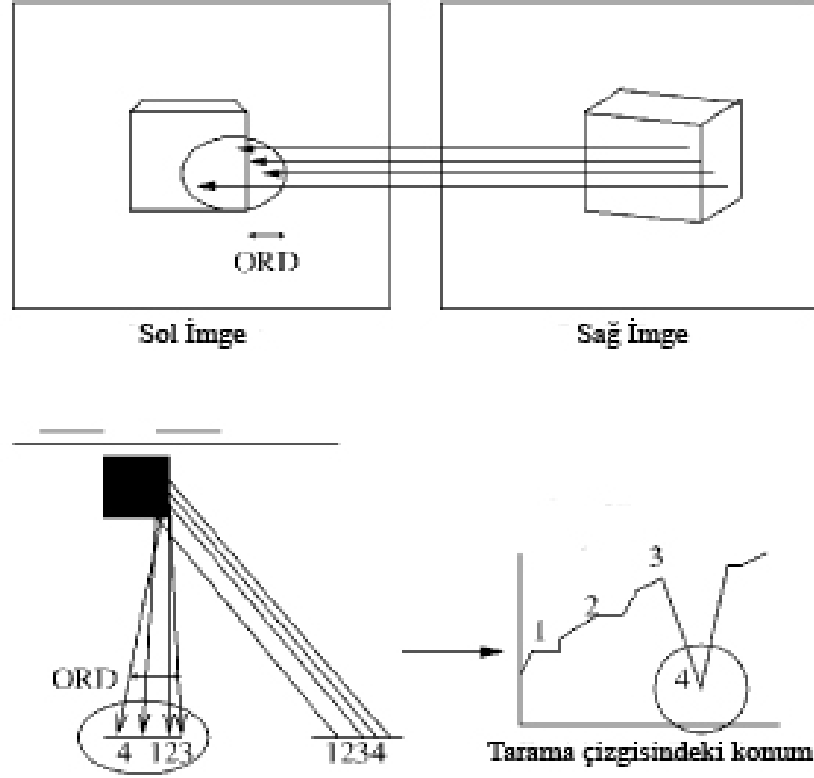
Burada x , pikselin yatay koordinatı ve C , belli w boyutundaki pencere içindeki eşleşme değerlerinin toplamıdır.

2.3.3.3. Sağ/Sol Kontrolü (LRC)

Sol ve sağ görüntü aynı sahneye ait olduğundan, soldan sağa ve sağdan sola hesap yapılarak oluşturulan yatay aykırılık imgelerindeki değerlerin, birbirlerinin negatifi olması beklenir. Aksi durumda bu noktalar örtülmüş noktalardır. Eğer x'_L kestirilen nokta ve sağ resime dayalı aykırılık değeri $d^R_{x_R}$ ise, sağ koordinat ile $x'_L = x_R + d^R_{x_R}$ şeklinde eşleşir. Hata aşağıdaki biçimde hesaplanabilir.

$$Hata = x_R - (x'_L + d^L_{x_L}), \quad (2.4.11)$$

Burada $d^L_{x_L}$ sol tabanlı aykırılık değeridir.



Şekil 2.20 : Sıralama Kısıtı

2.3.3.4. Sıralama Kısıtı (ORD)

Eğer birinci resimde A noktası B noktasının solunda ise diğer resimde de A noktası B noktasının solunda olmalıdır (Şekil 2.19). Örtülmüş bölgelerde hatalı eşleşmelerden dolayı bu sıra bozulur. ORD yönteminde, bu noktalar örtülmüş nokta olarak işaretlenir. Önceki bölümdeki notasyonu kullanarak bu durumdaki hata,

$$Hata = \max(0, d^R_{x_R} - d^R_{x''_R}), \quad (2.4.12)$$

olarak gösterilebilir. Burada x'' , en sağdaki eşleşmedir.

2.3.3.5. Örtme Kısıtı (OCC)

İdeal bir eşleştirici, iki sürekli yüzey arasında sürekli bir aykırılık haritası çıkarmalıdır. Ancak örtülmüş bölge sınırlarında, aykırılık değeri, noktayı diğer resimde eşleşmesiz bırakacak şekilde, nesne ve arkaplan arasında zıplamalar yapar. Eşleşme işleminin değeri diğer resimde atlama yaptığı zaman, OCC diğer resimdeki eşleşmeyen noktayı örtülmüş olarak işaretler. Örtülmüş bölge sınırlarındaki bu kısıt analitik olarak aşağıdaki şekilde gösterilir.

$$Hata = \max(0, d^L_{x_{L+1}} - d^L_{x_L}), \quad (2.4.13)$$

2.3.3.6. Yöntemlerin Karşılaştırılması

Yukarıdaki yöntemler karşılaştırıldığında, çoğu durumda LRC yönteminin tutarlı sonuçlar verdiği görülmektedir. Özellikle eşleşme fonksiyonlarının başarılı olduğu, sahne geometrisinin frontoparalel olduğu durumlarda, en iyi sonucu LRC vermektedir. Ancak 2 görüntü üstünde de işlem yaptığından dolayı hesaplaması diğerlerine göre daha fazla zaman alan bir çözümdür. Bu sorun paralel donanım kullanılarak çözülebilir. Gürültüye karşı diğerlerinden daha duyarlı olması, LRC yönteminin bir diğer dezavantajıdır. Bunun nedeni algoritmanın performansının eşleştiricinin performansına bağlı olmasıdır. Eşleştiricinin hata yaptığı noktalar örtülmüş olarak algılanmaktadır.

ORD yöntemi bazı hatalara karşı LRC yöntemine göre daha dayanıklıdır. Bir noktanın örtülmüş olarak işaretlenmesi için hem hatalı olması hemde sırayı bozmuş olması gerekmektedir. Bunun yanında, ORD uygulaması için daha az kod ve parametre yeterlidir. Ancak sağda büyük hataların olduğu durumlarda veya görüntünün ince bir çizgi içerdiği durumlarda algoritmada problemler yaşanmaktadır.

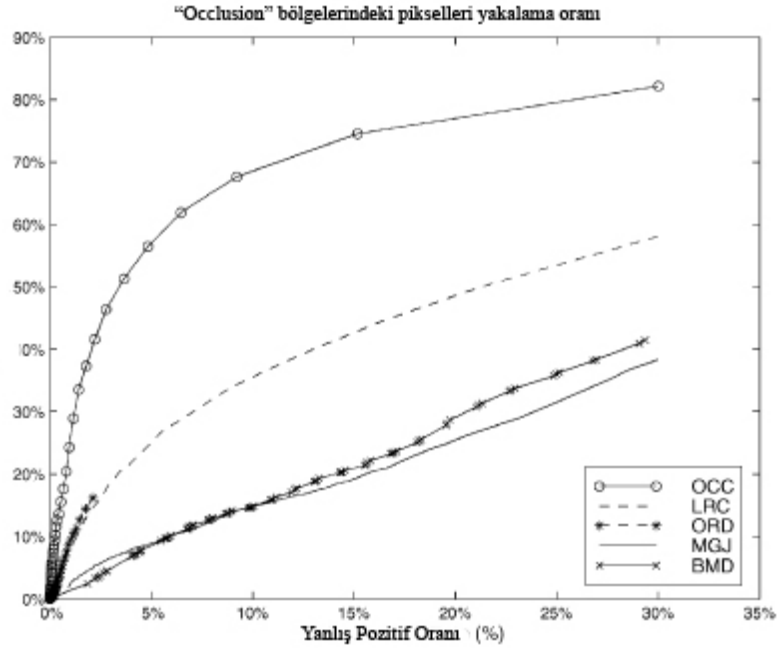
OCC yöntemi örtülmüş bölgeleri, LRC yönteminden bile daha tutarlı bir şekilde işaretlemektedir. Hem nitelik hemde nicelik olarak OCC'nin örtülmüş nokta belirleme yüzdesi LRC'den daha yüksektir. Fakat, diğer resimdeki örtülmüş bölgede çalışırken hatalı sonuçlar vermektedir. Bu problem iki yönlü OCC uygulanarak çözülebilir. Ancak bu durumda LRC kadar zaman alan bir algoritma oluşacaktır.

BMD yönteminin avantajı, epipolar çizgilere ortogonal, süreksizlik bölgeleri dahil tüm örtülmüş bölge kenarlarını saptayabilmesidir. Bu nedenle diğer yöntemlere göre

örtülmüş bölge sınırlarını nitel olarak daha doğru belirler ve daha çok 3B sınırları gösteren bir indikatör olarak yararlanır. Ancak pencere boyutu gibi parametrelerine karşı çok hassastır ve sahneye göre parametrelerinin ayarlanması başarısını önemli oranda artırır.

Diğer yöntemlerden farklı olarak, MGJ, eşleşme fonksiyonun o anki performansı yerine eşleşme algoritmasının iç yapısına bağlıdır. Bu özellik yöntemin bazı hatalı eşleşmeleri örtülmüş piksellerden ayırmasına olanak sağlar. Örnek olarak: çimlerin, yaprakların olduğu düşük uzamsal sıklığa sahip bölümlerde, diğer algoritmaların doğru eşleşme olarak algılayacağı piksellerin örtülmüş olduğunu veya bazı hatalı eşleşmelerin örtülmüş olmadığını belirleyecektir.

Sonuç olarak tüm yöntemler karşılaştırıldığında, yöntemin iyiliği belirleyecek tek boyutlu bir kriter olmadığı görülür. Alınan sonuçlara niteliksel açıdan bakıldığında, en düşük hatalı-pozitif oranını ve en yüksek doğru örtme tahmini sağlayan OCC en iyi algoritmadır. LRC algoritması OCC'ye yakın sonuçlar vererek ikinci sırayı alır. (Şekil 2.21). Ancak, bu ölçüler herşeyi ifade etmez. OCC yöntemi, sol örtülmüş bölgedeyken, sağ örtülmüş bölgeleri tahmin etmede veya tersi durumlarda başarısızdır.



Şekil 2.21: Örtülmüş bölge belirleme yöntemleri karşılaştırması

Öte yandan, LRC algortimasının düşük uzamsal sıklığa sahip bölgelerdeki performansı ciddi oranda düşmektedir. Uzamsal sıklığın düşük olduğu dışarı ortam resimlerinde, MGJ algoritması, diğerlerine göre daha gürbüz olan örtülmüş bölge belirleme yöntemiyle, çok daha başarılı sonuçlar vermektedir. Sadece örtülmüş bölgelerin sınırlarının önemli olduğu durumlar için BMD algoritması daha uygundur. Genel olarak, ORD en ihtiyatlı algoritma olmasına rağmen, yinede hatalı-pozitif sonuçlar üretebilmektedir ve “iki çivi ilüzyonuna” [61] karşı hassastır.

Yöntemlerin karşılaştırılmasından anlaşılacağı üzere, tek başına bir yöntem kullanmak yerine sahnenin durumuna göre değişik yöntemlerden yararlanmak daha verimli olacaktır. Bu tez kapsamında önerilen algortmada LRC, OCC ve MGJ yöntemlerinden eşleşme skorunun güvenilirliğinin test edilmesi ve örtme haritalarının çıkarılması aşamalarında yararlanılmıştır.

2.3.4. Optimizasyon ve Aykırılık Hesabı

Eşleşme fonksiyonun sonuçları alındıktan sonra oluşan verinin doğru biçimde yorumlanması gerekmektedir. Bu amaçla iki farklı yaklaşım kullanılır: lokal ve global yöntemler.

Lokal yöntemlerde, uygulamanın başarısı tamamen eşleşme fonksiyonuna ve kurgulamaya bağlıdır. Her piksel için en düşük maliyeti veren aykırılık değeri seçilir. Bu yöntemlerde her piksel için “kazanan-hepsini-alır” (KHA) optimizasyonu kullanılır. Bu yöntemin (ve birçok ilişkilendirme yönteminin) limiti, eşleşme tekilliğinin sadece referans görüntüsü için sağlanabilmesidir. Diğer görüntüde bir nokta birçok noktayla eşleşebilir.

Global yöntemlerde, lokal yöntemlerin aksine tüm işlem aykırılık hesaplama evresinde yapılır. Tüm global yöntemler enerji minimizasyonu çâtısına [62] göre formülendir. Amaç aşağıda genel formülü (2.4.13) verilmiş olan global enerjiyi minimize edecek aykırılık değerini bulmaktır.

$$E(d) = E_{veri}(d) + \lambda E_{düzlük}(d), \quad (2.4.13)$$

Veri terimi, $E_{veri}(d)$, aykırılık değerinin imge çiftine ne kadar uyumlu olduğunu ölçer. Aykırılık uzayı formülasyonu kullanılarak,

$$E_{veri}(d) = \sum_{(x,y)} C(x, y, d(x, y)), \quad (2.4.14)$$

elde edilir. Burada C , eşleşme fonksiyonunun aykırılık uzayı imgesidir.

Düzlük terimi $E_{düzlük}(d)$, belirlenen düzlük varsayımlarının enerji hesabına eklenebilmesini sağlar. Optimizasyon işleminin, hesaplama açısından altından kalkılabilir olması amacıyla, çoğunlukla sadece komşu pikseller arasındaki aykırılık değeri farkları kullanılır.

$$E_{düzlük}(d) = \sum_{(x,y)} \rho(d(x, y) - d(x + 1, y)) + \rho(d(x, y) - d(x, y + 1)), \quad (2.4.15)$$

Burada ρ , aykırılık farkının tek düze artan fonksiyonudur.

$E_{veri}(d)$ ve $E_{düzlük}(d)$ terimlerinin yanısıra, önceki bölümde açıklanan örtme problemini kontrol edebilmek amacıyla, global enerji formülüne E_{occ} terimi eklenebilir. E_{occ} teriminin amacı gereksiz olarak örtme yaratan piksellerin cezalandırılmasıdır.

$$E_{occ}(d) = \lambda_{occ}, \quad (2.4.16)$$

Burada λ_{occ} sabit bir katsayıdır. Örtme terimi eklendiğinde global enerji formülü aşağıdaki son şeklini alır.

$$E(d) = E_{veri}(d) + \lambda E_{düzlük}(d) + E_{occ}(d), \quad (2.4.17)$$

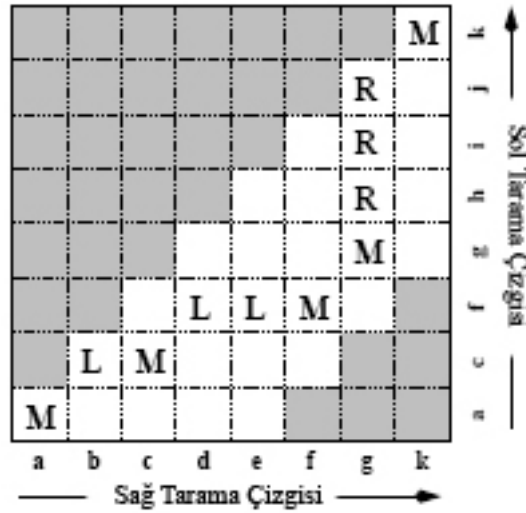
Global enerji tanımlandıktan sonra, tanımlanan enerji fonksiyonunun minimum noktasını bulmak için birçok algoritma kullanılabilir. Yeni optimizasyon yöntemleri sunuldukça, bu yöntemlerin (2.4.17)'deki minimizasyon problemini çözecek uygulamaları da yapılmıştır. Optimizasyon algoritmalarındaki gelişime bağlı olarak 2000'li yılların başlangıcında *Dinamik Programlama*, *İşbirliği Yaklaşımı* ve *Grafik-Kesme* algoritmaları popülerken, günümüzde en çok *İnanç Yayılımı* yöntemi tercih edilmektedir. Araştırmaların bu şekilde belli yönlere doğru ilerlemesinde Scharstein ve

Sleziski'nin karşılaştırma ortamındaki sonuçların [43] araştırmacılara yol göstermesinin etkisi büyüktür. Bu bölümün devamında, en sık kullanılan enerji minimizasyonu algoritmaları hakkında kısa bilgiler verilmiştir.

2.3.4.1. Dinamik Programlama

(2.4.17) denkleminin 2B optimizasyonunun birçok düzlük fonksiyonu için NP-Zor olduğu gösterilebilir[63]. Dinamik programlama ile polinomsal süre içinde bağımsız yatay tarama hatları için global minimum bulunabilir. Daha çok doksanlı yıllarda sıklıkla tercih edilen dinamik programlama, stereo görüde ilk olarak seyrek, kenar bazlı yöntemlerde [64, 65] kullanılmıştır. Daha sonra kullanım alanı, yoğun (yeğinlik bazlı) tarama çizgisi optimizasyonu problemlerine kaymıştır [66].

Bu yöntemlerin çalışma prensibi, iki ilişkili tarama hattı arasındaki eşleşme maliyetlerinden oluşan ikili matris (Şekil 2.22) dahilindeki minimum maliyete sahip yolun bulunmasıdır. Enerji fonksiyonunda göz önüne alınabilecek kısmi örtme durumu, birinci imgedeki bir grup pikselin diğer imgedeki tek bir pikselle ilişkilendirilmesi yoluyla modellenir.



Şekil 2.22 : Dinamik programlama kullanarak stereo eşleme.

Dinamik programlamanın en önemli problemleri, örtülmüş bölgedeki piksel için sağ maliyetin belirlenmesi ve iç tarama çizgisinin tutarlılığının sağlanmasıdır. Bu problemleri aşmak için birçok yöntem önerilmiştir[65, 67].

Gözardı edilmemesi gereken bir diğer sorun ise, tek düzelilik ve sıralama kısıtlarının karşılanmasıdır. Bu kısıt, tarama hattı üzerindeki piksellerin sırasının her iki imgede de aynı olmasını gerektirir. Ancak sahnenin birbirine yakın önplan nesnelere içerdiği durumlarda bu kısıtın karşılanmasında sorunlar yaşanacaktır.

2.3.4.2. Grafik-Kesme Yöntemi(*Graph Cut*)

Dinamik programlamanın en belirgin limiti, yatay ve dikey tarama hatlarının devamlılık kısıtlarını birleştirememesidir. Bu durumu düzeltmek için birçok yöntem önerilse de tam olarak 2 yönlü tutarlılık kısıtını sağlamak mümkün olmaz. Bu noktadaki güçlerinden dolayı sadece tek bir tarama hattında değil iki yönde devamlılık kısıtını sağlayan *Grafik-Kesme* yöntemi önerilmiştir.

Grafik kesme yöntemleri, ilişkilendirme problemini grafik içinde bir maksimum akış (minimum kesme) problemine dönüştürerek çözer. İlk olarak Greig ve diğ.[78], minimum kesme/max-akış algoritmalarının bilgisayarlı görme uygulamalarındaki önemli enerji minimizasyonu problemlerini çözebileceğini keşfetti. İlk uygulamalarında sadece bir minimum maliyet kesmesi kullanılırken, daha sonraları döngüsel yöntemler kullanılarak güçlü lokal kesmeler elde edildi.

Önceleri stereo ilişkilendirme algoritmaları arasında çok popüler olmasına rağmen, geçtiğimiz yıllarda Grafik-Kesme yöntemine dayalı birkaç yeni geriçatım yöntemi geliştirilmiştir [74, 75, 76]. Bu yöntemler, objektif fonksiyonunu, etiketleme problemi olarak formüle edip minimize etmeye çalışır. Bu işlem, ilgili grafikteki en küçük çok-yönlü kesit bulunarak gerçekleştirilir. Genelde çok-yönlü grafik kesimi NP-Zor bir problemdir ve buna uygun tekniklerle çözülmesi gerekir. Bu amaçla, ana problem genellikle iki bağlantılı, verimli ve optimal çözümü mevcut, alt probleme bölünür. Bu mantıkla yapılandırılan sistemler 2005 yılına kadar diğer sistemlerden daha iyi sonuçlar vermiş olmasına rağmen, günümüzde yerini daha çok inanç yayılımına bırakmıştır.

2.3.4.3. İşbirliği Yaklaşımı

Marr ve Poggio [79, 80], biyolojik sinir sisteminden esinlenerek, *işbirliği yaklaşımını* ortaya attı. Yaklaşımlarında, düzlük ve teklik kısıtlarını lokal olarak birbirine bağlanmış döngülü paralel sisteme uyguladılar.

İşbirliği yaklaşımında, doğru eşleşmenin yüksek eşleşme değerine sahip olacağı varsayılmıştır. Bu amaçla normalize edilmiş çapraz ilinti (NCC) veya SSD gibi maliyet fonksiyonları kullanılır. Düzlük ve teklik kısıtları gözönüne alınarak, döngüsel biçimde komşuların değerleri güncellenir. Döngüsel güncelleme yakınsadığı zaman, referans imgesindeki her piksel için maksimum eşleşme değerini veren aday seçilir. Marr ve Poggio [79, 80], makalelerinde sabit bir destek bölgesini, ikili imgelere uygulamışlardır. Zitnick ve Kanade [81], işbirliği algoritmasını geliştirerek gerçek imgelere uygulamıştır. Yöntemlerinde, orijinal yöntemdeki 2B destek yerine 3B destek kullanmışlardır. Buna ek olarak örtülmüş bölgeleri belirlemek amacıyla bir eşik belirleme yöntemi önermişlerdir. Bu önerinin arkasındaki fikir, örtülmüş bölgede bulunan piksellerin benzer derinlik değerleri içermeyeceği ve bu bölgelerdeki eşleşme fonksiyonu değerlerinin küçük olacağıdır. Bu fikir doğrultusunda, eşleşme değeri belli bir eşğin altında kalan pikseller örtülmüş bölgede kabul edilir.

Bu algoritmadaki hata oranından dolayı yeni bir işbirliği yaklaşımı önerilmiştir [82]. Bu yaklaşımdaki birinci değişiklik, güncelleme fonksiyonunun başlangıç değerlerine karşı hassaslığından kaynaklı hataları azaltmak amacıyla, başlangıç eşleşme değerlerinin doğru eşleşmenin değerini yükseltecek bir yöntemin dizayn edilmiş olmasıdır. İkinci olarak, lokal destek bölgeleri bölütleme bilgisi kullanılarak seçilmektedir.

Stereo ilişkilendirme algoritmalarına işbirliği yaklaşımının bir başka kullanım şekli Wang ve Zeng [83] tarafından ortaya atılmıştır. Önerdikleri yöntemde, işbirliği algoritması, bölütlere ait düzlemlerin parametrelerini belli bir enerji fonksiyonunu minimize edecek şekilde değiştirmede kullanılmıştır.

2.3.4.4. İnanç Yayılımı (Belief Propagation)

Stereo ilişkilendirme problemi Markov ağı şeklinde modellenenebildiği takdirde, *Bayes İnanç Yayılımı* kullanılarak çözülebilir. Bu amaçla problem: aykırılık için düz bir alan, derinlik süreksizliği ve örtülmüş bölge belirleme için bir yordam içeren Markov Rastgele alanıyla gösterilebilir. Daha sonra Markov ağında MAP kestirimi, bayes inanç yayılımı algoritması yardımıyla yapılır.

İnanç yayılımı, herbir sahne değişkeninin döngüsel olarak, lokal mesaj geçişleriyle güncellendiği global enerji minimizasyon tekniğidir. Mesajlar, komşu değişkenlerden

gelen güncel bilgilere göre alıcının belli bir durumda olma olasılığını barındırır. Bu tekniğin popülerliği, sağladığı başarılı sonuçlarından dolayı her geçen gün artmaktadır [45, 68, 70, 72, 73]. Bu tezin yazıldığı dönemde “Middlebury” test ortamındaki [71] en başarılı 5 algoritmadan 4 tanesi bu yöntemi kullanmaktadır. Bu tekniğin detayları bölüm 3.2 ve Ek-A’da verilmiştir.

