

# DERİNLİK ALGILAYICIYLA 3B EL POZU BULMA VE SINIFLAMA 3D HAND POSE ESTIMATION AND CLASSIFICATION USING DEPTH SENSORS

Cem Keskin, Furkan Kırac, Yunus Emre Kara, Lale Akarun

Bilgisayar Mühendisliği Bölümü  
Boğaziçi Üniversitesi, İstanbul

keskinc@cmpe.boun.edu.tr, {kiracmus, yunus.kara, akarun}@boun.edu.tr

## ÖZETÇE

*Bu makale, derinlik imgesi kullanarak insan eline 3B iskelet oturtmak üzere geliştirdiğimiz yöntemi anlatmaktadır. İnsanı elini temsil etmek üzere 21 parçadan oluşan bir 3B el iskeleti yaratılmıştır. Bu model kullanılarak oluşturulan sentetik el derinlik imgeleri, piksel başına el parçası sınıflandırma yapacak rassal karar ormanları (RDF) eğitmek üzere kullanılmıştır. El parçası sınıflandırma sonuçları, ortalama kayma algoritması ile iskelet eklemlerinin 3B koordinatlarını bulmak üzere kullanılmaktadır. Sistem, Kinect imgelerini kullanarak gerçek zamanlı olarak, 30 fps hızında çalışabilmektedir. Uygulama olarak seçtiğimiz Amerikan İşaret Dili (AİD) rakamlarını tanımadaki, geliştirdiğimiz yöntemle destek vektör makinesi (DVM) kullanarak % 99.9 başarı elde ettik.*

## ABSTRACT

This paper describes our method to fit a 3D skeleton to the human hand using depth images. The human hand is represented by a 3D skeleton with 21 parts. This model is used to generate synthetic depth images, that are used to train Random Decision Forests (RDF), which are used to assign each pixel to a hand part. Mean-shift method is used on the classification results and joint locations are estimated. The system can run in real time at 30 fps on Kinect depth images. We use this method and Support Vector Machines for classification and obtain 99.9% recognition rate on the American Sign Language (ASL) digit recognition problem.

## 1. GİRİŞ

El hareketleri ile kontrol edilen insan-bilgisayar etkileşim yöntemleri geliştirmek, araştırmacıların 20 senedir ilgisini çeken bir konu olmuştur. Ancak karmaşık bir geri planda elin yerinin ve tüm eklemleriyle pozunun bulunması, çözülemeyen zor bir problem olarak kalmıştır. Işık durumundan bağımsız olarak derinlik ve renk imgesi sağlayabilen Kinect derinlik sensörünün geliştirilmesiyle, el yeri bulma ve geri plandan bölütleme problemleri büyük ölçüde çözülmüştür; ancak 3B el iskeletinin bulunması hala çözülmemiş bir problemdir.

Bu çalışma Tubitak 108E161, BU-BAP 09M101, BU-BAP 6061 ve DPT 2007K120610 TAM projeleri tarafından desteklenmektedir.  
978-1-4673-0056-8/12/\$26.00 ©2012 IEEE

Kinect derinlik imgesini kullanarak insan vücudu iskeleti çıkaran bir yöntemin geliştirilmesi, doğal insan-bilgisayar etkileşim yöntemleri için çığır açıcı olmuştur [1]. Bu yöntemin el iskeletinin bulunmasında da kullanılması, şu nedenlerle daha zordur: i) El imgesi, insan vücudu imgesinden oldukça küçüktür; ii) İnsan vücudunun bilgisayarla etkileşeceği normal durumlarda hep baş-yukarı olduğu varsayılabilir; ancak el, her oryantasyonda olabilir; iii) Parmakların beraberce alabileceği konfigürasyon sayısı çok daha fazladır ve birbirini örtmeye daha fazla rastlanır. Öte yandan, insan vücut ve giysilerindeki değişkenlik düşünüldüğünde, eller arası farklar çok daha azdır.

El pozu kestirimi probleminde yaklaşımların bir özeti, [2]'de verilmiştir. Kısıtlı sayıda çalışmanın çoğu RGB imgeleri ile çalışırken, derinlik imgeleri ile çalışan birkaç örnek vardır [3, 4, 5, 6, 7, 8, 10, 11]. [8] 26 eklemli bir modeli büyük başarıyla izlemek için elde tutulan işaretçi cisimlerden yararlanmaktadır. Pek çok çalışmada, 3B bir el modelinden yararlanarak sentetik imgeler yaratılmış ve bunlar sınıflandırıcıların eğitimi için kullanılmıştır [1, 4].

