

**T.C.**  
**AKDENİZ ÜNİVERSİTESİ**  
**SAĞLIK BİLİMLERİ ENSTİTÜSÜ**  
**BİYOİSTATİSTİK VE TIBBİ BİLİŞİM ANABİLİM DALI**



**BEZIER EĞRİLERİ İLE DEFORME EL MODELİNİN**  
**GELİŞTİRİLMESİ**

**Buket ŞİMŞEK**

**YÜKSEK LİSANS TEZİ**

**2016-ANTALYA**

**T.C.**  
**AKDENİZ ÜNİVERSİTESİ**  
**SAĞLIK BİLİMLERİ ENSTİTÜSÜ**  
**BİYOİSTATİSTİK VE TIBBİ BİLİŞİM ANABİLİM DALI**

# **BEZİER EĞRİLERİ İLE DEFORME EL MODELİNİN GELİŞTİRİLMESİ**

**Buket ŞİMŞEK**

**YÜKSEK LİSANS TEZİ**

**DANIŞMAN**  
**Doç. Dr. Ahmet YARDIMCI**

“Kaynakça gösterilerek tezimden yararlanılabilir”

2016-ANTALYA

**Sağlık Bilimleri Enstitüsü Müdürlüğüne;**

Bu çalışma jürimiz tarafından Biyoistatistik ve Tıbbi Bilişim Anabilim Dalı Tıp Bilişimi Programında yüksek lisans tezi olarak kabul edilmiştir. 25 Kasım 2016

İmza

Tez Danışmanı : Doç. Dr. Ahmet YARDIMCI .....

Akdeniz Üniversitesi

Üye : Prof. Dr. Mustafa ALKAN .....

Akdeniz Üniversitesi

Üye : Doç. Dr. Tuncay AYDOĞAN .....

Süleyman Demirel Üniversitesi

Bu tez, Enstitü Yönetim Kurulunca belirlenen yukarıdaki jüri üyeleri tarafından uygun görülmüş ve Enstitü Yönetim Kurulu'nun ...../...../.....tarih ve...../..... sayılı kararıyla kabul edilmiştir.

**Enstitü Müdürü**

**Prof. Dr. Narin DERİN**

## ETİK BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün safhalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı beyan ederim.

Öğrencinin  
Buket ŞİMŞEK  
İmza

Tez Danışmanı  
Doç. Dr. Ahmet YARDIMCI  
İmza

## TEŞEKKÜR

Yüksek lisans çalışmamda bana sabırla yol gösteren, desteklerini esirgemeyen, her zaman bana yardım eden saygı değer hocam Doç. Dr. Ahmet YARDIMCI'ya en içten saygı, sevgi ve teşekkürlerimi sunuyorum.

Ayrıca tez jürisinde bulunan değerli hocalarıma katkıları için teşekkür ediyorum.

En büyük destekçim babam Prof. Dr. Yılmaz ŞİMŞEK tez çalışmama vermiş olduğu katkılarından dolayı sonsuz teşekkür ediyorum.

Tez çalışmama vermiş olduğu katkılardan dolayı Dr. İrem KÜÇÜKOĞLU'na sonsuz teşekkür ediyorum.

Her zaman yanımda olup bana destek olan annem, ablam ve tüm arkadaşlarıma teşekkür ediyorum.

## ÖZET

Son zamanlarda bilgisayarların hızlı gelişimi, matematik tıp, fizik ve mühendislik alanında kolaylık sağlamaya başladı. Bu kolaylıkların birincisi, hızlı işlem yapabilme kabiliyetidir. İkincisi ise artan depolama kapasiteleri yoğun veri giriş ve çıkışlı sistemlerin, bilgisayarlar üzerinde incelenmesine imkan sağlamaktadır. Bu gelişmeler sonucu, doğadaki sistemlerin gerçeğe daha yakın bir biçimde benzetim modelleri kurulabilmekte, daha hassas ve gerçekle neredeyse aynı hesaplar yapılabilmektedir.

Bu tezin amacı Bernstein polinomları ile belirlenen Bezier eğrileri yardımıyla normal el ve deforme el modelleri tanımlayarak, yapılacak girişimlerin sonuçlarını model üzerinde incelemektir.

Bu tezde Bezier eğrilerinin, tıp alanında kullanımını gösteren önceki çalışmaların bazıları özetlenmiştir. Ayrıca kullanılan yöntem, matematikte, istatistikte ve programlamada kullanılan yöntemlere dayanmaktadır. Bernstein polinomları, Bezier eğrileri, B-spline eğrilerinin tanımları verilmiş ve aralarındaki ilişkiler incelenmiştir. Bezier eğrileri ve B-Spline eğrileri yardımıyla C++ ve Autocad programlarında el modelleri çizilmiş ve elde edilen modeller arasında karşılaştırma yapılmış ve bu modellerin matematiksel denklemleri de verilmiştir. Son olarak tezde elde ettiğimiz modeller yardımıyla bir makine öğrenme algoritması ve akış diyagramı verilmiştir.

**Anahtar Kelimeler:** Bezier eğrileri, C++, B-Spline eğrileri, Dağılım fonksiyonları, Matematiksel modelleme

## ABSTRACT

**Objective:**In recent years, the rapid developments of computers have provided important convenience in mathematics, medicine, physics and engineering. The first convenience is the ability of performing mathematical computations in a fast way. The second convenience is that the increasing storage capacities of computers allow evaluations of big and intensive data. As a result of these developments in computers, simulation models can be constructed to capture the nature of the systems in unique way and accurate way. Moreover, these simulation models yield more sensitive and realistic results that are as close as to its nature.

**Method:**In this thesis, simulation models is specifically used to study the nature of human hands. The aim of this thesis is to define mathematical models of deformed and undeformed human hands by using the Bernstein polynomials and the Bezier curves. Investigation of these mathematical models is helpful to understand the nature of the deformed and undeformed human hands.

**Results:**Briefly literature review of the mathematical models is presented in this thesis by summarizing some of its usage in medicine and related areas. The method that is used in this thesis is quite well-known because it is developed by the standard methods from mathematics, statistics, and computer science. After introducing basic definitions and properties of the Bernoulli polynomials, the Bezier curves, and the B-spline curves, the relationship between them are examined. By using C++ and AutoCAD, the Bezier curves and B-spline curves are drawn to construct the models for deformed and undeformed hand. Comparisons of these constructed models are also summarized. In addition, the mathematical equations of these models are provided to illustrate the details of the models.

**Conclusion:**Finally, by using the results of our models that we obtained as a part of this thesis, we construct a new algorithm which can be used in future studies in related areas.

**Key words:**Bezier Curves,C++, B-Spline Curves, Distrubition Function, Mathematical modelling

# İÇİNDEKİLER

|                                                                                               |            |
|-----------------------------------------------------------------------------------------------|------------|
| <b>ÖZET</b>                                                                                   | <b>i</b>   |
| <b>ABSTRACT</b>                                                                               | <b>ii</b>  |
| <b>İÇİNDEKİLER</b>                                                                            | <b>iii</b> |
| <b>ŞEKİLLER DİZİNİ</b>                                                                        | <b>iv</b>  |
| <b>SİMGELER ve KISALTMALAR</b>                                                                | <b>vi</b>  |
| <b>1.GİRİŞ</b>                                                                                | <b>1</b>   |
| <b>2. GENEL BİLGİLER</b>                                                                      | <b>3</b>   |
| 2.1. Bezier Eğrileri                                                                          | 4          |
| 2.2. Bernstein Polinomları                                                                    | 5          |
| 2.3. Bernstein Polinomları İle Olasılık İstatistik Dağılım Fonksiyonları Arasındaki İlişkiler | 6          |
| 2.4. Bezier Eğrilerinin Bernstein Polinomları İle Belirlenmesi                                | 7          |
| 2.4.1. Lineer (Doğrusal) Bezier Eğrileri                                                      | 8          |
| 2.4.2. Kuadratik Bezier Eğrileri(2D)                                                          | 8          |
| 2.4.3. Kübik Bezier Eğrileri (3D)                                                             | 9          |
| 2.5. Bezier Yüzeyleri                                                                         | 10         |
| 2.6. B-Spline Eğrileri                                                                        | 12         |
| <b>3. GEREÇ ve YÖNTEM</b>                                                                     | <b>14</b>  |
| <b>4. BULGULAR</b>                                                                            | <b>24</b>  |
| 4.1. AutoCAD’de çizilen el modelleri                                                          | 25         |
| 4.2.Normal El Modeli                                                                          | 27         |
| 4.3.Birinci Deforme El Modeli                                                                 | 29         |
| 4.4.İkinci Deforme El Modeli                                                                  | 30         |
| 4.5.Üçüncü Deforme El Modeli                                                                  | 31         |
| 4.6. Bir Eğrinin Eğriliği                                                                     | 33         |
| <b>5. TARTIŞMA</b>                                                                            | <b>35</b>  |
| <b>6. SONUÇ ve ÖNERİLER</b>                                                                   | <b>36</b>  |
| <b>KAYNAKLAR</b>                                                                              | <b>38</b>  |
| <b>ÖZGEÇMİŞ</b>                                                                               | <b>41</b>  |

## ŞEKİLLER DİZİNİ

|                                                                                       |    |
|---------------------------------------------------------------------------------------|----|
| Şekil2.1: Lineer Bezier eğrisi                                                        | 8  |
| Şekil2.2: Kuadratik Bezier eğrisi                                                     | 9  |
| Şekil2.3: Kübik Bezier eğrisi                                                         | 10 |
| Şekil2.4: 16 kontrol noktası ile tanımlanmış bir Bezier yüzey                         | 11 |
| Şekil2.5: 16 kontrol noktasıyla tanımlanmış üç boyutlu Beziér yüzeyi                  | 11 |
| Şekil2.6: Kuadratik Düzgün B-Spline Eğrisi                                            | 12 |
| Şekil2.7: İki Bezier eğri segmentinin birleştirilmesi ile elde edilen B-spline eğrisi | 13 |
| Şekil3.1:Dört kontrol noktası olan bir Bezier eğrisi uygulamasının ekran görüntüsü    | 23 |
| Şekil4.1:Soldan 1. Bezier eğrisi                                                      | 24 |
| Şekil4.2: Soldan 2.Bezier eğrisi                                                      | 24 |
| Şekil4.3:Soldan 3.Bezier Eğrisi                                                       | 24 |
| Şekil4.4: Soldan 4.Bezier Eğrisi                                                      | 24 |
| Şekil4.5: Tüm elin Bezier modeli                                                      | 25 |
| Şekil4.6: AutoCAD programında çizilmiş el eğri modelinin ekran görüntüsü              | 26 |
| Şekil4.7: AutoCAD programında çizilmiş parmak arası eğri modeli                       | 26 |
| Şekil4.8: AutoCAD programında çizilmiş el modeli                                      | 27 |
| Şekil4.9:C++ programında çizilmiş normal el modelinin ekran görüntüsü                 | 28 |
| Şekil4.10: Normal el için Bezier Eğrisi                                               | 28 |
| Şekil4.11:C++ programında çizilmiş birinci deforme el modelinin ekran görüntüsü       | 29 |

|                                                                                        |    |
|----------------------------------------------------------------------------------------|----|
| <b>Şekil4.12:</b> Birinci deforme el için Bezier Eğrisi                                | 29 |
| <b>Şekil4.13:</b> C++ programında çizilmiş ikinci deforme el modelinin ekran görüntüsü | 30 |
| <b>Şekil4.14:</b> İkinci deforme el için Bezier Eğrisi                                 | 30 |
| <b>Şekil4.15:</b> C++ programında çizilmiş üçüncü deforme el modelinin ekran görüntüsü | 31 |
| <b>Şekil4.16:</b> Üçüncü deforme el için Bezier Eğrisi                                 | 32 |
| <b>Şekil4.17:</b> Bir Eğrinin Eğriliği                                                 | 33 |
| <b>Şekil4.18:</b> Deforme el tespiti için bir makine öğrenme modeli                    | 34 |

## SİMGELER ve KISALTMALAR

### SİMGELER:

$B_k^n(t)$  : Bernstein Polinomu

$B(t)$  : Bezier Eğrisi

$E(X)$  : Beklenen değer

$\binom{n}{k}$  : Binom katsayısı

2D: 2 Boyutlu (Kuadratik)

3D: 3 Boyutlu (Kübik)

$K$  : Eğrinin Eğriliği

$P(u, v)$ : Bezier Yüzeyi

$Var(X)$  : Varyans

## 1.GİRİŞ

Günümüzde bilgisayarların hızlı gelişimi tıp, fizik ve mühendislik alanında kolaylık sağlamaya başladı. Bu kolaylıkların birincisi, hızlı işlem yapabilme kabiliyetidir. İkinci kolaylık ise artan depolama kapasiteleri yoğun veri giriş ve çıkışlı sistemlerin, bilgisayarlar üzerinde incelenmesine olanak vermektedir. Bu gelişmeler sonucunda, doğadaki sistemlerin gerçeğe daha yakın şekilde benzetim modelleri kurulabilmektedir. Ayrıca daha hassas ve gerçekle hemen hemen aynı hesaplar yapılabilmektedir.

Mimarlık ve mühendislik alanlarında, el yazısı tanıma, bilgisayar grafikleri, özellikle tasarım geliştirme aşamasında zaman kazandıran bir etkidir. Neredeyse geliştirilen tüm modeller, bilgisayar üzerinde yapılan modellemelerin sonucunda en iyi biçimde gerçekleştirilebilir. Bilgisayar grafikleri mimarlık, mühendislik alanlarının yanı sıra tıp alanında da kullanılmaktadır. Bir objenin 2 boyutlu kesitlerinden 3 boyutlu görünümünün elde edilebilmesine sağladığı olanak ile medikal fizik ve medikal informatik gibi alanlarda çok sayıda uygulama alanında kullanılmıştır (Park ve Kim, 1996; Li ve ark., 2006; Goldman, 2002;Farouki, 2012).