2.3.5. Daha Önce Yapılmış Çalışmalar

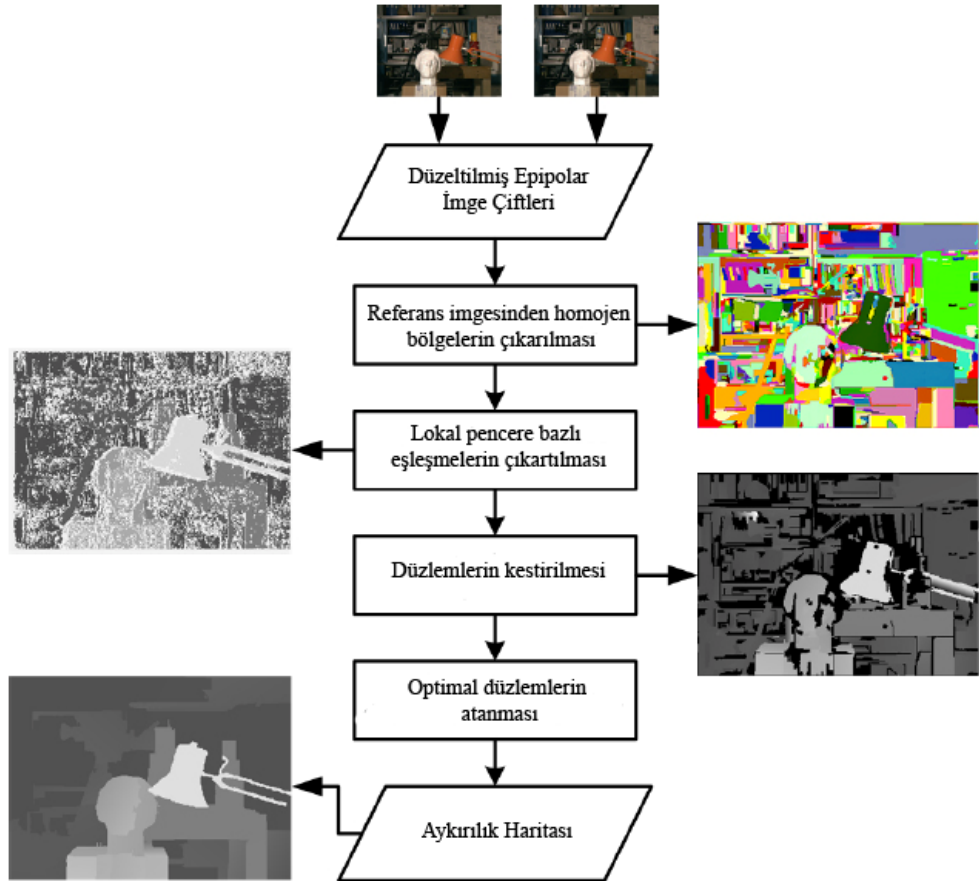
Araştırmacıların yoğun olarak ilgisini çeken bir konu olması nedeniyle yetmişli yıllardan beri stereo ilişkilendirme algoritmaları üzerine birçok çalışma yapılmıştır. Bu açıdan bilgisayarlı görme uygulamaları arasında üzerinde en çok araştırma yapılan konulardan biridir. Önerilen sistemlerin bolluğu, bu konuda sınıflandırma yapılmasını gerektirmiştir. Bu amaçla Scharstein ve Szeliski tarafından yapılan çalışma büyük önem taşımaktadır. Tıpkı daha önceki sınıflandırmalarda [84] olduğu gibi, bu çalışmanın arkasından araştırmalar belli bir yöne doğru kaymıştır. Yazarların sınıflandırmasının güncelliğini korumasında, çalışmalarına paralel olarak hizmete soktukları internet sitesi üzerinden araştırmacılara yöntemlerini karşılaştırma şansı vermelerinin payı büyüktür. “Middlebury” test ortamı olarak adlandırılan bu yapıda[71], ortak imge setleri ve bu imge setleri kullanılarak alınan algoritma sonuçlarını karşılaştıran tablolar mevcuttur. Sürekli güncellenen bu ortam incelendiğinde, Mayıs 2008 tarihi itibarıyla, başarı oranlarıyla ve önerdikleri farklı yaklaşımlarla öne çıkan bazı yöntemler ve ortak özellikleri bu bölümün devamında sunulmuştur.

Öne çıkan yöntemler incelendiğinde, lokal pencere bazlı, devamında KHA (kazanan hepsini alır) yöntemi uygulanan yöntemlerin artık sadece algoritmaların başında veya hazırlık aşamasında kullanıldığı tespit edilmiştir.

Tüm sistemlerde belirli bir enerji fonksiyonu tanımlanmış ve daha sonra çeşitli optimizasyon yöntemleriyle bu enerji fonksiyonunu minimum yapacak değerler seçilmiştir. Bu amaçla yaygın olarak inanç yayılımı algoritması kullanılmaktadır. Sistemlerin bir başka ortak özelliği, hemen hepsinde renk bazlı bölütleme kullanılmış olmasıdır. İmgeleri bölütlere ayırıp, bölüt boyunca derinliğin düz olduğunu varsaymak, yöntemlerin başarısı önemli ölçüde arttırmıştır.

Bölüt bazlı yöntemlerin diğerlerine göre başarısı yüksek olmasına rağmen, başlangıç bölüt şekillerinin sabit olması, bölütlemeyi dolaylı olarak oluşturabilecek hataları beraberinde getirir. Bu tip hataları azaltmak için kullanılan yaygın yöntem, bölütlere tek bir derinlik değeri atanması yerine düzlemler kullanılarak, bölüt başına düşen derinlik değerinin artırılmasıdır. Bu amaçla farklı düzlem kestirimleri önerilmiştir.

Sistemlerin enerji fonksiyonları incelendiğinde, hepsinde örtme bölgeleri için ek önlemler alındığı ve bu sayede algoritma performanslarının artırıldığı gözlemlenmektedir. Çoğu sistemde sadece referans imgesinin enerjisi minimize edilirken, J. Sun ve diğ.[85] önerdiği yöntem gibi, her iki imgenin enerji değerlerini kullanan stereo yöntemlerde mevcuttur. Bu yöntemlerin güçlü yanı, örtülmüş bölgeleri tahmin etmekteki başarılarıdır.



Şekil 2.23: Klaus ve diğ. [45] sunduğu algoritmanın temel yapısı

Klaus ve diğ. [45] öne sürdükleri yöntemde, ilk olarak, mutlak farkların toplamı ve gradyenin uyarlanabilir şekilde karışımını kullanan maliyet fonksiyonu yardımıyla piksel bazında eşleşmeler hesaplanır.

$$C(x, y, d) = (1 - w) * C_{SAD}(x, y, d) + w * C_{GRAD}(x, y, d) \quad (2.4.18)$$

Her iki imge içinde hesaplama yapıldıktan sonra, sağ/sol kontrolüyle tutarlı pikseller ayıklanır. *Ortalama-kaydırmalı renk bölütleme*[77] yardımıyla, imge bölütlere ayrılır. Her bölüt için, içerdiği tutarlı değerler kullanılarak gürbüz düzlem kestirimi yapılır. Oluşan birçok düzlemden, en iyi sonucu verenler, inanç yayılımı bazlı enerji minimizasyonu ile belirlenir. Bu algoritmanın temel adımları Şekil 2.23'te gösterilmiştir.

David Nister ve diğ. [68] tarafından önerilen sistemde, eşleşme maliyet fonksiyonu olarak *uyarlanabilir destek ağırlıklı algoritma* [69] kullanılır. Bölüm 4.1'de yapılan karşılaştırmalarda da kullanılan bu algortmada, Birchfield ve Tomasi'nin piksel benzeşmezlik ölçüsü, renk farkları ve pikseller arasındaki mesafeler kullanılarak ağırlıklandırılmıştır. Aynı imge içindeki x ve y pikselleri arasındaki renk farkı ∇_{xy} aşağıdaki biçimde belirtilmiştir.

$$\nabla_{xy} = \sum_{c \in \{r, g, b\}} |I_c(x) - I_c(y)| \quad (2.4.19)$$

Burada I_c , renk kanalı c'nin yeğlilik ölçüsüdür. Destek penceresi içinde x pikselinin ağırlığı, renk ve uzamsal farklar kullanılarak aşağıdaki şekilde belirlenir.

$$w_{xy} = e^{-\left(\beta^{-1} \nabla_{xy} + \gamma_{cw}^{-1} \|x - y\|_2\right)} \quad (2.4.20)$$

Son olarak x pikseli için destek penceresi W_x içindeki veri maliyeti aşağıdaki şekilde belirtilir.

$$C(x_L, x_R) = \frac{\sum_{(y_L, y_R) \in W_{x_L} \times W_{x_R}} w_{x_L y_L} w_{x_R y_R} d(y_L, y_R)}{\sum_{(y_L, y_R) \in W_{x_L} \times W_{x_R}} w_{x_L y_L} w_{x_R y_R}} \quad (2.4.21)$$

Sistemin ilk aşamasında, yukarıdaki ilinti fonksiyonu kullanılarak her iki imge içinde ilinti hacimleri hesaplanır ve inanç yayılımı yöntemiyle derinlik haritaları çıkarılır. Ayrıca bu aşamada imgelere ait düzlük maliyetleride aşağıdaki şekilde hesaplanır.

$$\delta_{xy} = |I(x) - I(y)| \quad (2.4.22)$$

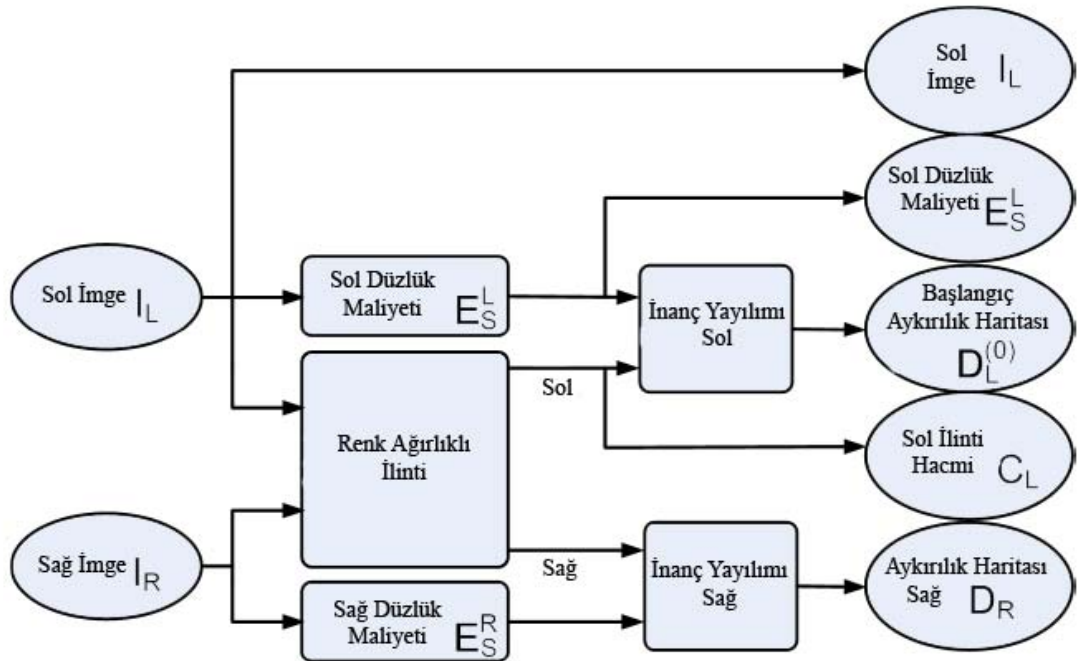
Burada $[0,1]$ aralığında olacak şekilde normalize edilmiştir. Tüm karedeki değerlerin ortalaması çıkarıldıktan sonra, δ_{norm} elde edilir. Buradan maliyet katsayısı hesaplanır.

$$\rho_s = 1 - \delta_{norm} \quad (2.4.23)$$

Hesaplanan katsayı doğrultusunda düzlük fonksiyonu aşağıdaki gibidir.

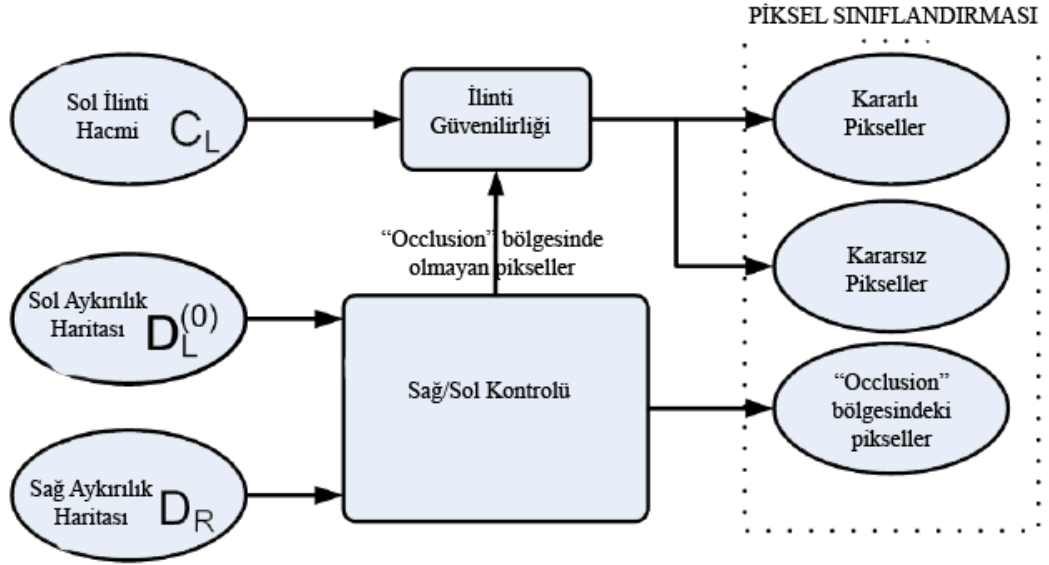
$$E_{düzlük} = \rho_s |D(x) - D(y)| \quad (2.4.24)$$

Bu sistemin başlangıç modülü Şekil 2.24'te özetlenmiştir.



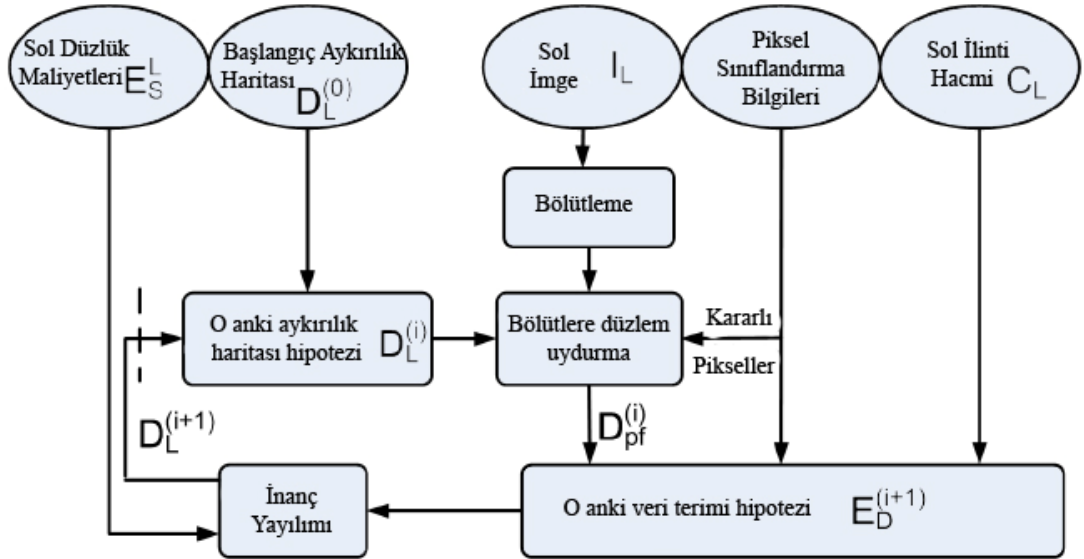
Şekil 2.24: [68] algoritmasının giriş modülü

Sistemin ikinci aşamasında, başlangıç modülünde oluşturulan haritalar, sağ/sol kontrolü ve BMD benzeri bir ölçü kullanılarak sınıflandırılır. (Şekil 2.25)



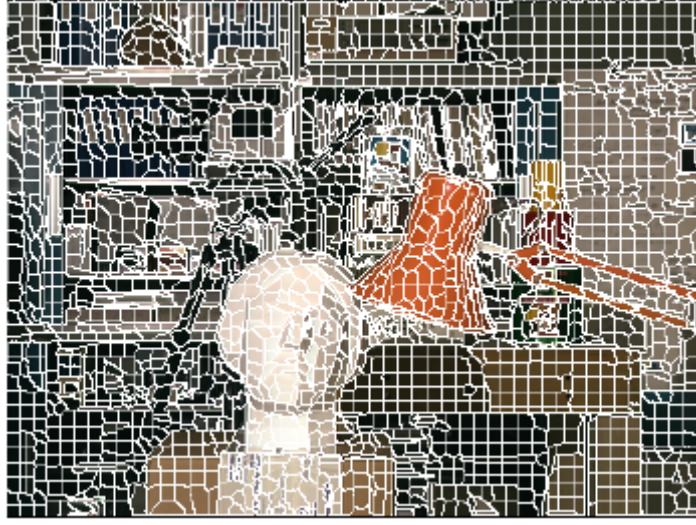
Şekil 2.25: Piksel sınıflandırma modülü [68]

Son modülde, referans imgesi ortalama-kaydırmalı renk bölütlemesi yöntemi yardımıyla, bölütlere ayrılır. Her bölüt için, sadece tutarlı pikseller kullanılarak, düzlemler kestirilir. Kesitirilen düzlemler, bir başka inanç yayılımı döngüsü içinde (Şekil 2.26) sonucun iyileştirilmesinde kullanılır.



Şekil 2.26: Döngülü iyileştirme modülü [68]

Kullanılmakta olan yöntemlerin çoğunda, bölütlemeye bağlı hataları gidermek için bir yöntem sunulmamıştır. Ancak Zitnick ve diğ. [70], bölütleri markov rastgele alanında düğümler olarak kabul ederken, rengin arkaplan ve önplan renginin karışımı olduğu varsayımıyla yeni bir olasılık modeli önermişlerdir.

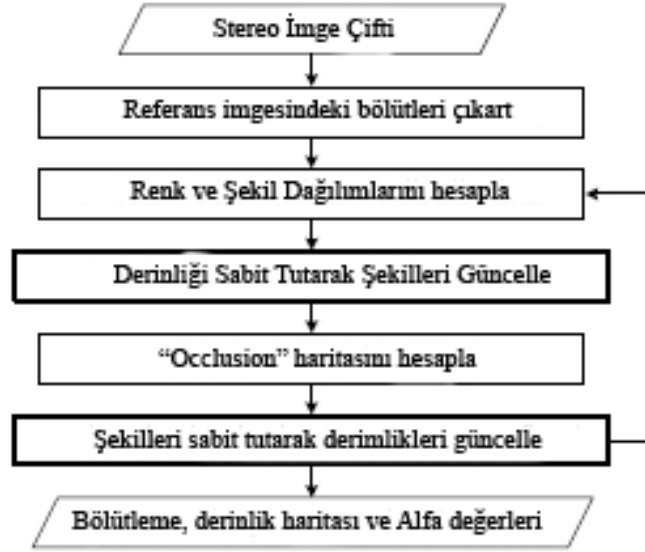


Şekil 2.27: Zitnick ve diğ.[70] algoritmasının “Tsukaba” imgesi üzerinde oluşturduğu bölütleme

Bu olasılık fonksiyonun MAP kestirimi yapılarak, piksellerin hangi bölüte ait olduklarına yönelik bilgiler güncellenebilir. Daha sonra şekli değişen bölütler için yeni aykırılık kestirimleri yapılır. Algoritmanın sonunda, derinlik haritasının yanı sıra, piksellerin ön ve arka planları arasındaki α değerleri de elde edilir. Bu yöntemin aşamaları şekil 2.28’de gösterilmiştir.

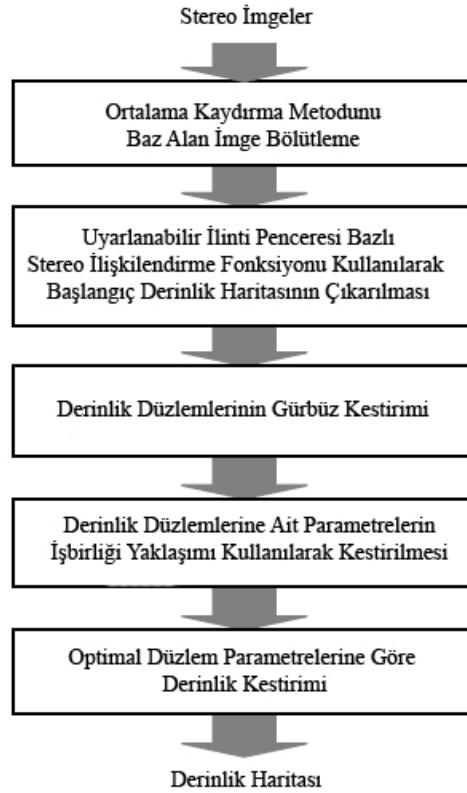
Wang ve Zeng [83], yöntemlerinde (2.4.21)’deki eşleşme maliyet fonksiyonunu kullanmıştır. Diğer yöntemlerdeki gibi ortalama-kaydırmalı renk bölütlemesi yardımıyla referans imgesini homojen bölgelere ayırmışlar ve ,her bölüt için gürbüz kestirim yöntemiyle düzlemler belirlemişlerdir. Ancak diğer yöntemlerden farklı olarak, enerji minimizasyonunu aykırılık değerleri yerine, bölütler için kestirilen düzlemlerin parametrelerini değiştirmek için kullanmışlardır. Daha sonra optimal düzlem parametreleri yardımıyla aykırılık haritasını oluşturmuşlardır. Bu yöntemin adımları Şekil 2.28’de verilmiştir.

Mevcut başarılı yöntemler incelendiğinde bölütlemeyen doğan hataların azaltılmasının ve dolayısıyla düzlem kestiriminin giderek daha fazla önem kazandığı görülmektedir.



Şekil 2.28 : Hem bölüt şekillerini hemde derinliği güncelleyen algoritma [70].

Başarılı bir yöntem yaratmak için, eşleşme maliyet fonksiyonunun seçimi ve düzlem kestiriminde kullanılacak noktaların belirlenmesi kritik öneme sahiptir. Bu tez kapsamında her iki konu üzerine yeni öneriler getirilmiştir.



Şekil 2.29 : Enerji minimizasyonunu düzlem parametrelerini güncellemede kullanan algoritma [83]

3. YÖNTEM

Aykırlık kestirimine dayalı stereo ilişkilendirme yöntemleri, birden fazla görüntüden yapı elde etme arařtırmalarının önemli bir kısmını oluřturmaktadır. Yapay sinir aęları gibi birçok yeni yöntem geliřtirilirken insan fizyolojisi örnek alınmıřtır. Bu nedenle, insanın iki gözünün olması stereo yöntemlere olan ilgiyi arttırmaktadır.

Bölüm 2.3.1’de açıkladıęı üzere, stereo sistemlerin “epipolar” olarak adlandırılan geometrisi, bilgisayarlı görü konularının en anlařılmıř olanıdır. Optimizasyon yöntemlerindeki geliřmelerin çoęunun stereo algoritmalarda uygulanabilmesi ve bu konuda çalıřan arařtırmacılar için, güncel bir karřılařtırma ortamının[71] sunulmuř olması, yayınlanan makale sayısını önemli oranda etkilemektedir.

Bu bölümde, tez kapsamında önerilen stereo ilişkilendirme algoritmasının detaylarına yer verilmiřtir. Bu algortmada kullanılan eřleřme fonksiyonu ve ikinci ařamasında yararlanılan örtme haritası çıkartma yöntemi, önerilen yeni yaklařımların en önemlileridir.

3.1. PİKSEL ALANINDA LOKAL EŐLEŐME

Tüm stereo algoritmalarda, sisteme giren imgelerin düzeltilmiř olduęu varsayılr. Bu varsayım doęrultusunda saęlanan ideal geometride (Bölüm 2.3.1), epipolar çizgiler paralel konumlanacaęından, iki imge arasındaki eřleřen pikselleri bulmak için yatay çizgiler üzerinde tarama yapılması yeterli olacaktır. Bu amaçla ilk olarak tutarlı bir eřleřme maliyet fonksiyonu tanımlanmalıdır. Bölüm 2.3.2’de örnekleri verildięi üzere seçilebilecek birçok maliyet fonksiyonu mevcuttur. Günümüzdeki başarılı algoritmaları incelendięinde, mutlak farklar toplamına dayanan yöntemlerin (SAD) daha fazla tercih edildięi görülmektedir. Bu tez kapsamında önerilen eřleřme maliyet fonksiyonunda, temeli mutlak farklar toplamına dayanan *piksel benzeřmezlik ölçüsü*, önemli bir rol oynamaktadır.