Benzer şekilde, bu çalışmada da, el parçaları işaretlenmiş bir sentetik 3B modeli kullanarak, sentetik imgeler yarattık. Sentetik el imgelerinde, her bir parçanın merkezi, bir eklemi gösterir. Bu modelleri değişik parametrelerini değiştirip oynatarak, büyük veri setleri elde ettik. Bu verisetlerini, çeşitli Rassal Karar Ağaçları (RKA) eğitmek için kullandık. Bu RKA'lar, piksel başına el parçası sınıflandırma için kullanıldı. Son olarak, ortalama kayma algoritması ile el eklemlerini takip ettik. Bu şekilde, gerçek zamanda 3B el pozunu bulabilen bir sistem geliştirdik. Bu sistemi, Amerikan İşaret Dili (AİD) rakamlarını tanımadaki kullanarak hem sentetik hem de gerçek veride % 99.9 başarı elde ettik.

Kısım 2, el iskeleti oturtma yöntemimizi anlatmaktadır. Kısım 3'te deneylerimizi özetliyor ve AİD rakamları sınıflandırma sonuçlarımızı veriyoruz. Kısım 4'te vargılarla makale sonlandırılmaktadır.

## 2. YÖNTEM

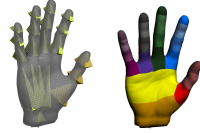
Yaklaşımımız, el piksellerinin hangi el parçasına ait olduğunu sınıflandıracak çoklu karar ağaçlarının eğitiminde, sentetik bir el modelinin değişik parametrelerle canlandırılmasıyla oluşan bir verisetinin kullanılmasına dayalıdır. Canlandırmada, aradeğerleme ve eklem parametrelerine güdültü ekleme ile

çeşitleme sağlayarak, büyük bir veriseti elde ettik. Aynı veritabanları üzerinde, sistemin piksel sınıflandırma başarımını arttıracak şekilde çoklu karar ağaçları eğittik. Sınıflandırma sonucunda, her pikselin ardıl olasılık değerleri 3B iskelet eklem koordinatlarını kestirmek için kullanılır. Uygulama, Kinect kullanılarak, saniyede 30 çerçeve hızında kare atlamadan 3B el iskeletini oluşturabilmektedir.

Sistemin başarımı her bir RKA'nın verimine bağlıdır. Ağaç derinliği fazlaysa veya eğitim imgeleri gerçek hayatta karşılaşılan el pozlarının çeşitliliğini yansıtmıyorsa, ağaçlar aşırı uyum sağlar ve iyi genellemez. Bunun bir çözümü sisteme çok sayıda gerçek eğitim imgesi eklemektir. Shotton ve diğerleri [1] gerçek veritabanlarında birçok yapılanışın olmamasından dolayı sentetik imgelerin genelleme için daha uygun olduğunu belirtmiştir. İnsanların elleri arasındaki farklılığın fazla olmamasından dolayı bu değişimi yansıtacak gerçekçi sentetik imgeler üretmenin daha uygun olacağına karar verdik ve sistemi bu imgeleri kullanarak eğittik.

### 2.1. Veri

El modeli olarak, 19 kemik, 15 eklem ve 21 parça içeren 3 boyutlu kaplanmış bir örgü modeli (Şekil 1) kullanıyoruz. El parçaları, merkezleri eklemler olacak şekilde seçildi. Baş parmak 3 parçaya, diğer parmaklar 4 parçaya bölündü. Avuç, üst ve alt avuç olmak üzere ikiye bölündü. Modeli tonlamalı olarak gerçekleştireceğimiz derinlik imgeleri, etiket değerleriyle gerçekleştireceğimiz etiket imgeleri oluşturulur.



Şekil 1: Sıradüzensel iskeletli 3 boyutlu el modeli ve etiketlenmiş parçalar

Bu şekilde, 60 bin ila 200 bin imge içeren sentetik veritabanları oluşturduk. Bu veritabanları üzerinde bir çok RKA eğittik. Bu veritabanları, eğitilmiş ağaçların yaygın el hareketlerinde rastlanabilen ama veritabanında olmayan el pozlarına da uygun olabilmesini sağlayacak şekilde tasarlanmıştır.