Bilgisayar grafikleri, tasarlanan bir parçanın bütünüyle tanımlanarak bir Bilgisayar Destekli Tasarım (BDT) sisteminde korunmasını sağlamaktadır. Bu işlem, nesnelerin oluşturulmasında hem görsel hem de parçaya ait sayısal değerlerin korunmasını kapsamaktadır. Geometrik modelleme bir nesnenin matematiksel ve kuramsal ilişkilerini açıklar ve geometrinin verimli bir şekilde bilgisayarda görselleştirilmesine odaklanır. Bu bağlamda bilgisayar yardımıyla tıpta özellikle estetik cerrahi alanında yüz, el ve meme gibi uzuvlar üzerinde yapılacak cerrahi işlemlerin beklenen sonuçlarının oluşturulan görüntü üzerinde işlemiden önce değerlendirilebilmesinin büyük avantajı vardır. İşte bu noktada nesnelerin bilgisayarda nasıl ifade edileceği meselesi karşımıza çıkar.

Bu konu hakkında çalışmalar 1960 yılında başlamıştır. Son zamanlarda birçok yöntem geliştirilmiş, Bilgisayar Destekli Tasarım paketlerinde daha verimli ve gerçeğe yakın olarak geometrik şekillerin oluşturulması için çalışmalar yapılmaya devam edilmiştir. Bézier Eğrileri, Spline, B-Spline, T-Spline ve en sonunda NURBS gibi modelleme

yapmak için kullanılan, eğri oluşturma yöntemleri Bilgisayar Destekli Tasarım programlarının çok sık kullandığı matematiksel modellerdir (Farin, 1993).

Farklı madde özelliklerine sahip nesneleri bilgisayarda tanımlamak için birden fazla gösterim biçimi vardır. Daha gerçekçi görüntüler oluşturmak için bu nesne özelliklerini doğru bir biçimde modellemek gerekmektedir. Çokgenler ve ikinci dereceden yüzeyler, çokyüzlüler ve elipsoidlerin elde edilmesinde kullanılırlar. Modelleme yapmak için birçok üç boyutlu görüntüleme tekniği vardır(Hearn, 1997).

Bu tezin amacı “ Bernstein polinomları ile belirlenen Bezier eğrileri yardımıyla normal el ve deforme el modelleri tanımlayarak, yapılacak girişimlerin sonuçlarını model üzerinde inceleyebilmektir.”

Bu tez çalışmasının içeriği aşağıda bahsedilen adımlardan oluşturmaktadır.

- Bu çalışmanın Bezier eğrileri üzerine tıp alanında yapılan bazı çalışmaların özeti ve Bezier eğrilerinin tıptaki bazı uygulamaları verilmiştir.
- Bu çalışmada kullanılacak matematiksel eğri modelleri, fonksiyonları, denklemleri, bu eğri modelleri hakkında bilgi ve aralarındaki ilişki verilecektir.
- İnternet üzerinden alınan normal el ve deforme olmuş el modelleri için C++ programlama dilinde oluşturulan Bezier eğrisi ve bilgisayar destekli tasarım programı olan AutoCAD programında oluşturulan B-spline eğrisi uygulamaları verilmiştir ve elde edilen bu eğri modelleri karşılaştırılmıştır.
- Son olarak üzerinde çalıştığımız el modelleri için elde edilen sonuçlar karşılaştırılıp bulunan sonuçlar değerlendirilmiştir.

## 2. GENEL BİLGİLER

Sevinç, üzüntü, kızgınlık, korku iğrenme, şaşırma olmak üzere altı temel yüz ifadesinin tanınmasını hedefleyen Özmen çalışmasında AdaBoost algoritmasını kullanarak yüz tanıma tespiti yapmıştır. Yüz tespitini yaptıktan sonra yüz ifadelerinin sebep olduğu değişiklikleri çıkarmıştır. Bunu Geometrik Özellik Tabanlı yöntem ile yapmıştır. Bu yöntemde yüz ifadelerindeki yerleşimleri ve şekilleri (ağız, gözler, burun, kaşlar) bulmuştur. Daha sonra Haar Özelliklerini kullanarak yüze ait ya da yüzde belirli bölgeler için özellik vektörünü çıkarmıştır. Yüzdeki yerleşim ve şekilleri bulunduktan sonra Bezier eğrilerini kullanarak FREETUM veritabanı üzerinde rastgele seçtiği resimlerde dudak ve göz çevresinde eğriler oluşturup bu eğriler ile yüz ifadesi belirlemiştir(Özmen, 2012).

Yüz ifadesinin tıptaki yeri;

- Tıbbi teşhis (ağrı değerlendirmek için)
- Duygu ile ilgili araştırma (davranış bilimleri, klinik psikoloji, nöroloji, psikiyatri, gibi alanlarda)
- Ayrıca insan-bilgisayar etkileşiminde yüz ifadesi analizi motor nöron rahatsızlıkları nedeniyle hareket edemeyen felçli hastaların yüz ifadelerindeki değişim takip edilip, hasta ile bilgisayar arasında iletişim kurmak için yapılmıştır. Burada felçli hastaların göz kırpmaya sıklığına göre bilgisayara farklı komutlar göndermektedir(Güneş, 2009).

Khan ve Bhuiyan'nın Bezier eğrileri yaklaşım tekniği ve yüz geometrisi tekniği kullanarak tendeki renk tonu tabanlı ve yüz ifadelerindeki değişimleri incelenmiş ve bu inceleme sonucu bu tekniklerin %100'e yakın oranda başarı elde ettiği görülmüştür. Bu teknik ile elde edilen yüz ifadeleri komut olarak manipülatör robota aktarılıp test edilmiştir (Khan, M., İ., Bhuiyan A,2010).

Tendeki renk tonu tabanlı yüz bulma çalışması yapan Khan ve Bhuiyan, yüz bulmada daha iyi sonuç elde edebilmek için bulanık mantık kullanmışlardır. Yüz ifadeleri tanıma işlemi Bezier eğrileri kullanılarak dudak ve göz çevresinde eğri modelleri yapılmıştır.

Bu eğri modelleri yardımıyla yüzdeki hareket ve ifade değişimleri verilmiştir (Khan, M., İ., Bhuiyan A,2009).

Taner ve arkadaşlarının yapmış olduğu çalışmada ağızdaki kavislerin uzunlamasına genişliği, formdaki değişiklikleri ve kavis formlarının tipleri yeni bir bilgisayar metodu ile değerlendirilmiştir. Bu kavis formlarını tanımlamak için matematiksel olarak çeşitli eğri modelleri ve denklemler kullanmışlardır. Bu eğri modelleri elips ve parabol eğrileridir. Denklemler ise örneğin polinom fonksiyonları, kübik eğriler, konik kesitler, B-fonksiyonları ve kübik Bezier denklemlerdir. Bu çalışmada Kübik Bezier eğri fonksiyonu kullanılarak öncelikle bilgisayar programı Borland Turbo Pascal program dili ile geliştirilmiştir. Bu program Microsoft Visual C++ yeniden yazmışlar ve her diş kasti için Bezier eğri formu ve diğer eğri formlarını üretmişlerdir. Daha sonra üretilen bu eğri formlarını istatistiksel olarak birbirleri ile karşılaştırmışlardır (Taner ve ark., 2004 ).

Roth'un yapmış olduğu çalışmada sonlu elemanlar metodu ve statik lineer elastisite simülasyon uygulaması çalışmalarının uzun bir araştırma geçmişi vardır. Aynı uygulamalar Bernstein-Bezier eğriler ve yüzeylerin bilgisayar destekli geometrik tasarım yüzeylerinde gösterimleri içinde geçerlidir. Bu çalışmada yüz cerrahisi simülasyon çerçevesinin teorisi ve bu unsurların uygulanması üzerinde odaklanmıştır. Kısa bir motivasyon ve mevcut deforme olan modelleri bilgisayar grafikleri ve cerrahi simülasyonlarını gözden geçirmiştir. Yüzeylerin ve hacimlerin temsili Bernstein-Bezier eğri yamaları sayesinde elde etmiştir. Barisentrik temsilde ve yapımında pürüzsüz yama geçişlerini almıştır. Roth çalışmasında yöntem ve çözüm önerileri için gereken modeli geliştirilmiş ve cerrahi değişimin ve gerçekte olan değişimin otomatik hesaplanmasını tanımlamıştır. Sonuç olarak bir test hastasını örnek alarak elde edilen sonuçlarını bu tezde sunmuştur(Roth, 2002).

## **2.1. Bezier Eğrileri**

Bezier eğrileri ilk olarak 1959 yılında Paul de Faget, Citroen araba fabrikasında çalışan bir Fransız otomotiv mühendis tarafından tanımlanmıştır. Aynı dönemde, Renault araba fabrikasında silindir parçalarının kesişimi üzerinde çalışma yapan bir başka Fransız otomotiv mühendisi Pierre Bezier benzer bir çalışmayla araştırmalarını devam ettirmiştir. Paul de Faget ve Pierre Bezier birbirlerinden bağımsız olarak benzer

sonuçları elde etmiş olmalarına rağmen, bu konu hakkında yayınlanan ilk makale Bézier tarafından yazıldığından ve kabul edildiğinden dolayı, günümüzde bu eğrilere, Bézier eğrisi denildiği bilinmektedir (Faruki, 2012; wikipedia).

Aşağıda detaylı olarak vereceğimiz Bezier eğrileri kısaca kendini belirleyecek olan poligonun köşeleri ile birebir bağlantılı olduğu bilinmektedir. Poligonun yalnızca ilk ve son köşeleri eğri üzerinde yer almaktadır. Diğer köşeler ise eğrinin derecesini ve şeklini belirlemek için kullanılmaktadır (Özmen ve Kandemir, 2012).

Bézier eğrisi, matematiksel olarak, genellikle Bernstein polinomları baz alınarak ifade edilir. Bernstein polinomları Bezier eğrilerinin temelini oluştururlar. Buna göre,  $n$ 'inci dereceden temel fonksiyon, kontrol noktaları  $k$  ile parametrize edilmek üzere, aşağıdaki şekilde gösterilir (Bernstein, 1912; Farouki, 2012; Goldman 1995).

## 2.2. Bernstein Polinomları

$t, [0,1]$  kapalı aralığın bir elemanı olsun.  $n$  ve  $k$  negatif olmayan tam sayılar olsun. Bernstein polinomları;

$$B_k^n(t) = \binom{n}{k} t^k (1-t)^{n-k}$$

dır. Burada

$$k = 0, 1, 2, \dots, n$$

ve

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

şeklinde verilir (Goldman, 2002; Farouki, 2012; Salomon, 2006).

### 2.3. Bernstein Polinomları İle Olasılık İstatistik Dağılım Fonksiyonları Arasındaki İlişkiler

Bernstein polinomları ile Binom ya da Newton dağılımı arasında çok yakın bir bağlantı olduğunu söyleyebiliriz. Bernstein polinomlarının derecesini yeterince büyüttüğümüzde yani sonsuza götürdüğümüzde Poisson dağılımına yaklaştığını görebiliriz. Bu dağılımların lazer ışınlarının bir noktada odaklanmasında kullanıldığını biliyoruz. Günümüzde lazer başta tıpta kullanılmak üzere birçok alanda kullanıldığını biliyoruz.  $a$  ve  $b$  iki negatif olmayan reel sayı ve  $a=b$  olsun.  $y \in [a,b]$  olmak üzere

$$x = \frac{y-a}{b-a}$$

dönüşümü yapılırsa Bernstein polinomlar herhangi bir  $[a,b]$  kapalı aralığı üzerinde aşağıdaki şekilde tanımlanır (Farouki, 1996; Goldman, 2009):

$$G(y; a, b; k, n) = \binom{n}{k} \left( \frac{y-a}{b-a} \right)^k \left( \frac{b-y}{b-a} \right)^{n-k}$$

Burada varsayalım ki

$$0 \leq \frac{y-a}{b-a} \leq 1$$

ve

$$0 \leq \frac{b-y}{b-a} \leq 1$$

olsun.

Yukarıdaki varsayım alındığı zaman genelleştirilmiş binom ya da (Newton) dağılımı  $G(y; a, b; k, n)$  ile verilen bağıntıdan dağılım fonksiyonu elde edilir. Eğer  $a = 0$  ve

$b = 1$  alınırsa, bu dağılım  $0 \leq X \leq 1$  olmak üzere Bernstein polinomlarının özel bir durumu olan aşağıdaki binom dağılımına indirgenir:

$$F(X; k, n) = \binom{n}{k} X^k (1 - X)^{n-k}$$

$0 \leq X \leq 1$  olsun. Xrastgele değişkeninin beklenen değeri  $E(X)$  ve varyansı  $Var(X)$  aşağıdaki şekilde hesaplanır (Lorentz, 1986):

$$E(X) = \sum_{k=0}^n k F(X; k, n) = n \frac{X - a}{b - a}$$

ve

$$Var(X) = \sum_{k=0}^n k^2 F(X; k, n) - E^2(X) = \frac{n(X - a)(b - X)}{(b - a)^2}$$

$n \rightarrow \infty$  olduğunda Bernstein polinomları ile elde edilen normal dağılım Poisson dağılımına indirgenir (Jetter and Stöckler, 2003; Şimşek, 2014)

$$G\left(\frac{b-a}{n} \mu + a; a, b; k, n\right) \rightarrow \frac{\mu^k e^{-\mu}}{k!}$$

#### 2.4. Bezier Eğrilerinin Bernstein Polinomları İle Belirlenmesi

$n$ -inci dereceden  $P_0, P_1, P_2, \dots, P_n$  kontrol noktaları verilen Bezier eğrileri aşağıdaki şekilde tanımlanır;

$$B(t) = \sum_{k=0}^n p_k B_k^n(t),$$

Burada  $0 \leq t \leq 1$  dir.