3.1.1. Piksel Benzeşmezlik Ölçüsü

Karşılaştırılan imgeler düzeltildiği için [42], tarama hatları epipolar çizgilerdir. İki tarama hattı boyunca, tek boyutlu sürekli yeşinlik fonksiyonları i_L ve i_R tanımlanabilir. Bu fonksiyonlar, sensörlere gelen ışık miktarının evriştirilmesinin sonucunda oluşan, desteği bir piksel genişliğinde kutu fonksiyonlarıdır. Gerçek imge sensörleri, her piksel üzerinden yeşinliğin integralinin alınıp, arkasından ideal örnekleyiciden geçirilmesiyle modellenebilir. İdeal örnekleme üzerinde çalışıldığı için integral işlemi kaldırılabilir. Sonuç olarak elimizdeki değerler, sürekli i_L ve i_R fonksiyonlarının (Şekil 2.23) ideal örnekleyiciyle elde edilmiş ayrık değerleridir.

Sol tarama hattındaki x_L ile sağ tarama hattındaki x_R arasındaki benzeşmezliği ölçmek için ilk olarak, örneklenmiş noktalar arasında lineer aradeğerleme fonksiyonu \hat{I}_R tanımlanır. Daha sonra x_L yeşinlik değerinin, x_R çevresinde lineer olarak ara değerlemesi yapılmış bölgeye ne kadar uyduğu ölçülür:

$$\bar{d}(x_L, x_R, I_L, I_R) = \min_{x_R - \frac{1}{2} \leq x \leq x_R + \frac{1}{2}} |I_L(x_L) - \hat{I}_R(x)| \quad (3.1)$$

Aynı şekilde $\hat{I}_L(x)$ tanımlanır:

$$\bar{d}(x_R, x_L, I_R, I_L) = \min_{x_L - \frac{1}{2} \leq x \leq x_L + \frac{1}{2}} |\hat{I}_L(x) - I_R(x_R)| \quad (3.2)$$

Pikseller arasındaki benzeşmezlik d , bu iki fonksiyondan minimum değere sahip olanıdır.

$$d(x_R, x_L) = \min\{\bar{d}(x_L, x_R, I_L, I_R), \bar{d}(x_R, x_L, I_R, I_L)\} \quad (3.3)$$

Parçalı lineer fonksiyonun uç noktalarının, kesme noktaları olmasına dayanarak, d aşağıdaki gibi hesaplanabilir. İlk olarak, x_R ve soldaki komşusu ortasında bulunan noktadaki, lineer aradeğerlenmiş yeşinlik değeri bulunur.

$$I_R^- \equiv \hat{I}_R\left(x_R - \frac{1}{2}\right) = \frac{1}{2}(I_R(x_R) + I_R(x_R - 1)) \quad (3.4)$$

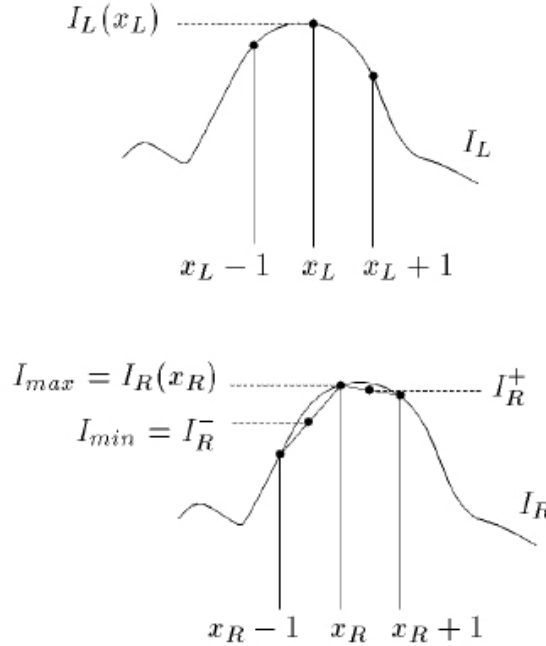
Aynı işlem sağdaki komşu içinde tekrarlanır.

$$I_R^+ \equiv \hat{I}_R\left(x_R + \frac{1}{2}\right) = \frac{1}{2}(I_R(x_R) + I_R(x_R + 1)) \quad (3.5)$$

Daha sonra , $I_{\min} = \min\{I_R^-, I_R^+, I_R(x_R)\}$ ve $I_{\max} = \max\{I_R^-, I_R^+, I_R(x_R)\}$ elde edilir. Bu değerler doğrultusunda benzeşmezlik ölçüsü aşağıdaki gibi formüle edilir.

$$\bar{d}(x_R, x_L, I_R, I_L) = \max\{0, |I_L(x_L) - I_{\max}|, |I_{\min} - I_L(x_L)|\} \quad (3.6)$$

Bu ifadenin, simetrik karşılığı $\bar{d}(x_L, x_R, I_R, I_L)$ ile beraber hesaplanmasıyla mutlak farklar yönteminin hesaplama süresi arasında az bir fark vardır. Yapılan hesaplamalarda, piksel benzeşmezlik ölçüsünün %10 daha yavaş olduğu görülmüştür. Buna karşın, başarı oranları bu zaman kaybına değecek şekilde, piksel benzeşmezlik ölçüsü lehinedir.



Şekil 3.1 : $\bar{d}(x_R, x_L, I_R, I_L)$ tanım ve hesaplaması

3.1.2. Temel Eşleşme Maliyet Fonksiyonu

Bölüm 2.4.1'den incelenebileceği üzere, stereo eşleştirme algoritmalarında kullanılabilecek birçok maliyet fonksiyonu vardır. Ayrıca maliyet fonksiyonun tutarlılığını arttırmak amacıyla, iki farklı yöntemin belirli ağırlıklarla çarpılıp beraber

kullanılması düşünülebilir. Örneğin, yeğinlik farklarının karesi veya mutlak yeğinlik farkları gibi yöntemlerle, kameranın hatalarına veya lambert olmayan yüzeylerden dolayı oluşan hatalara karşı daha dayanıklı olan, gradyen temelli yöntemler birleştirilebilir. Bu yaklaşımı ilk olarak Klaus ve diğ. [45] tarafından mutlak farkların toplamı ve gradyenleri kullanarak önerilmiştir. Bu tezde ortaya konan yöntemde, mutlak farkların toplamı yerine piksel benzeşmezlik ölçüsü kullanılmıştır. Yaklaşımlar arasındaki bir başka fark ise, hem benzeşmezlik ölçüsü hemde gradyen değerleri için aykırı değerlere ve gürültüye karşı dayanıklı bir hata fonksiyonunun hesaplamalara eklenmiş olmasıdır. İki yöntemin performanslarının karşılaştırmasına bölüm 4.1'den ulaşılabilir.

Maliyet fonksiyonunun genel formülü aşağıdaki gibidir:

$$C(x, y, d) = (1 - w) * C_{BNZ}(x, y, d) + w * C_{GRAD}(x, y, d) \quad (3.7)$$

Genel formüldeki benzeşmezlik ölçüsü aşağıdaki şekildedir:

$$C_{BNZ}(x, y, d) = \sum_{(i,j) \in N(x,y)} CVL(i, j, d) \quad (3.8)$$

Burada $CVL(i, j, d)$, önceki bölümde sunulan piksel benzeşmezlik ölçüsü kullanılarak her aykırılık düzeyi için hesaplanmış eşleşme maliyetlerinin oluşturduğu fonksiyondur.

Genel formüldeki gradyent terimi,

$$C_{GRAD}(x, y, d) = \sum_{(i,j) \in N_x(x,y)} |\nabla_x I_1(i, j) - \nabla_x I_2(i+d, j)| + \sum_{(i,j) \in N_y(x,y)} |\nabla_y I_1(i, j) - \nabla_y I_2(i+d, j)| \quad (3.9)$$

şeklinindedir. Burada $N(x,y)$, pikseli çevreleyen pencere, ∇_x sağa doğru gradyent ve ∇_y aşağı doğru gradyenttir. (3.7) denklemindeki w terimi, benzeşmezlik maliyeti ile gradyent arasındaki ağırlığı ayarlamak amacıyla kullanılmaktadır. Klaus ve diğ. [45] uygulamasında olduğu gibi, w terimi, sağ/sol kontrol sonucu en fazla pikseli verecek şekilde seçilmektedir.

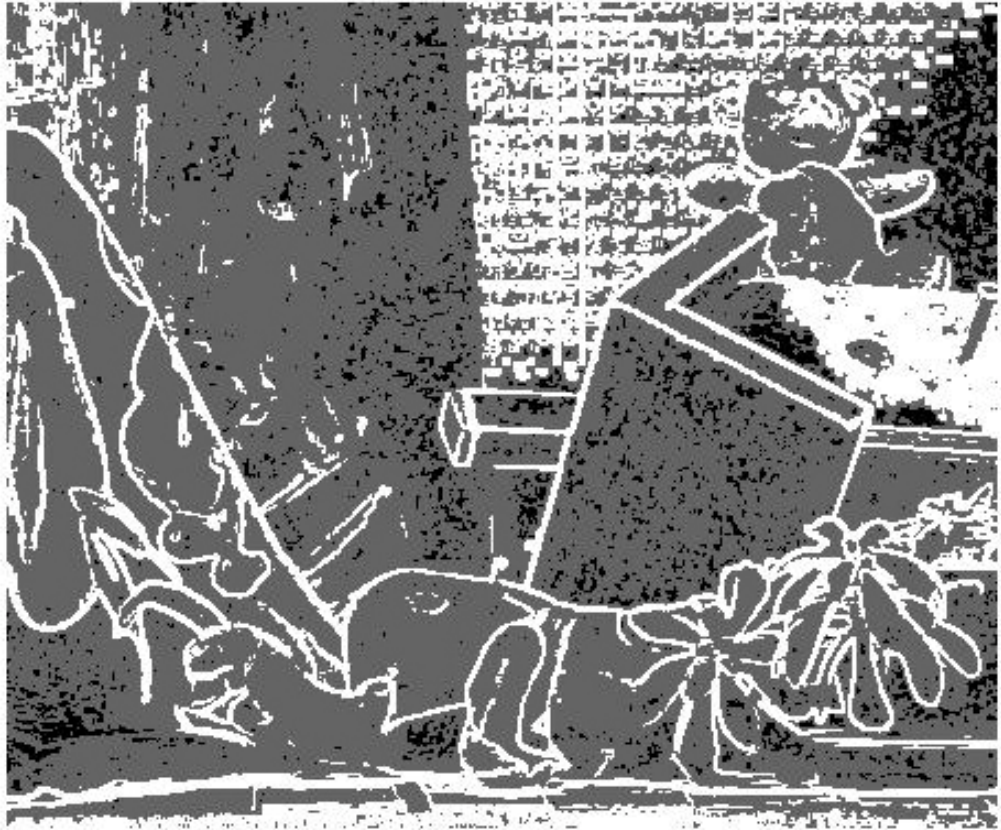
Bu yönteme ek olarak, imgenin farklı uzlamsal sıklıktaki bölgelerinde iki ölçü arasında ince ayar yapmak amacıyla, imgenin 3 farklı bölgeye ayrılması önerilmiştir. Bu amaçla ilk olarak, girdi imgesindeki her bir pikselin, kendisini çevreleyen komşu piksellerle

oluşturduğu pencere içindeki yeğlilik değerlerinin varyansı hesaplanır. Hesaplanan değerlerle imgenin varyans haritası çıkarılır. Daha sonra tüm imge içindeki varyansların medyanı kullanılarak değerler normalize edilir. Normalize edilen varyanslar, belli değer aralıkları kullanılarak üç farklı şekilde etiketlenir. Bu şekilde elde edilen “Teddy” imgesine ait varyans haritası Şekil 3.2’de görülmektedir. Şekildeki beyaz renkli bölümler, normalize varyans değerinin 6’dan yüksek olduğu, dokunun az olduğu siyah bölgeler ise, normalize varyans değerinin 0.08’den küçük olduğu bölümlerdir. Bu bölümlerde eşleşme fonksiyonlarının çok farklı hata oranları verdiği saptanmıştır. Bu nedenle, sağ/sol kontrolü ile yapılan w ağırlığı seçimi, her 3 bölge için ayrı ayrı yapılır.

Hem benzeşizlik hemde gradyen fonksiyonları, aşağıdaki, aykırı verilere karşı gürbüz hata fonksiyonu yardımıyla kırılmıştır.

$$\rho_d(x) = -\ln((1 - e_d)\exp(-|x|/\sigma_d) + e_d) \quad (3.10)$$

Burada, e_d ve σ_d , hata fonksiyonun şeklini kontrol eden parametrelerdir.

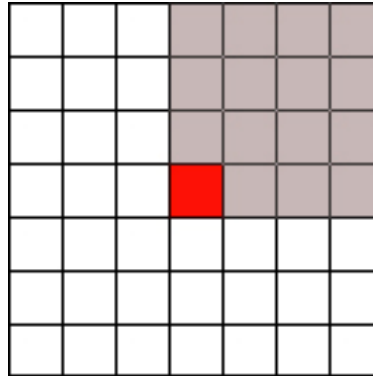


Şekil 3.2 : “Teddy” imgesi için varyans sınıflandırması sonucu

Hesaplama da kullanılacak pencerenin boyutu, resimdeki dokulu yüzeylerin yoğunluğuna göre seçilmelidir. Düşük yoğunluklu imgelerde, daha geniş pencereler kullanılmalıdır. Doku yoğunluğu homojen olmayan imgelerde ise, pencere boyutu seçimi problem haline gelebilir. Bu sorunu çözebilmek amacıyla, bu maliyet fonksiyonunun *çoklu pencereli* sürümü önerilmiştir.

3.1.3. Çoklu Pencereli Eşleşme Maliyet Fonksiyonu

İmgedeki doku yoğunlukları arası farklardan dolayı oluşabilecek hataların önüne geçmek için, maliyet fonksiyonu hesabında kullanılan pencerenin boyutunun değişken olması gerekmektedir. Bu nedenle, hem farklı boyutlarda pencerelerin hesaplarını bünyesinde barındıran hemde en düşük maliyete sahip aykırılık değerinin tutarlılığını artıran bir yöntem önerilmiştir. Bu amaçla ilk olarak, maliyet hesabı sonuçlarının güvenilirlik ölçüsü belirlenir. Maliyet fonksiyonu sonuçları her piksel için en düşükten en yükseye doğru sıralanır. İlk iki sıradaki sonuçlar arasındaki fark belli bir oranı geçemeyenler “problemlı” olarak işaretlenir.



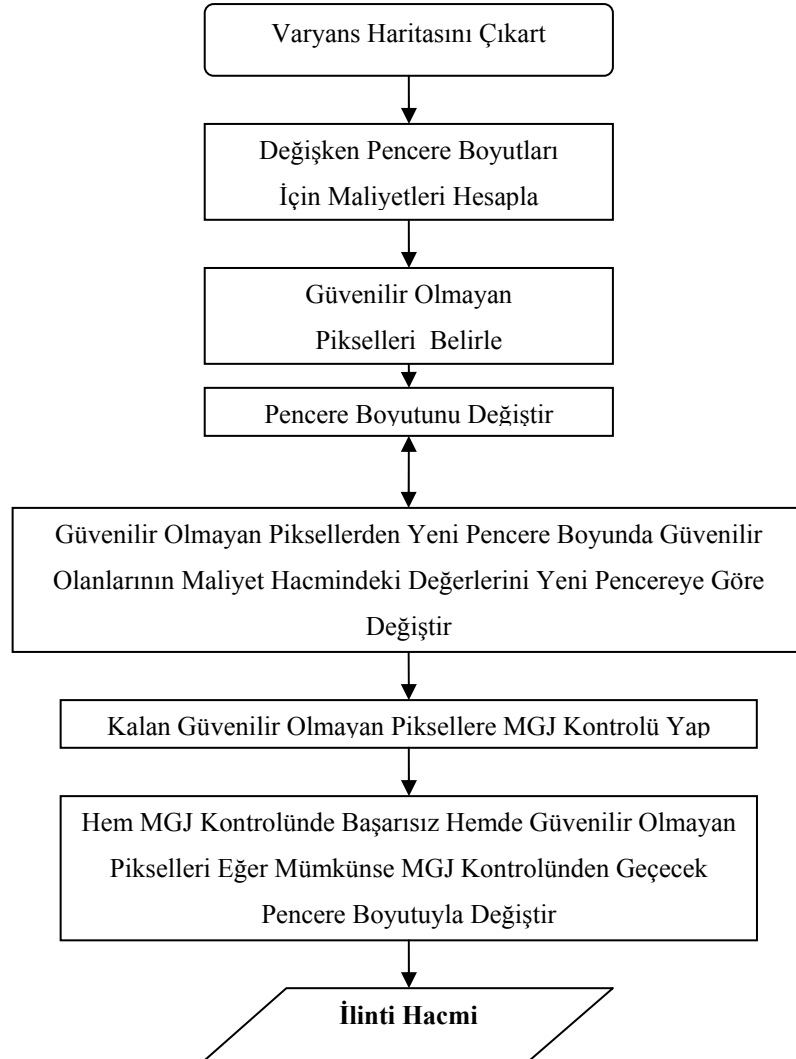
Şekil 3.3 : Standart pencerelere ek olarak kullanılan destek penceresi: kırmızı nokta merkez piksel, gri pikseller ortalamaya katılan diğer pikseller

Algoritmanın birinci adımında 3x3, 5x5, 7x7, 11x11 ve Şekil 3.3’deki pencereler için önceki bölümde kullanılan yöntem yardımıyla maliyetler hesaplanır. Belli bir pencere boyutundan başlayarak yukarıda bahsedilen şekilde problemlı pikseller işaretlenir. Daha sonra diğer pencerelerde işaretli pikseller için maliyetlere bakılır. Eğer yeni maliyet değeri güvenilir ise, o piksel için artık yeni penceredeki değerler kullanılır. Şekil 3.3’deki pencerenin kullanılmasındaki amaç, nesne kenarlarındaki maliyetlerin daha verimli bir şekilde hesaplanabilmesidir.

İkinci aşamada, kalan problemlili pikseller için daha iyi bir pencere boyutu olup olmadığını belirlemek amacıyla, bölüm 2.3.2’de sunulan MGJ yöntemi kullanılır. Bu yöntemde ilk olarak, piksellerin tüm derinlikler için maliyetlerinin ortalaması alınır. Oluşan matris, 3x3 boyutlarında bir pencere yardımıyla taranarak aşağıdaki hesaplama yapılır.

$$Hata = \max(\bar{C}_x - \bar{C}_{x+3}, \bar{C}_x - \bar{C}_{x-3}), \quad (3.11)$$

Burada x , pikselin yatay koordinatı ve C , pencere içindeki eşleşme değerlerinin toplamıdır. Bu işlemin amacı, eşleşme maliyetlerini, çevresiyle uyumunu kontrol etmektir ve hata oranı belli bir limitin üstünde olan pikseller işaretlenir. .



Şekil 3.4 : Çoklu Pencere Eşleşme Maliyet Fonksiyonu Algoritmasının Adımları

Hem MGJ yöntemiyle hemde güvenilirlik yöntemiyle problemlili olan pikseller, diğer pencere boyutlarında tekrar incelenir. Eğer başka bir pencere boyutunda MGJ ölçüsüne göre problemsiz olarak işaretleniyorsa, söz konusu pikseller için artık o pencere boyutu kullanılır.

Çoklu pencere yaklaşımı ile normal hesaplamanın başarı oranları bölüm 4.1.'de sunulmuştur.

3.2. BÖLÜTLEME VE ENERJİ MİNİMİZASYONU

Piksellerin eşleşme maliyetleri çıkarıldıktan sonra, her piksel için uygun aykırılık değerinin seçilmesi gerekir. Bu problem genellikle, bir enerji fonksiyonu belirlenip daha sonra tüm imgenin bu fonksiyona göre enerjisinin minimize edilmesiyle çözülür. Ancak NP-Zor olan bu problem, döngülü minimizasyon yöntemlerinin kullanılmasını gerektirir. Bu noktada bölüm 2.3.4'te belirtilen optimizasyon yöntemlerinden biri kullanılır. Son dönemlerin başarılı algoritmalarında genellikle *inanç yayılımı* yöntemi kullanılmaktadır. Standart inanç yayılımı yönteminde hesaplama süresi yaklaşık,

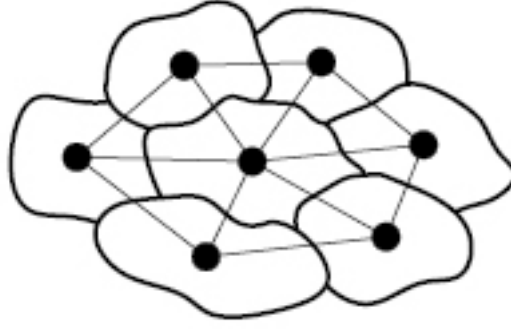
$$O(nk^2T) \quad (3.11)$$

olarak hesaplanır. Burada n, düğüm sayısı k, etiket sayısı ve T, iterasyon sayısıdır. Ancak ekte önerilen yöntemler kullanılarak bu sürenin $O(n^{1/2}kT)$ 'ye kadar düşürülmesi mümkündür. Hem bu hesaplama süresini düşüren, hemde algoritmanın genel başarısını arttıran bir başka yaklaşım, renk bazlı bölütlemedir. “Middlebury” test ortamındaki[71] karşılaştırma havuzunda önde olan tüm sistemlerin ortak özelliği, bölütleme kullanılmış olmasıdır. Renk bazlı bölütleme için kullanılan popüler yöntem *ortalama-kaymalı* bölütlemedir [77].

Bölütleme bazlı yaklaşımlarda, aykırılık değerinin bölüt boyunca düz olduğu varsayılır. Piksel yerine çok daha az sayıdaki bölütlerin kullanılması, enerji minimizasyonu algoritmalarında önemli ölçüde işlem tasarrufu sağlamaktadır.

3.2.1. Bölüt Bazlı İnanç Yayılımı

Stereo ve düzlük kısıtları, Şekil 3.3'de grafiksel olarak gösterilen, düğüm noktalarının bölütler olduğu, markov rastgele alanıyla tanımlanabilir.



Şekil 3.5 : Stereo ve düzlük kısıtları MRA kullanarak modellenmiştir. Her düğüm bir bölütü temsil eder ve komşu bölütler kenarlarla birleştirilmiştir.

Her düğüm bir s bölütüne aittir ve $(s, t) \in N$ dahilindeki komşu t bölütleriyle kenarlarını paylaşır. Burada N tüm komşu bölütlerin kümesidir. Düğümlerin durum sayıları, derinlik düzeyi sayısına eşittir. MRA için derinlik atamalarının kalitesi aşağıdaki enerji fonksiyonu kullanılarak ölçülebilir.

$$E(f) = \sum_{(p,q) \in N} V(f_p, f_q) + \sum_{p \in P} D_p(f_p) \quad (3.12)$$

Burada $D_p(f_p)$, p bölütüne f değerini vermenin maliyeti ve $V(f_p, f_q)$ f_p ile f_q değerlerini komşu bölütlere atama maliyetidir. $V(f_p, f_q)$ ayrıca düzlük fonksiyonu olarak anılır.