Oyunlarda, doğal arayüzlerde ve işaret dillerinde sıkça rastlanılan el pozları elle modellenmiştir. Bu modellerin sıradüzensel iskelet modeli üzerindeki aradeğerlemeleri her karede eklemlere hafif gürültü eklenerek ve kamera pozisyonu değiştirilerek gerçekleştirilmiştir.

### 2.2. Öznitelikler

Piksel sınıflandırma yaptığımız için özniteliklerin karmaşıklığı sistemin verimliliği üzerinde büyük bir etki yaratmaktadır. Hızlı ve etkili oldukları için basit derinlik karşılaştırması özniteliklerini kullanmaktayız [1]. Piksel koordinatı  $\mathbf{x}$ , derinlik değeri  $I(\mathbf{x})$ ,  $\mathbf{u}$  ve  $\mathbf{v}$  ise piksele bağlı kaydırma vektörleri olmak üzere öznitelikler

$$F_{\mathbf{u},\mathbf{v}}(I, \mathbf{x}) = I\left(\mathbf{x} + \frac{\mathbf{u}}{I(\mathbf{x})}\right) - I\left(\mathbf{x} + \frac{\mathbf{v}}{I(\mathbf{x})}\right) \quad (1)$$

şeklinde hesaplanır. Kaydırmaların normalizasyonu 3B ötelemeye karşı değişimsizlik sağlar. Fakat, dönme ve ölçeklemeye karşı değişimsizlik sağlamaz. Eğitim imgeleri bu dikkate alınarak oluşturulur. Derinlik imgesinde arkaplan çıkarımı daha kolay olduğu için el piksellerinin bilindiğini varsayar ve geri kalan bölgenin derinliğini çok yüksek bir değere sabitleriz.

### 2.3. Piksel Sınıflandırma

Piksel sınıflandırma rassal karar ormanı (RKO) [9] kullanılarak yapılır. RKO rassal karar ağaçlarının birleşimidir. Eğitim sırasında, RKA'ların boğumları veriyi sol ve sağ çocuklarına daha saf boğumlar oluşturacak şekilde böler. Saf bir boğumda sadece tek bir sınıfa ait örnekler vardır. Bizim durumumuzda, her bir boğum  $\mathbf{u}$ ,  $\mathbf{v}$ , ve  $\tau$  değerleri öğrenir ve veriyi sağ ve sol çocuklarına

$$C_L(\mathbf{u}, \mathbf{v}, \tau) = \{(I, \mathbf{x}) | F_{\mathbf{u},\mathbf{v}}(I, \mathbf{x}) < \tau\} \quad (2)$$

$$C_R(\mathbf{u}, \mathbf{v}, \tau) = \{(I, \mathbf{x}) | F_{\mathbf{u},\mathbf{v}}(I, \mathbf{x}) \geq \tau\} \quad (3)$$

şeklinde böler.

Eğitim esnasında, öznitelikler ve eşik değerleri her boğumda rasgele seçilir ve ayırım entropi azalımı üzerinden skorlanır:

$$S(\mathbf{u}, \mathbf{v}, \tau) = H(C) - \sum_{s \in \{L, R\}} \frac{|C_s(\mathbf{u}, \mathbf{v}, \tau)|}{|C|} H(C_s(\mathbf{u}, \mathbf{v}, \tau)) \quad (4)$$

Burada  $H(K)$ ,  $K$  kümesindeki etiketlerin normalize edilmiş histogramlarından hesaplanmış Shannon entropi değerini gösterir. Her boğumda bir çok rasgele  $(\mathbf{u}, \mathbf{v}, \tau)$  değerleri seçilir, en yüksek skoru veren grup seçilir.

Her ağaç 160x160 boyutlarındaki her bir eğitim imgesinden seçilen rasgele 2500 piksel ile eğitilir.  $\mathbf{u}$ ,  $\mathbf{v}$  kaydırma vektörlerinin değerleri 0-60 piksel aralığında rasgele seçilir.  $\tau$  eşik değeri kabaca el boyutlarına yakın olan -200mm ile 200mm derinlik değerleri arasında seçilir. En yüksek skoru veren  $\{\mathbf{u}, \mathbf{v}, \tau\}$  grubunu bulmak için 4000'e kadar farklı deneme yapıyoruz.