Bezier eğrileri aşağıdaki şekilde sınıflandırılır:

1. Lineer (doğrusal) Bezier eğrisi
2. Kuadratik Bezier eğrisi
3. Kübik Bezier eğrisi

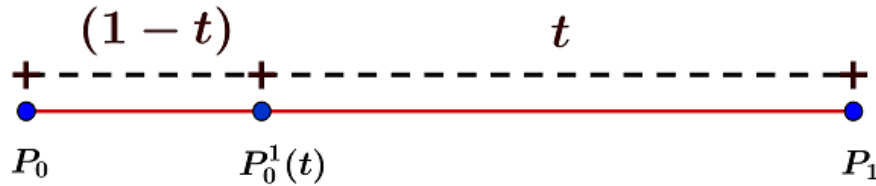
Şimdi bu eğrileri tanımlayalım.

#### 2.4.1. Lineer (Doğrusal) Bezier Eğrileri

$P_0$  ve  $P_1$  iki kontrol noktası verilsin.  $t \in [0,1]$  olsun. Bu kontrol noktalarına bağlı lineer Bezier eğrisi aşağıdaki şekilde tanımlanır (Goldman, 2009; Farouki, 2012; wikipedia):

$$B(t) = (1-t) P_0 + t P_1$$

$P_0$  ve  $P_1$  kontrol noktalarından geçen lineer Bezier eğrisinin gösterimi Şekil2.1'deki gibidir.



Şekil2.1: Lineer bezier eğrisi

#### 2.4.2. Kuadratik Bezier Eğrileri (2D)

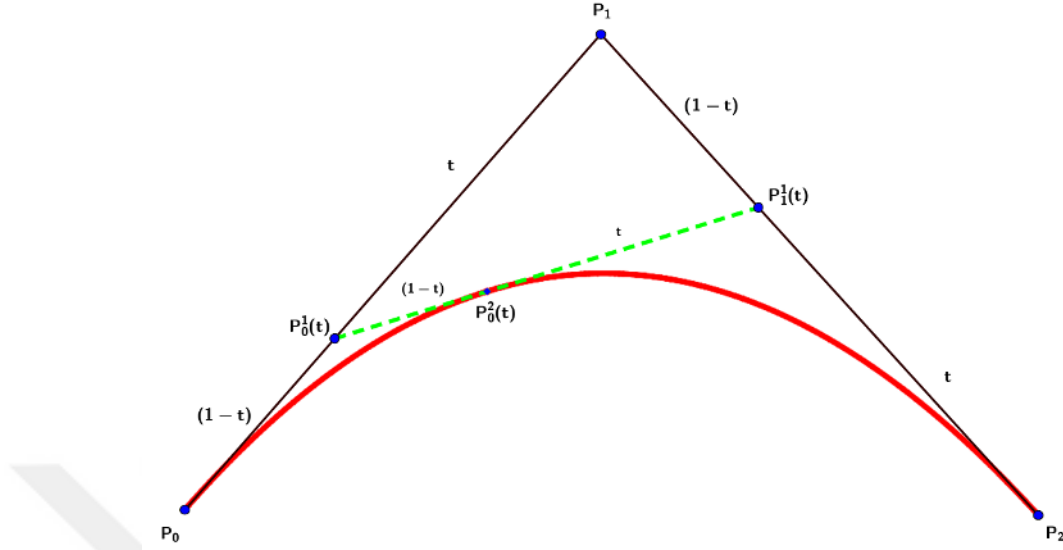
$P_0$ ,  $P_1$  ve  $P_2$  üç kontrol noktası verilsin. Bu noktalara bağlı 2D Bezier eğrisi aşağıdaki şekilde tanımlanır:

$$B(t) = (1-t)^2 P_0 + 2t(1-t) P_1 + t^2 P_2,$$

burada

$$0 \leq t \leq 1$$

dir (Goldman, 2009; Farouki, 2012; wikipedia ).



**Şekil2.2:** Kuadratik Bezier eğrisi

$P_0 = (a_0, b_0), P_1 = (a_1, b_1)$  ve  $P_2 = (a_2, b_2)$  olsun. Şekil2.2'deki eğri aşağıdaki gibi yazılır.

$$B(t) = B(x(t), y(t)),$$

burada  $0 \leq t \leq 1$  olmak üzere aşağıdaki denklem elde edilir:

$$x(t) = a_0 + 2(a_1 - a_0)t + (a_0 + a_2 - 2a_1)t^2$$

$$y(t) = b_0 + 2(b_1 - b_0)t + (b_0 + b_2 - 2b_1)t^2.$$

### 2.4.3. Kübik Bezier Eğrileri (3D)

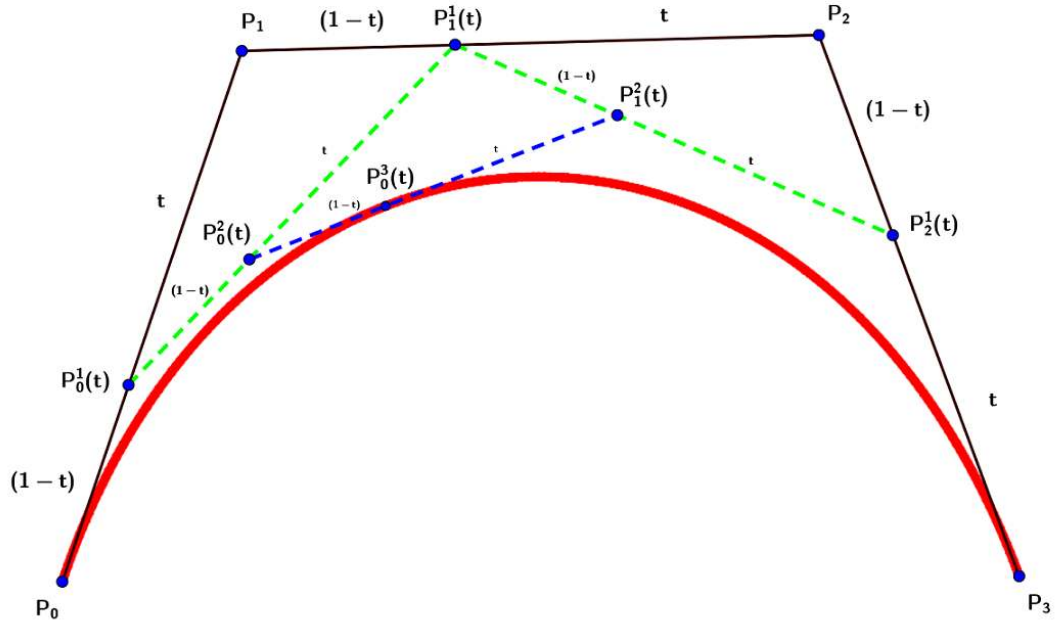
$P_0, P_1, P_2$  ve  $P_3$  dört kontrol noktası verilsin. Bu noktalara bağlı 3D Bezier eğrisi aşağıdaki şekilde tanımlanır:

$$B(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3,$$

burada

$$0 \leq t \leq 1$$

dir (Farouki 2012; wikipedia).



Şekil2.3: Kübik Bezier eğrisi

$P_0 = (a_0, b_0)$ ,  $P_1 = (a_1, b_1)$ ,  $P_2 = (a_2, b_2)$  ve  $P_3 = (a_3, b_3)$  olsun. Şekil2.3’de Bezier eğrisi açık bir şekilde aşağıdaki koordinat sistemi ile verilir.

$$B(t) = (x(t), y(t), z(t))$$

olmak üzere

$$x(t) = a_0 + 3(a_1 - a_0)t + (3a_0 + 3a_2 - 6a_1)t^2 + (3a_3 + 3a_1 - a_0 - 3a_2)t^3,$$

$$y(t) = b_0 + 3(b_1 - b_0)t + (3b_0 + 3b_2 - 6b_1)t^2 + (3b_3 + 3b_1 - b_0 - 3b_2)t^3,$$

$$z(t) = c_0 + 3(c_1 - c_0)t + (3c_0 + 3c_2 - 6c_1)t^2 + (3c_3 + 3c_1 - c_0 - 3c_2)t^3$$

dır.

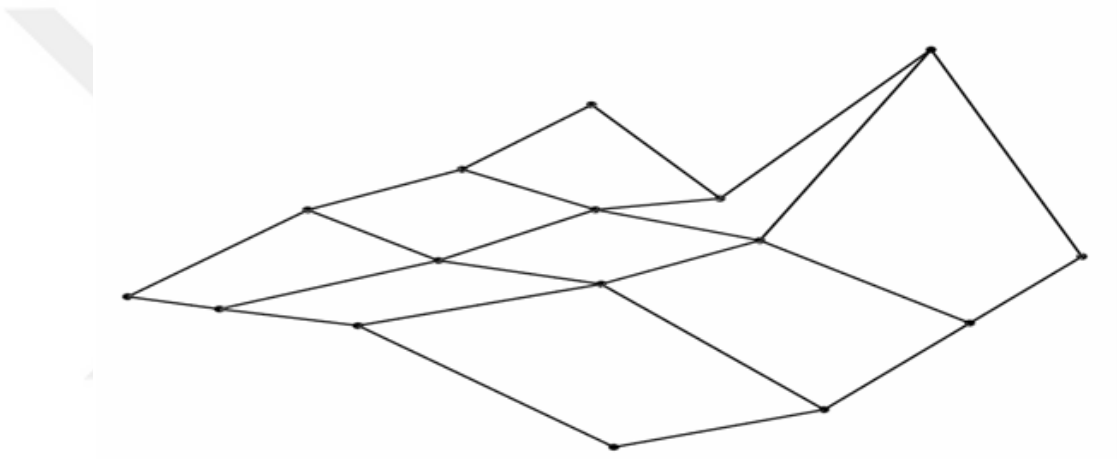
## 2.5. Bezier Yüzeyleri

Beziér yüzeyleri, Beziér eğrilerinin gösteriminin genelleştirilmesi sonucu ortaya çıkan yüzeylerdir. Matematiksel olarak, üç boyutlu (3D) yüzeyleri aşağıdaki şekilde verilir (Hearn ve Baker, 1997; Çetin ve Gündükbay, 2006).

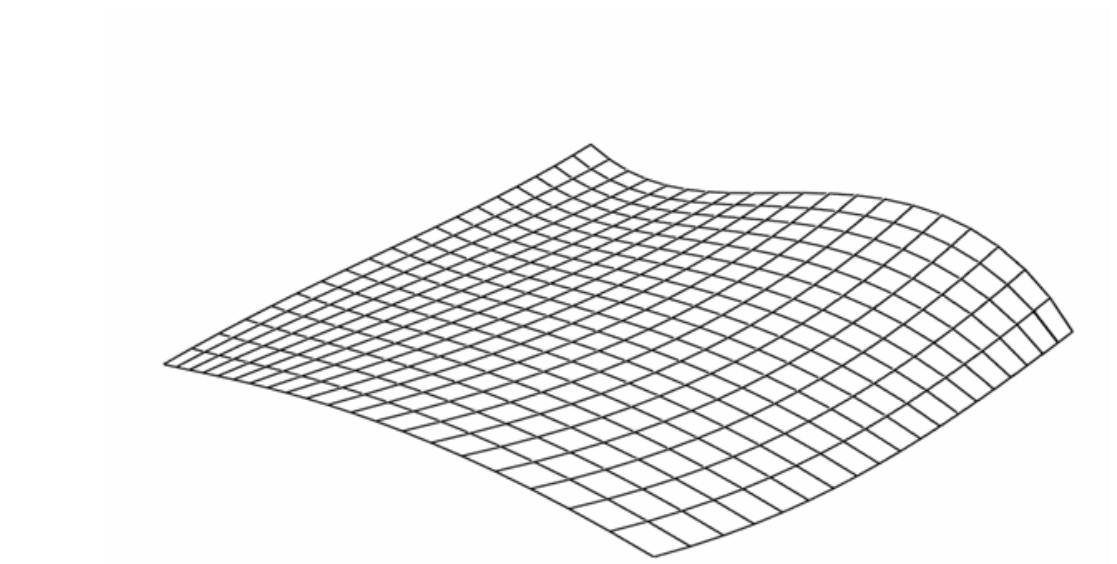
$$P(u, v) = \sum_{i=0}^n \sum_{j=0}^m P_{i,j} B_i^n(u) B_j^m(v)$$

Burada  $P_{i,j}$  kontrol noktalarıdır.

16 kontrol noktası verilen bir Bezier yüzeyinin şekilleri Şekil2.4 ve Şekil2.5'teki gibidir.