Enerji minimizasyonundaki en önemli adım, düzlük fonksiyonunun belirlenmesidir. Derinliğin, bölütler arasında yumuşak bir şekilde değiştiği varsayılır. İki bölüt arasındaki değişimin fazla olması durumu cezalandırmak amacıyla aşağıdaki terim kullanılabilir

$$\min(|d_p - d_q|, T_d) \quad (3.13)$$

Burada d_p ve d_q , p ve q bölütlerinin derinlik değerleridir. Belirlenen cezanın değeri T_d ile kontrol edilir. Bu terime ek olarak, benzer renkteki bölütlerin aynı derinlik bilgisine sahip olacağı, tersi düşünüldüğünde, farklı renkteki bölütlerin farklı derinlik değerlerine sahip olabileceği varsayımıyla, derinlik farklarına göre verilen cezanın büyüklüğü, renk benzerlikleriyle kontrol edilebilir. Bu amaçla ilk olarak, ele alınan bölütün tüm komşu bölütler ile renk yeğinliği farkları hesaplanır. Daha sonra bu farklar, tüm resimdeki

maksimum farka bölünerek normalize edilir. Elde edilen değer ρ_{pq} , aşağıdaki terimle düzlük fonksiyonuna katılır.

$$\min(|1 - \rho_{pq}|, T_c) \quad (3.14)$$

Burada T_c , renk farkı değerini kontrol amaçlıdır. Yukarıdaki terimler birleştirildiğinde düzlük fonksiyonu aşağıdaki şekli alır.

$$V(f_p, f_q) = \lambda_{düz} \min(|dp - dq|, T_d) \min(|1 - \rho_{pq}|, T_c) \quad (3.15)$$

Oluşan enerji fonksiyonu, inanç yayılımı algoritmasıyla minimize edilir. Bu yöntem, enerjinin mesajlar aracılığıyla komşu düğümlere yayıldığı varsayar. Herhangi bir t anında, p düğümünün q düğümüne gönderdiği mesaj,

$$m_{pq}^t(f, q) = \min_{f_p} \left(V(f_p, f_q) + D_p(f_p) + \sum_{s \in N(p) \setminus q} m_{sq}^{t-1}(f_p) \right) \quad (3.16)$$

şeklindedir. Burada $N(p) \setminus q$, p 'nin q dışındaki komşularını simgeler. T döngüden sonra inanç vektörü aşağıdaki şekli alır.

$$b_q(f_q) = D_p(f_q) + \sum_{p \in N(q)} m_{pq}^T(f_q) \quad (3.17)$$

Daha sonra $b_q(f_q)$ terimini minimize edecek derinlik değeri seçilir. Seçilen derinlik değerlerinden aykırılık haritası oluşturulur.

3.3. ÖRTME HARİTASININ ÇIKARILMASI

Stereo ilişkilendirme algoritması tasarlarken göz önünde bulundurulması gereken en önemli problemlerden biri örtme bölgeleridir. Bu bölgeler, aynı sahneye ait iki görüntü arasında, nesnelerin birbirlerinin üzerine gelerek görüntülerini kapatmasıyla oluşur. Bu bölgeye denk gelen bir pikselin, doğal olarak diğer resimde eşi olamayacaktır.

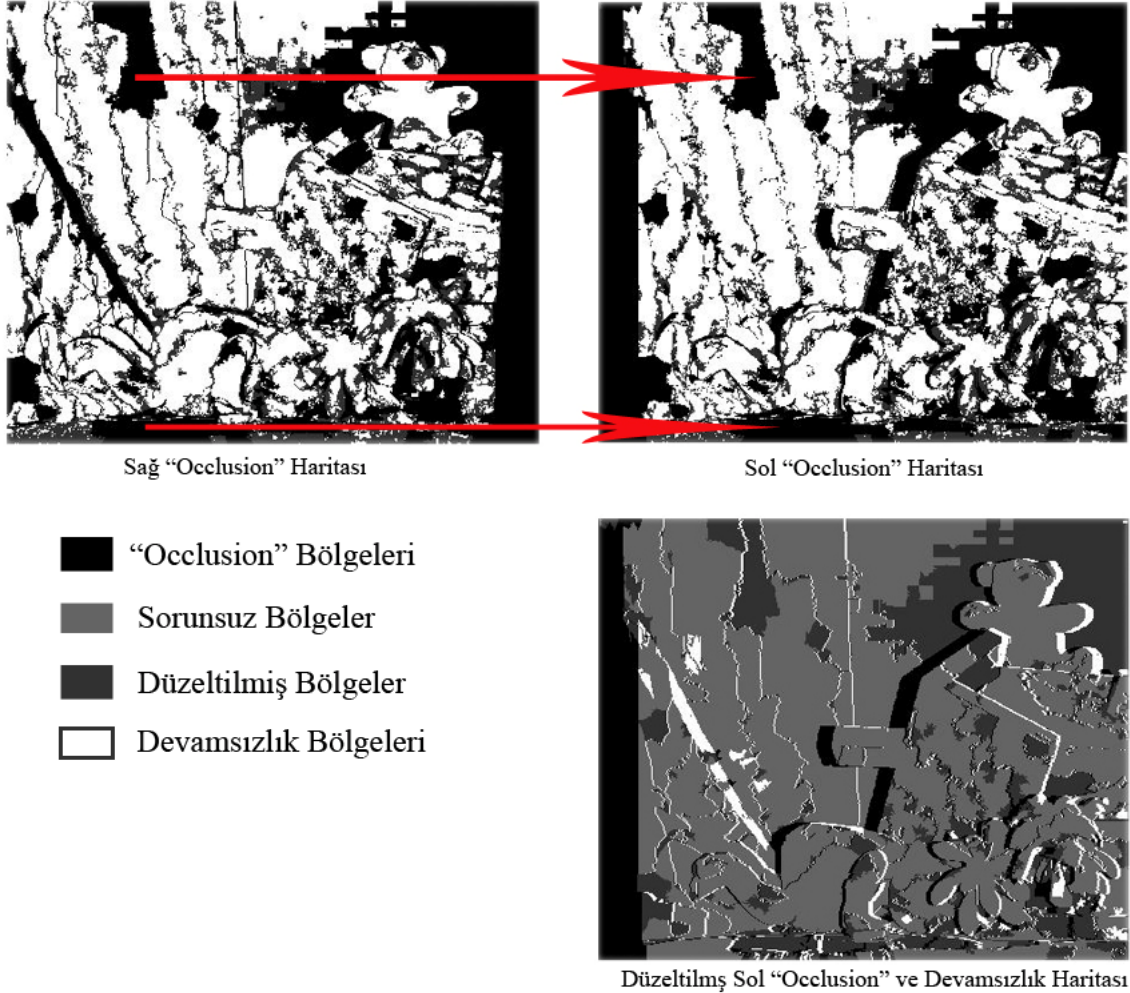
Genel algoritmanın başarısını arttırmak için, bu bölgelerin etkin bir şekilde belirlenmesi yararlı olur. Bu amaçla farklı bir yaklaşım önerilmiştir. Bölüm 2.3.3.'de özetlenen örtülmüş bölgeleri belirleme algoritmalarını baz alan bu yaklaşımın en önemli özelliği,

her iki imgeden çıkarılan örtme haritalarını, birbirlerinin hata oranlarını düşürecek şekilde birleştirmesidir.

İlk olarak imgelerin örtme haritaları, sağ/sol kontrol yöntemiyle çıkarılır. Bu yöntemde, sol imgedeki noktanın derinlik değeri, noktaya ait x koordinatından çıkartılır ve sağ imgede bu koordinat değerlerine sahip nokta ile derinlikleri karşılaştırılır. Eğer değerler eşit değilse bu nokta örtülmüş olarak işaretlenir.

$$Occ = \min\left(\left(d^R_{(x'_L+d^L_{X_L})} - d^L_{X_L}\right), 1\right), \quad (3.18)$$

Bir imgedeki örtme bölgesinin kenarının, diğer imgede devamsızlık bölgesi olduğu varsayımına [57] ek olarak, bir imgede örtülmüş olarak işaretlenmiş bir pikselin diğerinde örtülmüş olamayacağı varsayılmıştır.



Şekil 3.6 : Hatalı örtülmüş bölgelerin ayıklanması

Sağ haritadaki örtülmüş noktaları, sol haritaya yansıtılır. Yeni varsayım doğrultusunda, yansıtılan nokta sol haritada normal bir piksel ise devamsızlık noktası olarak işaretlenir. Eğer, sol haritadaki nokta örtülmüş olarak işaretli ise bu işaret kaldırılır. Bu yaklaşımın örtme haritası kestirimine yaptığı katkı bölüm 4.2’de sayısal olarak gösterilmiştir.

Şekil 3.4’te görüldüğü üzere, yukarıdaki yaklaşım sayesinde, hem örtme hemde devamsızlık bölgelerini bulmak mümkündür. Sadece örtme bölgelerine ihtiyaç duyulduğunda, bölütlerin kenarlarında oluşan çizgilerin temizlenmesi, sonucu olumlu yönde etkilemektedir. Bu noktada yapının bölütlere ayrılmış olmasından faydalanılarak, içinde az sayıda örtülmüş pikseli içeren bölütlerdeki pikseller temizlenir.

3.4. EĞİMLİ DÜZLEMLERLE AYKIRILIK KESTİRİMİ

Görüntüdeki tüm nesnelere, imge düzlemine paralel derinlik düzlemleri oluşturamaz. Bu tip yüzeylerin, bölütlere sabit derinlik değerleri verilerek kestirilmesi, hatalara neden olabilir. Bu hataların giderilmesi için kullanılan en yaygın yöntem, bölütlere düzlemler atanmasıdır. Bu düzlemlerin kestirilmesi için, yüksek doğruluk yüzdeleri noktalar kullanılmalıdır. Bu amaçla, bölüt dahilindeki piksel değerleri arasındaki aykırı değerlerin ayıklanması için hem istatistiksel hemde mantıksal bir yaklaşım önerilmiştir.

3.4.1. Piksellerin Sınıflandırılması

Yang ve diğ. [68] makalesindeki yaklaşıma benzer bir şekilde, eşleşme maliyet fonksiyonunda kullandığımız güvenilirlik ölçüsününü, pikselleri sınıflandırmak için de kullanabiliriz. İlk olarak, görüntülerden elde edilen eşleşme maliyetleri hacmine, “kazanan hepsini alır” algoritması uygulanır. Bu yöntemde, uygulanan piksel için en düşük maliyete sahip değer seçilir. Oluşan derinlik haritalarına, sal/sol kontrolü uygulanır. Ancak bu sefer, örtülmüş olarak işaretlenmeyen noktaların güvenilirlik oranına bakılır. Belli bir oranın üstündekiler, tutarlı, diğerleri tutarsız olarak işaretlenir.



Şekil 3.7 : PİKSEL SINIFLANDIRMASI

3.4.2. Aykırı Piksellerin Ayıklanması

Tutarlı olarak işaretlenen pikseller arasında, aykırı değerler olabilir. Aykırı değerleri temizlemek için, 2 aşamalı bir algoritma önerilmiştir.

Birinci aşamada, her bölüt içindeki tutarlı piksellerin ortalama mutlak sapması hesaplanır. Ortalama ile farkı, ortalama mutlak sapmasının 4 katından büyük olan pikseller elenir.

İkinci aşamada, bölüt içindeki tutarlı değerlerin aynı veya ardışık sayılar olabileceği varsayıma dayanarak, içerdiği değerler arasındaki fark birden fazla olan bölütler işaretlenir. Daha sonra, işaretli bölütler teker teker ele alınarak, içerdiği tutarlı değerler komşu bölütlerde aranır. Eğer aranan değer komşularında da yoksa, o değere sahip piksellerin işareti tutarsızla çevrilir. Değerlerin komşu bölütlerde aranmasının amacı, renk bölütlemesi hatası nedeniyle, incelenen bölüte denk gelmiş tutarlı piksellerin korunmasıdır. Bu yöntemle birçok aykırı değer saptanabilmekte ve bölütler için kestirilecek düzlemlerin hata oranları düşürülmektedir.

3.4.3. Mevcut Bölütlerin Yeniden Yapılandırılması

Sistemin bir sonraki aşamasında, her bölüt için, tutarlı piksellerin değerleri kullanılarak düzlemler kestirilecektir. Kestirilen düzlem ve bölüt sayıları, enerji minimizasyonu aşamasındaki işlem sayısını doğrudan etkilemektedir. Hatta bazı teknikler için yapılacak işlem sayısı, düzlem sayısının karesi ile doğru orantılıdır. Bu nedenle, yeteri kadar bilgi

içermeyen ya da içindeki verilerin hata oranı yüksek olabilecek bölütlerin kaldırılması, hem işlem yükünü hafifletecek hemde sistemin doğruluk oranını arttıracaktır.

Bölütlerin içerdiği verinin kalitesini, önceki bölümde işaretlediğimiz tutarlı piksellerin oranıyla ölçmek mümkündür. Düşük tutarlı piksel oranına sahip bölütlerin, genel hatanın büyük bir kısmını içerdiği gözlemlenmiştir. Bu bölütlerin belirlenmesi ve mümkün olduğu durumlarda komşu bölütlere dağıtılması amacıyla 4 aşamalı bir algoritma önerilmiştir.

- Birinci aşamada, içinde sadece bir tutarlı derinlik değeri barındıran 50 pikselden küçük veya tutarlı piksel oranı %50'nin üzerinde 200 pikselden küçük bölütler, kendilere komşu, tutarlı piksel oranı %50 üzerinde olan ve aynı derinlik değerini barındıran bölütlerle birleştirilir.
- İkinci aşamada, önceki bölümdesine benzer bir şekilde, içerdiği değerler arasında birden fazla fark olan veya bu değere sahip tutarlı piksel oranı bölüt içindeki tüm tutarlı piksellerin %20'sinden küçük olan bölüt içindeki problemlili değere sahip pikseller kaldırılır ve birinci aşama tekrarlanır.
- Üçüncü aşamada, 50 piksel altı, değerleri arasında boşluk veya tutarlı piksel oranı %5'in altında olan bölütlerin içindeki tutarlı piksellerin işaretleri kaldırılır.
- Son aşamada, sol resim için resmin sol tarafında, sağ resim için resmin sağ tarafındaki örtülmüş piksel değeri %80'nin üzerindeki bölütler komşuları ile birleştirilir.

Son aşamadaki birleştirme işleminin amacı, diğer imgede doğru eşleşmesinin bulunması imkansız olan piksellerin hesaba katılmayarak, tutarlı pikseller aracılığıyla atanacak düzlemler kullanılarak kestirilmesidir. Bu yaklaşım referans imgesinin solundaki bölgelerin doğru biçimde kestirilebilmesine olanak sağlamalıdır.

3.4.4. Düzlemlerin Kestirimi

Tutarlı noktaların belirlenmesinden sonra, bu noktalar yardımıyla düzlemler kestirilir. Düzlemlerin kestirimi için RANSAC algoritması kullanılmıştır.

Bazı durumlarda aynı düzlem, komşu bölütler boyunca ilerleyebilir. Bu tip düzlemleri kestirebilmek için aşağıdaki algoritma önerilmiştir.

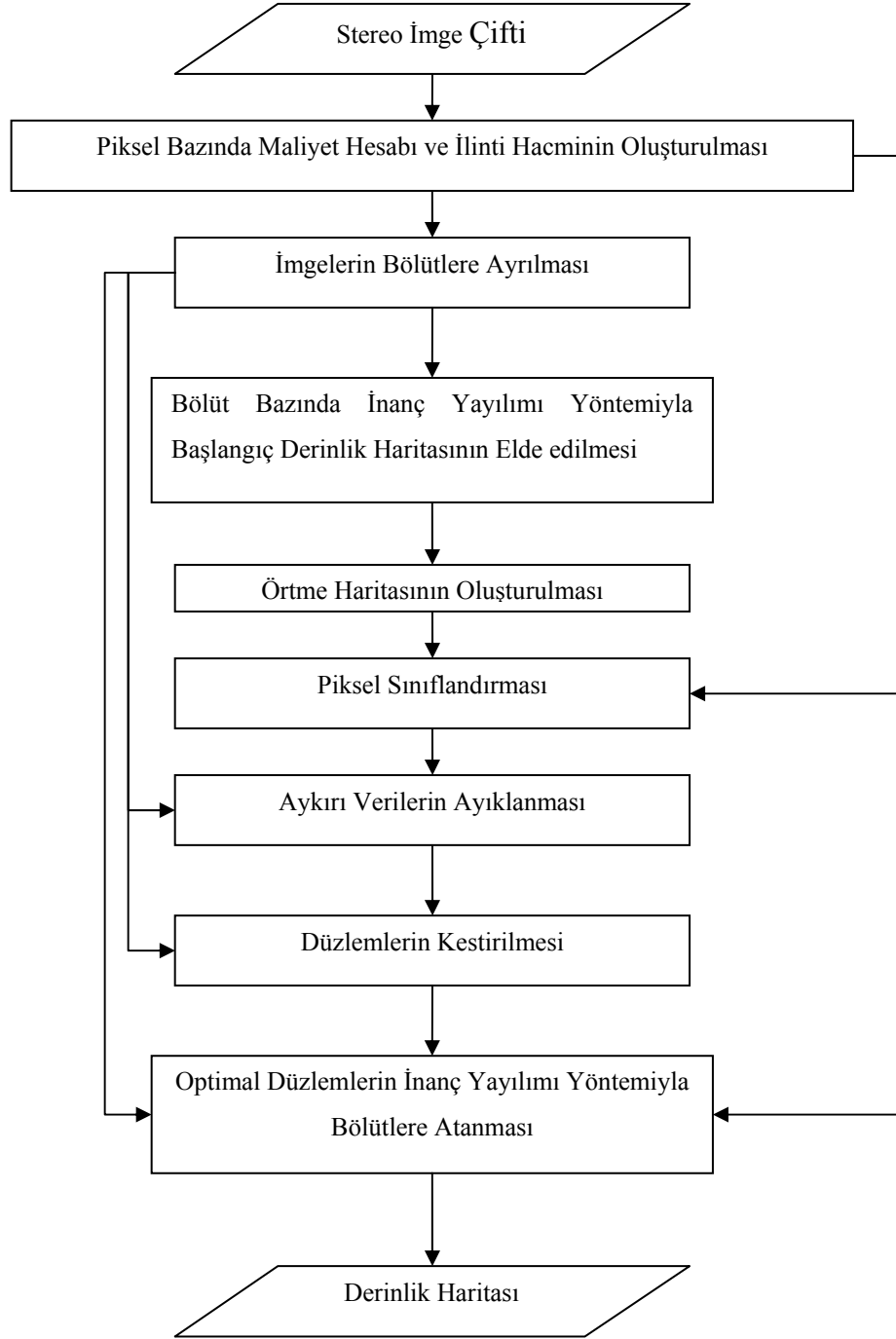
- Her bölüt için, tutarlı pikseller derinlik değerlerine göre gruplanır ve (x, y) düzleminde ağırlık merkezleri alınır.

- Bölütün içerdiği ağırlık merkezleri sıralandığında, derinlik değerleri herhangi bir tepe veya dip oluşturmuyorsa devam edilir. Aksi durumda bu bölütte başka işlem yapılmaz.
- Üzerinde işlem yapılan bölüt, ikişer ikişer komşu bölütlerle gruplanır, ağırlık merkezleri x yada y düzleminde, tepe veya dip oluşturmuyorsa, iki bölütün içerdiği tüm tutarlı değerler yardımıyla yeni düzlem kestirilir.
- Son olarak, tepe veya dip oluşturmayan, maksimum sayıda komşu ile birleştirilip, oluşan değerler kümesinden yeni düzlem kestirilir.

3.4.5. Enerji Minimizasyonu

Oluşan sistemdeki ideal düzlemleri bulmak amacıyla bölüm 3.1.1'dekine benzer şekilde inanç yayılımı algoritması kullanılır. Öte yandan, ilk seferdekine göre elde çok daha fazla veri vardır. Örneğin, bölüm 3.1.2'de oluşturduğumuz örtme haritası aracılığıyla, düzlük fonksiyonun şekli ve katsayıları bu bölgelerde farklı ayarlanabilir. Bu nedenle (3.15)'teki düzlük fonksiyonuna ek koşullar getirilmiştir. Komşu bölütler arasındaki örtülmüş piksel farkının belli bir oranın altında olup olmamasına göre T_c , T_d ve $\lambda_{düz}$ parametrelerinin değerleri değiştirilir. Ancak birçok düzlem arasından seçim yapılacak olması, minimizasyonun işlem yükünü arttırmaktadır.

Son enerji minimizasyonu adımı eklendikten sonra önerilen algoritma aşağıdaki şekli alır.



Şekil 3.8 : Önerilen Stereo İlişkilendirme Algoritmasının Temel Yapısı

4. BULGULAR

Üçüncü bölüm boyunca, stereo ilişkilendirme algoritmalarında kullanılacak birçok yeni teknik önerilmiştir. Bu bölümde önerilen tekniklerin, Matlab ortamında hazırlanmış kodlar yardımıyla mevcut yöntemlerle karşılaştırılması yapılacaktır. Karşılaştırma amacıyla “Middlebury” test ortamındaki [71] imgelerden yararlanılmıştır. Bu imgeler ve derinlik haritaları üzerinde belirlenmiş devamsızlık bölgeleri ve örtülmüş bölgeler şekil 4.1’de gösterilmiştir.

Performans kriteri olarak iki ayrı bölgedeki hata oranları çıkarılmıştır. Bu bölgeler; *devamsızlık* ve *örtülmüş olmayan* olarak belirlenmiştir. Değerlendirme yapılırken, olması gereken değer ile kestirilen değer arasındaki fark birden büyük olan pikseller hatalı olarak kabul edilmiştir. “Tsukaba” imgelerinde 15, “Teddy” ve “Cones” imgelerinde 59 derinlik düzeyi vardır.

4.1. EŞLEŞME MALİYET FONKSİYONLARI KARŞILAŞTIRMASI

Önerilen maliyet fonksiyonlarının performanslarını ölçmek amacıyla, 2008 Haziran itibariyle, “Middlebury” test ortamında ilk sırada bulunan algoritmanın eşleşme maliyet fonksiyonu, piksel benzeşmezlik ölçüsü, mutlak farkların toplamı, renk ağırlıklı destek penceresi yöntemi ve önerilen yöntemler karşılaştırılmıştır. Karşılaştırmalarda dört farklı yöntem kullanılmıştır.

İlk olarak, eşleşme maliyeti hacimlerine KHA (Kazanan Hepsini Alır) algoritması uygulanmış ve sonuçlar karşılaştırılmıştır. (Şekil 4.1, 4.5, 4.9, Tablo 4.2, 4.6, 4.10)

Daha sonra kestirilen her derinlik haritasına sağ/sol kontrolü yapılarak hatalı pikseller elenmiş ve kalan piksellerin hata oranları ayrıca verilmiştir. (Şekil 4.2, 4.6, 4.10, Tablo 4.3, 4.7, 4.11)

Üçüncü olarak, piksel bazında inanç yayılımı algoritması kullanılarak, enerji fonksiyonun minimum değeri bulunmaya çalışılmış ve minimumu veren derinlik değerlerinden aykırılık haritası oluşturulmuştur. (Şekil 4.3, 4.7, 4.11, Tablo 4.4, 4.8, 4.12)

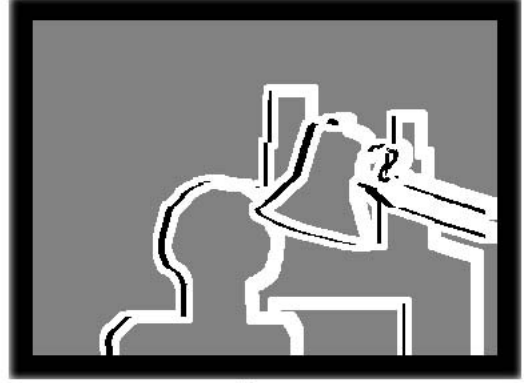
Son olarak, her imge seti için tablo 4.1’deki sabit parametreler kullanılarak, bölüm 3.2’de detayları verilmiş olan enerji minimizasyonu algoritması uygulanmış ve sonuçlar karşılaştırılmıştır. (Şekil 4.4, 4.8, 4.12, Tablo 4.5, 4.9, 4.13)

Tablo 4.1: İnançYayılmıAlgoritması Parametreleri: “N” iterasyonsayısı.

| N | T_d | T_c | λ |
|---|-------|-------|-----------|
| 5 | 2 | 0.5 | 0.2 |



(a)



(b)



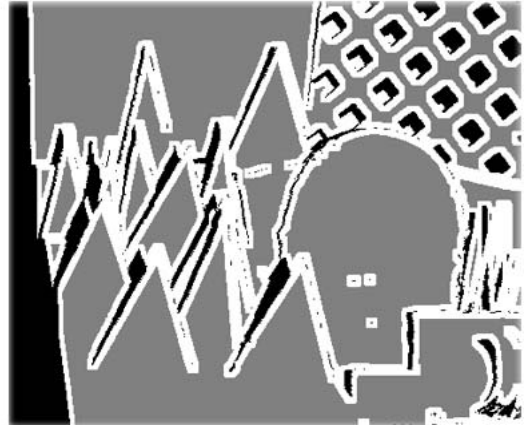
(c)



(d)



(e)



(f)

Şekil 4.1: Karşılaştırmalarda kullanılan imgeler ve derinlik haritalarının; örtülmüş (siyah), devamsızlık (beyaz) ve örtülmüş olmayan (gri) bölgelere ayrılmış hali: a-) “Tsukaba” imgesi b-) “Tsukaba” derinlik bölgeleri c-) “Teddy” imgesi d-) “Teddy” derinlik bölgeleri d-) “Cones” imgesi e-) “Cones” derinlik bölgeleri

Tez kapsamında ele alınan yeni maliyet fonksiyonları aşağıdaki şekilde tanımlanmıştır.