$(I, \mathbf{x})$  pikselinin sınıflandırması sırasında, kök boğumdan başlanır, uç boğuma ulaşıncaya kadar piksel döngüsel olarak sağ veya sol çocuklara taşınır. Her uç boğumda her  $c_i$  sınıfında denk gelen ve eğitim sırasında öğrenilmiş olan  $P(c_i | I, \mathbf{x})$  ardıl olasılıkları bulunur. Son kararı verirken,  $N$  toplam ağaç sayısını göstermek üzere bütün ağaçlarda ulaşılan ardıl olasılıkların ortalaması hesaplanır:

$$P(c_i | I, \mathbf{x}) = \frac{1}{N} \sum_{n=1}^N P_n(c_i | I, \mathbf{x}). \quad (5)$$

Sınıflandırma sonuçlarını güçlendirmek için, bütün pikselleri ağaçlarda değerlendirip uç boğumlardaki histogramları tekrar oluşturuyoruz. Bu işlem piksel sınıflandırma başarımının ağaç başına %2-3 ve orman için %6-8 civarında artmasını sağlar.

### 2.4. Eklem Pozisyonu Kestirimi

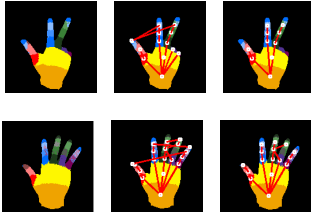
Her piksele ardıl olasılık atandıktan sonra, bu değerler eklem pozisyonlarını kestirmekte kullanılır. Bu yöntem çok sayıda yanlış kabul üretir. Bu aykırı değerler bir el parçasına denk

gelen piksellerin merkezini aşırı etkiler. Aykırı değerlerin etkisini düşürmek için ortalama kaydırma mod bulma yöntemi tercih edilir. Ortalama kaydırma yöntemi her sınıf etiketinin olasılık yoğunluğunu ağırlıklandırılmış Gauss çekirdekleri vasıtasıyla kestirir. Her ağırlık, pikselin ilgili sınıfa denk gelen ardıl olasılığının derinliğinin karesiyle çarpımıyla hesaplanır. Bu yöntemle bulunan eklem pozisyonları elin yüzeyinde çıkar ve iskeletle örtüşmesi için içeri itilmesi gerekir. Elin kalınlığı neredeyse düzgün olduğu için dönüşüm bütün elin ötelenmesiyle yapılır.

Ortalama kaydırma yöntemi en yakın maksimum noktayı bulmak için bayır çıkışı yaklaşımı kullanır. Bulunan maksimum nokta yerel olduğu için bir çok başlangıç noktası seçilir ve en büyük değere ulaşan seçilir.

## 2.5. Örtmeler

Ellerin çok eklemli ve esnek olmalarından dolayı bazı el kısımlarının diğer kısımlar tarafından örtülmesi sıklıkla görülen bir durumdur. İmgedeki her piksele, tüm etiketler için bir ardıl olasılık değeri atanır. Bir eklem görünüp görünmediği ile ilgili karar ortalama kaydırma adımı sırasında ulaşılan puan değerinin eşiklenmesi ile bulunmaktadır. Eşikleme adımının etkileri Şekil 2’de gösterilmiştir. Soldaki imgeler özgün piksel sınıflandırma sonuçlarıdır. Ortadaki imgeler eklem pozisyonları ile birlikte çizilmiş halleridir. Bu imgeler yanlış kabul edilmiş pikseller yüzünden oluşan sahte eklemeleri göstermektedir. Ancak eşikleme değerini çok arttırmak aynı zamanda doğru eklemeleri de eleyebilir. Üst sırada eşikleme değeri çok yüksek, alt sırada ise çok düşük tutulmuştur.



Şekil 2: El iskeleti oturtulması ve güven puanlarını eşiklemenin eklemeler üzerindeki etkisi

Bilinen eklemelerden gözükmeyen eklemelerin yerlerini kestirmek için çeşitli analitik veya otomatik öğrenme yöntemleri uygulanabilir. Eksik parmak eklemelerini parmak pozisyonlarından gerçekçi olarak kestirebilmek için sıkça kullanılan el şekillerinden Yapay Sinir Ağları (YSA) eğittik. Ancak iskelet bilgisinin açıklayıcı gücü, eksik eklemelere rağmen poz sınıflandırma işi için yeterince güçlü olduğundan, görünmeyen eklemelerin yerlerini en yakın görünen eklemelerin koordinatlarıyla değiştirmek hem basit hem de yeterince verimli bir buluşsal fonksiyon oldu.

