

Pinotator ile Hızlı İmge İşaretleme

Fast Image Annotation with Pinotator

İsmail Arı, Yunus Açıköz

Bilgisayar Mühendisliği Bölümü
Boğaziçi Üniversitesi, İstanbul, Türkiye
{ismailar,yunus.acikoz}@boun.edu.tr

Özetçe

Bu çalışma imge işleme ve bilgisayarla görü gibi alanlarda sıklıkla ihtiyaç duyulan imge işaretleme çabalarını ortak bir çatı altında toplamayı hedeflemektedir. Araştırmacıların önemli mesaisini gerektiren imge işaretleme ve veritabanı oluşturma sürecini hızlandırmak için genel bir yöntem önerilmiş ve destek olarak geliştirilen işaretleme aracı sunulmuştur. Farklı kullanım durumları için çizilen örnek senaryolarla aracın nasıl kullanılacağı gösterilmiştir. Açık kaynak olarak araştırmacıların hizmetine sunulan araç ile işaretleme sürecinin otomatize edilmesi amaçlanmıştır.

Abstract

This work aims to combine the image annotation efforts commonly spent in fields such as image processing and computer vision. A generic flow is proposed to improve the process of annotating images and gathering databases, which takes considerable time of the researchers, and an accompanying tool is presented. Sample scenarios for different use cases are taken into consideration to show the usage of the tool. The process of image annotation is planned to be automatized with the free tool, open for the use of researchers.

1. Giriş

Bilgisayarla görü ve imge işleme araştırmalarının güvenilirliği, bu amaçla toplanmış veritabanlarında sergiledikleri performansları ile ölçülmektedir. Veritabanı oluşturmak ve gerçeklik verilerini işaretleme, yoğun bir emek istemekte, araştırmada önemli bir işgücü ve zaman teşkil etmektedir.

Ortak bir araç ile farklı kullanımlar için çoklu ortam (imge, video, ses) işaretleme hedefleyen çalışmalar yapılmaktadır [1]. Fakat bu çalışmalar genellikle semantik işaretleme odaklanmış, ses veya videoya ait zaman çizelgesinde metin eklemek amacıyla tasarlanmıştır.

Bu çalışma; farklı amaçlar için sarf edilen imge işaretleme çabalarını ortak bir çatıda toplamayı ve bu süreci otomatize etmeyi önermektedir. Bu amaçla platformdan bağımsız ve açık kaynaklı yeni bir imge işaretleme aracı sunulmakta, örnek kullanım senaryoları ile desteklenerek işaretleme sürecine katkıları gösterilmektedir.

2. Pinotator: İmge İşaretleme Aracı

Pinotator iki temel bileşenden oluşmaktadır. İlk nirengi noktası, ikincisi ise dikdörtgen işaretleme işlevleridir.

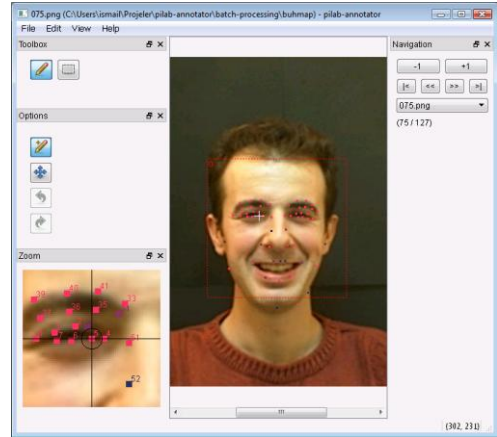
2.1. Nirengi noktası işaretleme

Yüz işleme, biyomedikal imge işleme, nesne tanıma ve takibi, 2B'den 3B bilginin geri-çakımı gibi birçok alanda nirengi noktalarına ihtiyaç duyulur. Pinotator bu alanlardaki araştırmacıların genel kullanımına imkân verecek özellikleri içerir. Bu özellikler şöyle sıralanabilir: *Nokta ekleme*: Nirengi noktalarını eklemek için kullanılır. Grafik tablet ile bu işlemi kolaylaştırmak mümkündür (bkz. Şekil 1). *Nokta sürüklenme*: İşaretleme noktalarının yer değiştirilmesi için kullanılır.