- **Öneri 1:** Benzeşmezlik ölçüsünü ve gradyenti kullanan fonksiyon (3.7)
- **Öneri 2:** Birinci önerideki fonksiyonun, imgenin varyansına göre 3 farklı bölgede ince ayarı yapılmış şekli
- **Öneri 3:** İkinci önerilen fonksiyonun, çoklu pencere yöntemi ile geliştirilmiş versiyonu. (Bölüm 3.1.3.)

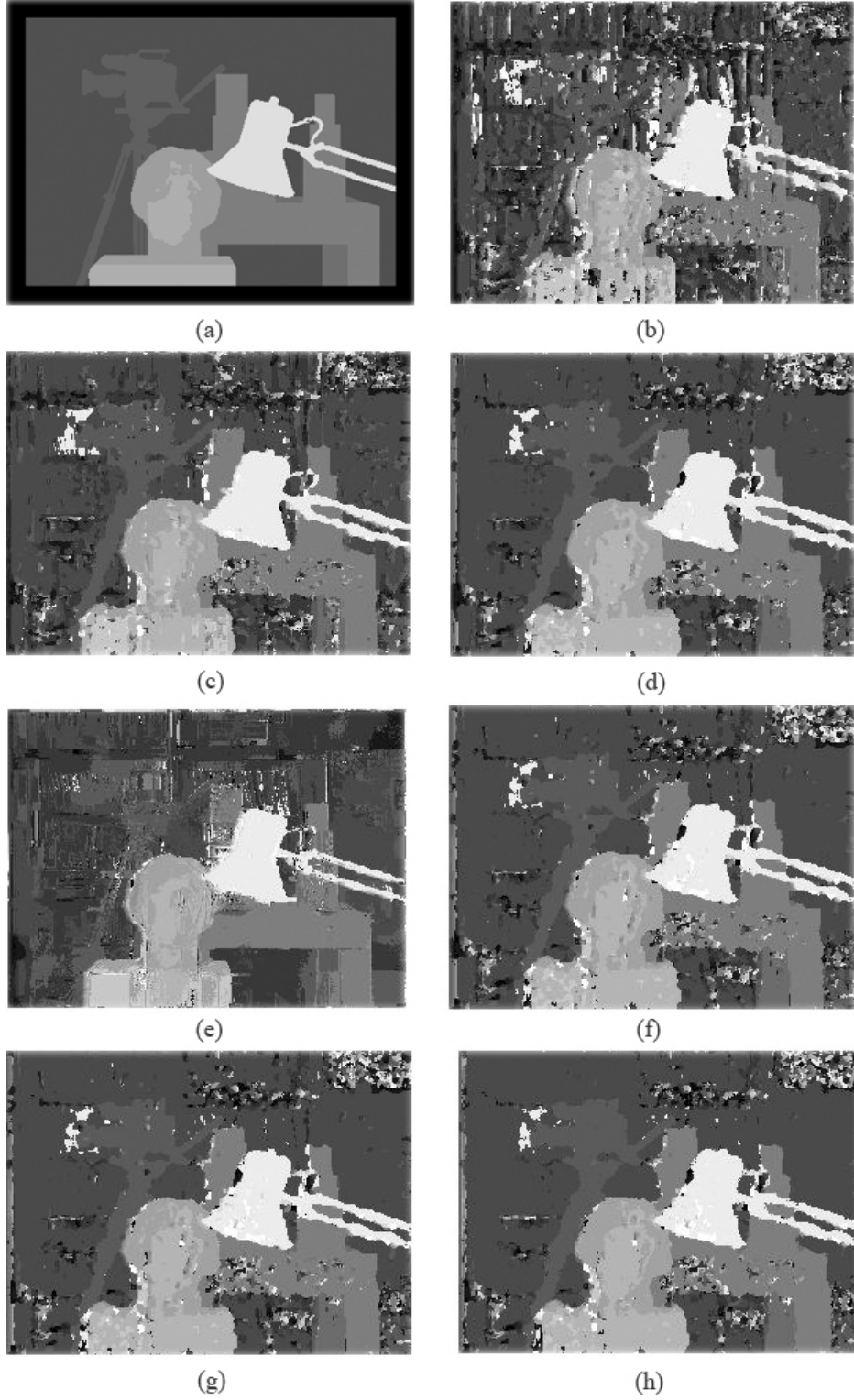
4.1.1. “Tsukaba” İmgelerine ait Bulgular

Tablo 4.2 : KHA algoritmasına göre hata oranları

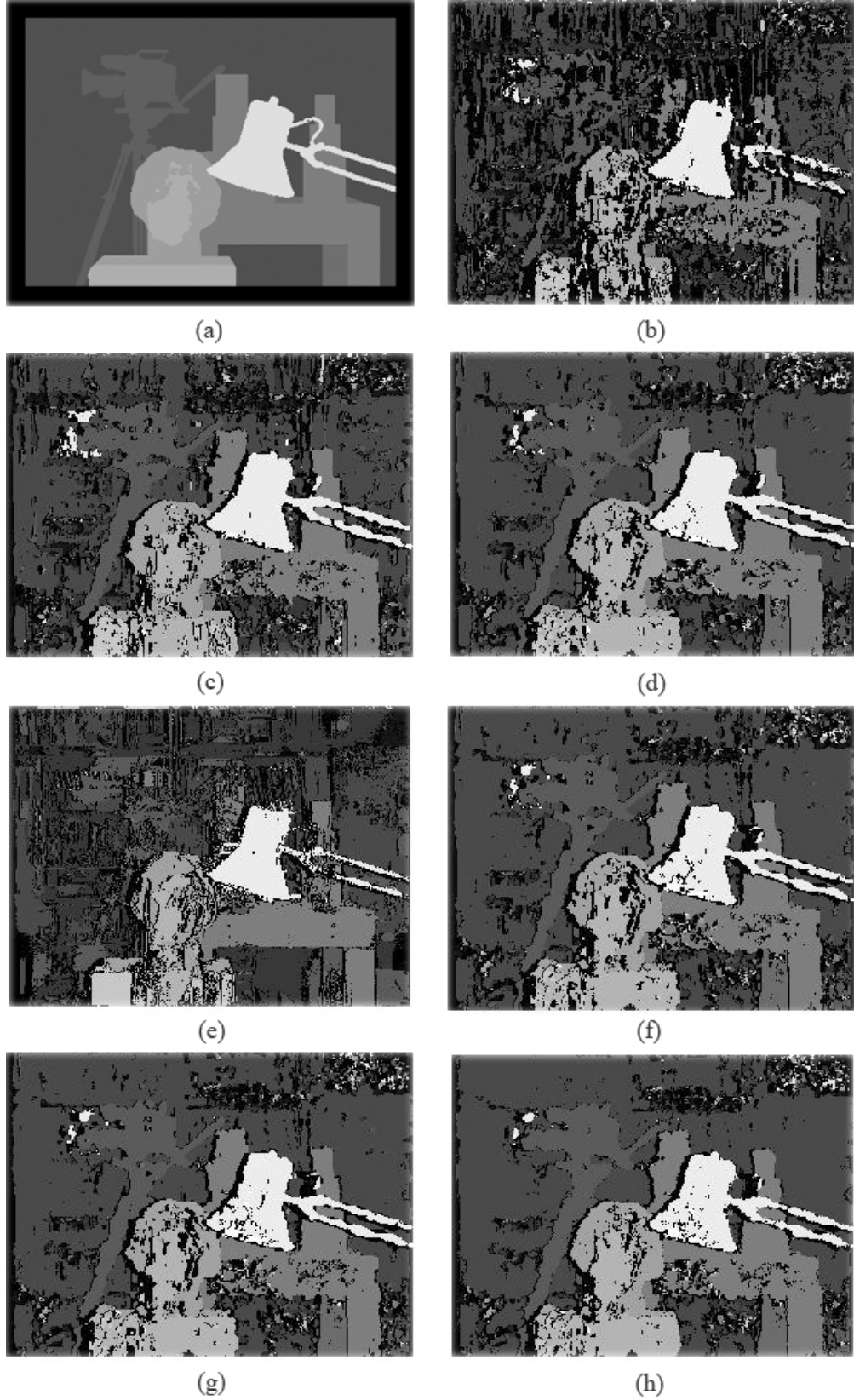
| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 12,18 | 10,11 | 16,03 |
| Öneri 2 | 12,41 | 10,34 | 16,2 |
| Öneri 3 | 11,26 | 9,22 | 16,01 |
| SadGrad[47] | 11,58 | 9,52 | 15,35 |
| SAD | 13,55 | 11,66 | 15,33 |
| Benzeşmezlik | 22,32 | 20,64 | 25,4 |
| Renk Ağırlıklı[56] | 9,45 | 7,62 | 13,05 |

Tablo 4.3 : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri

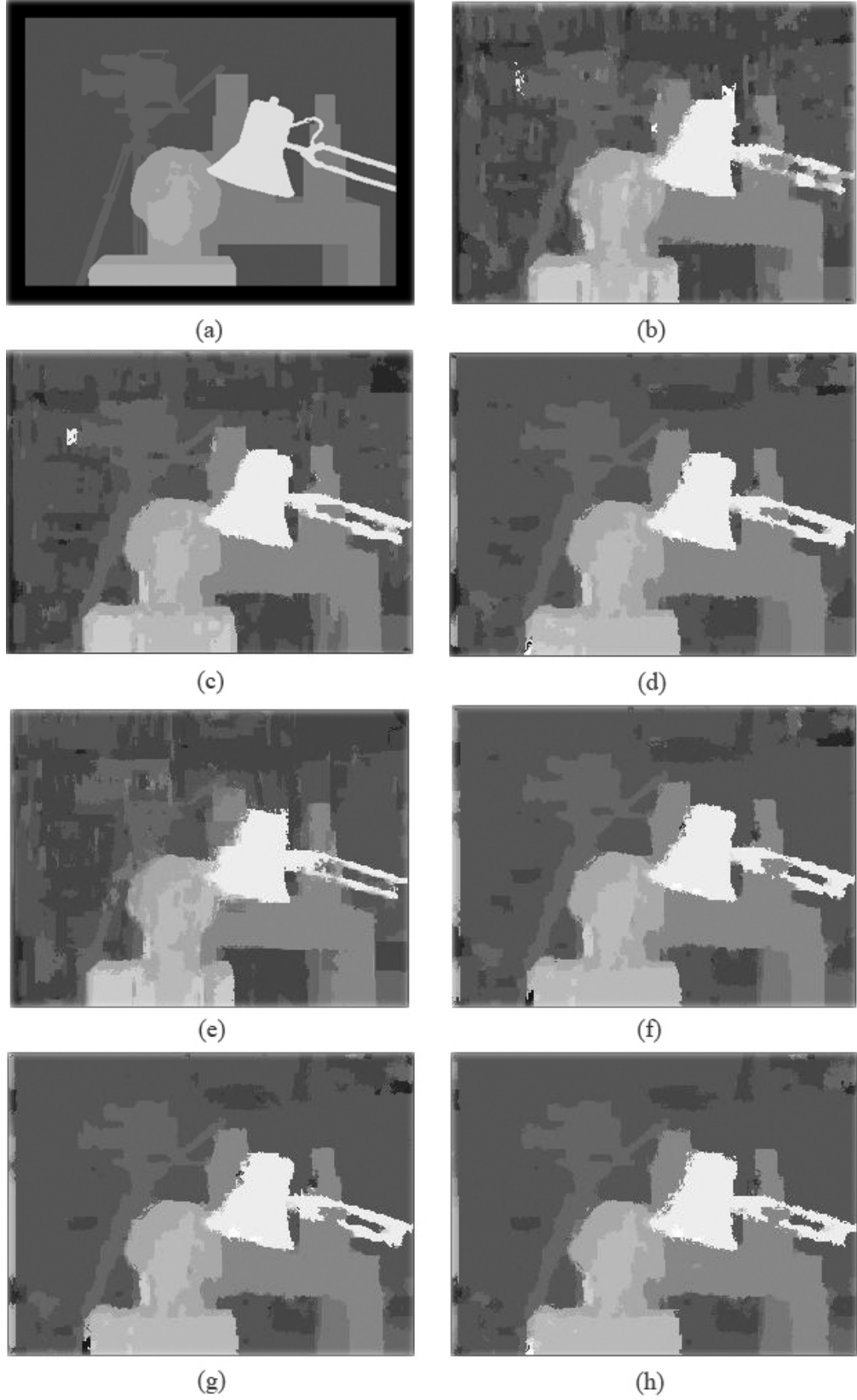
| | Tüm Pikseller | Kestirim(%) | Örtülmemiş | Kestirim(%) | Devamsızlık Böl | Kestirim(%) |
|--------------------|---------------|-------------|------------|-------------|-----------------|-------------|
| Öneri 1 | 5,45 | 82,54 | 4,87 | 84,14 | 6,33 | 78,96 |
| Öneri 2 | 6 | 83 | 4,96 | 84,58 | 8,24 | 79,25 |
| Öneri 3 | 5,14 | 86,63 | 4,33 | 88,06 | 10,06 | 81,36 |
| SadGrad[47] | 5,56 | 83,73 | 4,94 | 85,3 | 8,33 | 81,34 |
| SAD | 4,48 | 67,33 | 3,65 | 68,43 | 7,64 | 72,97 |
| Benzeşmezlik | 4,21 | 47,95 | 3,55 | 48,84 | 5,39 | 51,15 |
| Renk Ağırlıklı[56] | 3,95 | 72,59 | 3,57 | 73,97 | 6,48 | 68,34 |



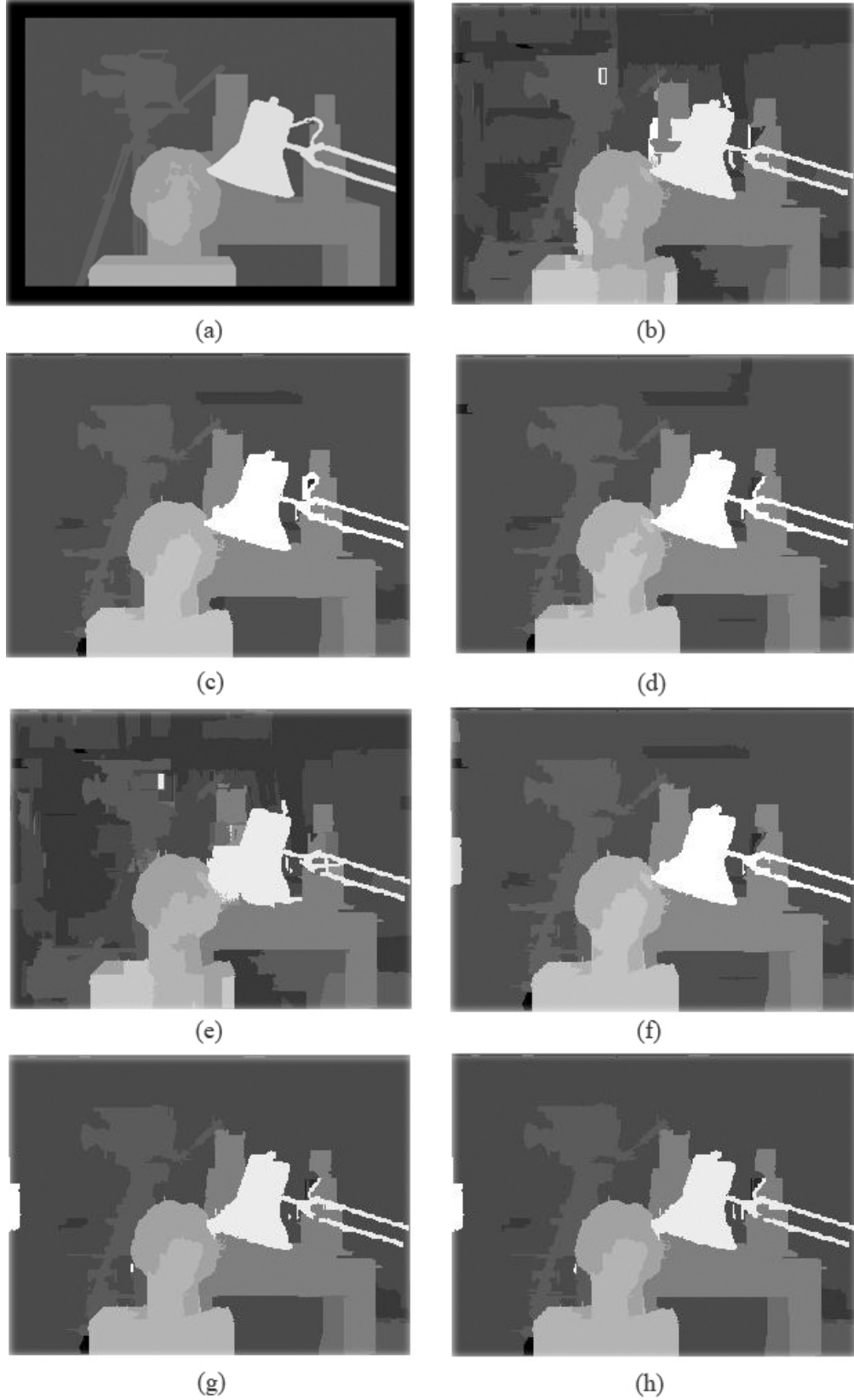
Şekil 4.2 : KHA algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.3 : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.4 : Pksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.5 : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**

Tablo 4.4 : Piksel Bazlı İnanç Yayılımı algoritmasına göre hata oranları

| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 4,99 | 3,05 | 14,85 |
| Öneri 2 | 4,91 | 2,94 | 14,86 |
| Öneri 3 | 4,98 | 3,11 | 16,65 |
| SadGrad[47] | 4,93 | 2,94 | 14,69 |
| SAD | 5,49 | 3,5 | 13,25 |
| Benzeşmezlik | 6,49 | 4,57 | 17,7 |
| Renk Ağırlıklı[56] | 5,31 | 3,53 | 15,08 |

Tablo 4.5 : Bölüt Bazlı İnanç Yayılımı algoritmasına göre hata oranları

| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 1,92 | 1,53 | 8,04 |
| Öneri 2 | 2,44 | 2,08 | 9,27 |
| Öneri 3 | 2,1 | 1,68 | 8,84 |
| SadGrad[47] | 2,01 | 1,56 | 8,2 |
| SAD | 2,27 | 1,85 | 7,77 |
| Benzeşmezlik | 3,58 | 2,8 | 12 |
| Renk Ağırlıklı[56] | 3,51 | 3,04 | 11,98 |

4.1.2. “Teddy” İmgelerine ait Bulgular

Tablo 4.6 : KHA algoritmasına göre hata oranları

| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 30,25 | 22,58 | 32,14 |
| Öneri 2 | 30,25 | 22,55 | 32,25 |
| Öneri 3 | 25,09 | 17,24 | 29,62 |
| SadGrad[47] | 29,42 | 21,69 | 30,28 |
| SAD | 36,61 | 29,53 | 36,13 |
| Benzeşmezlik | 46,31 | 40,19 | 50,51 |
| Renk Ağırlıklı[56] | 28,50 | 20,38 | 31,94 |

Tablo 4.7 : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri

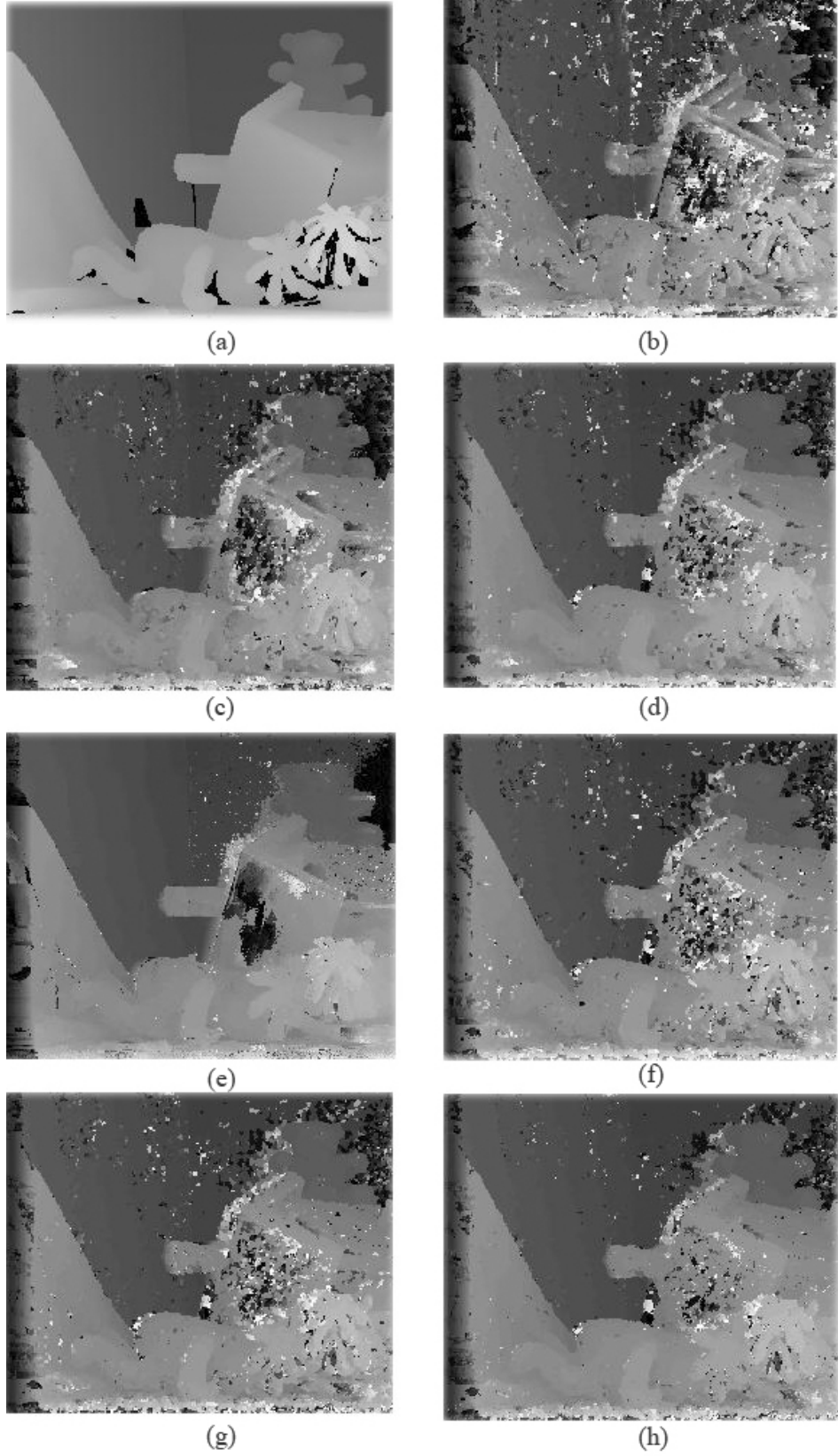
| | Tüm Pikseller | Kestirim(%) | Örtülmemiş | Kestirim(%) | Devamsızlık Böl | Kestirim(%) |
|--------------------|---------------|-------------|------------|-------------|-----------------|-------------|
| Öneri 1 | 10,99 | 67,03 | 10,06 | 74,08 | 17 | 66,96 |
| Öneri 2 | 11,18 | 67,21 | 10,24 | 74,29 | 17,28 | 67,02 |
| Öneri 3 | 8,31 | 71,48 | 7,32 | 78,94 | 15,56 | 68,89 |
| SadGrad[47] | 10,18 | 62,81 | 9,23 | 69,37 | 16,71 | 64,45 |
| SAD | 16,65 | 61,88 | 15,43 | 68,01 | 22,10 | 64,34 |
| Benzeşmezlik | 18,75 | 47,05 | 17,82 | 51,95 | 28,62 | 43,64 |
| Renk Ağırlıklı[56] | 10,67 | 64,97 | 9,55 | 71,64 | 21,11 | 62,79 |