## 3. DENEYLER

### 3.1. Veritabanları

Piksel sınıflandırma yapacak RKA’ların eğitimi için sentetik veritabanları kullanıldı. RKA’ların performansları eğitim kümesine yakından bağlıdır. Her eğitim imgesi bir derinlik imgesi ve

etiket imgesinden oluşur. Bellek limitlerinden dolayı veritabanı boyutlarını 200 bin imgede sınırladık.

Poz sınıflandırma için, 10 farklı kişinin 10 işaret dili rakamını yaparken çekilmiş derinlik ve etiket imgelerini topladık. Etiket imgelerinin oluşturulması için sentetik veritabanları üzerinde eğitilen RKA’lar kullanıldı. Ortalama kaydırma yöntemi ile gerçek imgelerin eklem pozisyonları tahmin edildi ve kapatılmış eklemeler Kısım 2.5’te anlatıldığı gibi en yakın eklemeye ilişkilendirildi. Sonuç olarak ulaşılan iskeletler poz sınıflandırma işleri için veritabanı olarak kullanıldı.

### 3.2. Piksel sınıflandırma sonuçları

Ağaç derinliği, piksel sayısı,  $u$ ,  $v$ , ve  $\tau$  parametrelerinin piksel sınıflandırma başarımı üzerindeki etkilerini inceledik. Ağaç derinliğini artırmak, beklendiği gibi piksel sınıflandırma başarımını artırmakta, fakat her seviye ile birlikte bellek kullanımını da katlanmaktadır. Sistemdeki ağaç derinliği 20 ile sınırlandırıldı, eğitim kümesinde piksel sınıflandırma başarımı %67, histogram güncellemesi sonrası %70 olarak ölçüldü.  $u$  ve  $v$  için en uygun değer 60 olarak bulundu, ve  $\tau$  değeri 20’den büyük olduğu sürece etkisi gözardı edilebilir bulundu.

### 3.3. Örnek Uygulama: Amerikan İşaret Dili Rakam Tanıyıcısı

Sistemimizi denemek için Amerikan İşaret Dili (AİD) rakamlarını tanıyacak şekilde eğittik. Kısım 2’de anlatılan yöntemle el iskeleti tahminleri bulundu ve poz sınıflandırıcısı bu iskeletleri rakamları tahmin etmek için kullanıldı.

İlk olarak, eğitilmiş RKA’lar kullanılarak Kinect ile toplanmış AİD imgeleri etiketlendi. Daha sonra, bu etiketler kullanılarak eklem yeri tahminleri yapıldı ve sonuç el yapıları AİD pozları olarak sınıflandırıldı.

#### 3.3.1. El Pozu Sınıflandırıcıları

Gerçek zamanlı kullanım için, tanıma hız oranı da önemli bir kriterdir. Sınıflandırma için Yapay Sinir Ağı (YSA) ve Destek Vektör Makinası (DVM) kullandık. Denenen bütün sınıflandırıcılar için ilk olarak parametre eniyileme yapıldı ve seçilen parametrelerle 5x2 çapraz sağlama yapıldı.

Sistemi hem gerçek hem de sentetik veri üzerinde eğittik. Sentetik veri sonuçları parametre eniyileme ve Kısım 2.5’te anlatıldığı gibi kapatmanın etkilerini sınamak için kullanıldı. Sentetik veri üzerinde YSA’lar, DVM’ler ve RKA’lar eğittik. Gerçek veri üzerinde sadece YSA ve DVM eğittik.

#### 3.3.2. Sentetik Veri Üzerinde AİD Rakam Sınıflandırma Sonuçları

Eğitim için kullanılan sentetik veritabanı her bir rakam için 2000’er örnek olmak üzere toplam 20000 örnekten oluşmaktadır. Örnek imgeler, elle tasarlanmış poz şablonlarının her eklemine Gauss gürültüsü eklenmiş hallerinin çeşitli kamera açılarından gerçekleşmesiyle elde edilmiştir.

YSA için en uygun gizli düğüm sayısı 20 olarak bulunmuştur. DVM’de en uygun masraf değeri  $C = 2^6$  ve Gauss yayılımı değeri  $\gamma = 2^{-4}$  olarak bulunmuştur. Karar ormanı olarak, 128 RKA kullanılmıştır.