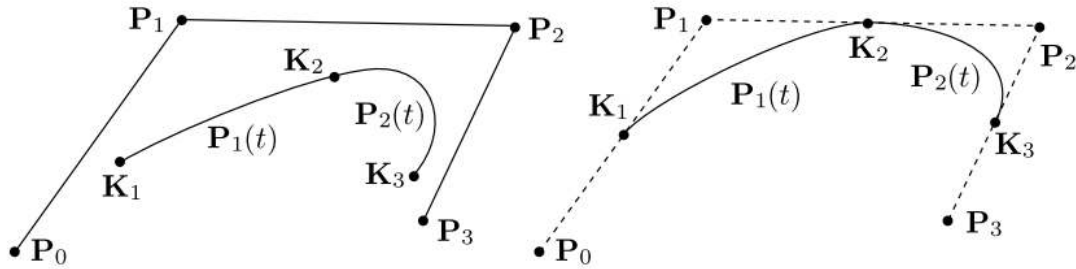
**Şekil2.4:** 16 kontrol noktası ile tanımlanmış bir Bezier yüzey (Çetin ve Güdükbay, 2006)



**Şekil2.5:** 16 kontrol noktasıyla tanımlanmış üç boyutlu Beziér yüzeyi (Çetin ve Güdükbay, 2006)

## 2.6. B-Spline Eğrileri

Bezier metotları eğriler ve yüzeyler için çok kullanılan bir tekniktir. Farklı operatörler ve Bernstein polinomları kullanılarak bu metot inşa edilir. Dikdörtgensel ve üçgensel yüzeyler kullanılarak Bezier yüzeyleri oluşturulabilir. Yüzeyler için B-spline metotları 1940'lı yıllarda başlamış olup 1970'li yıllarda ciddi anlamda araştırmacılar tarafından çalışılmaya başlanmıştır. Bezier eğrilerinin B-Spline eğrilerinden dezavantajlara sahip olmasına rağmen farklı kullanım alanları vardır. Modelleme ve tasarım aşamasında sıkça kullanılmaktadır. Bezier eğrilerinin dezavantajlarından biri Bezier eğrilerinin derecesinin kontrol noktalarına bağlı olmasıdır. Bu dezavantajı kaldırmak için B-spline eğrileri kullanılmaktadır. Bezier eğrilerinde olduğu gibi B-spline eğrilerinde kontrol noktaları ile tanımlanır fakat düğümler aracılığıyla B-spline eğrisi belirlenir. Bu da tasarım ve modelleme sırasında kolaylıklar sağlar. B-spline eğrileri de kendi içlerinde farklı sınıflara ayrılırlar. Bunlar düzdün ve düzgün olmayan B-spline eğrileri olarak adlandırılır. Düzgün olmayan B-spline eğrileri aynı zamanda NURBS olarak adlandırılır (Salomon, 2006).



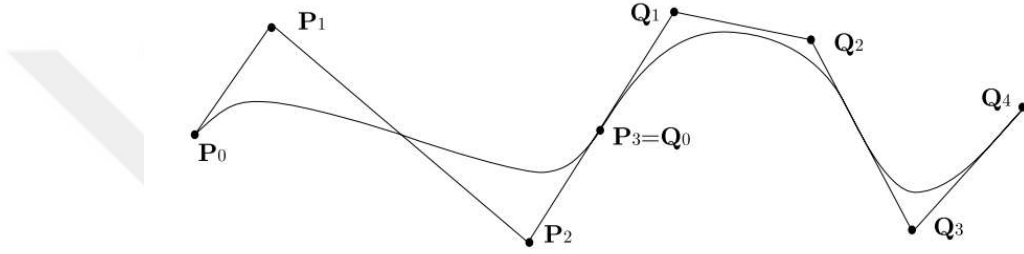
Şekil2.6: Kuadratik Düzgün B-Spline Eğrisi (Salomon, 2006)

(İçaan, 2007)' a göre B-Spline eğrileri ve Bezier Eğrileri Arasındaki İlişkiler aşağıdaki gibi özetlenebilir:

- B-spline eğrileri Bézier eğrilerinin geliştirilmesi sonucu ortaya çıkan eğrilerdir.
- B-spline eğrileri genelde Bézier eğrisi gibi tek para bir eğri değildir iki veya daha fazla eğriden oluşur.

- B-spline eğrileri en az bir ya da daha çok polinom segmentinin birleşmesinden meydana gelmektedir.
- B-spline eğrisinin tek bir parçasından oluşması durumunda bu eğri aynı zamanda Bézier eğrisidir (İçaan, 2007).

İki Bezier eğri segmentinin birleştirilmesi ile elde edilen B-spline eğrisi Şekil2.7’de verilmiştir:



**Şekil2.7:** İki Bezier eğri segmentinin birleştirilmesi ile elde edilen B-spline eğrisi (Salomon, 2006)

### 3. GEREÇ ve YÖNTEM

Bu çalışmada Bezier eğrileri ve Bezier eğrilerinin geliştirilmesi sonucu elde edilen B-spline eğrilerini kullanarak normal el ve deforme olmuş ellerin eğrileri oluşturulup karşılaştırma yapıldı. Lineer, Kuadratik ve Kübik Bezier eğrileri örnek çizimleri Geogebra programında yapıldı. C++ ve AutoCAD programları kullanılarak el modelleri elde edildi. C++ programlama dilinde Bezier eğrileri ve AutoCAD2014 programında modelleme yaparken B-spline eğrileri kullanıldı.

Uygulamamızın C++ kodları:

```
#include<opencv2/opencv.hpp>

#include<opencv2/highgui/highgui.hpp>

#include<string>

#include<iostream>

#include<algorithm>

#include<vector>

#include<math.h>

#include<fstream>

#include<windows.h>

usingnamespace std;

usingnamespace cv;

Mat BEZIERCIZ;

vector<CvPoint> BEZIERCIZBezierParametricControlpnts;

vector<CvPoint> KontrolIsaretle;

// Resim üzerinde tıkladığım noktaları tespit icin mouseclick fonksiyonum:
```

```

void on_mouseBEZIERCIZ(int ee, int x, int y, int flags, void* param)

{

    switch (ee) {

        case CV_EVENT_LBUTTONDOWN:

            CvPoint clickPoint = cvPoint(x, y);

            printf("(x,y)--> %d %d ve (R, G, B)-->", clickPoint.x,
clickPoint.y);

            CvPoint BEZIERCIZPoint = { clickPoint.x, clickPoint.y };

            BEZIERCIZBezierParametricControlpnts.push_back(BEZIERCIZPoint);

            CvPoint KontrolPoint = { clickPoint.x, clickPoint.y };

            KontrolIsaretle.push_back(KontrolPoint);

            circle(BEZIERCIZ, Point(KontrolPoint.x, KontrolPoint.y), 3,
Scalar(0, 0, 255), CV_FILLED, 8, 0);

            imshow("Kontrol_Nokta_Sec", BEZIERCIZ);

            for (int i = 0; i < BEZIERCIZ.channels(); i++)

            {

                printf("%4i ", BEZIERCIZ.at<uchar>(y,
x*BEZIERCIZ.channels() + i));

            }

            printf("\n");

        }

    }

}

```

```

// Binom katsayisini hesaplayan fonksiyonum:

double BinomKatsayisi(int n, int k) {

    double sonuc = 1;

    for (int i = 1; i <= k; i++)

    {

        sonuc *= n - (k - i);

        sonuc /= i;

    }

    return sonuc;

}

// Bernstein baz fonksiyonunu bu fonksiyonla olusturuyorum...

double Bernstein(int n, int k, double t) {

    double Binom = BinomKatsayisi(n, k);

    double BernsteinSonuc = Binom * pow(t, k) * pow(1.0 - t, n - k);

    return BernsteinSonuc;

}

// Bernstein baz fonksiyonunun string ifadesini bu fonksiyonla
olusturuyorum...

String BernsteinEquation1(int n, int k) {

    double fakt = BinomKatsayisi(n, k);

    String BernsteinSonuc = std::to_string(int(fakt)) + " * ( t^"+
std::to_string(k) + " ) * ( (1 - t)^( "+ std::to_string(n) + "- "+
std::to_string(k) + " ) )";

```

```

        return BernsteinSonuc;
    }

    //Bernstein baz fonksiyonunun ara string ifadesini bu fonksiyonla
    olusturuyorum...

    String BernsteinEquation2(int n, int k) {

        double fakt = BinomKatsayisi(n, k);

        String BernsteinSonuc = std::to_string(int(fakt)) + " * ( t^"+
std::to_string(k) + " ) * ( (1 - t)^( "+ std::to_string(n) + "-" +
std::to_string(k) + " ) ) + ";

        return BernsteinSonuc;
    }

    //Bezier Egrisinin parametrik denklemini string olarak ekrana yazmak icin bu
    fonksiyonu olusturuyorum...

    String ParametricEquation(vector<CvPoint> p)

    {

        vector<CvPoint> RR;

        String Bezier_t;

        CvPoint r = {};

        String X_t = "";

        String Y_t = "";

        int n = p.size();

        for (int i = 0; i < n; i++) {

            //n tane nokta varsa (n-1).dereceden Bernstein polinomu
            çiziyoruz...

```

```

String Berns = "";

if (i == n - 1)

    Berns =BernsteinEquation1(n - 1, i);

else Berns = BernsteinEquation2(n - 1, i);

X_t = X_t + std::to_string(p.at(i).x) + " * " + Berns; //p ler
kontrol noktaları ve r.X= x(t) yani parametrik denklemin x(t)si

Y_t = Y_t + std::to_string(p.at(i).y) + " * " + Berns; //p ler
kontrol noktaları ve r.Y= y(t) yani parametrik denklemin y(t)si
}

Bezier_t = "{" + X_t + ";" + Y_t + "}";

return Bezier_t; //r:=(x(t), y(t)), Düzlemsel egrinin parametrik
denklemini....
}

//Bezier Egrisinin parametrik denklemini bu fonksiyonla olusturuyorum...

CvPointParametricBezier(vector<CvPoint>p, doublet)

{

    CvPoint r = {};

    double X_t = 0;

    double Y_t = 0;

    int n = p.size();

    for (int i = 0; i < n; i++) {

        //n tane nokta varsa (n-1).dereceden Bernstein polinomu
        çiziyoruz...
    }
}

```

```

        double Berns = Bernstein(n - 1, i, t);

        X_t = X_t + Berns * p.at(i).x; //p ler kontrol noktaları ve r.X=
x(t), yani parametrik denklemin x(t)si

        Y_t = Y_t + Berns * p.at(i).y; //p ler kontrol noktaları ve r.Y=
y(t), yani parametrik denklem y(t)si

    }

    r.x = (int)X_t;

    r.y = (int)Y_t;

    return r; //r:=(x(t), y(t)), Düzlemsel egrinin parametrik denklemini....
}

// Parametrik denklemini x(t) ve y(t) olarak ayrı ayrı ekrana yazdırmak için
kullandığım stringi parse etme fonksiyonum:

vector<string> ARA_BULS(StringAranacak, std::stringayrac) {

    size_t pos = 0;

    std::string token;

    vector<string> DualEquation;

    while ((pos = Aranacak.find(ayrac)) != std::string::npos) {

        token=Aranacak.substr(0, pos);

        DualEquation.push_back(token);

        Aranacak.erase(0, pos + ayrac.length());

    }

    DualEquation.push_back(Aranacak);

    return DualEquation;
}

```

```

}

int main() {

    char working_directory[MAX_PATH + 1];

    GetCurrentDirectoryA(sizeof(working_directory), working_directory);

    Stringpath(string(working_directory) + "\\Resimler\\Normal.jpg");

    BEZIERCIZ = imread(path.c_str(), IMREAD_COLOR);

    cout<<"Secilen Kontrol Noktalarının Koordinatları ve Renk Bilgileri"<<
    std::endl;

    cv::resizeWindow("Kontrol_Nokta_Sec", 644, 529);

    cvNamedWindow("Kontrol_Nokta_Sec", 1);

    cvSetMouseCallback("Kontrol_Nokta_Sec", on_mouseBEZIERCIZ, 0);

    imshow("Kontrol_Nokta_Sec", BEZIERCIZ);

    cvWaitKey(0);

    MatWhiteimage(BEZIERCIZ.rows, BEZIERCIZ.cols, CV_8UC3, Scalar(255, 255,
255));

    cout<< std::endl <<"BEZIER Eğrisi İçin Secilen Kontrol Noktaları"<<
    std::endl;

    //BEZIER eğrisi için kontrol noktaları vektörü:

    vector<CvPoint> BEZIERCIZpoints;

    //Kontrol noktalarını ekrana yazdırıyorum...

    for (int i = 0; i < BEZIERCIZBezierParametricControlpnts.size(); i++)

    {

        cout<<"( "+
        std::to_string(BEZIERCIZBezierParametricControlpnts.at(i).x) + " ,

```

```

"+std::to_string(BEZIERCIZBezierParametricControlpnts.at(i).y) +" )"<<
std::endl;

    }

    cout<< std::endl <<"Bezier Egrisinin Parametrik Denklemini";

    String BEZIERCIZ_Bezier =
ParametricEquation(BEZIERCIZBezierParametricControlpnts);

    std::string delimiter = ";";

    vector<string> Koordinat = ARA_BULS(BEZIERCIZ_Bezier, delimiter);
    delimiter="{";

    vector<string> Koordinat_x = ARA_BULS(Koordinat.at(0), delimiter);
    delimiter="}";

    vector<string> Koordinat_y = ARA_BULS(Koordinat.at(1), delimiter);

    std::string x_T = Koordinat_x.at(1);

    std::string y_T = Koordinat_y.at(0);

    //Bezier egrisinin parametrik denklemini ekrana yazdırıyorum...

    cout << std::endl;

    cout <<"x(t)= " + x_T << std::endl<< std::endl;

    cout <<"y(t)= " + y_T << std::endl<< std::endl;

    ofstream
fout(string(working_directory)+"\\Parametrik_Sonuc\\Parametrik_Sonuc.txt")
;

    //Bezier egrisinin parametrik denklemini dosyaya yazdırıyorum...

    fout<<"x(t)= " + x_T << std::endl<< std::endl;