2.2. Dikdörtgen işaretleme

İmge işleme ve bilgisayarla görü çalışmalarında kullanılan diğer önemli işaret bilgisi imgeler üstünde belirlenen belli kutucuklardır. Araç şu işlevleri sunar:

Dikdörtgen ekleme: İmge üstünde kutular seçmeyi sağlar. *Dikdörtgen sürüklenme*: Nokta sürüklemeye benzer işlevedir. *Dikdörtgeni yeniden boyutlandırma*: Seçili kutuların boyutlarını güncellemek için kullanılır.



Şekil 1: İşaretleme anındaki ekran görüntüsü

2.3. Güvenilirlik ve İşaretleme Uyumumu

Her ne kadar gerçeklik verisi olarak insanların işaretlediği verileri kullansak da, bu işaretleme küçük hatalar içerebilir. Genelde nirengi noktası konumları ışık şiddeti varyasyonunun çok olduğu bölgelerde seçilmektedir. Bu düşünceyle imge yatay ve dikey Sobel filtreleriyle evrştirilmiş (konvolüsyon), elde edilen değerlerle doğru orantılı olarak Şekil 1'de sol alttaki yakın imgede görüldüğü gibi noktanın rengi kırmızılaştırılmıştır.

2.4. Diğer bileşenler

İşaretlemeye yardımcı olması amacıyla araca yakın ekran penceresi, resim gezgincisi, geçmiş işlemler bilgisi eklenmiştir.

2.5. Teknik özellikler, Dosya Protokolü ve Erişilebilirlik

Pinotator, Python programlama dilinde geliştirilmiştir. Araç ve ek olarak sunulan otomatik imge işaretleyici bileşenler, <http://code.google.com/p/pilab-annotator> adresinde özgür kullanıma açıktır. Teknik ayrıntılara ilgili adresten erişilebilir. Pinotator, her imge için elde edilen işaretleme bilgilerini ilgili bir *xml* dosyasında şu biçimde kaydetmektedir:

```
<?xml version="1.0" encoding="UTF-8"?>
<annotation>
  <objects count="N" type="points">
    <![CDATA[x1 y1
              x2 y2
              ...
              xN yN]]>
  </objects>
  <objects count="M" type="rectangles">
    <![CDATA[x1 y1 en1 boy1
              x2 y2 en2 boy2
              ...
              xM yM enM boyM]]>
  </objects>
</annotation>
```

2.6. Kullanıma uygunluk

Yapılan işaretleme diğer çalışmalarda kullanılan formatlarla uyumlu olabilmesi ve yeniden kullanılabilirliğini arttırmak amacıyla format dönüştürücü fonksiyonlar eklenmiştir. Stegmann *v.d.*'nin geliştirmiş olduğu aktif görünüm modeli araç kutusunun (AAM Toolbox) kullandığı *asf* (active shape file) [2] ve aktif şekil modeli tabanlı takipçilerin (*asm-library*) kullandığı *pts* (points file) [3] uzantılı dosya formatları için dönüştürücüler tedarik edilmektedir. OpenCV dâhilinde kullanılabilen yüz bulucusu için de dikdörtgen kutu bilgileri dönüştürülebilmektedir [4].

3. Otomatik İmge İşaretleme

Veritabanlarındaki örnekler arasında ilişileşim (korelasyon) bulunmaktadır ve öznelilikler birbirinden tamamen bağımsız değildir. Tümünün elle işaretleme yerine bir kısmının elle işaretleme ve elde edilen modelin diğer kısmı işaretleme kullanılması işlem süresini kısaltacaktır. Örneğin, bir yüz veritabanına eklenecek birkaç resmi elle işaretleme yerine bir modelle otomatik işaretleme, sonra ince düzeltmeleri yapmak daha akıllıca olacaktır.

