

GAZİ ÜNİVERSİTESİ

BİLGİSAYAR MÜHENDİSLİĞİ

MAKİNE ÖĞRENMESİ PROJESİ

Ders Sorumlusu: Öğr. Gör. Dr. Oktay YILDIZ

Hazırlayanlar:

Cengizhan ÖZCAN

Mehmet Emre AKTAŞ

Onur Gökberk ÖZERDAL

Mayıs, 2015

İçindekiler Tablosu

[Leap Motion Destekli Örüntü Tanıma Projesi 1](#_Toc420595738)

[Giriş 1](#_Toc420595739)

[Proje Tanımı 1](#_Toc420595740)

[Projenin Amacı 1](#_Toc420595741)

[Projede Kullanılan Teknolojiler 1](#_Toc420595742)

[Leap Motion 1](#_Toc420595743)

[Yapay Sinir Ağları 1](#_Toc420595744)

[FeedForward Ağ Mimarisi 2](#_Toc420595745)

[Backpropagation 2](#_Toc420595746)

[Projenin Geliştirilmesi 3](#_Toc420595747)

[SONUÇ 4](#_Toc420595748)

[Kaynakça 5](#_Toc420595749)

# Leap Motion Destekli Örüntü Tanıma Projesi

# Giriş

## Proje Tanımı

Projemiz Leap Motion Destekli donanım sayesinde belirli bir mesafedeki el hareketleri ile çizilen rakamları sistemde kayıtlı olan örüntülerle eşleştirip sonucu ekrana yazdıran makine öğrenmesi projesidir.

## Projenin Amacı

Projenin temel amacı, el hareketlerini algılamayı destekleyen donanımlar sayesinde el hareketleriyle oluşturulan pattern’ leri bilgisayar ortamına aktarıp, klavye, mouse vb. donanımlara alternatif bir yöntem geliştirmektir. Bu amaca yönelik ilk aşamada el ile çizilen rakamların Leap Motion destekli donanım tarafından algılanması ve bilgisayara aktarılması amaçlanmıştır. Hazırlanacak bu proje sonuçlandıktan sonra bu projeden ilham alınıp geliştirilmesiyle klavye, mouse gibi donanımlara alternatif donanımlar kazandırılmış olacaktır.

## Projede Kullanılan Teknolojiler

### Leap Motion

Amerika merkezli Leap Motion şirketi tarafından geliştirilen ve el hareketlerini 3 boyutlu olarak algılayabilen bir donanımdır. Bilgisayara USB üzerinden bağlanan cihaz, üzerinde bulunan üç kızılötesi sensör yardımıyla 10 parmağın hareketini milimetrenin 100’ de 1’ ine kadar algılayabilmektedir. Cihaz bir el üzerindeki 29 noktayı x, y ve z eksenlerinde 150° açıyla algılayabilmektedir. Ayrıca el hareketlerini saniyede 200 frame ile izlemektedir.

### Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), insanlara özgü yeni bilgiler türetebilme, yeni bilgiler oluşturabilme ve keşfedebilme gibi özellikleri herhangi bir yardım almadan otomatik olarak gerçekleştiren bilgisayar sistemlerine denir.

Yapay sinir ağları oluşturulurken biyolojik sistemin çalışma mekanizması örnek alınır. Hiç karşılaşmadığı durumlar karşısında var olan örneklere bakılarak durumlar hakkında genellemeler yapıp çıkarım yapabilmektedir. YSA gözetimli öğrenme kullanmaktadır. Bu öğrenme ağdaki nöronların bağlantılarının istenilen değere ulaşana kadar bağlantı ağırlıklarının tekrar tekrar değiştirilmesiyle olur.

Yapay Sinir Ağları şu alanlarda kullanılmaktadır:

♦ Regresyon  
♦ Tahmin Yürütme   
♦ Karar Verme  
♦ Zaman Serisi Analizi  
♦ Sınıflandırma  
♦ Kontrol Benzetimi  
♦ Optimizasyon  
♦ Doğrusal olmayan sinyal işleme  
♦ Doğrusal olmayan sistem modelleme (Alkan, 2013)

Yapay sinir ağlarında Feedforward, RadialBasis, Dynamic Networks, Learning vectorquantization vb. ağ mimarileri vardır. Projede Feedforward mimarisi tercih edilmiştir.