Tablo 4.8 : Piksel Bazlı İnanç Yayılımı algoritmasına göre hata oranları

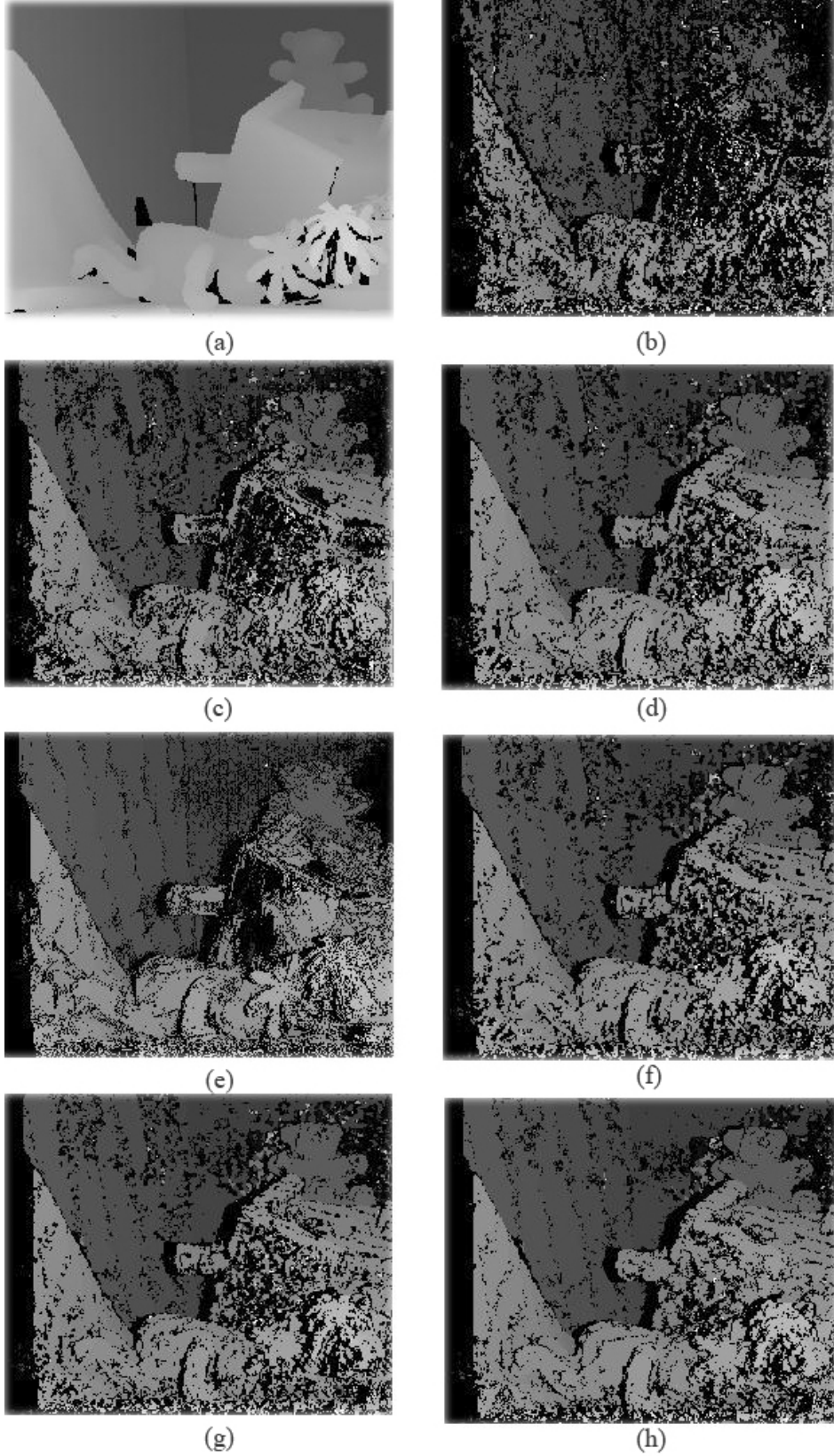
| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 17,82 | 11,54 | 24,04 |
| Öneri 2 | 18,14 | 11,74 | 24,05 |
| Öneri 3 | 18,53 | 12,33 | 26,64 |
| SadGrad[47] | 18,76 | 11,54 | 23,84 |
| SAD | 25,18 | 16,96 | 27,69 |
| Benzeşmezlik | 29,82 | 22,05 | 34,6 |
| Renk Ağırlıklı[56] | 21,06 | 12,13 | 24,23 |

Tablo 4.9 : Bölüt Bazlı İnanç Yayılımı algoritmasına göre hata oranları

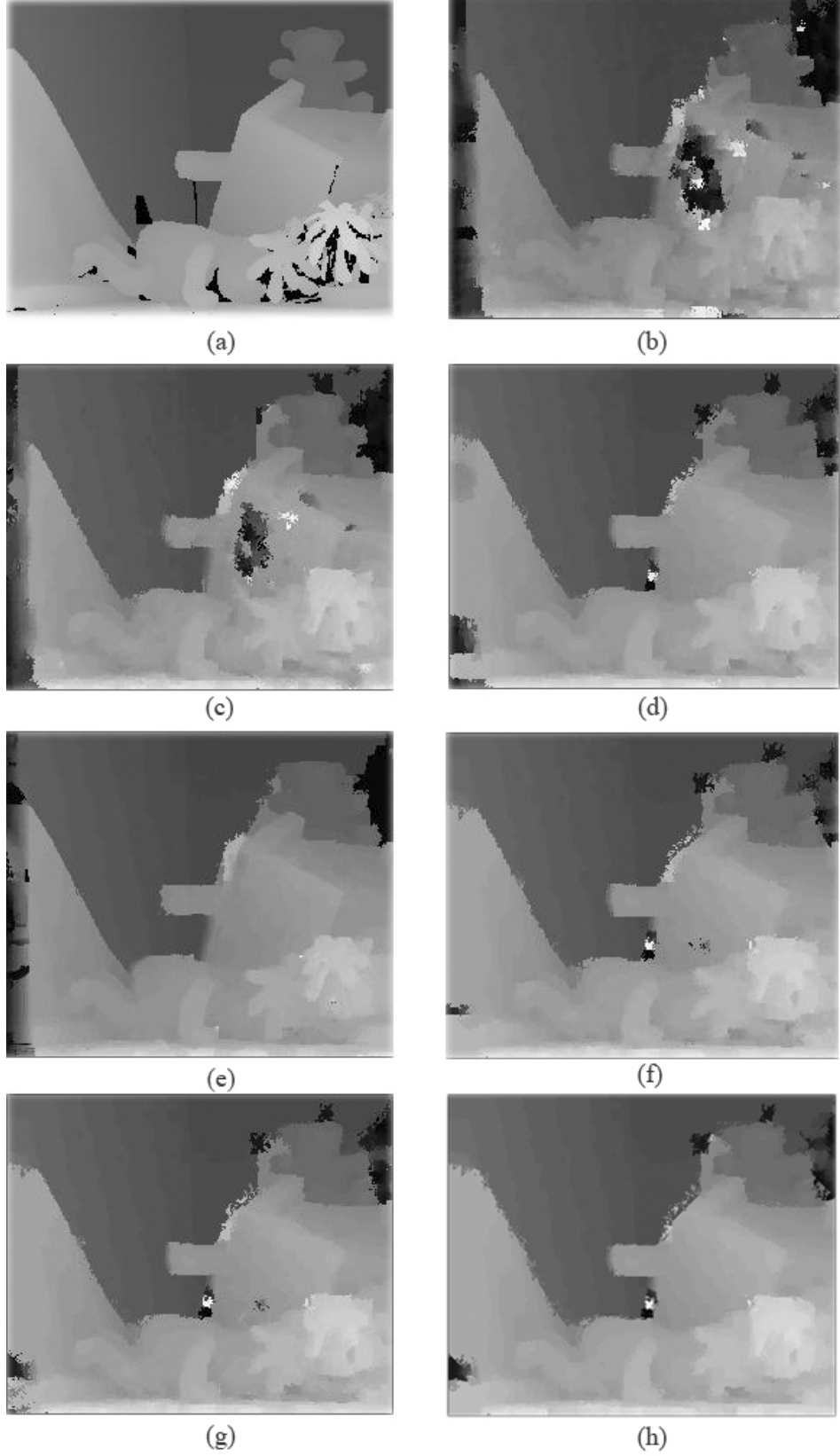
| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|---------------|------------|------------------|
| Öneri 1 | 13,4 | 8,79 | 19,2 |
| Öneri 2 | 12,94 | 8,28 | 18,88 |
| Öneri 3 | 13,11 | 8,37 | 18,98 |
| SadGrad[47] | 13,1 | 8,25 | 18,65 |
| SAD | 14,74 | 9,69 | 20,45 |
| Benzeşmezlik | 18,6 | 13,15 | 25,32 |
| Renk Ağırlıklı[56] | 16,08 | 11,43 | 24,69 |



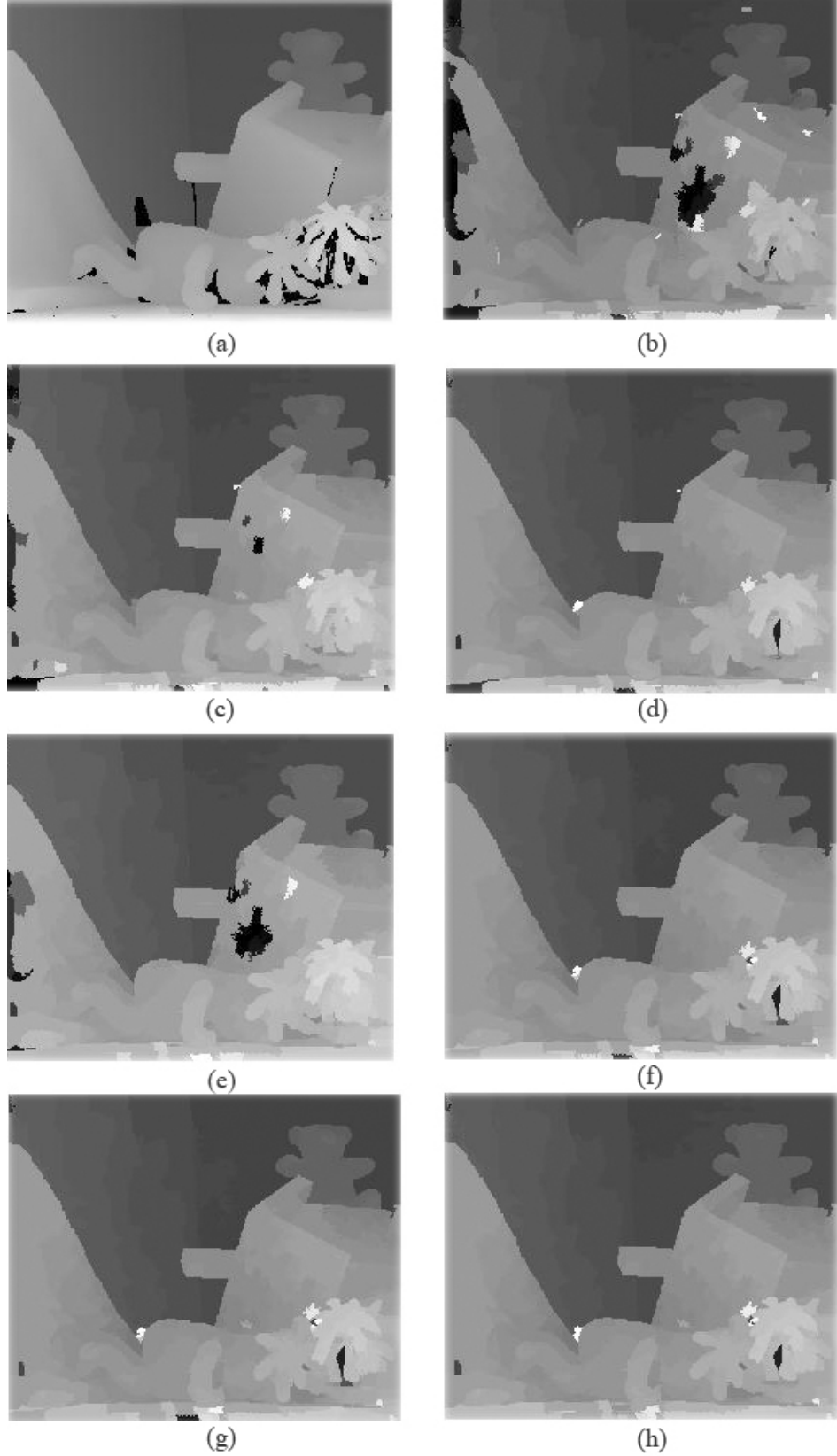
Şekil 4.6 : KHA algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.7 : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.8 : Pksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) Öneri 1 (g) Öneri 2 (h) Öneri 3



Şekil 4.9 : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**

4.1.1. “Cones” İmgelerine ait Bulgular

Tablo 4.10 : KHA algoritmasına göre hata oranları

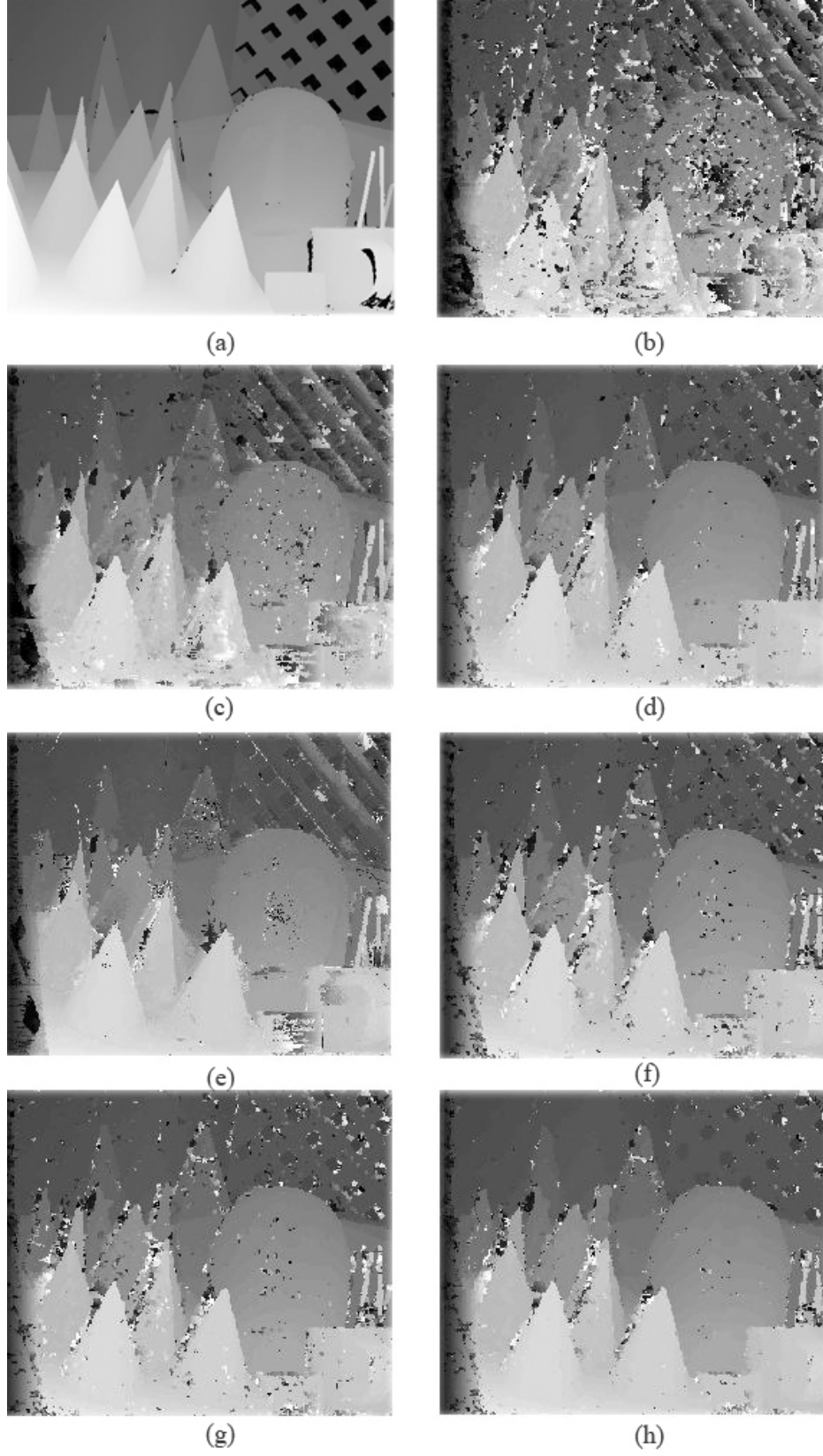
| | Tüm Pkseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|--------------|------------|------------------|
| Öneri 1 | 20,73 | 11,17 | 20,65 |
| Öneri 2 | 20,53 | 11 | 20,67 |
| Öneri 3 | 16,43 | 7,03 | 15,95 |
| SadGrad[47] | 19,29 | 9,63 | 17,69 |
| SAD | 36,62 | 28,95 | 33,67 |
| Benzeşmezlik | 51,56 | 45,61 | 51,75 |
| Renk Ağırlıklı[56] | 28,94 | 20,19 | 30,77 |

Tablo 4.11 : KHA algoritmasına sağ/sol kontrolü yapıldıktan sonra kalan piksel oranı ve hata yüzdeleri

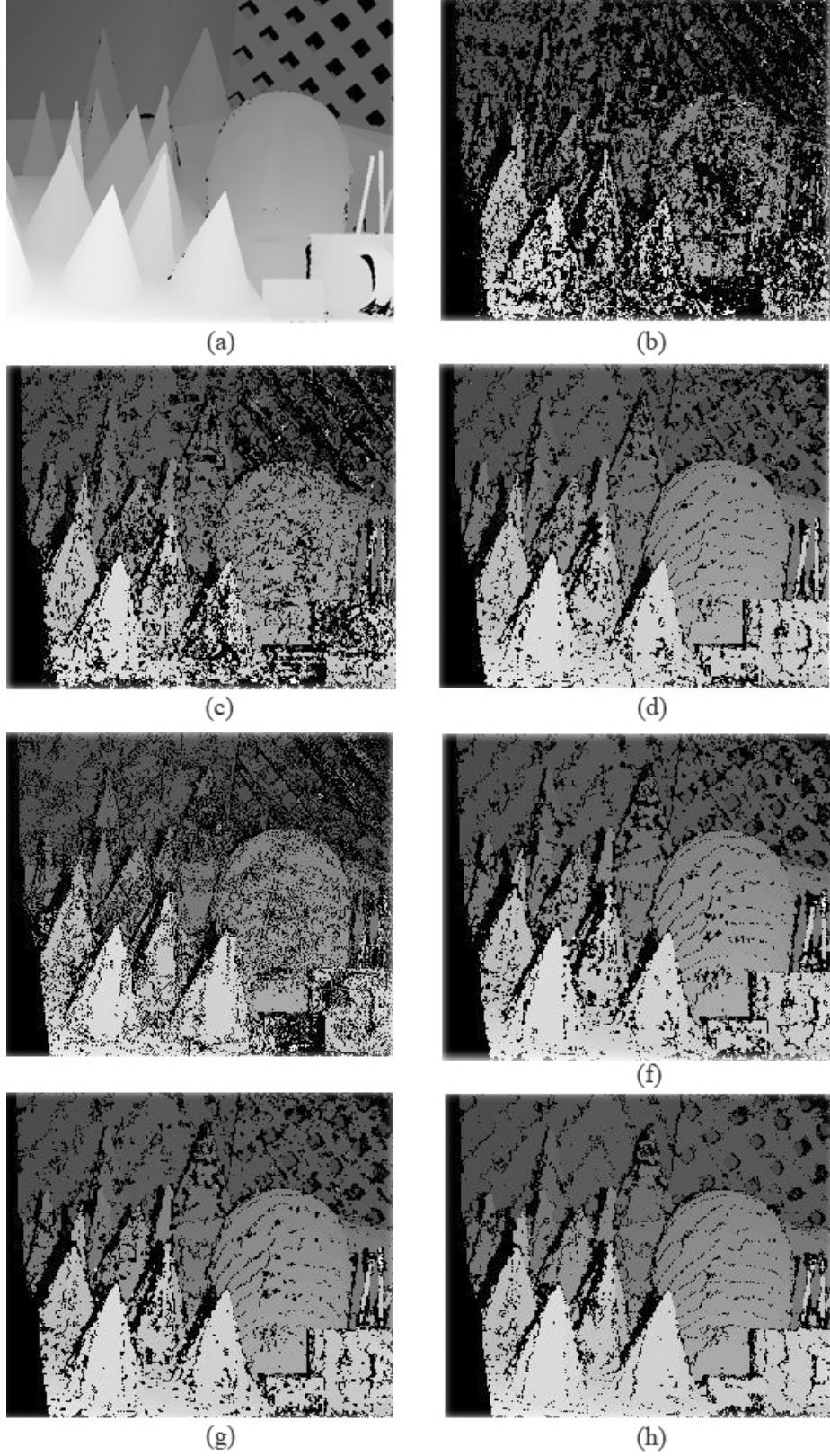
| | Tüm Pkseller | Kestirim(%) | Örtülmemiş | Kestirim(%) | Devamsızlık Böl | Kestirim(%) |
|--------------------|--------------|-------------|------------|-------------|-----------------|-------------|
| Öneri 1 | 3,80 | 72,41 | 2,9 | 81,16 | 6,31 | 70,79 |
| Öneri 2 | 3,63 | 72,98 | 2,77 | 81,84 | 6,14 | 71,32 |
| Öneri 3 | 3,22 | 78,10 | 2,23 | 87,36 | 6,12 | 77,12 |
| SadGrad[47] | 4,47 | 75,06 | 3,31 | 83,87 | 7,17 | 75,69 |
| SAD | 15,92 | 60,55 | 14,45 | 67,17 | 18,78 | 62,98 |
| Benzeşmezlik | 22,98 | 40,72 | 21,87 | 45,33 | 27,48 | 40,01 |
| Renk Ağırlıklı[56] | 12,12 | 59,38 | 10,89 | 66,16 | 19,75 | 56,82 |

Tablo 4.12 : Piksel Bazlı İnanç Yayılımı algoritmasına göre hata oranları

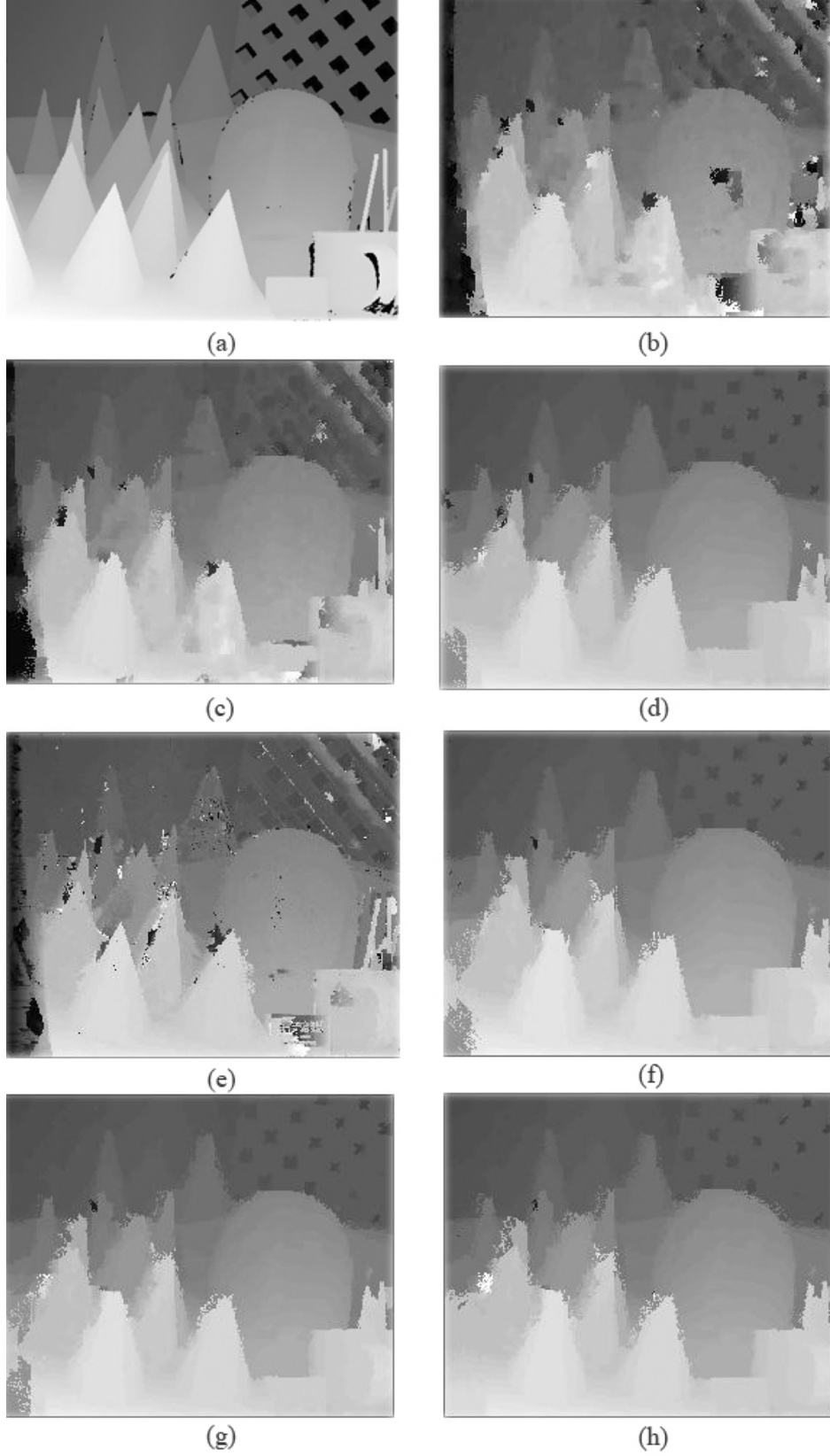
| | Tüm Pkseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|--------------|------------|------------------|
| Öneri 1 | 12,67 | 5,83 | 15,22 |
| Öneri 2 | 12,66 | 5,77 | 15,13 |
| Öneri 3 | 13,23 | 6,67 | 17,71 |
| SadGrad[47] | 12,54 | 5,59 | 14,48 |
| SAD | 26,37 | 18,21 | 26,4 |
| Benzeşmezlik | 33,6 | 26,12 | 35,99 |
| Renk Ağırlıklı[56] | 23,51 | 16,82 | 30,29 |