```

```

fout <<"y(t)= " + y_T << std::endl<< std::endl;

fout.close();

//Bezier egrisini tıklanan resmin üzerine çizdiriyorum...

for (double t = 0; t <= 1; t += 0.001)
{
    CvPoint PPP =
ParametricBezier(BEZIERCIZBezierParametricControlpnts, t);
    BEZIERCIZpoints.push_back(PPP);

    cv::rectangle(BEZIERCIZ, cv::Rect(PPP.x, PPP.y, 1, 1), CV_RGB(0,
255, 0), 2, 8, 0);

}

cvNamedWindow("Bezier Egrisi Sonuc", 1);

cvSetMouseCallback("Bezier Egrisi Sonuc", on_mouseBEZIERCIZ, 0);

imshow("Bezier Egrisi Sonuc", BEZIERCIZ);

//Bezier egrisi cizilen resmi kaydediyorum...

imwrite(string(working_directory)+"\\Bezier_Sonuc\\BEZIERCIZ_SONUC.j
pg", BEZIERCIZ);

cvWaitKey(0);

int bekle;

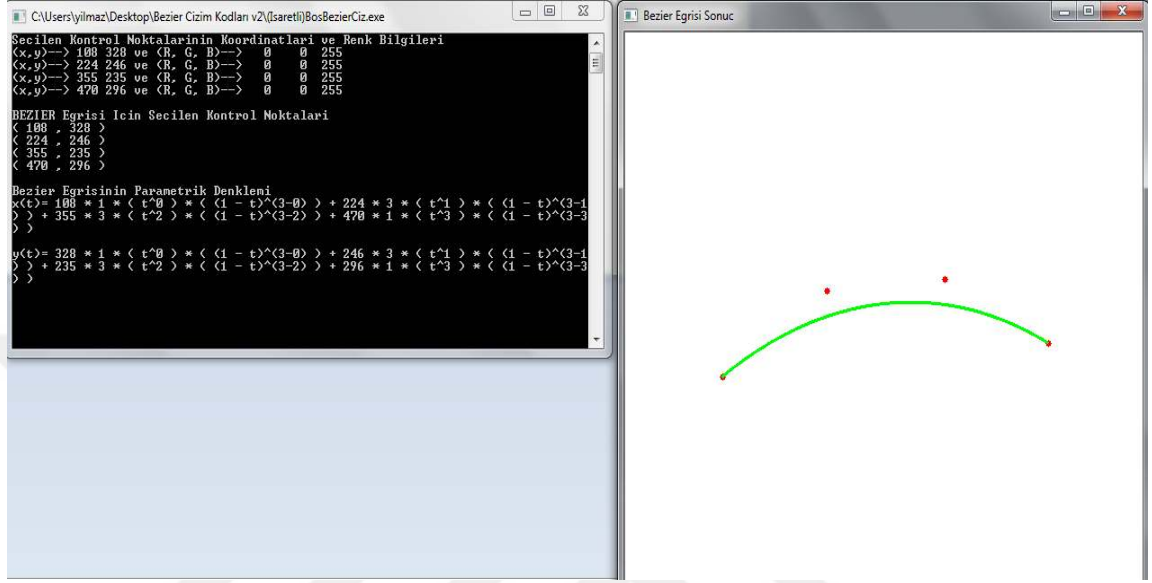
cin>> bekle;

return 0;}

```

Yukarıda verilen C++ kodları ile elde edilen dört kontrol noktasına sahip Bezier eğrisi ve çizim metodu Şekil3.1’de verilmiştir. Uygulamamız içerisinde kullanılan bu metot, sol üst köşeyi (0,0) başlangıç noktası kabul eden koordinat sistemini kullanmaktadır.

Buradaki kontrol noktaları uygulamamız içerisinde Mouse yardımıyla tıklanarak elde edilmiştir.



**Şekil3.1:** Dört kontrol noktası olan bir Bezier eğrisi uygulamasının ekran görüntüsü

Şekil 3.1’deki Bezier eğrisinin elde edilebilmesi için tıklanan kontrol noktalarının koordinatları,(108, 328), (224, 246), (355, 235), (470, 296) dır. Ayrıca bu Bezier eğrisinin parametrik denklemleri aşağıdaki gibidir:

$$x(t) = 470t^3 + 1065(1-t)t^2 + 672(1-t)^2t + 108(1-t)^3$$

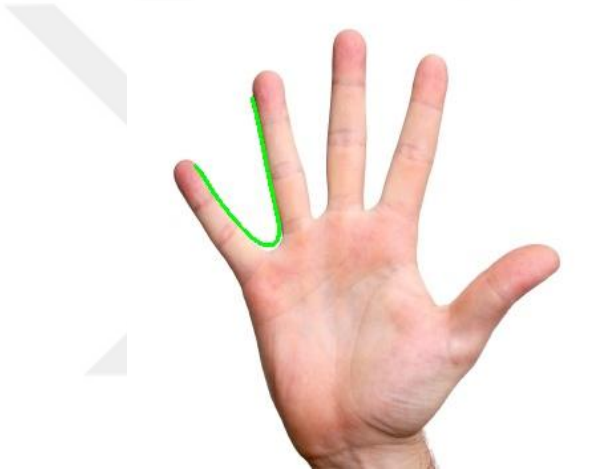
$$y(t) = 296t^3 + 705(1-t)t^2 + 738(1-t)^2t + 328(1-t)^3$$

Şekil8’deki gibi Bezier eğri modelleri bize normal el ve deforme el arasındaki farklılıkların tespiti için bir yöntem sağlayacaktır.

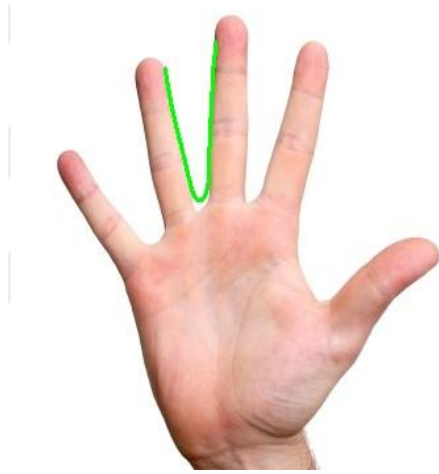
#### 4. BULGULAR

Bu bölümde önceki bölümlerde bahsettiğimiz Bezier eğrileri ve B-spline eğrileri kullanılarak normal el ve deforme el için oluşturulmuş eğri modelleri yer almaktadır. Oluşturulan bu eğriler birbirleri ile karşılaştırılıp deforme el ile normal el arasındaki farklar verilmiştir.

Bezier eğrileri kullanılarak C++ programında oluşturulmuş parmak arası eğri modelleri;



Şekil4.1: Soldan 1. Bezier eğrisi



Şekil4.2: Soldan 2. Bezier eğrisi

(1-<http://www.graciebarra.com/uk/2014/03/tips-bjj/five-hand-image/>)

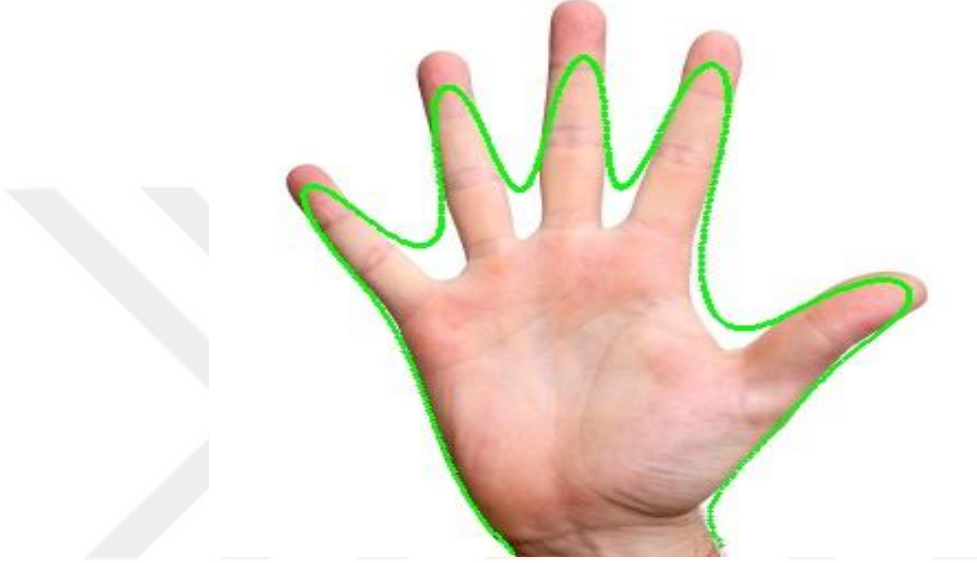


Şekil4.3: Soldan 3. Bezier Eğrisi



Şekil4.4: Soldan 4. Bezier Eğrisi

Şekil4.1, Şekil4.2, Şekil4.3, Şekil4.4'deki eğrileri elde ederken her bir parmak arası için seçtiğimiz ayrı ayrı kontrol noktaları ile Beizer eğrilerini elde ettik. Örneğin Şekil4.1'daki eğri için 50, Şekil4.2'daki eğri için 53, Şekil4.3'deki eğri için 51 ve Şekil4.4'deki eğri için ise 47 tane kontrol noktası seçilmiştir.

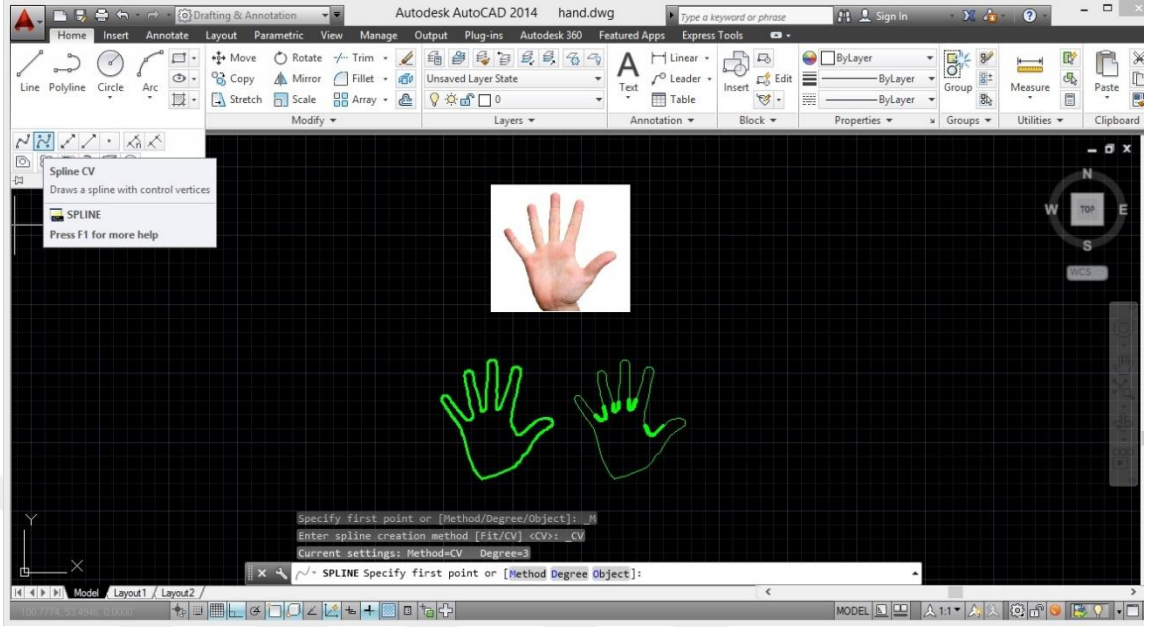


**Şekil4.5:** Tüm elin Bezier modeli

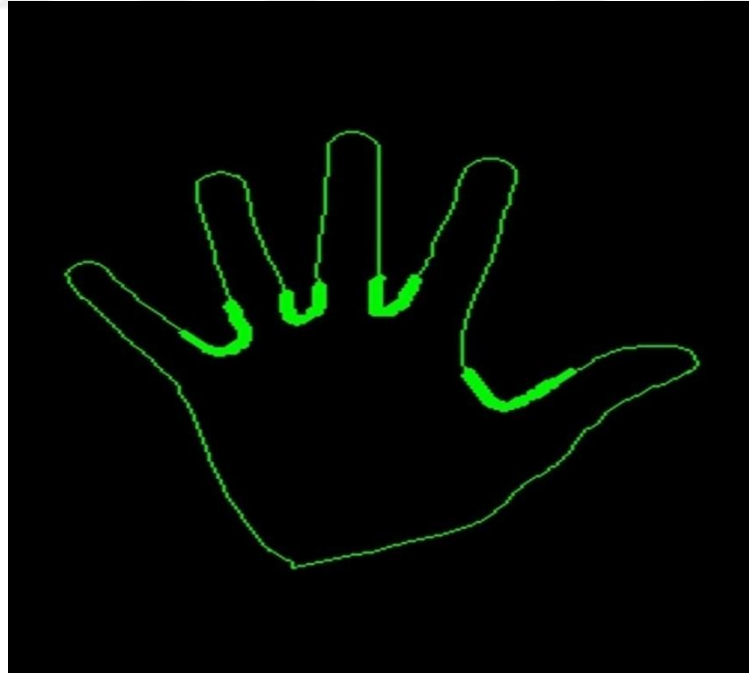
Şekil4.5'de görülen el modelinde 361 tane kontrol noktası alınarak eğri oluşturulmuştur. Burada eğrinin derecesi 360 olan bir polinom eğrisi elde edilmiştir. Kontrol noktası arttıkça Bezier eğrisinin eli tam olarak modelleyemediği görülmektedir. Çünkü Bezier eğrilerinde kontrol noktaları ne kadar fazla seçilirse eğrinin eğriliği o kadar bozulur. Bu yüzden Bezier eğrilerinin geliştirilmesi sonucu elde edilen B-spline eğrilerine ihtiyaç duyulmuştur.