#### FeedForward Ağ Mimarisi

Feedforward ağları nöronları arasında directedcycle bulunmayan bir sinir ağı yapısıdır. Bu yapı birden fazla katmandan oluşmaktadır. İlk katman input katmanıdır. Aralarda HiddenLayer bulunabilmektedir. Her katman bir önceki katman ile bağlantılıdır. Son katman output’u üretir.Single-Layer ve Multi-LayerPerceptron şeklinde iki farklı yapısı mevcuttur. Single-Layer’dainput katmanı direkt output katmanına bağlıyken Multi-Layer’dainput ve output arasında HiddenLayer’lar bulunabilmektedir.

Feedforward ağlardan birden fazla öğrenme tekniği bulunmaktadır. Bunlardan biri olan Backpropagation projede tercih edilmiştir.

##### Backpropagation

Backpropagation bir öğrenme algoritmasıdır. Algoritma, nöronların ağırlıklarını optimize etme amaçlı bir çalışma mekanizmasına sahiptir. Backpropagation öğrenme yapabilmek için her input’a karşılık gelen beklenen bir değer (output) istemektedir. Bu algoritma iki aşamadan oluşmaktadır.

1. Propagation: Bu aşamada nöronlar arası yolların oluşturulması ve her bir nörona delta değerleri verilmektedir.
2. Ağırlık optimizasyonu: Bu aşamada ise nöronlardaki delta değerlerini ve eğim değerlerini kullanarak her bir nöron için optimal ağırlık değeri bulmak hedeflenir.

## Projenin Geliştirilmesi

Leap Motion donanımının hareketleri algılamasıyla oluşturduğu sinyallerin bilgisayar ortamında matrise dönüştürülmesi ve bu dönüştürülen matrislerin normalizasyon gibi işlemler sonucunda elde edilen işlenmiş matrisin öğrenim kümesindeki verilerle kıyaslanıp eşleşen örüntünün bulunması sağlanmıştır.

Proje geliştirilirken EclipseJuno ve MatLab ortamları kullanılmıştır. LeapMotion’dan veri alınarak matrise dönüştürülmesi işlemi ve normalizasyon işlemleri EclipseJuno üzerinde gerçekleşirken, oluşturulan matrisler üzerinden yapay sinir ağı oluşturulma işlemi MatLab üzerinden yapılmıştır. Bu işlemler gerçekleştirilirken şu kütüphaneler kullanılmıştır:

* Java’nın standart kütüphaneleri,
* Java.IO,
* Java.lang,
* Java.Util,
* LeapSDK (Leap Motion),
* MatLabController (MatLab).

Proje gerçekleştirilirken ilk olarak LeapMotion’ı daha kolay yönetebilmek için bazı hareketler tanımladık. Bu hareketlerden ilki, sol el ile (el açık olmak koşulu ile) x ekseni boyunca yapılan kaydırma işlemi (swipegesture) programın rakam çizme bölümünün pasif ise aktife, aktif ise pasife geçmesini sağlamaktadır. Swipegesture ile rakam çizme aktif olduktan sonra sağ el ile çizilmek istenen rakamın çizilmesiyle matris oluşturulmuş olur. Çizim yapılırken verilerin daha doğru algılanabilmesi için sadece sağ el işaret parmağından okuma işlemi yapılmıştır. Ayrıca rakamların çizilmesi LeapMotion’ın x ekseninin + kısmında yapılmaktadır.

LeapMotion’dan gelen veriler üç boyutlu koordinat verileri olmakla birlikte sadece x ve y eksenindeki veriler alınmıştır. Alınan bu veriler ise 0 ile 400 arasında sınırlandırılmıştır. Çizim sonucu oluşturulan veriler 400x400 boyutundaki matrislere atanmıştır. Matris oluşturulurken hareket algılanan alanlara ‘1’ diğer alanlara ise ‘0’ değeri girilmiştir (Hareket algılanan x ve y değerleri 2 boyutlu matriste bir hücreye denk gelmektedir.). Oluşturulan matrisler üzerinde daha verimli sonuç alabilmek için yerleştirilen her ‘1’ değeri x ve y ekseninde 45’e 45 oranında genişletilmiştir.