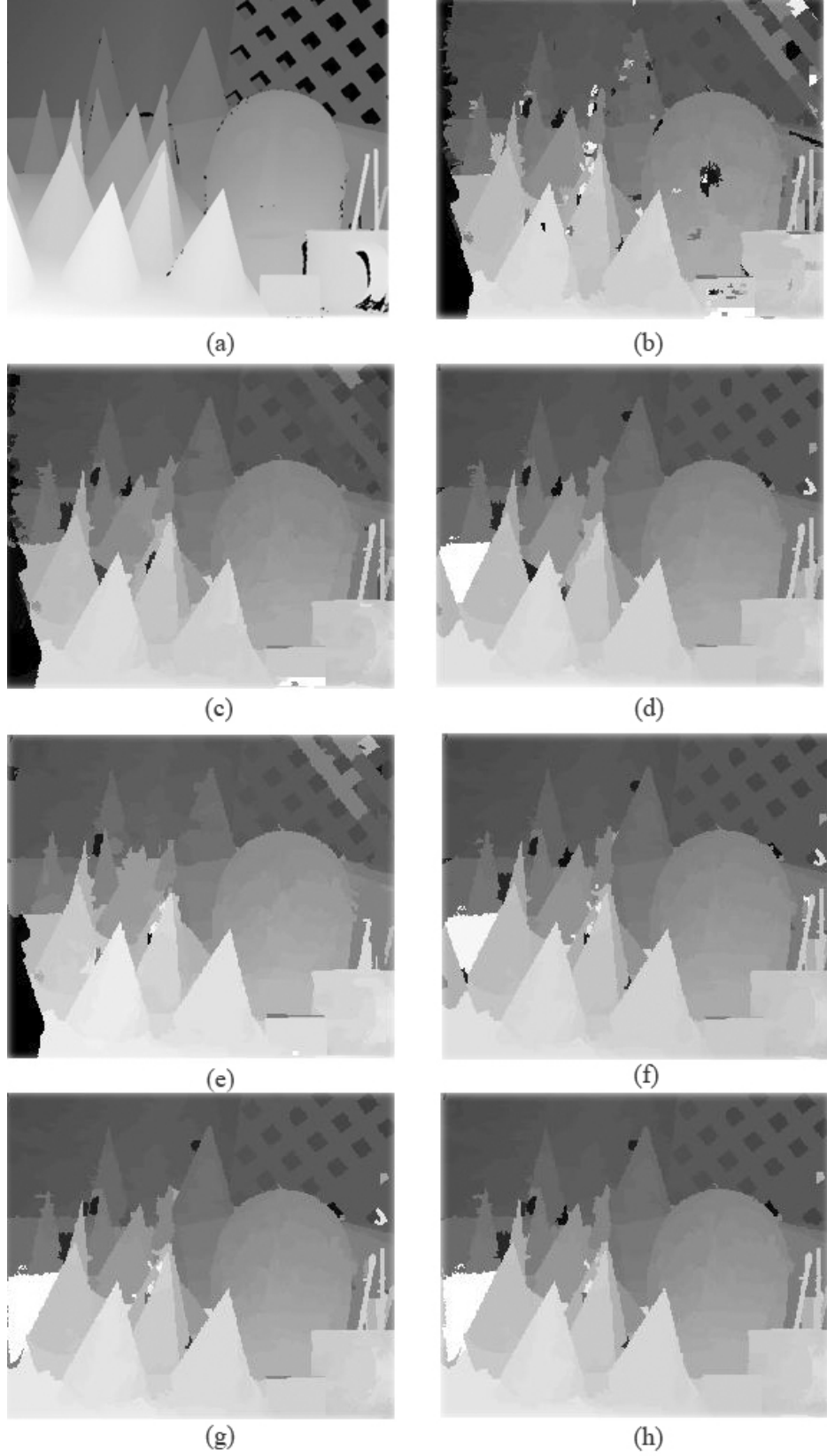
Şekil 4.10 : KHA algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**



Şekil 4.11 : KHA algoritmasına sağ/sol kontrolü yapılarak elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) Öneri 1 (g) Öneri 2 (h) Öneri 3



Şekil 4.12 : Piksel bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) Öneri 1 (g) Öneri 2 (h) Öneri 3



Şekil 4.13 : Bölüt bazında inanç yayılımı algoritmasıyla elde edilen derinlik haritaları: (a) Orjinal Harita (b) Benzeşmezlik Ölçüsü (c) SAD (d) SadGrad [47] (e) Renk Ağırlıklı[56] Yöntem (f) **Öneri 1** (g) **Öneri 2** (h) **Öneri 3**

Tablo 4.13 : Bölüt Bazlı İnanç Yayılımı algoritmasına göre hata oranları

| | Tüm Pkseller | Örtülmemiş | Devamsızlık Böl. |
|--------------------|--------------|------------|------------------|
| Öneri 1 | 11,68 | 6,19 | 12,96 |
| Öneri 2 | 11,58 | 6,23 | 13,07 |
| Öneri 3 | 11,55 | 6,2 | 12,73 |
| SadGrad[47] | 11,74 | 6,29 | 13,15 |
| SAD | 20,63 | 14,25 | 22,32 |
| Benzeşmezlik | 28,03 | 21,79 | 31,8 |
| Renk Ağırlıklı[56] | 21,54 | 16,5 | 26,52 |

4.2. ÖRTME HARİTASI ÇIKARMA BULGULARI

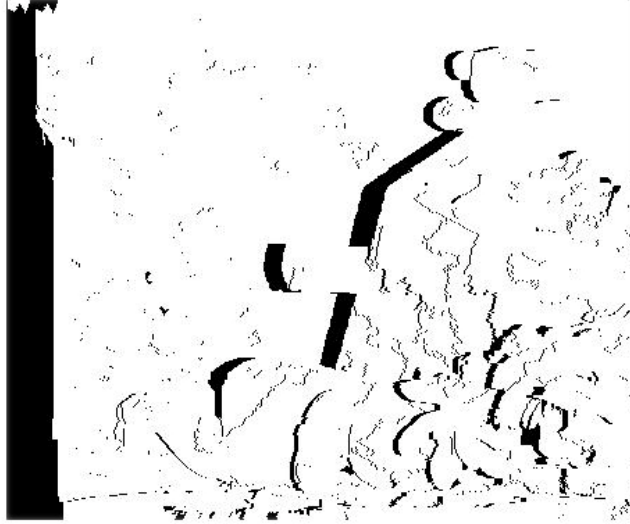
Örtülmüş bölgelerin saptanması konusundaki öneriyi test etmek amacıyla, önceki bölümde derinlik haritalarını elde ettiğimiz imge setlerine, bölüt bazlı inanç yayılımı algoritması uygulanmış ve referans imgesine ilk olarak bölüm 2.3.3.3'te gösterilen LRC yöntemi uygulanmıştır. Daha sonra aynı imgeler üzerinde bölüm 3.3'te önerilen yöntem ve aynı yöntemin her bölüt için belli bir yüzdenin altındaki örtülmüş pikselleri yok sayan ikinci versiyonu karşılaştırılmıştır. Şekil 4.13'te bu yaklaşım adımları görsel olarak gösterilmiştir. Tablo 4.14'te sistemlerin örtülmüş bölgeleri doğru tahmin etme oranları ve hatalı olarak örtülmüş olarak işaretlenen piksellerin toplam tahmin edilen piksellere oranı (hatalı pozitif) verilmiştir.

Tablo 4.14 : Örtme haritası çıkarmada önerilen yöntemlerin başarı yüzdeleri

| | LRC Yöntemi | | 1.Önerilen Yöntem | | 2.Önerilen Yöntem | |
|-----------|--------------|----------------|-------------------|----------------|-------------------|----------------|
| | Doğru Tahmin | Hatalı Pozitif | Doğru Tahmin | Hatalı Pozitif | Doğru Tahmin | Hatalı Pozitif |
| “Teddy” | 93.86 | 69.97 | 88.54 | 19.90 | 87.87 | 11.69 |
| “Cones” | 93.85 | 54.44 | 89.04 | 25.82 | 88.61 | 17.85 |
| “Tsukaba” | 75.55 | 79.22 | 68.60 | 46.84 | 68.11 | 35.86 |



(a)

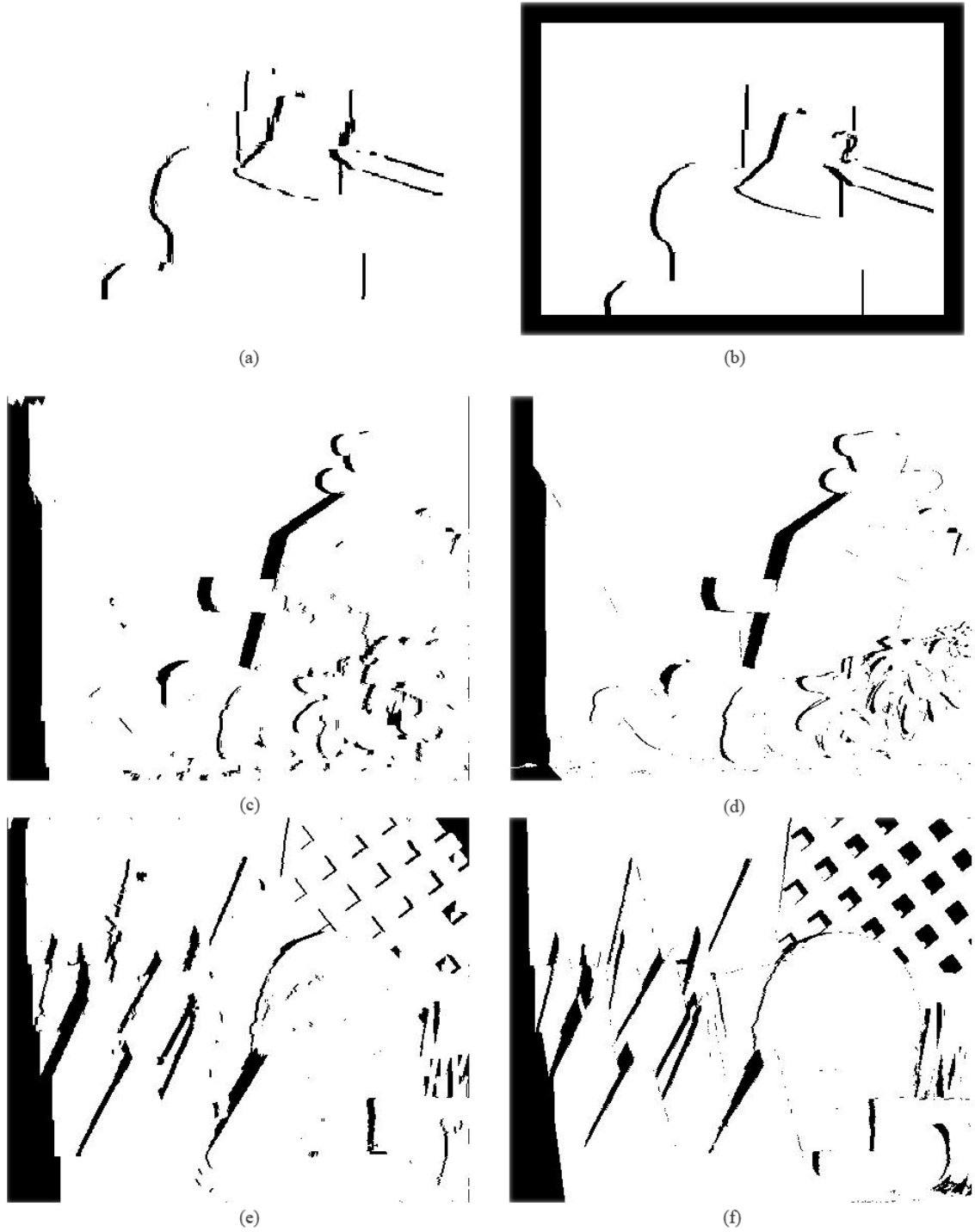


(b)



(c)

Şekil 4.14 : (a) LRC sonucu (b) Önerilen Yöntem (c) 2.Önerilen Yöntem



Şekil 4.15 : Kestirilen örtme haritaları ve orjinal haritalar: (a) Tsukaba (c) Teddy (e) Cones

4.3. HATALI VERİ AYIKLAMA VE ÖNERİLEN ALGORİTMA BULGULARI

Bölüm 3.4.2’de detayları verilen “aykırı verileri ayıklama” yöntemini test etmek amacıyla, daha önce önerilen maliyet fonksiyonuna önce sağ/sol kontrol yapılmış daha sonra elde edilen sonuçlara önerilen “aykırı veri ayıklama” yöntemi uygulanmıştır. Elde edilen sonuçlar Tablo 4.15’te karşılaştırılmıştır.

Tablo 4.15 : Aykırı verileri ayıklama algoritması uygulandıktan sonra kalan kararlı pikseller ve hata oranları

| | Tüm Pikseller | Kestirim(%) | Örtülmemiş | Kestirim(%) | Devamsızlık Böl | Kestirim(%) |
|---------------------|---------------|--------------|-------------|--------------|-----------------|--------------|
| Tsukaba | 5,45 | 82,54 | 4,87 | 84,14 | 6,33 | 78,96 |
| Tsukaba(ayıklanmış) | 0.83 | 57.74 | 0.63 | 59.13 | 4.49 | 43.39 |
| Teddy | 10,99 | 67,03 | 10,06 | 74,08 | 17 | 66,96 |
| Teddy(ayıklanmış) | 2,41 | 56,65 | 2,02 | 63,05 | 6,61 | 47,05 |
| Cones | 3,80 | 72,41 | 2,9 | 81,16 | 6,31 | 70,79 |
| Cones(ayıklanmış) | 1.02 | 54,60 | 0,74 | 64,72 | 2,41 | 47,12 |

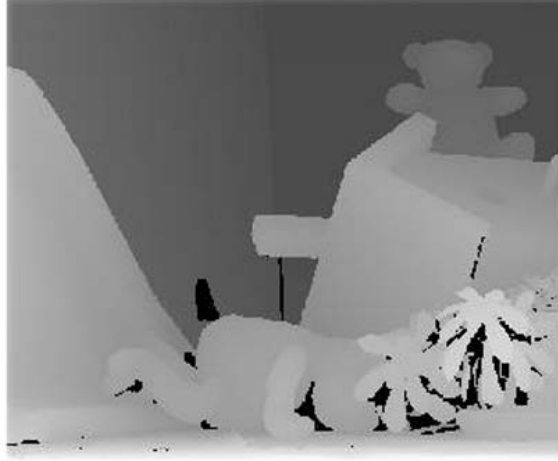
Son olarak Şekil 3.6’da özeti verilmiş olan içinde düzlem kestiriminde barındıran derinlik haritası elde etme yöntemi “teddy” ve “cones” imge setlerine uygulanmıştır. Elde edilen sonuçlar, “Middlebury” test ortamında [71] ilk iki sırayı paylaşan algoritmayla karşılaştırılmıştır. Bu karşılaştırmalar Tablo 4.16 ve Tablo 4.17’den incelenebilir. Ayrıca elde edilen derinlik haritalı ve bu haritalar üzerindeki hatalı bölgeler Şekil 4.15 ve Şekil 4.16 da gösterilmiştir.

Tablo 4.16 : “Teddy” imgesi için önerilen yöntemin “Middlebury” test ortamının ilk sıralarındaki algoritmalarla karşılaştırması

| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|------------------------|---------------|------------|------------------|
| Önerilen Yöntem | 10,39 | 5,82 | 14,58 |
| AdaptingBP[47] | 7,06 | 4,22 | 11,8 |
| DoubleBP2[56] | 8,30 | 3,53 | 9,63 |

Tablo 4.17 : “Cones” imgesi için önerilen yöntemin “Middlebury” test ortamının ilk sıralarındaki algoritmalarla karşılaştırması

| | Tüm Pikseller | Örtülmemiş | Devamsızlık Böl. |
|------------------------|---------------|------------|------------------|
| Önerilen Yöntem | 9,83 | 4,35 | 11,13 |
| AdaptingBP[47] | 7,92 | 2,48 | 7,32 |
| DoubleBP2[56] | 8,78 | 2,90 | 7,79 |



(a)

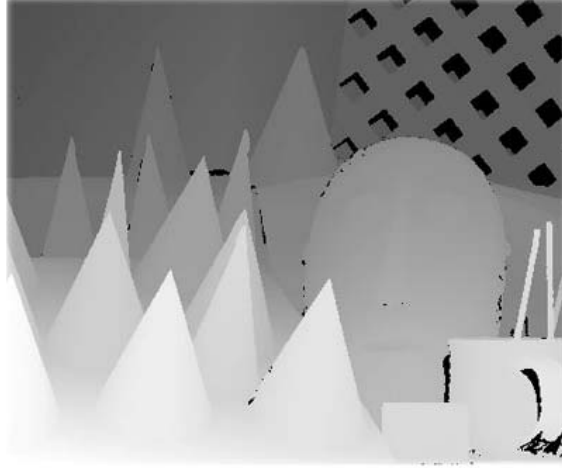


(b)



(c)

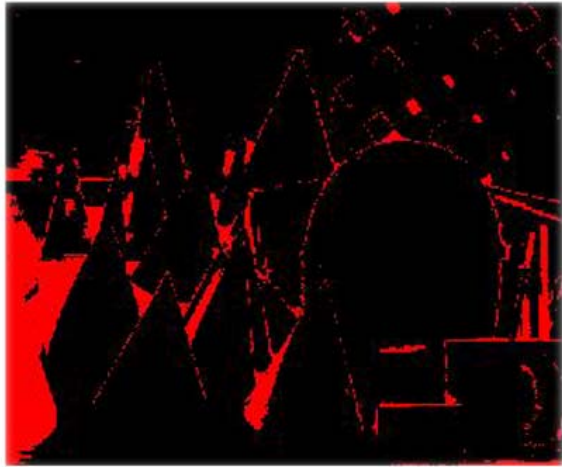
Şekil 4.16 : “Teddy” imgelerine ait kestirim sonuçları: (a) orjinal derinlik haritası (b) kestirilen derinlik haritası (c) hatalı kestirilen bölgeler(kırmızı renkli bölgeler)



(a)



(b)



Şekil 4.17 : “Cones” imgelerine ait kestirim sonuçları: (a) original derinlik haritası (b) kestirilen derinlik haritası (c) hatalı kestirilen bölgeler(kırmızı renkli bölgeler)

5. TARTIŞMA VE SONUÇ

Bu tez kapsamında, stereo ilişkilendirme algoritmalarında kullanılacak birçok yeni yöntem sunulmuştur. İlk olarak önerilen eşleşme maliyet fonksiyonu, diğer emsalleriyle karşılaştırıldığında daha fazla güvenilir piksel üretmekte ve genel olarak hata oranı diğerlerine göre daha düşüktür. Tablo 4.2, 4.6 ve 4.10 incelendiğinde sadece Tablo 4.2’de sonuçları verilen “Tsukaba” imge seti için KHA algoritması uygulandığında başka bir maliyet fonksiyonunun daha başarılı olduğu görülmektedir. Bu imgede en iyi sonucu veren renk ağırlıklı yöntemin, 33 piksel genişliğinde bir destek penceresi olmasının sonuca etkisi büyüktür. Ancak fonksiyonlar tarafından oluşturulan maliyet hacimleri, bölüt bazlı inanç yayılımı gibi sistemlerde kullanıldığında tüm imge setlerinde önerilen maliyet fonksiyonları en iyi sonucu vermektedir.

Fonksiyonun birden fazla pencere kullanan sürümü, diğer algoritmalara göre %10’a yakın daha fazla piksel için kararlı çözüm üretmektedir. Ancak, ilinti hacmi olarak kullanıldığında diğer versiyonlarına göre başarısı düşmektedir. Bunda, kararlı olmayan pikseller için daha geniş pencereler kullanılmasının etkisi büyüktür. Mevcut haliyle kullanılabilceği algoritmalar da mevcuttur. Sadece KHA algoritması kullanılarak imgedeki piksellerin %80’lik bölümünün kararlı olarak kestirilebilmesi, derinlik haritasının tümüne ihtiyaç duymayan ve çabuk işlem yapılmasını gerektiren stereo imgelerden elde edilen bilgileri mono imgelerden elde edilenlerle birleştiren uygulamalarda [86] başarıyla kullanılabilir.

İkinci olarak, imgedeki örtülmüş bölgeleri başarılı bir şekilde belirleyen bir yöntem sunulmuştur. İki imgenin örtülmüş bölge hesaplarının birden kullanılması ve önerilen varsayım sayesinde, yöntemin doğru bölge belirleme oranı LRC algoritmasına göre %5 düşerken, hatalı pozitif belirleme oranı %30 civarı azalmaktadır. Bu sonuçlar ışığında, önerilen yöntemin diğerlerine göre daha gürbüz olduğu ve stereo algoritmalarda rahatlıkla kullanılabilceği söylenebilir. Öte yandan, tüm diğer örtülmüş bölge belirleme algoritmaları gibi eşleşme maliyeti fonksiyonunun başarısına karşı hassastır. “Tsukaba” imge seti için alınan sonuçlardan bu durum gözlemlenebilir. Bölüt bazlı inanç yayılımı yönteminin “Tsukaba” için performansının düşük olması, örtülmüş bölgelerin haritasının başarısını ciddi oranda düşürmüştür.

Düzlem kestirimlerinin doğruluğunu arttırmak için önerilen aykırı veri ayıklama yöntemi, sağ/sol kontrolü sonrasında elde edilen pikseller içinden hatalı olanları başarılı bir şekilde ayıklamaktadır. Tablo 4.15'teki sonuçlar incelendiğinde bazı imge setleri için aykırı pikseller ayıklandıktan sonra hata oranının ilk durumun beşte birine indiği görülmektedir. Ayrıca imge içindeki piksellerin %50'den fazlası %1'e yakın bir hata oranıyla kestirilebilmektedir. Bu özellikleriyle önerilen algoritmanın düzlem kestirimine dayalı yöntemlerde başarıyla kullanılabilmesi söylenilebilir.

Yukarıda belirtilen yaklaşımlar teker teker ele alındığında gözlemlenen başarıya rağmen, Tablo 4.16 ve 4.17 incelendiğinde, tüm bu yöntemleri kullanarak oluşturulan stereo ilişkilendirme algoritmasının, mevcut algoritmalara göre ortalama bir sonuç verdiği gözükmektedir. Alınan sonuçlar arasındaki fark birkaç açıdan irdelenebilir.