#### **4.1. AutoCAD'de çizilen el modelleri**

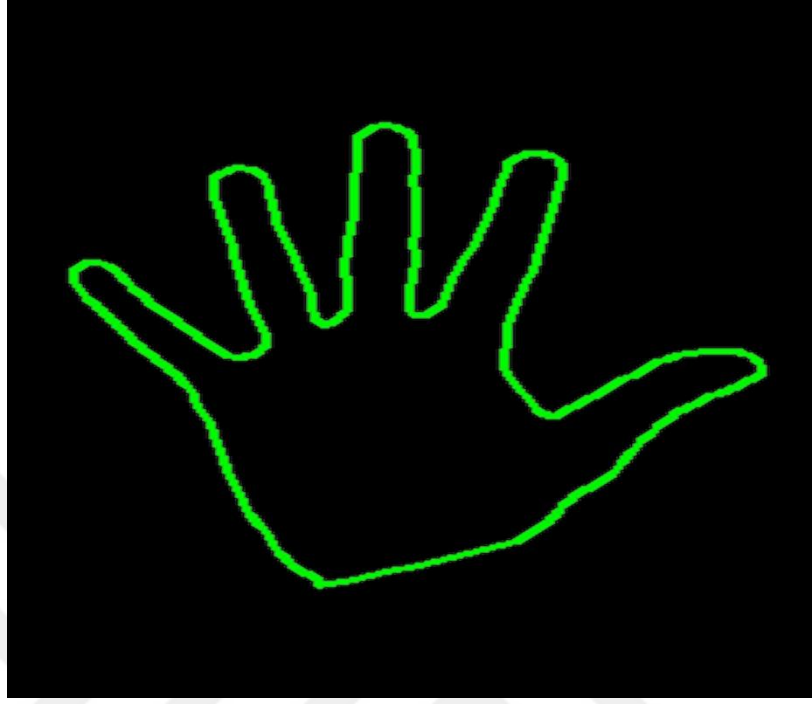
Bu bölümde, bilgisayar destekli tasarım programı olan AutoCAD programından yararlanılmıştır. Şekil4.6'te AutoCAD programında çizilmiş parmak arası eğri modeli ve el için oluşturulan eğri modeli çizimi sırasındaki ekran görüntüsü yer almaktadır.



Şekil4.6: AutoCAD programında çizilmiş el eğri modelinin ekran görüntüsü



Şekil4.7: AutoCAD programında çizilmiş parmak arası eğri modeli



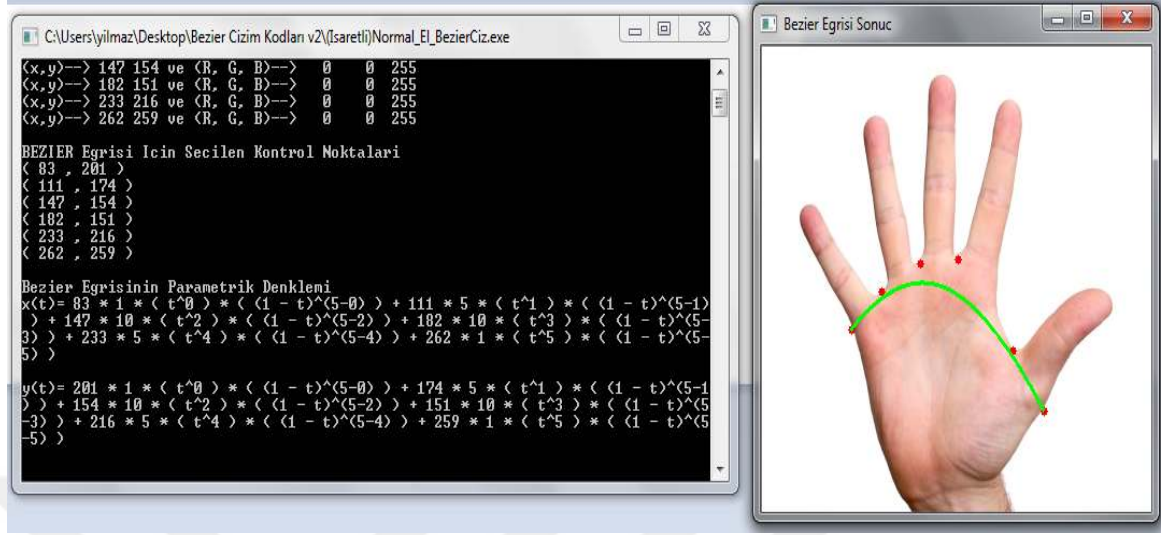
**Şekil4.8:** AutoCAD programında çizilmiş el modeli

Bezier eğrilerinde olduğu gibi B-spline eğrileri de kontrol noktaları ve polinomlar ile inşa edilen eğrilerdir. Bu B-spline eğrilerini kullanan Bilgisayar Destekli Tasarım yazılımlarından biri olan AutoCAD programı yardımıyla daha uygun el modeli Şekil4.8’de görülebilir. Şekil4.5 ve Şekil4.8 karşılaştırıldığında B-spline eğrileri ile oluşturulan eğrilerin Bezier eğrilerine göre daha iyi bir modelleme yaptığı görülmektedir.

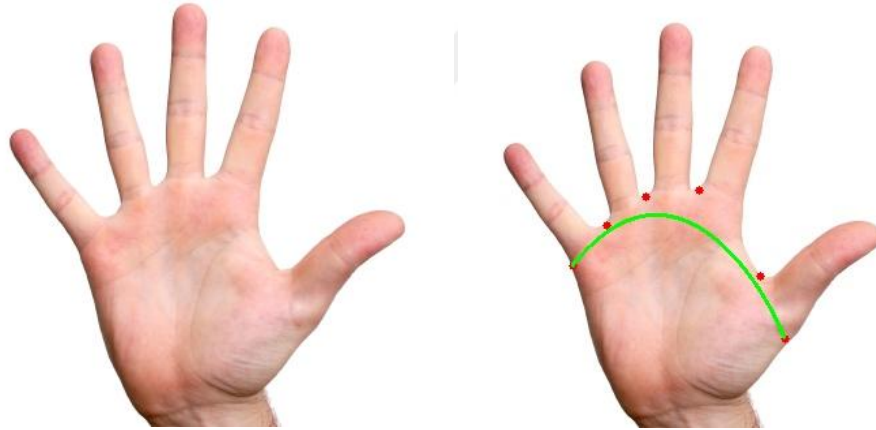
#### **4.2.Normal El Modeli**

Normal bir elde parmak ara kesitlerinin avuç içinde oluşturulan Bezier eğrisi üzerinde olduğu görülür. Fakat deformasyona uğramış bir elde eğri bozulur ve ara kesitler eğrinin üzerinde yer almaz. Bu eğride kontrol noktaları elin parmak aralarındaki orta noktalardan seçilen noktalar olmak üzere normal bir elde ve deformasyona uğramış bir el için çizilmiş Bezier eğrileri aşağıdaki şekillerde verilmiştir.

C++ programında çizilmiş normal el modelinin ekran görüntüsü Şekil4.9’da verilmiştir.



Şekil4.9: C++ programında çizilmiş normal el modelinin ekran görüntüsü



Şekil4.10: Normal el için Bezier Eğrisi

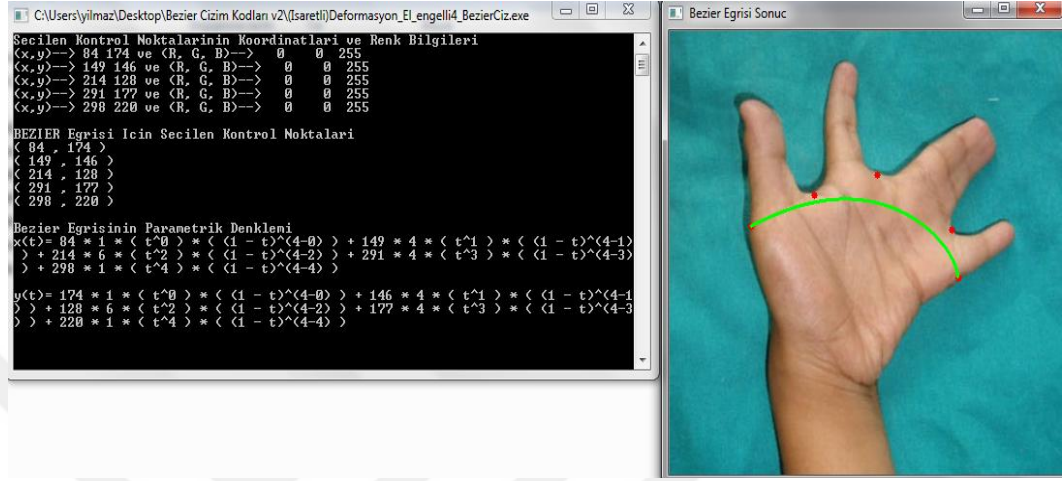
Şekil4.10'da normal el için kontrol noktaları seçilirken uygulanan teknik elin parmak aralarındaki orta noktalardan seçilen noktalar ile belirlenmiştir. Bu nedenle normal el için 6 kontrol noktası seçilmiştir. Bezier eğrisinin denklemi 5. dereceden bir polinom ile temsil edildiği görülmektedir.

Bu noktalar ile elde edilen Bezier eğrisi için parametrik denklemler;

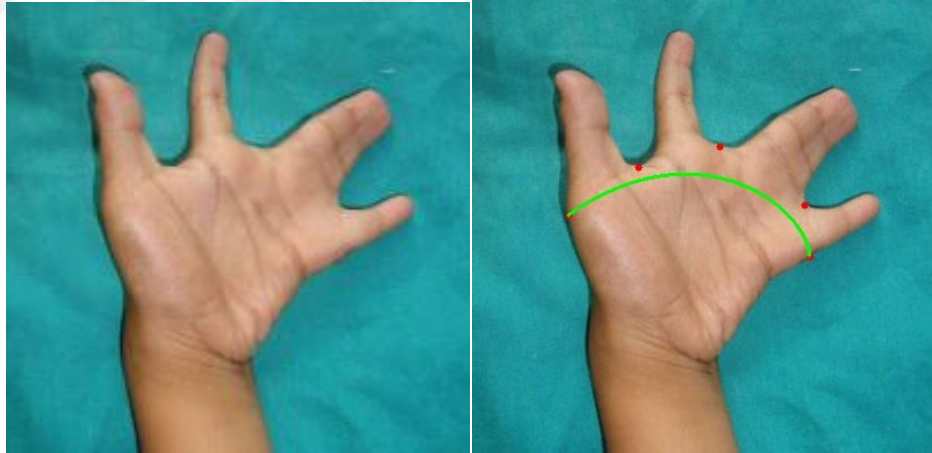
$$\begin{aligned}
 x(t) &= 247t^5 + 1110(1-t)t^4 + 1800(1-t)^2t^3 + 1420(1-t)^3t^2 + 550(1-t)^4t + 86(1-t)^5 \\
 y(t) &= 272t^5 + 990(1-t)t^4 + 1490(1-t)^2t^3 + 1550(1-t)^3t^2 + 875(1-t)^4t + 201(1-t)^5
 \end{aligned}$$

### 4.3.Birinci Deforme El Modeli

C++ programında çizilmiş normal el modelinin ekran görüntüsü;



Şekil4.11: C++ programında çizilmiş birinci deforme el modelinin ekran görüntüsü



Şekil4.12: Birinci deforme el için Bezier Eğrisi

(2-[http://www.plastangel.com/p/hand-surgery\\_17.html](http://www.plastangel.com/p/hand-surgery_17.html))

Benzer şekilde bir deforme el için kontrol noktaları seçilirken uygulanan teknik elin parmak aralarındaki orta noktalardan seçilen noktalar ile belirlendi. Bu nedenle Şekil4.12’dekideforme el için 5 kontrol noktası seçildi.Bu noktalar ile elde edilen Bezier eğrisi için parametrik denklemler;