Tablo 1: Sentetik AİD rakam veritabanı üzerinde sınıflandırma oranları ve süreleri

Yöntem Adı	Başarım	Sınıflandırma Süresi (ms)
YSA	<b>99.89</b>	<b>0.0045</b>
DVM	<b>99.96</b>	0.3
RKA-128	<b>99.94</b>	0.32

Sınıflandırma başarımları ve süreleri Tablo 1’de verilmiştir. Şekil 3’te bazı örnekler için ara faz ve son iskelet sonuçları gösterilmiştir.

### 3.3.3. Gerçek Veri Üzerinde AİD Rakam Sınıflandırma Sonuçları

Gerçek veri üzerinde yöntemlerin eniyi parametrelerini bulmak için ızgara üzerinde 5x2 çapraz sağlama yaptık. Tablo 2’de, YSA ve DVM parametrelerinin en uygun değerleri ve ilişkili başarımlar verilmiştir. DVM’ler YSA’lardan daha iyi performans sağlamış ve sağlama kümesi üzerinde mükemmel yakın başarımlar vermiştir.

Tablo 2: Eniyi parametreler ile ortalama eğitim ve sağlama başarımları

Yöntem Adı	Eniyi Parametreler	Eğitim Başarımı	Sağlama Başarımı
YSA	H=40	<b>99.27</b>	<b>98.81</b>
DVM	$C=2^5, \Gamma = 2^{-2}$	<b>100</b>	<b>99.90</b>



Şekil 3: Sentetik AİD örnekleri ve oturtulan iskeletler: En solda derinlik imgeleri, sonra sırasıyla etiket imgeleri, eklem pozisyonları ve bulunan 3B iskeletler görülmektedir.

## 4. SONUÇLAR

Bu çalışmada, derinlik imgeleri üzerinden RKO kullanarak 3B el iskeleti kestiren bir yöntem önerdik. Bu yöntem için çok sayıda sentetik el imgesi yarattık ve otomatik olarak etiketle-

dik. Sentetik imgeler üzerinden eğitilen RKO’ların yeterince iyi sonuçlar verdiğini gözlemledik. Örnek çalışma olarak, AİD rakamlarını sentezleyerek bir RKO eğittik, ve bu RKO’nun kestirdiği iskeletleri DVM ile sınıflandırdık. Yöntemimizin gerçek AİD rakamları üzerindeki gerçek zamanlı başarımlar oranı %99.9 olarak bulundu.

Sistemimiz çok etkin ve gerçek zamanlı bir el iskeleti kestirme ve el şekli tanıma yöntemi olarak önerilmektedir. Bu çalışmada zaman içinde bir süzme işlemi gerçekleştirilmemiştir. Kestirilen iskelet, Kalman veya parçacık süzgeci ile izlenerek geliştirilmeye açıktır.

## 5. KAYNAKÇA

- [1] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.
- [2] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. Vision-based hand pose estimation: A review. *Computer Vision and Image Understanding*, 108(1-2):52–73, Oct. 2007.
- [3] M. de La Gorce, D. J. Fleet, and N. Paragios. Model-Based 3D Hand Pose Estimation from Monocular Video. *IEEE transactions on pattern analysis and machine intelligence*, pages 1–14, Feb. 2011.
- [4] S. Malassiotis and M. Srinivasan. Real-time hand posture recognition using range data. *Image and Vision Computing*, 26(7):1027–1037, July 2008.
- [5] P. Suryanarayan, A. Subramanian, and D. Mandalapu. Dynamic Hand Pose Recognition Using Depth Data. *2010 20th International Conference on Pattern Recognition*, pages 3105–3108, Aug. 2010.
- [6] Z. Mo and U. Neumann. Real-time Hand Pose Recognition Using Low-Resolution Depth Images. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR’06)*, pages 1499–1505, 2006.
- [7] X. Liu and K. Fujimura. Hand gesture recognition using depth data. *Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings.*, pages 529–534, 2004.
- [8] I. Oikonomidis, N. Kyriazis, and A. Argyros. Markerless and efficient 26-DOF hand pose recovery. In *Proceedings of the 10th Asian conference on Computer vision-Volume Part III*, pages 744–757. Springer, 2011.
- [9] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.
- [10] E. Stergiopoulou and N. Papamarkos. Hand gesture recognition using a neural network shape fitting technique. *Engineering Applications of Artificial Intelligence*, 22(8):1141–1158, Dec. 2009.
- [11] T. de Campos and D. Murray. Regression-based Hand Pose Estimation from Multiple Cameras. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR’06)*, pages 782–789, 2006.