

XML로 변환하기

-Image set from Fddb의 전처리

YOLO 프로그램은 VOC라는 대회가 썼던 dataset의 XML을 기준으로 삼고 있습니다.

이에 따라 Fddb의 Annotation 형태인 txt에서 VOC의 XML로 변환해야 합니다.

VOC의 XML 형태

```
<?xml version="1.0"?>
- <annotation>
  <folder>VOC2007</folder>
  <filename>000001.jpg</filename>
  - <source>
    <database>The VOC2007 Database</database>
    <annotation>PASCAL VOC2007</annotation>
    <image>flickr</image>
    <flickrid>341012865</flickrid>
  </source>
  - <owner>
    <flickrid>Fried Camels</flickrid>
    <name>Jinky the Fruit Bat</name>
  </owner>
  - <size>
    <width>353</width>
    <height>500</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  - <object>
    <name>dog</name>
    <pose>Left</pose>
    <truncated>1</truncated>
    <difficult>0</difficult>
    - <bndbox>
      <xmin>48</xmin>
      <ymin>240</ymin>
      <xmax>195</xmax>
      <ymax>371</ymax>
    </bndbox>
  </object>
  - <object>
    <name>person</name>
    <pose>Left</pose>
    <truncated>1</truncated>
    <difficult>0</difficult>
    - <bndbox>
      <xmin>8</xmin>
      <ymin>12</ymin>
      <xmax>352</xmax>
      <ymax>498</ymax>
    </bndbox>
  </object>
</annotation>
```

이 사진은 VOC의 XML 형식을 따르는 파일의 예를 보여줍니다. 현재 두 object인 dog, person이 존재하는 사진임을 알 수 있습니다. 필수적인 부분은 다음과 같이 정리됩니다.

<folder>	사진이 위치한 폴더를 말합니다.
<filename>	파일의 이름입니다.
<source> <database>	Data를 어디에서 가져왔느냐 하는 것입니다.
<size> <width>, <height>, <depth>	이미지의 크기입니다.
<segmented>	0으로 설정합니다.
<object> <name>	분류한 class의 이름입니다. 저희는 person만을 사용합니다.
<object> <pose>	인물의 자세나 표정 등을 나타내는 것 같습니다. 사용하지 않으므로 Unspecified로 해주세요.
<object> <truncated>	0으로 설정합니다.
<object> <difficult>	0으로 설정합니다.
<object> <bndbox> <xmin>, <ymin>, <xmax>, <ymax>	Bounding box 직사각형의 네 꼭지점을 나타냅니다.

FDDB의 TXT -> VOC 형태의 XML 변환 작업

이제 FDDB의 Annotation을 VOC 형태의 XML로 변환해봅시다.

FDDB의 Annotation은 txt 파일이고 이미지의 width, height를 가지고 있지 않습니다. 이에 따라 Annotation뿐 아니라 width, height를 알기 위한 Image 파일이 함께 제공됩니다.

Dataset-FDDB-Annotation에는 많은 txt 파일들이 있습니다. 아래 사진 1은 어떤 이미지에 대한 txt 파일을 열었을 때의 모습입니다. 우선 줄이 두 개라는 것은 그 이미지 안에 두 개의 얼굴이 있다는 것을 뜻해요.

```
0 0.348145471642 0.329012226667 0.285596352239 0.349046266667
0 0.813598597015 0.187957902222 0.24627881791 0.298561662222
```

사진 1. 2002_07_19_img_135.txt

그리고 각 얼굴 당 다섯 개의 필드를 가지고 있는데, 각 필드는 다음과 같은 의미를 가집니다.

{constant,

box의 center_x(사진 2에서 묘사됨),

box의 center_y(사진 2에서 묘사됨),

box의 width,

box의 height}

이때 유의해야 할 점은 center_x, center_y, width, height가 픽셀을 나타내는 것이 아니라 이미지 전체의 width, height에 대한 비율이라는 것입니다. 그래서 이미지 전체의 width, height를 반드시 알아야 합니다. Jpeg header에서 width, height를 가져오는 C 코드는 다음 링크로 들어가면 보실 수 있습니다. <https://web.archive.org/web/20131016210645/http://www.64lines.com/jpeg-width-height>

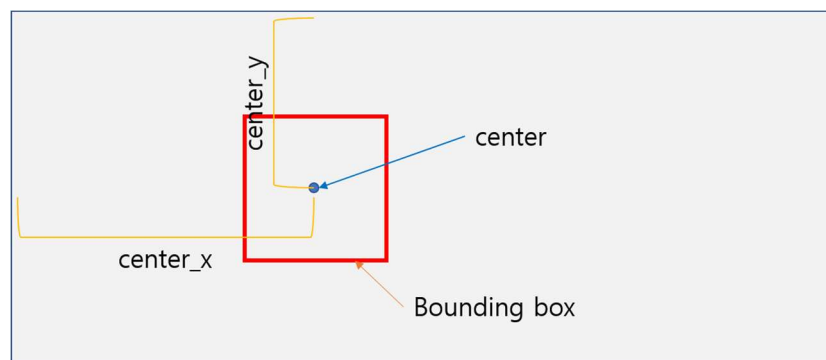


사진 2. Bounding box의 center_x와 center_y

이제 XML을 만들어 봅시다. 가령 2002_07_19_img_135.txt의 XML을 만들려고 합니다. 참고로 2002_07_19_img_135.jpg는 335*450의 크기를 가집니다.

XML에는 box의 xmin, ymin, xmax, ymax가 integer 형태로 들어갑니다. 각각은 이미지 전체의 width(=W), height(=H), 그리고 txt file의 center_x, center_y, width(w), height(h)으로 나타낼 수 있습니다.

$$xmin = W \left(center_x - \frac{w}{2} \right)$$

$$ymin = H \left(center_y - \frac{h}{2} \right)$$

$$xmax = W \left(center_x + \frac{w}{2} \right)$$

$$ymax = H \left(center_y + \frac{h}{2} \right)$$

이를 이용하여 2002_07_19_img_135.txt를 XML로 변환해보면 다음과 같습니다. 이미지마다 달라져야 하는 부분을 볼드 처리하였습니다. 열심히 계산하였지만 구체적인 숫자는 틀릴 수 있음을 알려드립니다.

```
<?xml version="1.0"?>
<annotation>
  <folder>Image</folder>
  <filename>2002_07_19_img_135.jpg</filename>
  <source>
    <database>Fddb</database>
  </source>
  <size>
    <width>335</width>
    <height>450</height>
    <depth>3</depth>
  </size>
  <segmented>0</segmented>
  <object>
    <name>person</name>
    <pose>Unspecified</pose>
    <truncated>0</truncated>
    <difficult>0</difficult>
    <bndbox>
      <xmin>69</xmin>
      <ymin>69</ymin>
      <xmax>164</xmax>
      <ymax>227</ymax>
    </bndbox>
  </object>
</annotation>
```

```
</object>
<object>
  <name>person</name>
  <pose>Unspecified</pose>
  <truncated>0</truncated>
  <difficult>0</difficult>
  <bndbox>
    <xmin>231</xmin>
    <ymin>18</ymin>
    <xmax>314</xmax>
    <ymax>152</ymax>
  </bndbox>
</object>
</annotation>
```