Stereo ilişkilendirme algoritmalarının başarısı, içinde kullanılan renk bazlı bölütlemeye karşı çok hassastır. Diğer algoritmaların hesaplarında kullanılan parametreler tam olarak bilinmediğinden sağlıklı bir karşılaştırma yapılamamaktadır.

Öte yandan, düzlem kestirimi sırasında RANSAC yöntemi kullanılmıştır. Bilindiği üzere bu yöntem her defasında aynı sonucu vermemektedir. Bu durum sistemin performansında %2'lere varan oynamalar oluşmasına neden olmaktadır.

Sonuçlar incelendiğinde göze çarpan bir başka nokta, örtülmüş bölgelerin başarılı bir şekilde kestirilebilmesine rağmen, genel algoritmanın bu bölgelerde beklenen başarıyı gösterememiş olmasıdır. Bu durum görsel olarak Şekil 4.17'de, hunilerin sol taraflarında oluşan hatalı bölgelerden gözlemlenebilmektedir. Ayrıca, referans imgesinin solunda oluşan örtülmüş bölgede, düzlem kestirimine rağmen başarılı bir sonuç alınamamıştır.

Yukarıdaki bilgiler ışığında, önerilen maliyet fonksiyonunun diğer algoritmalarda kullanılabilmesi söylenebilir. Ayrıca elde edilen yüksek doğrulukta örtülmüş bölge haritası, mevcut yöntemlerin başarısını arttırmada kullanılabilir.

Gelecekte yapılacak çalışmalarda, önerilen aykırı veri ayıklama yöntemiyle elde edilen yüksek doğruluktaki noktaları kullanarak düzlem kesitirimi yapmak amacıyla, RANSAC yöntemine alternatif yöntemler incelenmelidir. Ayrıca, çoklu pencere yönteminin ilinti hacmi olarak kullanıldığı durumlardaki başarıyı arttırmak için, farklı boyuttaki pencerelerin maliyetlerini daha tutarlı bir şekilde birleştirecek yöntemler araştırılmalıdır.

Elde edilen örtülmüş bölgeler haritasını daha verimli biçimde kullanarak enerji minimizasyonu formülünün içine yerleştirilmesi genel başarıyı arttıracak yöntemlerden biri olabilir.

KAYNAKLAR

1. Hartley R. ve Zisserman A., 2000, *Multiple view geometry in Computer Vision*, Cambridge University Press, New York.
2. Faugeras O., 1993, *Three Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, Massachusetts, London, England.
3. Maybank S. J ve Faugeras O., A theory of self-calibration of a moving camera. *International Journal of Computer Vision*, 8(2): 123-152, Aug 1992
4. Tsai R.Y. , 1987, A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323-344.
5. Hartley R., 1994, Self calibration from multiple views with a rotating camera. *Proc. Third European Conference on Computer Vision*, J.-O. Eklundh, ed., vol. 800-801, pp.471-478.
6. Zhang, Z., 2000, A flexible new technique for camera calibration. *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334.
7. Zhang, Z., 2002, Camera calibration with one-dimensional objects, *Proc. European Conf. Computer Vision*, vol. 4, pp. 161-174.
8. Zhang, Z., 2004, Camera calibration with one-dimensional objects, *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol.26, no.7.
9. Horn, B. K. P., 1986, *Robot Vision*, MIT Press, Cambridge, Massachusetts and McGraw-Hill, New York
10. Tapper M., McKerrow P.J., Abrantes J., 2002, Problems encountered in the implementation of Tsai's algorithm for camera calibration. *Proc. 2002 Australasian Conference on Robotics and Automation*, Auckland.
11. Faugeras Olivier, Luong Quang-Tuan, Maybank S.J., 1992, Camera Self-Calibration: Theory and Experiments, *Proceedings of the 2nd European Conference on Computer Vision*, 321-334.

12. Luong Quang-Tuan, Faugeras Olivier, 1997, Self-Calibration of a Moving Camera from Point Correspondences and Fundamental Matrices, *International Journal of Computer Vision*, 22(3), 261-289.
13. Zeller Cyril, Faugeras Olivier, 1996, Camera Self-Calibration from Video Sequences: the Kruppa Equations Revisited, *INRIA*.
14. Lei Cheng, 2002, A New Approach to Solving Kruppa Equations for Camera Self-Calibration, *ICPR*.
15. Fusiello A., 2000, Uncalibrated Euclidean Reconstruction: A Review”, *Image and Vision Computing*, 18, 555-563.
16. Heyden A., Astrom K., 1996, Euclidean Reconstruction from Constant Intrinsic Parameters, *Proc. 13th International Conference on Pattern Recognition*, 339-343.
17. Pollefeys Marc, Gool L. Van, 1997, Self-calibration from the absolute conic on the plane at infinity, *Proc. Computer Analysis of Images and Patterns*, 175-182
18. Pollefeys Marc, Gool L. Van, 1997, A stratified approach to selfcalibration, *Proc. 1997 Conference on Computer Vision and Pattern Recognition*, 407-412
19. Pollefeys Marc, Gool L. Van, Oosterlinck A. , 1996, The Modulus Constraint: A new Constraint for Self Calibration, *Proc 13th International Conference on Pattern Recognition*, 349-357.
20. G. Xu ve Z. Zhang, 1996, Epipolar geometry in stereo, motion and object recognition. *Kluwer academic publishers*.
21. H. C. Longuet-Higgins, 1981, A computer algorithm for reconstructing a scene from two projections. *Nature*. 293:133-135.
22. Harris, C., Stephens, 1988, M. A Combined Corner and Edge Detector. *In 4th Alvey Vision Conference*, S. 147–151.
23. K. Mikolajczyk ve C. Schmid., 2002, An affine invariant interest point detector. *Proceedings of the 7th European Conference on Computer Vision*, Copenhagen, Denmark, volume I, pages 128–142.
24. Schmid, C., Mohr, R., Bauckhage C., 1998, Comparing and Evaluating Interest Points. *IEEE International Conference on Computer Vision*, S. 230–235.
25. Canny, J. F.,1986, A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Bd. 8(6):679–698.

26. Deriche, R., Faugeras, O. 1993, A Computational Approach for Corner and Vertex Detection. *International Journal of Computer Vision*, Bd. 1(2):167–187.
27. Tekalp, A. M., 1995, *Digital Video Processing*, Prentice Hall.
28. B. Triggs, P. McLauchlan, R. Hartley, A. Fitzgibbon, 2000, Bundle Adjustment – A Modern Synthesis, In B. Triggs, A. Zisserman, R. Szeliski (Eds.), *Vision Algorithms: Theory and Practice*, LNCS Vol.1883, pp.298-372, Springer-Verlag.
29. Bayram I., 2004, *Interest point matching across arbitrary views*, M.S. Thesis.
30. Z. Zhang, R. Deriche, O. Faugeras, Q.T. Luong, 1994, A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry, *INRIA*, Report No. 2273.
31. Dufournaud Y., Schmid C., Horaud R. Image matching with scale adjustment. *Research Report*, no. 4428, 2002
32. Kanazawa Y., Kanatani K., 2002, Robust image matching under a large disparity. *In proceedings of Workshop on Science of Computer Vision*, Okayama, Japan, 46-52.
33. P. H. S. Torr ve A. Zisserman., 2000, MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156.
34. Sturm P., Maybank S., 1999, On plane based camera calibration: A general algorithm, Singularities, Applications. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 432-437.
35. Weng J, Huang T.S., Ahuja N. , 1993, “Motion and Structure from Image Sequences”, *Springer Series in Information Sciences 29*.
36. G. Xu ve Z. Zhang, 1996, Epipolar geometry in stereo, motion and object recognition. *Kluwer academic publishers*.
37. Hartley, R. I, Gupta, R., Chang, T. , 1992, Stereo from Uncalibrated cameras, *IEEE Computer Society Conference on Proc. CVPR'92*.
38. Hartley, R. I, and Sturm, P. Triangulation, 1997, *Computer Vision and Image Understanding*, Vol. 68, No.2, 146-157 Article No.IV970547
39. Beardsley, P. A., Zisserman, A. and Murray, D. W. , 1994, Navigation using affine structure from motion, in *Computer Vision, ECCV'94*, LNCS Series 801, 85-96, Springer-Verlag, Berlin/New York.

40. Press, W. H. , 1992, *Numerical recipes in C: the art of scientific computing*, Cambridge University Press, New York
41. Lourakis M., 2004, Levenberg-Marquardt non-linear least squares algorithms in C/C++, <http://www.ics.forth.gr/~lourakis/levmar/>
42. A. Fusiello, E. Trucco, A. Verri, 2000, A compact algorithm for rectification of stereo pairs., *Machine Vision and Applications*, 12: 16-22.
43. D. Scharstein, R. Szeliski, 2002, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 47(1):7–42.
44. S. Birchfield ve C. Tomasi., 1998, A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Trans. PAMI*, 20(4):401–406.
45. A. Klaus, M. Sormann, K. Karner., 2006, Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. *ICPR*, 15–18.
46. T. Kanade ve M. Okutomi, 1994, A Stereo Matching Algorithm with an Adaptive Window: Theory and Experiments, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 16, no. 9, pp. 920-932.
47. Y. Boykov, O. Veksler, and R. Zabih, 1998, “A Variable Window Approach to Early Vision,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, no.12, 1283-1294.
48. O. Veksler, 2002, “Stereo Correspondence with Compact Windows via Minimum Ratio Cycle,” *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1654-1660.
49. O. Veksler, 2003, Fast Variable Window for Stereo Correspondence using Integral Images, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, 556-561.
50. A. Fusiello, V. Roberto, ve E. Trucco, 1997, Efficient Stereo with Multiple Windowing, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 858-863.
51. A.F. Bobick ve S.S. Intille, 1999, Large Occlusion Stereo, *Int’l J. Computer Vision*, vol. 33, no. 3, 181-200.
52. S.B. Kang, R. Szeliski, ve C. Jinxjang, 2001, Handling Occlusions in Dense Multi-View Stereo, *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 1, 103-110.

53. K. Prazdny, 1985, "Detection of Binocular Disparities," *Biological Cybernetics*, vol. 52, 93-99.
54. T. Darrel, 1998, "A Radial Cumulative Similarity Transform for Robust Image Correspondence," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 656-662.
55. Y. Xu, D. Wang, T. Feng, and H.-Y. Shum, 2002, Stereo Computation using Radial Adaptive Windows, *Proc. Int'l Conf. Pattern Recognition*, vol. 3, 595-598.
56. J. Yoon ve I.-S. Kweon, 2006, Adaptive support-weight approach for correspondence search., *TPAMI*, 28(4):650–656.
57. G. Egnal ve R. Wildes., 2002, Detecting binocular halfocclusions: empirical comparisons of five approaches. *PAMI*, 24(8):1127–1133.
58. A. Spoerri ve S. Ullman, 1987, The Early Detection of Motion Boundaries, *Proc. Int'l Conf. Computer Vision*, 209-218.
59. T. Smitley ve R. Bajcsy, 1984, Stereo Processing of Aerial Urban Images, *Proc. Int'l Joint Conf. Pattern Recognition*, 405-409.
60. B. Anderson ve K. Nakayama, 1994, Toward a General Theory of Stereopsis: Binocular Matching, Occluding Contours, and Fusion, *Psychological Rev.*, vol. 101, no. 3, 414-445.
61. J.D. Krol ve W.A. van der Grind, 1982, The Double-Nail Illusion, *Perception*, vol. 11, 615-619.
62. D. Terzopoulos. , 1986, Regularization of inverse visual problems involving discontinuities. *IEEE TPAMI*, 8(4):413–424.
63. O. Veksler., 1999, *Efficient Graph-based Energy Minimization Methods in Computer Vision*. PhD thesis, Cornell University.
64. H. Baker ve T. Binford., 1981, Depth from edge and intensity based stereo, *IJCAI81*, 631–636.
65. Y. Ohta ve T. Kanade. Stereo by intra- and interscanline search using dynamic programming. *IEEE TPAMI*, 7(2):139–154, 1985.
66. P. N. Belhumeur ve D. Mumford., 1992, A Bayesian treatment of the stereo correspondence problem using half-occluded regions. *CVPR*, 506–512.
67. P. N. Belhumeur., 1996, A Bayesian approach to binocular stereopsis. *IJCV*, 19(3):237 260.

68. Q. Yang, L. Wang, R. Yang, H. Stewenius, ve D. Nister., 2006, Stereo matching with color-weighted correlation, hierarchical belief propagation and occlusion handling. *CVPR*, volume 2, 2347–2354.
69. K.-J. Yoon ve I.-S. Kweoni, 2005, Locally Adaptive Support Weight Approach for Visual Correspondence Search, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. II:924-931,
70. Y. Taguchi, B. Wilburn ve L. Zitnick., 2008, Stereo reconstruction with mixed pixels using adaptive over-segmentation. *CVPR 2008*.
71. D. Scharstein, R. Szelisk., 2008, *Middlebury stereo vision research page*. <http://bj.middlebury.edu/schar/stereo/newEval/php/results.php> [Ziyaret Tarihi: 3 Mayıs 2008].
72. Sun J., Zheng N.N. ve Shum H.Y., 2003, Stereo matching using belief propagation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 25, No. 7, July, pp. 787–800.
73. Sun, J. Li, Y. Kang, S.B. ve Shum H.Y., 2005, Symmetric stereo matching for occlusion handling, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June, pp. 399–406.
74. Lin M.H. ve Tomasi C., 2004, Surfaces with occlusions from layered stereo, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No. 8, pp. 1073–1078.
75. Kim J., Kolmogorov V. ve Zabih R., 2003, Visual correspondence using energy minimization and mutual information, *Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV)*, IEEE Computer Society, Washington, DC, USA, 1033–1040.
76. Snow D., Viola P. ve Zabih R., 2000, Exact voxel occupancy with graph-cuts, *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June, 345–352.
77. D. Comaniciu ve P. Meer, 2002, Mean shift: A Robust Approach Toward Feature Space Analysis, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 4.
78. D. Greig, B. Porteous ve A. Seheult, 1989, Exact maximum a posteriori estimation for binary images. *Journal of the Royal Statistical Society, Series B*, 51(2):271-279.

79. D. Marr ve T. Poggio, 1976, Cooperative computation of stereo disparity. *Science*, 194:209-236.
80. D. Marr ve T. Poggio, 1979, A computational theory of human stereo vision. *Proc. Royal Soc. London B*, 204:301-328.
81. C.L. Zitnick ve T. Kanade. 2000, A cooperative algorithm for stereo matching and occlusion detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7): 675-684.
82. Y. Zhang ve C. Kambhamettu, 2002, Stereo matching with segmentation-based cooperation. *LECT NOTES COMPUT SC*, 2351, 556-571.
83. Z. Wang, Z. Zheng, 2008, A Region Based Stereo Matching Algorithm Using Cooperative Optimization, *CVPR*.
84. Y. C. Hsieh, D. McKeown, ve F. P. Perlant, 1992, Performance evaluation of scene registration and stereo matching for cartographic feature extraction. *IEEE TPAMI*, 14(2):214–238.
85. Y. Deng, Q. Yang, X. Lin, 2005, A symmetric Patch-Based Correspondence Model for Occlusion Handling, *Proc. 10th IEEE International Conference on Computer Vision*. 12(4): 375-379.
86. Ashutosh Saxena, Min Sun, Andrew Y. Ng., 2007, 3-D Reconstruction from Sparse Views using Monocular Vision, *ICCV workshop on Virtual Representations and Modeling of Large-scale environments (VRML)*

EK

DÖNGÜSEL İNANÇ YAYILIMI

“Döngüsel İnanç Yayılımı” ile ilgili olarak öncelikle, “Markov Rasgele Alanları” uygulamalarında sonuca ulaşmak için “İnanç Yayılımı” yaklaşımının kullanılması konusunu kısaca değerlendirmek gerekmektedir. “Maksimum Çarpım Algoritması” özellikle, aşağıdaki denklem (ek-1) örneğinde olduğu gibi, yaklaşık minimum maliyete sahip etiketlemeyi sağlayacak enerji fonksiyonlarını bulmak için kullanılabilir.

$$E(f) = \sum_{(p,q) \in N} V(f_p, f_q) + \sum_{p \in P} D_p(f_p) \quad (\text{ek.1})$$

Normalde bu algoritma olasılık dağılımları çerçevesinde tanımlanmakta, fakat eşdeğer bir hesaplama *Maksimum Çarpım* algoritmasının *Minimum Toplam*’a dönüştüğü negatif logaritmik olasılıklar ile de yapılabilmektedir. Bu denklem, sayısal bozulmalara karşı daha az duyarlı olduğu ve enerji fonksiyonunu daha açık kullandığı için tercih edilmektedir.

Maksimum Çarpım İnanç Yayılımı Algoritması, dört bağlantılı imge dizilimi ile tanımlanan çizgenin etrafından iletiler geçirmek suretiyle çalışmaktadır. Her bir ileti, olası etiketlerin belirlediği boyutta vektörlerdir. p noktasının t zamanda komşu q noktasına gönderdiği ileti ‘ m_{pq}^t ’ olarak adlandırılmaktadır. Negatif logaritmik olasılıklar kullanılırken bütün m_{pq}^0 başlangıç girdileri sıfırdır ve her döngüde yeni iletiler aşağıdaki gibi hesaplanır;

$$m_{pq}^t(f, q) = \min_{f_p} \left(V(f_p, f_q) + D_p(f_p) + \sum_{s \in N(p) \setminus q} m_{sq}^{t-1}(f_p) \right) \quad (\text{ek.2})$$

$N(p) \setminus q$, q ’dan başka p ’ye komşu noktaları belirtmektedir. T döngülerinden sonra her bir nokta için bir inanç vektörü hesaplanmış olur.

$$b_q(f_q) = D_p(f_q) + \sum_{p \in N(q)} m_{pq}^T(f_q) \quad (\text{ek.3})$$

Sonuçta, ayrı olarak her bir nokta için $b_q(f_q)$ niceliğini en aza indirgeyen bir nokta seçilmektedir. Görüntü dizilimleri üzerinde ileti geçirilen algoritmanın standart uygulamasında süre $O(nk^2T)$ olarak hesaplanır. Burada n imgedeki piksel sayısını, k her bir piksel için etiket sayısını ve T ise döngülerin sayısını ifade etmektedir. Temel olarak her bir iletinin hesaplanması için $O(k^2)$ kadar zaman ve her bir döngü için de $O(n)$ kadar ileti gerekmektedir.

Bir ileti güncellemesinin hesaplama süresini makul düzeyde azaltan üç farklı teknik kullanılmaktadır. Birinci olarak, dönüştürme tekniklerinin yardımıyla bilinen $V(f_p, f_q)$ düzlük fonksiyonları kullanılarak, yeni bir iletinin $O(k)$ zamanda hesaplanmasını sağlamaktadır. İkinci olarak, çizge dizilimleri (iki kısımlı çizge) için esasen birinci teknikte anlatılan inançların aynılarının elde edildiği ama bunun için neredeyse birinci tekniktekinin yarısı kadar ileti güncellemesi yapıldığı belirtilmektedir. Daha hızlı bir teknik olmasının yanı sıra iletilerin, normal algoritmanın yarısı kadar bellek kullanılarak yerinde hesaplanmasını sağlamaktadır. Bu, her bir piksel için çoklu dağıtım yapan, yüksek bellek değerleri isteyen *İnanç Yayılımı* açısından hayati önem taşımaktadır. Son olarak da çoklu ölçekli algoritma tekniği ile inanç yayılımının ham halden ince ayarlamalara kadar gerçekleştirilmesi değerlendirilmektedir. Çoklu ölçekli algoritma tekniği yaklaşımında, iletilerin geçirdiği döngü sayısı (T) az olabilmektedir. Bunun nedeni, büyük aralıktaki etkileşimlerin, kaba ölçekteki çizgedeki kısa yollar sayesinde yakalanmasıdır. Bunun yanında problemlerde çoğunlukla, bir bilginin imge üzerinde yayılmasını sağlaması açısından uzaklığı sınırlayan T değerinin normal algoritmalar için yüksek olması gerekmektedir. Standart algoritmada imgenin tümüne dağılmadan belli bir kısımdan bilgi elde edebilmek için T değerinin $n^{1/2}$ şeklinde büyümesinin sağlanması gerektiği ortaya çıkmaktadır.

Bütün bu tekniklerin birleştirilmesi ile uygulamada oldukça hızlı olan $O(nk)$ algoritması elde edilmiştir. Bu algoritma ile denklemdeki enerji fonksiyonlarını en aza indirmek için elde edilen sonuçlar, maksimum çarpımlı inanç yayılımı veya çizge kesimli algoritmaları kullanarak elde edilen sonuçlar kadar kesindir.

İletilerin Hesaplanması

Bu bölümde, *Düşük Seviye Görüntü Uygulamalarında* kullanılmak üzere bir iletinin güncellenmesi için gereken süreyi $O(k^2)$ 'den $O(k)$ 'ye indirgeyen tekniğe değinilmektedir. Ek.2 yeniden yazıldığında aşağıdaki gibi düzenlenmektedir;

$$m_{pq}^t(f, q) = \min_{f_p} (V(f_p, f_q) + h(f_p)) \quad (\text{ek.4})$$

Burada, $h(f_p) = D_p(f_p) + \sum_{s \in N(p) \setminus q} m_{sq}^{t-1}(f_p)$ olarak belirtilmiştir. İletileri hesaplamının

standart yolu, her bir f_q seçeneği için ifadeyi f_p üzerinden minimize etmektir. k değişkeninin, etiket sayılarını belirttiği bu işlem $O(k^2)$ kadar zaman almaktadır. Bazı düşük seviye görüntü problemlerinde maliyet hesaplaması $V(f_p, f_q)$, etiketlerin ayrı ayrı parçalar halinde değil, f_q ve f_p etiketleri arasındaki farkın ölçülmesine dayanmaktadır. Bazı durumlarda iletler piktoriyal strüktürlere ve HMM'lere benzer tekniklerle $O(k)$ kadar zamanda hesaplanabilmektedir. Bu yöntem etiketlerin, k , sayısının yüzlerce ve hatta binlerce olduğu imge restorasyonu ve hareket kestirimi için önemlidir. Güncel algoritmalar, büyük ölçekteki etiket setlerini içeren bu tipteki problemler için çok kullanışlı olamamaktadır.

Potts modeli, etiketlerin parça başına sabit olduğu kabulüne dayanmaktadır. Bu model yalnızca etiketlerin eşit olması ve olmaması durumlarını değerlendirir. Eşit değerdeki etiketler için maliyet sıfırdır. Farklı değerdeki etiketler için ise pozitif bir değerdir.

$$V(f_p, f_q) = \begin{cases} 0 & \text{eger } f_p = f_q \\ d & \text{diger durumlar} \end{cases} \quad (\text{ek.5})$$

Bu maliyet fonksiyonu ile denklem aşağıdaki gibi açıklanabilir;

$$m_{pq}^t(f, q) = \min_{f_p} \left(h(f_p), \min_{f_p} h(f_p) + d \right) \quad (\text{ek.6})$$

Bu şekilde f_q değerinden bağımsız olarak f_p üzerinden minimizasyon gerçekleştirilebilir. Bu iletii hesaplamak için $O(k)$ kadar zaman yeterlidir. İlk olarak

$\min_{f_p} h(f_p)$ hesaplanır. Bu hesabın sonucu her bir f_q için ileti deęerini hesaplamakta kullanılır. Bu fikir çizgedeki her bir sınır için, basit bir sabit olan d yerine sabit d_{pq} olduęu zaman da uygulanabilmektedir. Bu teknik sınır tanımlama gibi başka işlemler için de yararlıdır.

ÖZGEÇMİŞ

Evren BUĞDAYCIOĞLU, 09.11.1979 yılında İstanbul'da doğdu. İlköğrenimini İlhami Ahmed Örnekal'da, tamamladı. Ortaokul ve liseyi Özel Saint-Joseph Fransız Lisesi'nde okudu. İstanbul Üniversitesi Elektronik Bölümü'nü bitirdi. 2005 yılında İstanbul Üniversitesi Fen Bilimleri Enstitüsü, Elektrik-Elektronik Müh. Anabilim Dalı'nda yüksek lisans eğitimine başladı. Tezinde stereo görüntülerden üç boyutlu yapının elde edilmesi üzerine çalışmalar yaptı ve Ekim 2008'de yüksek lisans öğrenimini tamamladı.