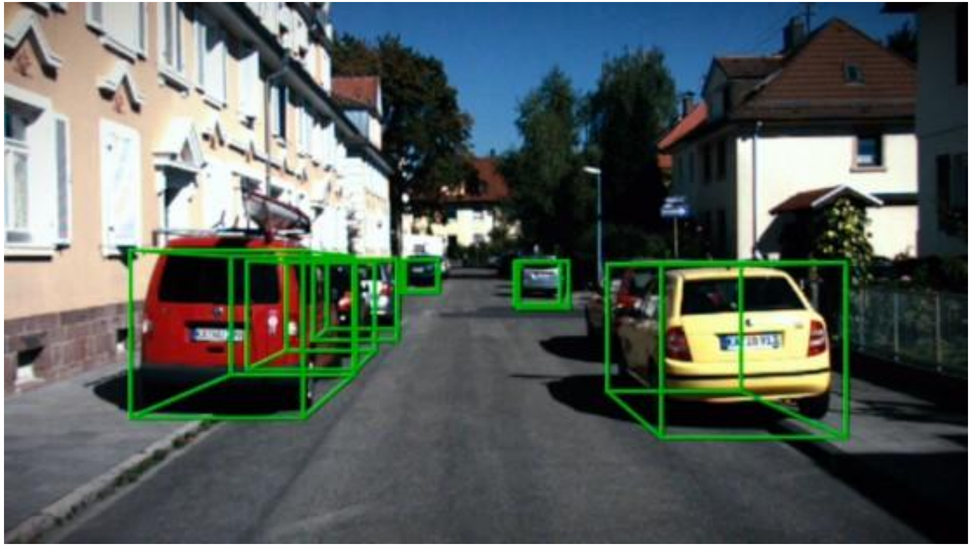
3D Bounding Box

Using Deep Learning and Geometry

이지민

3D Object Detection Problem의 중요성

- 실용성 - decision making / interactions with real world
  - 자율주행 자동차



3D Object Detection Problem에 대하여

- 3D object detection: pose + dimensions
  - Pose = position + orientation
- 효과적인 2D detection algorithms과 달리 open problem

State-of-the-art 3D Object Detectors

- Appearance & viewpoint function
    - Geometric approach 기하학적 접근
    - *Perspective n-point problem (PnP)*
    - 2D keypoints을 활용해서 3D model을 construct 하는 방법

State-of-the-art 3D Object Detectors

- Sampling + scoring of all hypotheses
  - 많은 3D models를 construct한 후 그 중에서 가장 object와 잘 맞는 모델을 고르는 방법
  - Existing 3D models를 활용하는 방법

State-of-the-art 3D Object Detectors

- Deep convolutional neural networks (CNN)이 2D object detection의 성능을 향상 시킴
  - 3D pose estimation에도 적용
- R-CNN detects objects → regions를 pose estimation network으로 보냄

논문에서 발표한 Method

- Pose를 directly regress하는 이전 방법들과 달리 object orientation과 dimension을 regress한 후 pose estimation하는 방법
- 2D object detection을 활용해서 3D bounding box를 구현
- 높은 실용성