Oluşturulan bu desenler üzerinde daha standart bir görüntü oluşturmak için çeşitli işlemler yapılmıştır. İlk olarak çizilen rakam matrisin ortasına kaydırılmış daha sonrasında matrisin boş kalan bölümleri ‘0’ ile doldurulmuştur. Bu işlemlerden sonra matris boyutunu 5’te 1 oranında küçültmek amacıyla her 25 birimlik karelere, karelerdeki 0-1 yoğunluğuna göre ‘0’ ya da ‘1’ değeri atanmıştır. Bunun sonucunda normalize edilmiş matrisler oluşturulmuş olur.Oluşturulan normalize edilen matrisler txt uzantılı dosyalara kaydedilir.

Yapay Sinir Ağının oluşturulması, eğitilmesi ve test edilmesi MatLab’ta yapılmaktadır. Eclipse ile MatLab arasındaki bağlantı MatLab Controller ile sağlanmaktadır.MatLab’takinewff fonksiyonu kullanarak Feedforward Yapay Sinir Ağı oluşturulmuştur. Oluşturulan ağın InputLayer’ında 10 nöron, OutputLayer’ında da 10 nöron bulunmaktadır. Öğrenim algoritması olarak Backpropagation kullanılmıştır. Bu algoritmada epoch sayısı 500’dür ve hata toleransı 1E-5 olarak belirlenmiştir.

Öğrenim kümesi oluşturulurken her rakam (0-9) için yukarıdaki matris oluşturma ve normalizasyon işlemleri uygulanır. MatLab’a gönderilen 80x80 boyutundaki matrisler 6400x1’lik matrislere dönüştürülür. Bu işlem her rakam için tekrarlanır ve 6400x10 boyutundaki öğrenim matrisinin her bir sütununa sırayla yerleştirilir. Oluşturulan bu öğrenim matrisi oluşturulan Yapay Sinir Ağının öğrenim kümesi olarak atanmıştır.

Projede oluşturulan öğrenim kümesinin test edilmesi sırasında Leap Motion üzerinde çizilen rakam,normalizasyon işlemleri sonucu MatLab’a aktarılır. MatLab’ta çizilen rakamın öğrenim kümesindeki elemanlara benzeme olasılıkları bulunur. Bu değerler Eclipse’e geri getirilerek en yüksek olasılığa sahip rakam ekranda gösterilir.

# SONUÇ

Bu projede Leap Motion kullanılarak çizilen rakamların bulunması amaçlanmıştır. Bunu yaparken yapay sinir ağları (Feedforward) kullanılmıştır. Bu yöntemin matris karşılaştırmasında doğruluk oranının daha iyi olduğu görülmüştür. Projedeki oluşturulan pattern’lere uygun şekilde çizim yapıldığında %80 oranında doğru sonuç alındığı görülmüştür. Fakat çizimlerde farklılıklar oluştuğunda doğruluk oranı azalmaktadır. Bu sonuç öğrenim kümesindeki verinin az olmasından kaynaklanmaktadır.

Projemiz destekleyen donanımlar aracılığıyla çizilen hareketleri bilgisayar ortamında anlamlandıran ve buna bağlı olarak işlemler yapan çalışmalara temel olacaktır.

# Kaynakça

Alkan, H. (2013, Eylül). *Yapay Sinir Ağları (YSA-Artificial Neural Networks)*. elektrikport.com: http://www.elektrikport.com/teknik-kutuphane/yapay-sinir-aglari/8874#ad-image-0 adresinden alındı

Çayıroğlu, İ. (2013, Nisan). *İLERİ ALGORİTMA ANALİZİ-5 Yapay Sinir Ağları.* ibrahimcayiroglu.com: http://www.ibrahimcayiroglu.com/Dokumanlar/IleriAlgoritmaAnalizi/IleriAlgoritmaAnalizi-5.Hafta-YapaySinirAglari.pdf adresinden alındı

*Java SDK Documentation*. (2015). developer.leapmotion.com: https://developer.leapmotion.com/documentation/java/index.html adresinden alındı

*MathWorks*. (2015). mathworks.com: http://www.mathworks.com/ adresinden alındı

*MatLab Control*. (2013). code.google.com: https://code.google.com/p/matlabcontrol/ adresinden alındı