3.1. Otomatik nirengi noktası işaretleme

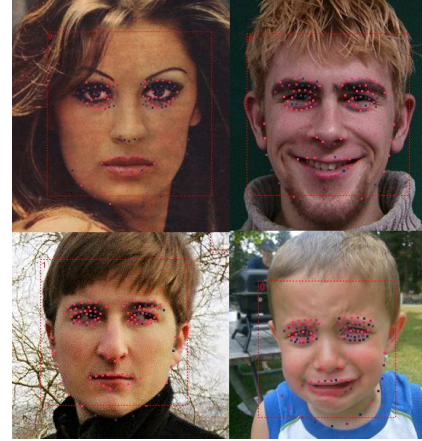
Bu senaryoda *asm-library* ile önceden oluşturulmuş genel bir model ile her imge için nirengi noktaları bulunmuştur [3]. İnce ayrıntılar *Son İşlem* altbölümünde anlatıldığı üzere kullanıcı tarafından yapılmaktadır.

3.2. Otomatik dikdörtgen işaretleme

Otomatik dikdörtgen işaretleme işlemine örnek oluşturması adına yaygın olarak kullanılan Viola-Jones yüz bulucusu kullanılmıştır.

3.3. Son İşlem

Şekil 2’te bir özne için verilerle eğitilmiş nirengi noktası modelinin ve Viola-Jones yüz bulucusunun sonuçları verilmiştir. Aydınlanma, cinsiyet, sakal, yüz ifadesi, yaş, ortam farklılıkları içeren ve modelin eğitildiği veritabanında bulunmayan bu örnekler için küçük değişikliklerle gerçeklik verilerine ulaşmak ve veritabanını geliştirmek mümkündür. Elbette bu sonuçlar kullanılan otomatik işaretleme modülünün performansına bağlıdır. Önerilen bileşenler nihai otomatik işaretleme olarak düşünülmemeli, örnek olarak ele alınmalıdır. Sonuçların güvenilirliği arttıkça, işaretleme süresi ters orantılı olarak azalacaktır.



Şekil 2: Örnek işaretleme sonuçları

4. Sonuç

Bu bildiriye, imge işleme ve bilgisayarla görü araştırmalarında sıklıkla kullanılan imge işaretleme sürecini iyileştirmek için genel bir yöntem önerilmiştir. Bu işlemi ortak bir çatıda toplamayı ve otomatize etmeyi desteklemek amacıyla platformdan bağımsız ve açık kaynaklı yeni bir imge işaretleme aracı sunulmuş, örnek kullanım senaryoları ile desteklenerek işaretleme sürecine katkıları gösterilmiştir.

5. Teşekkür

Bu çalışma, 108E161 numaralı proje kapsamında TÜBİTAK tarafından desteklenmiştir.

6. Kaynakça

- [1] C. Frisson, S. Alaçam, E. Coşkun, D. Ertl, C. Kayalar, L. Lawson, F. Lingenfelser ve J. Wagner, "CoMediAnnotate: towards more usable multimedia content annotation by adapting the UI", *Proc. of eINTERFACE'10 The Summer Workshop on Multimodal Interfaces*, 2010, sf. 10-17.
- [2] M. B. Stegmann, B. K. Ersbøll ve R. Larsen, "FAME - A Flexible Appearance Modelling Environment", *IEEE Transactions on Medical Imaging*, IEEE, 2003.
- [3] Y. Wei, "Research on Facial Expression Recognition and Synthesis", *Yüksek Lisans Tezi*, Bilgisayar Bilimleri ve Teknolojileri Bölümü, Nanjing Üniversitesi, Şubat 2009, <http://code.google.com/p/asmlibrary>.
- [4] G. Bradski ve A. Kaehler, "Learning OpenCV: Computer Vision with the OpenCV Library", *O'Reilly*, 2008.