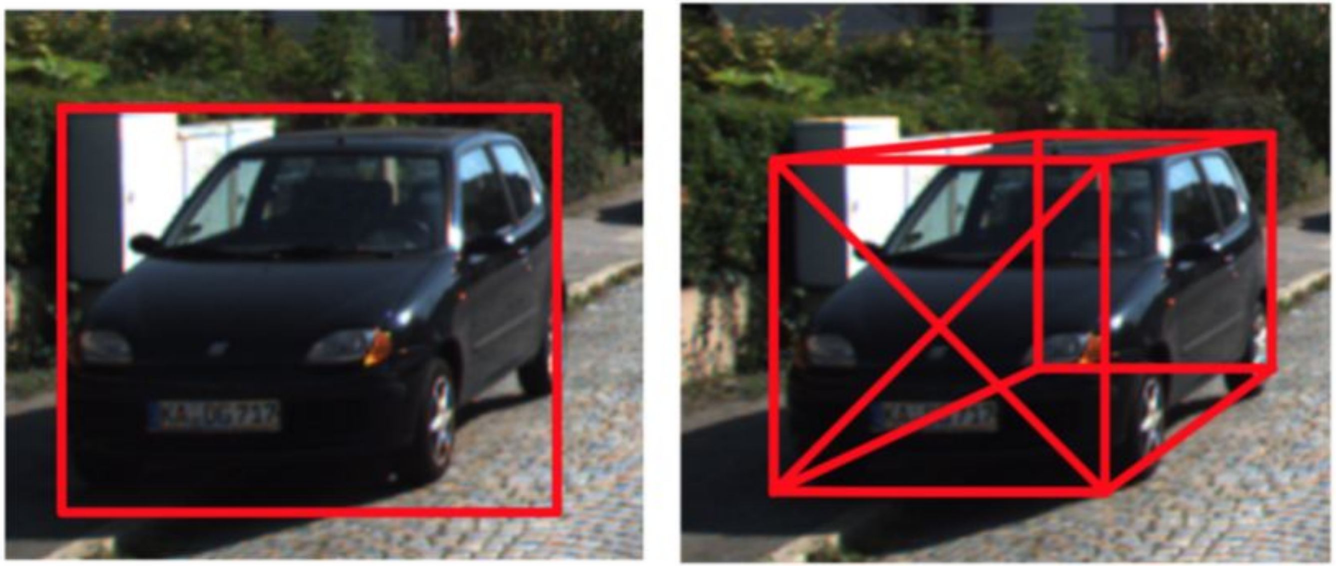


Figure 1. Our method takes the 2D detection bounding box and estimates a 3D bounding box.

3D Bounding Box Estimation

- 3D bounding box descriptors
  - center $T = [t_x, t_y, t_z]^T$
  - dimensions $D = [d_x, d_y, d_z]$
  - orientation $R(\theta, \phi, \alpha)$
    - Azimuth angle: angle relative to the north (compass bearing)
    - Elevation angle: angle formed by the line of sight and the horizontal plane for an object above the horizontal
    - Roll angle: angular displacement of a vehicle about its longitudinal axis

3D Bounding Box Estimation

- Pose, camera coordinate frame (rotation R, translation T), camera intrinsics matrix K가 주어졌을 때

$$\text{3D point } \mathbf{X}_o = [X, Y, Z, 1]^T$$

를 2D인 $\text{image } \mathbf{x} = [x, y, 1]^T$

로 project 하는 공식: $\mathbf{x} = K \begin{bmatrix} R & T \end{bmatrix} \mathbf{X}_o$

$$\mathbf{P} = \begin{bmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{ccc:c} r_1 & r_2 & r_3 & t_1 \\ r_4 & r_5 & r_6 & t_2 \\ r_7 & r_8 & r_9 & t_3 \end{array}\right]$$

intrinsic parameters

extrinsic parameters

$$\mathbf{R} = \begin{bmatrix} r_1 & r_2 & r_3 \\ r_4 & r_5 & r_6 \\ r_7 & r_8 & r_9 \end{bmatrix} \qquad \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

3D rotation

3D translation

3D Bounding Box Estimation

- Box center을 origin으로 삼고 dimensions *D*가 주어졌다는 가정하에 3D bounding box vertices는 아래와 같은식으로 표현할 수 있음

$$\mathbf{X}_1 = [d_x/2, d_y/2, d_z/2]^T$$

3D Bounding Box Estimation

- 3D bounding box가 2D detection window안에 꽉차게 들어감
  - 2D bounding box의 각 면에는 최소 하나의 projected 3D box corner이 있음
- 따라서 4 constraints가 생김
  - 남은 degrees of freedom은 geometric properties와 visual appearance로 생성

3D Bounding Box Estimation - Choice of Regression Parameters

- 3D bounding box에 영향을 많이 끼치는 parameters 중 하나는 orientation around axes $R(\theta, \phi, \alpha)$
- Box dimensions D도 regress함
  - Dimensions D는 variance가 적고 object의 orientation과 관계없음
- Translation T를 regress했을 경우 정확도가 떨어져서 regress안 함
- 따라서 orientation과 dimensions를 regress함

3D Bounding Box Estimation - Correspondence Constraints

- Regressed orientations과 dimensions를 사용해서 translation *T*를 계산함
- 2D box의 4개의 면이 3D box의 4개 중 8개의 모서리와 일치 → computationally fast
- 차량은 대부분 upright 방향 → 2D box의 윗면과 아랫면이 3D box의 윗부분과 아랫부분과 일치
- Relative object roll이 0이랑 가까울때 2D box 양쪽 옆면들이 3D vertical 면들과 일치, 윗면과 아랫면도 마찬가지로 3D horizontal 면들과 일치

Figure 2. Correspondence between the 3D box and 2D bounding box: Each figure shows a 3D bbox that surrounds an object. The front face is shown in blue and the rear face is in red. The 3D points that are active constraints in each of the images are shown with a circle (best viewed in color).

CNN Regression of 3D Box Parameters - *MultiBin* Orientation Estimation

- Global object orientation을 estimate하기 위해서는 전체적인 image에 속해있는 camera reference frame의 location이 필요함

Figure 4. Left: cropped image of a car passing by. Right: Image of whole scene. As it is shown the car in the cropped images rotates while the car direction is constant among all different rows.