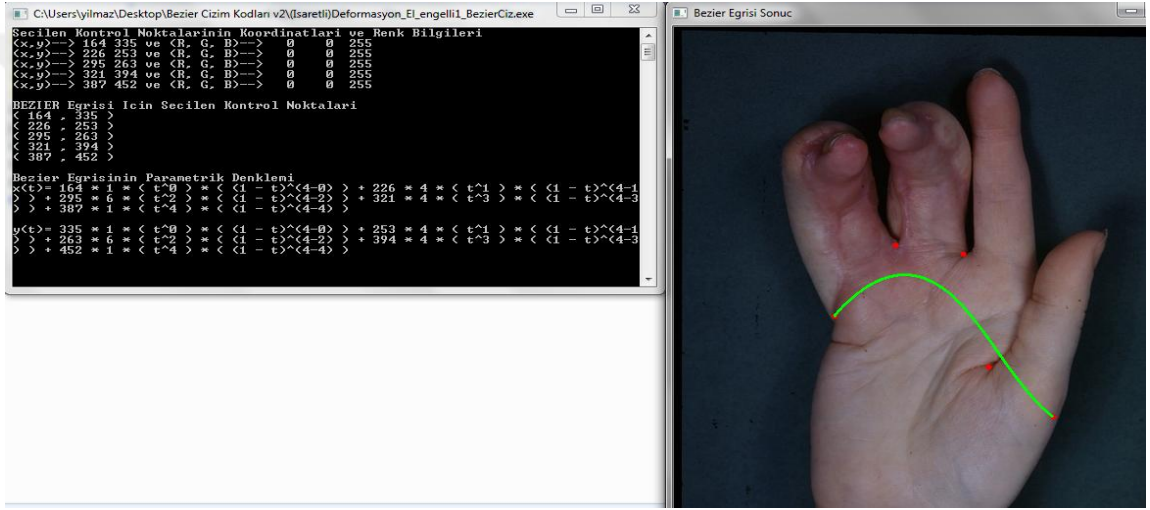
$$x(t) = 297t^4 + 1172(1-t)t^3 + 1302(1-t)^2t^2 + 588(1-t)^3t + 82(1-t)^4$$

$$y(t) = 227t^4 + 704(1-t)t^3 + 750(1-t)^2t^2 + 576(1-t)^3t + 180(1-t)^4$$

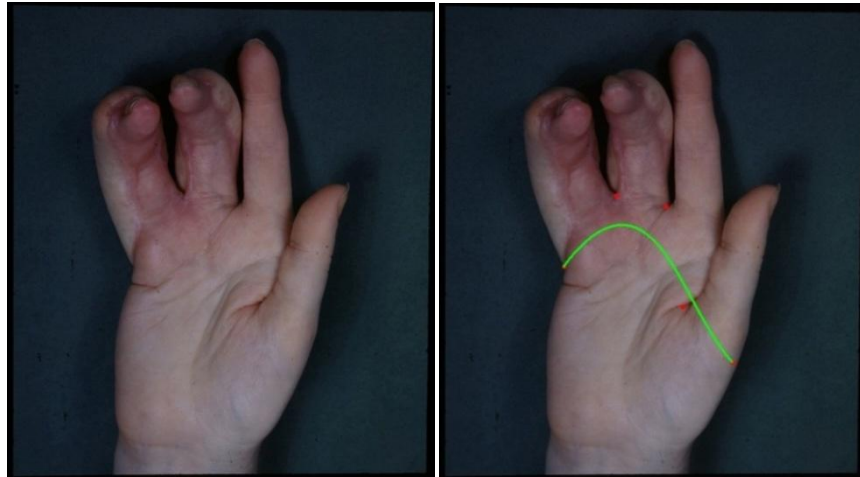
Bezier eğrisinin denklemi 4. dereceden bir polinom ile temsil edildiği görülmektedir.

#### 4.4.İkinci Deforme El Modeli

C++ programında çizilmiş normal el modelinin ekran görüntüsü;



Şekil4.13: C++ programında çizilmiş ikinci deforme el modelinin ekran görüntüsü



Şekil4.14:İkinci deforme el için Bezier Eğrisi

(3-[https://commons.wikimedia.org/wiki/File:Deformity\\_of\\_Hand\\_due\\_to\\_X-ray\\_burns.jpg](https://commons.wikimedia.org/wiki/File:Deformity_of_Hand_due_to_X-ray_burns.jpg))

Şekil4.14’de verilen bir deforme el için kontrol noktaları seçilirken uygulanan teknik elin parmak aralarındaki orta noktalardan seçilen noktalar ile belirlendi. Bu nedenle deforme el için 5 kontrol noktası seçildi.

Bu noktalar ile elde edilen Bezier eğrisi için parametrik denklemler;

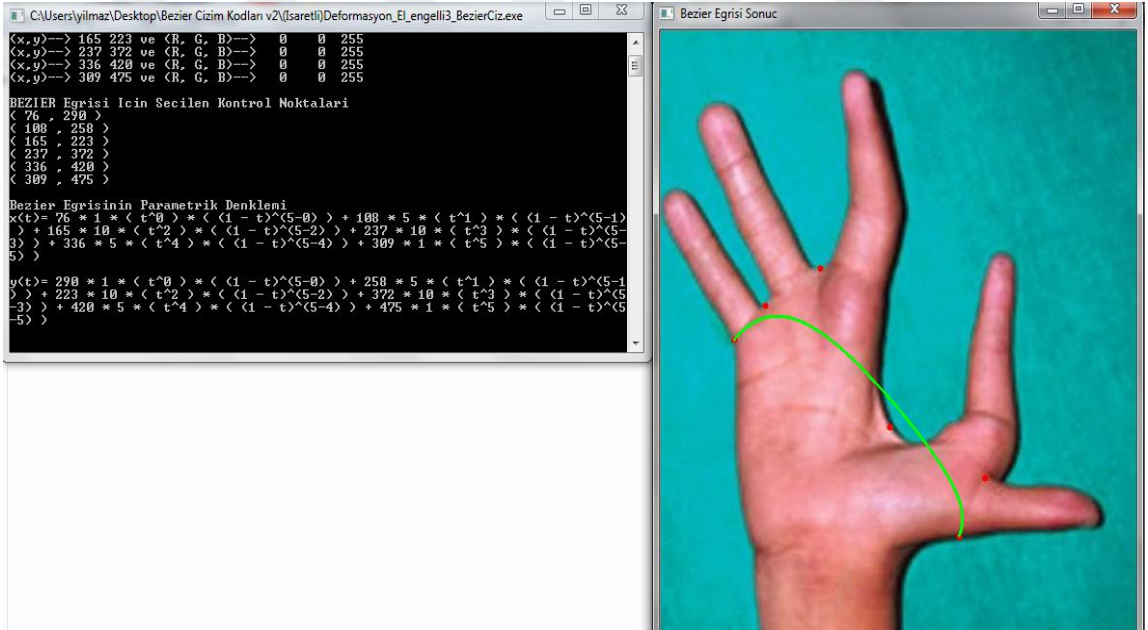
$$x(t) = 372t^4 + 1324(1-t)t^3 + 1788(1-t)^2t^2 + 920(1-t)^3t + 164(1-t)^4$$

$$y(t) = 498t^4 + 1548(1-t)t^3 + 1584(1-t)^2t^2 + 1004(1-t)^3t + 349(1-t)^4$$

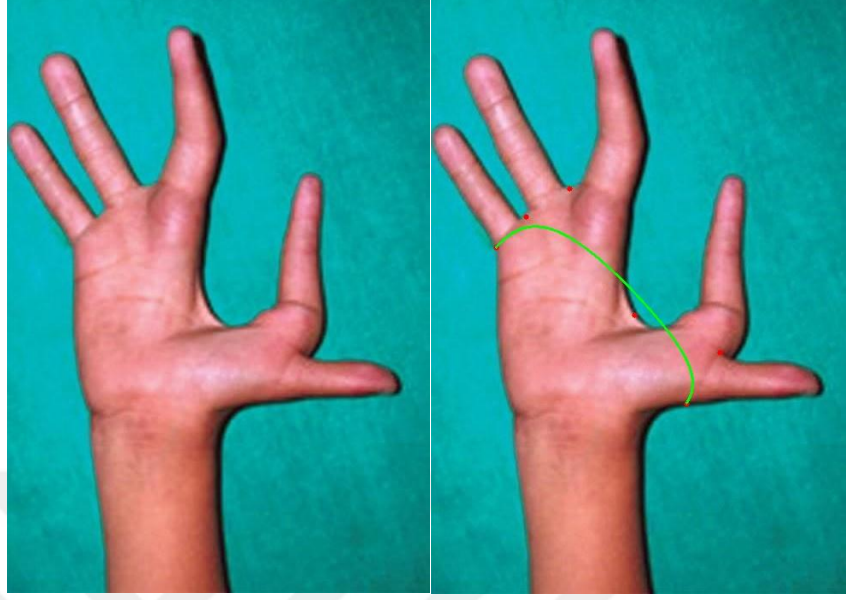
Deforme bir el üzerinde elde edilen Bezier eğrisinin denklemi 4. dereceden bir polinom ile temsil edildiği görülmektedir.

#### 4.5.Üçüncü Deforme El Modeli

C++ programında çizilmiş normal el modelinin ekran görüntüsü;



Şekil4.15: C++ programında çizilmiş üçüncü deforme el modelinin ekran görüntüsü



**Şekil4.16:**Üçüncü deforme el için Bezier Eğrisi

(4-<https://www.pinterest.com/pin/352758583284443385/>)

Şekil4.16’da verilen bir deforme el için kontrol noktaları seçilirken uygulanan teknik elin parmak aralarındaki orta noktalardan seçilen noktalar ile belirlendi. Bu nedenle deforme el için 6 kontrol noktası seçildi.

Bu noktalar ile elde edilen Bezier eğrisi için parametrik denklemler;

$$x(t) = 314t^5 + 1740(1-t)t^4 + 2400(1-t)^2t^3 + 1610(1-t)^3t^2 + 550(1-t)^4t + 76(1-t)^5$$

$$y(t) = 479t^5 + 2115(1-t)t^4 + 3750(1-t)^2t^3 + 2220(1-t)^3t^2 + 1300(1-t)^4t + 295(1-t)^5$$

Deforme bir el üzerinde elde edilen Bezier eğrisinin denklemi 5. dereceden bir polinom ile temsil edildiği görülmektedir.

Yukarıda ayrıntılı şekilde incelenen normal el ve deformasyona uğramış el için oluşturulan eğrilerin ve bunların parametrik denklemleri incelendiğinde şu sonuçlara ulaşıldı.

- Bulunan Bezier eğrilerinin normal el için elde edilen eğri modelinden farklı olduğu açıkça görülmektedir.
- Normal eldeki Bezier eğrisinin derecesi 5 olduğu tespit edildi.
- Deformasyona uğramış el için derece sayının parmak sayısı ile doğru orantılı olarak artıp azaldığı tespit edildi.

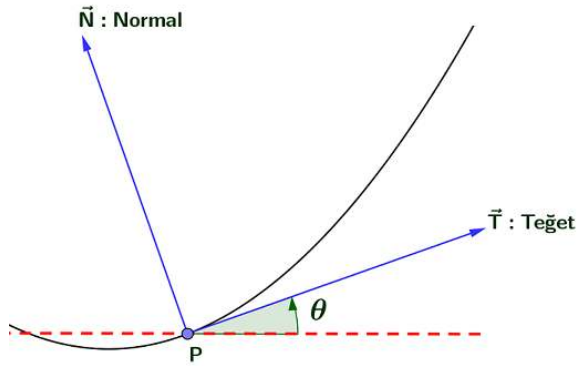
Şimdi de normal ve deformasyona uğramış eller için çizilen Bezier eğrilerini birbirinden ayırt etmek için kullanılabilecek özellik olan düzlemsel bir eğrinin eğriliği kısaca bahsedelim.

#### 4.6. Bir Eğrinin Eğriliği

Düzlemsel bir  $B(t) = (x(t), y(t))$  eğrisinin eğriliği,

$$\kappa = \frac{|x' y'' - x'' y'|}{\left((x')^2 + (y')^2\right)^{3/2}}$$

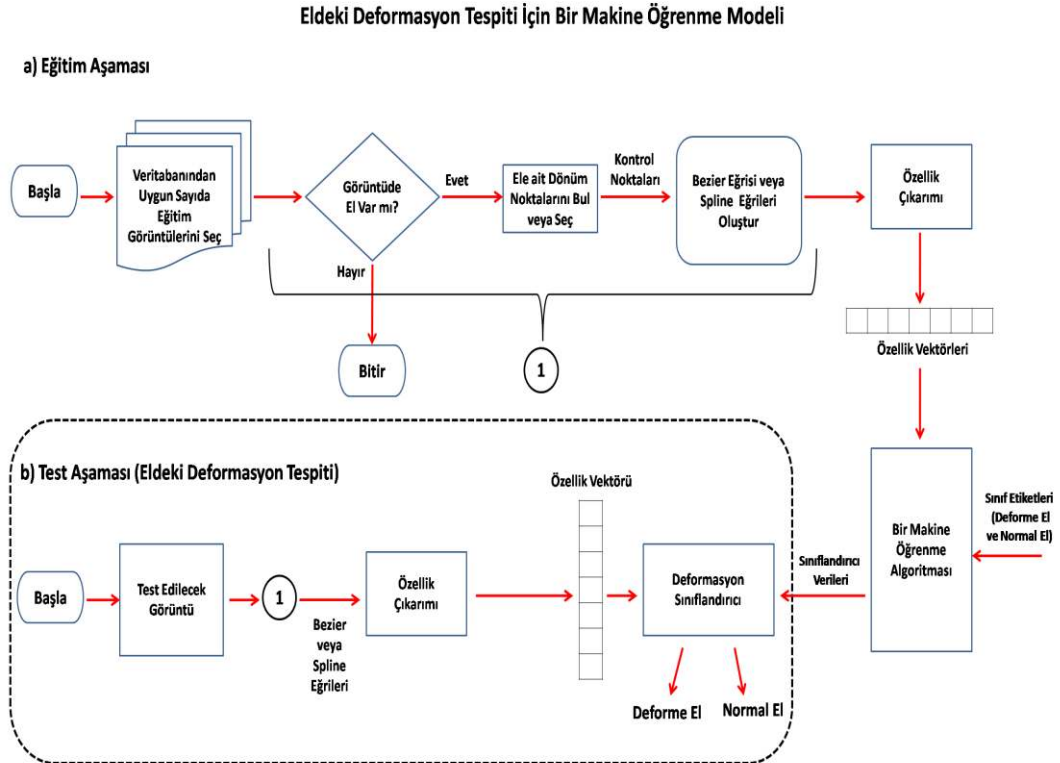
ile hesaplanır. Burada  $x' = \frac{dx}{dt}$  ve  $y' = \frac{dy}{dt}$  dir (Stoker, 1989).  $\kappa$ , eğrinin doğrudan ne kadar saptığını ölçer yani  $\kappa$ 'nın büyüklüğü, eğrinin bir noktadaki teğetinden ne kadar ayrıldığıнын ölçüsüdür. Şekil4.17'de gösterilen eğrinin eğriliği, eğri üzerindeki herhangi bir  $P$  noktasındaki teğet ile  $ox$ -ekseni arasındaki açı  $\theta$  açısı olmak üzere, birim yay boyunca  $\theta$  açısının değişiminin mutlak değeridir (Stoker, 1989).



Şekil4.17: Bir Eğrinin Eğriliği

Bu eğriden elde edilebilecek nümerik değerler ile ileride el deformasyonunu tespit edebilecek makine öğrenme programlarında özellik vektörleri oluşturulabilmesinde yardımcı olacaktır. Örneğin aşağıdaki akış diyagramı bir makine öğrenme modelinde taban oluşturabilir. Yukarıda verilen eğriliği kullanacak Şekil4.10 ve Şekil4.14'te incelenen normal el ve deformasyona uğramış el için oluşturulan eğriler incelendiğinde deformasyona uğramış elin eğriliğinin arttığı görüldü. Çünkü eğrinin kollarının birbirine yaklaştığı gözlemlendi. Normal el için elde edilen eğride ise eğrinin eğriliği azalır.

Algoritma matematik ve bilgisayar programlamada çok önemli bir yer tutmaktadır. Bir algoritma işlemleri kendi kendine yeten ve yapılacak işlemleri adım adım yaptıran ifadelerdir (wikipedia). Algoritmalar hesaplama işlemlerinde veri işlemlerinde programlama işlemlerinde ve problem çözme tekniklerinde kullanılırlar. Algoritmalar akış diyagramları ile de temsil edilebilirler. Bizim bu çalışmada ise Şekil4.18'deki gibi bir makine öğrenme algoritmasını temsil eden bir akış diyagramı elde edildi.



**Şekil4.18:** Deforme el tespiti için bir makine öğrenme modeli

## 5. TARTIŞMA

Literatür incelendiğinde Bezier eğrilerinin pek çok alanda kullanıldığı görüldü. Bu alanların başında matematik, endüstriyel tasarım, otomotiv, el yazısı tanıma, mimarlık ve mühendislik alanlarında bilgisayar grafikleri ile özellikle tasarım geliştirmede kullanıldığı gözlemlendi. Son yıllarda ise bu eğrilerin tıp, diş hekimliği gibi alanlarda da kullanılmış olduğu görüldü.

Bu çalışmada yapmış olduğumuz deformasyona uğramış bir el modeli tespiti için Bezier eğrilerinin daha önceki çalışmalarda şu ana kadar kullanılmadığı fark edildi.

Bu tezde, deformasyona uğramış bir el modeli tespitini yapan makine öğrenme algoritmalarında kullanılması için el üzerinde oluşturulan Bezier eğrilerinden çıkarılabilecek özellikler verildi. Bu özellikler normal bir elde çizilen Bezier eğrisinin parametrik denkleminin derecesinin beş olduğu tespit edildi ve eğrinin eğriliğine bakıldı. Fakat deforme bir el üzerinde çizilen Bezier eğrisinin parametrik denkleminin derecesinin değiştiği ve eğrinin eğriliğinin arttığı ya da azaldığı tespit edildi.

Çalışmamızda elde edilen sonuçların, daha önceki araştırmalarla karşılaştırıldığında farklı olan yanı deformasyon tespiti için Bezier eğrilerinin kullanılması, bu eğrilerin parametrik denklemleri, denklemlerin derecesi ve eğrinin eğriliğinin makine öğrenme ve bunun gibi başka algoritmalarda kullanılmasının önerilmesidir.

## 6. SONUÇ ve ÖNERİLER

Yapmış olduğumuz bu çalışmada Bezier eğrileri ve B-spline eğrileri ile elde ettiğimiz normal el ve deforme olmuş eller üzerinde çizdiğimiz eğriler birbirleri ile karşılaştırıldı.

Bu çalışmadan çıkan sonuçlar;

- Bernstein polinomları ile Binom dağılımı ve Poisson dağılımı arasındaki ilişkiler incelendi.
- Bernstein polinomları yardımıyla Bezier eğrilerinin tanımları verildi.
- Lineer Bezier eğrisi, Kuadratik Bezier eğrisi ve Kübik Bezier eğrilerinin parametrik denklemleri verildi ve grafikleri çizildi.
- B-spline eğrileri ile Bezier eğrileri arasındaki ilişkiler incelendi.
- C++ programında yazılan kodlar ve AutoCAD programı yardımıyla çizilen eğirlerin ekran görüntüleri verildi.
- Normal bir el için Bezier eğrileri oluşturuldu ve parametrik denklemleri ve dereceleri tanımladı.
- Deforme olmuş eller için Bezier eğrileri oluşturuldu ve parametrik denklemleri ve dereceleri tanımladı.
- Normal el ve deforme el modellerindeki eğrilerin birbirlerinden farklı olduğu tespit edildi.
- Normal eldeki Bezier eğrisinin derecesinin her zaman beş olduğu tespit edildi.
- Deformasyona uğramış el için derece sayının parmak sayısı ile doğru orantılı olarak artıp azaldığı tespit edildi.
- Normal el ve deforme eller için oluşturulan eğrilerin eğriliğine bakıldı ve deforme olan ellerin eğriliğinin eldeki deformasyona göre değiştiği tespit edildi.
- Son olarak deforme el modellemesi için yapılabilecek çalışmalara kaynak oluşturacak bir makine öğrenme algoritması oluşturuldu.

Bu tez çalışmasının sonuçlarına ek olarak Bezier eğrilerinin ve B-spline eğrilerinin yetersiz kaldığı matematiksel modeller olabileceği tespit edilmiştir. Bu nedenden dolayı

bu iki eğrinin daha genelleşmiş hali olan NURBS eğrilerine ihtiyaç olabilir. NURBS eğrileri kullanılarak insan uzuvlarında daha iyi modellemeler yapılmasını sağlayabilir.

Bu tezde elde edilen eğriler 3D modeller yapılmasına imkan sağlayabilirler. Aynı zamanda deformasyona uğramış elde yapılabilecek protez gibi biyomedikal tasarımlar kullanılabileceği düşünülmektedir.



## KAYNAKLAR

Atılım Çetin and Uğur Gdkbay. 3 Boyutlu Bilgisayar Grafikleri (3D Computer Graphics) (in Turkish). In Trkiye Biliřim Ansiklopedisi, pp. 211–220, Papatya Yayıncılık, 2006.

Bernstein S.N, D’emonstration du th’eor’eme de Weierstrass fond’ee sur la calcul des probabilit’es, Commun. Kharkov Math. Soc. 13 (1912), 1-2 .

Donald Hearn ve Pauline Baker. Computer Graphics. Prentice-Hall. Second Edition, C Version, 1997.

Farin G., Curves and Surfaces for Computer Aided Geometric Design, 3rd Ed., Academic Press, Boston, 1993.

Farouki, R. T. and Goodman, T. N. T. On the optimal stability of the Bernstein basis, Math. Comput. 1996; 65,15531566.

Farouki R. T.,The Bernstein polynomials basis: a centennial retrospective, Comput. Aided Geom. Des. 29 (2012), 379-419.

Goldman, Identities for the Univariate and Bivariate Bernstein Basis Functions, Graphics Gems V, (edited by Alan Paeth, Academic Press,). 1995; 149-162.

Goldman R, Pyramid Algorithms: A Dynamic Programming Approach to Curves and Surfaces for Geometric Modeling, Morgan Kaufmann Publishers, R. Academic Press, San Diego, (2002).

Goldman R, An Integrated Introduction to Computer Graphics and Geometric Modeling, CRC Press, Taylor and Francis, New York, (2009)

Gneř T., Polat, E., Yz İfade Analizinde znitelik Seęimi Ve Çoklu Svm Sınıflandırıcılarına Etkisi, Gazi niv. Mh. Mim. Fak. Der. 2009; Cilt 24, No: 1, 714

Hearn, D., Baker, M.P., “Computer Graphics C Version”, Prentice-Hall, 1997

Hossain MA, Hirokawa S, Ariyoshi S, Proposal to Improve Kinematic Estimation Algorithm of Knee Prosthesis, IFMBE Proceedings Vol. 14/4

İçaçan M.S. Maya’da İnsan Yüzünün Modellenmesi Ve Animasyonu. K.T.Ü. Fen Bilimleri Enstitüsü. Yüksek Lisans Tezi, 2007, Trabzon (Danışman: Öğr.Gör. Ömer ÇAKIR).

Jetter, K. and Stöckler, J. An identity for multivariate Bernstein polynomials, Comput. Aided Geom. Design. 2003; 20, 563-577.

Khan M. İ, Bhuiyan A, Facial Features Approximation for Expression Detection in Human-Robot Interface, Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. 2010

Khan M.İ, Bhuiyan A, Facial Expression Recognition for Human-Robot Interface, IJCSNS International Journal of Computer Science and Network Security. 2009; VOL.9 No.42009.

Li Z, Ma L, Tan W. Three-dimensional object reconstruction from contour lines. Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications; Hong Kong, China. 1128977: ACM; 2006. p. 319-22.

Lorentz, G. G. Bernstein Polynomials, Chelsea Pub. Comp. New York, N. Y, (1986)

Özmen G., Kandemir R., “Haar Dalgacıkları ve Kübik Bezier Eğrileri İle Yüz İfadesi Tespiti” ELECO '2012 Elektrik - Elektronik ve Bilgisayar Mühendisliği Sempozyumu, 29 Kasım - 01 Aralık 2012, Bursa

Park H, Kim K. Smooth surface approximation to serial cross-sections. Comput. Aided Geom. Des.28(12) (1996), 995-1005.

Salomon D, Curves and Surfaces for Computer Graphics. Springer, New York; 2006.

Simsek Y., Generating functions for the Bernstein type polynomials: a new approach to deriving identities and applications for the polynomials, hacettepe journal of mathematics and statistics. 2014; 43 (1), 1 -14.

Stoker J.J. Differential Geometry. Wiley-Interscience Wiley Classics Edition. 1989; 432.

Taner T, Ciğer S, El H, Germeç D, Es A, Evaluation of dental arch width and form changes after orthodontic treatment and retention with a new computerized method. 2004; Volume 126.

([https://en.wikipedia.org/wiki/B%C3%A9zier\\_curve](https://en.wikipedia.org/wiki/B%C3%A9zier_curve), Erişim Tarihi: 17 Ekim 2016)

(<http://bilgisayarkavramlari.sadievrenseker.com/2009/08/10/splines-seritler/>, Erişim tarihi: 18 Mayıs 2016)

<https://tr.wikipedia.org/wiki/Algoritma>( Erişim tarihi: 3 Kasım 2016)

1- <http://www.graciebarra.com/uk/2014/03/tips-bjj/five-hand-image/>, (12 Nisan 2016)

2-[http://www.plastangel.com/p/hand-surgery\\_17.html](http://www.plastangel.com/p/hand-surgery_17.html), (Erişim tarihi: 2 Kasım 2016)

3-[https://commons.wikimedia.org/wiki/File:Deformity\\_of\\_Hand\\_due\\_to\\_X-ray\\_burns.jpg](https://commons.wikimedia.org/wiki/File:Deformity_of_Hand_due_to_X-ray_burns.jpg), (Erişim tarihi: 2 Kasım 2016 )

4-<https://www.pinterest.com/pin/352758583284443385/>, (Erişim tarihi: 2 Kasım 2016)

## ÖZGEÇMİŞ

### Kişisel Bilgiler:

|                     |                   |                |                               |
|---------------------|-------------------|----------------|-------------------------------|
| <b>Adı</b>          | <b>Buket</b>      | <b>Uyruğu</b>  | <b>Türkiye Cumhuriyeti</b>    |
| <b>Soyadı</b>       | <b>ŞİMŞEK</b>     | <b>Tel no</b>  | <b>0506 111 37 13</b>         |
| <b>Doğum tarihi</b> | <b>20.02.1991</b> | <b>e-posta</b> | <b>simsekbukett@gmail.com</b> |

### Eğitim Bilgileri:

| <b>Mezun olduğu kurum</b> |                                    | <b>Mezuniyet yılı</b> |
|---------------------------|------------------------------------|-----------------------|
| <b>Lise</b>               | <b>Muratpaşa Lisesi</b>            | <b>2009</b>           |
| <b>Lisans</b>             | <b>Afyon Kocatepe Üniversitesi</b> | <b>2013</b>           |
| <b>Yüksek Lisans</b>      | <b>Akdeniz üniversitesi</b>        | <b>2016</b>           |

| <b>Yabancı Dilleri</b> | <b>Sınav türü</b> | <b>Puanı</b> |
|------------------------|-------------------|--------------|
| <b>İngilizce</b>       | <b>YDS</b>        | <b>55</b>    |

### **Yayınlar ve Bildiriler:**

Şimşek B, Yardımcı A, UsingBezierCurvesinMedicalApplications. Filomat 30:4 (2016), 937–943.