CNN Regression of 3D Box Parameters - *MultiBin* Orientation Estimation

- 따라서 local orientation으로 regress함
- Local orientation angle과 ray change angle이 합쳐져서 constant global orientation을 만듦

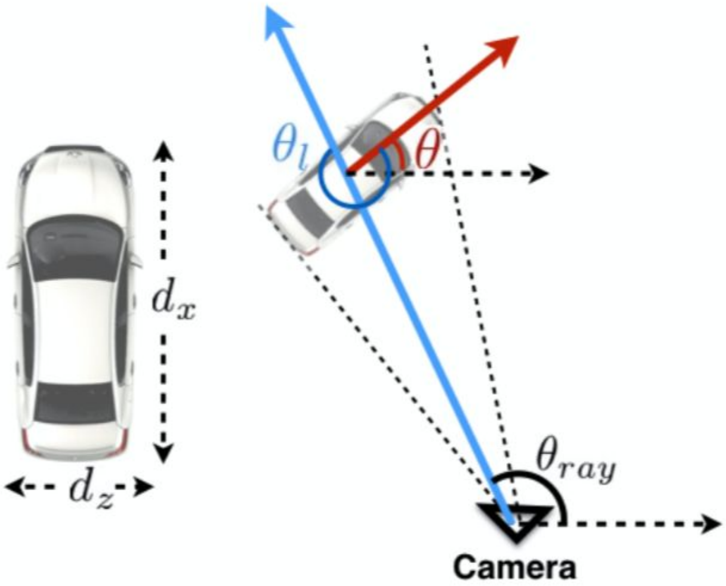


Figure 3. Left: Car dimensions, the height of the car equals d_y. Right: Illustration of local orientation θ_l, and global orientation of a car θ. The local orientation is computed with respect to the ray that goes through the center of the crop. The center ray of the crop is indicated by the blue arrow. Note that the center of crop may not go through the actual center of the object. Orientation of the car θ is equal to $\theta_{ray} + \theta_l$. The network is trained to estimate the local orientation θ_l.

CNN Regression of 3D Box Parameters - *MultiBin* Orientation Estimation

- L2 loss는 multi-modal regression problems에 부적합함
  - 모든 mode의 average loss를 줄이는 방식은 오히려 각각 mode estimate의 정확도를 떨어뜨림
- Faster R-CNN과 SSD도 마찬가지로 bounding box를 directly regress하지 않음
  - 대신 anchor boxes을 활용

CNN Regression of 3D Box Parameters - *MultiBin* Orientation Estimation

- *MultiBin* architecture도 anchor box와 비슷한 원리로 orientation angle를 먼저 discretize 시킨 다음 *n* overlapping bins로 나눔
- CNN이 각 bin마다 output angle이 i^{th} bin 안에 있을 confidence probability c_i와 residual rotation correction을 estimate함
- Bin 마다 output 형식은 따라서 $(c_i, \cos(\Delta\theta_i), \sin(\Delta\theta_i))$

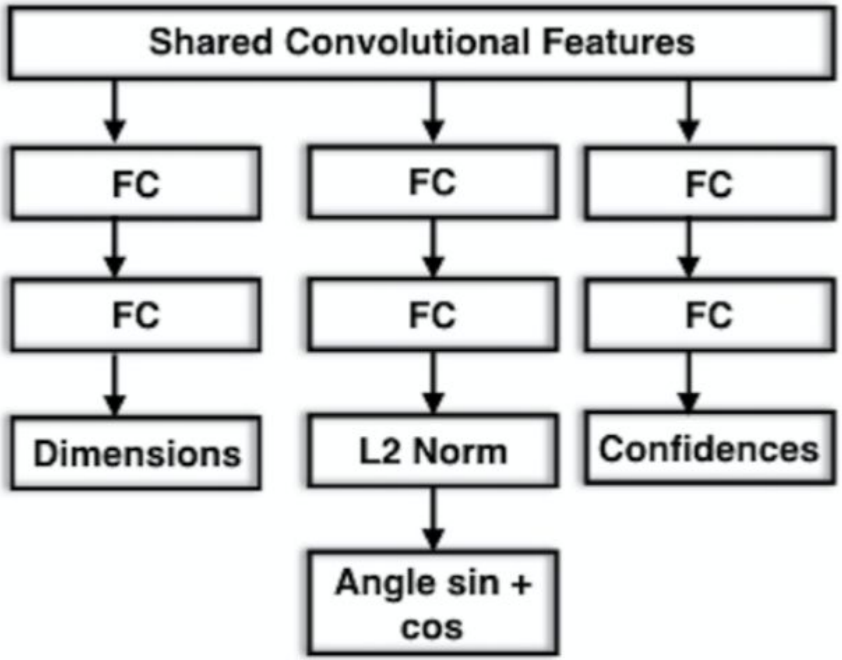


Figure 5. Proposed architecture for MultiBin estimation for orientation and dimension estimation. It consists of three branches. The left branch is for estimation of dimensions of the object of interest. The other branches are for computing the confidence for each bin and also compute the $cos(\Delta\theta)$ and $sin(\Delta\theta)$ of each bin

Limitations

- 다른 method들에 비해 training이 더 필요함

Works Cited

Mousavian, Arsalan, et al. "3D Bounding Box Estimation Using Deep Learning and Geometry." ArXiv, Cornell University, 10 Apr. 2017, https://arxiv.org/abs/1612.00496.