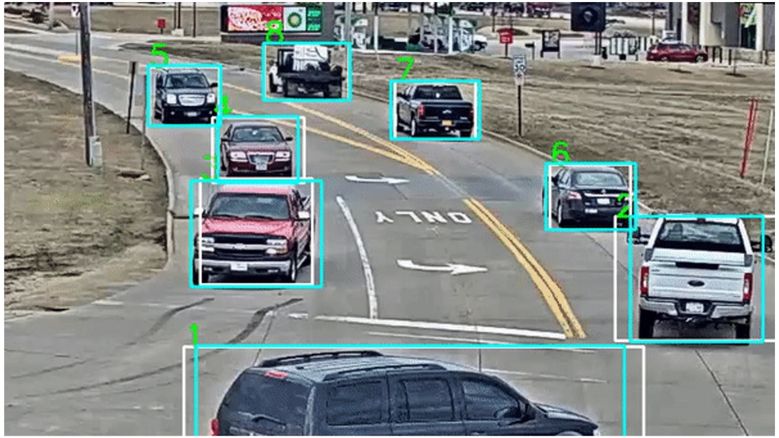
CenterTrack: Tracking Objects as Points

Xingyi Zhou[1], Vladlen Koltun[2], Philipp Krähenbühl[1]
[1]UT Austin, [2]Intel Labs

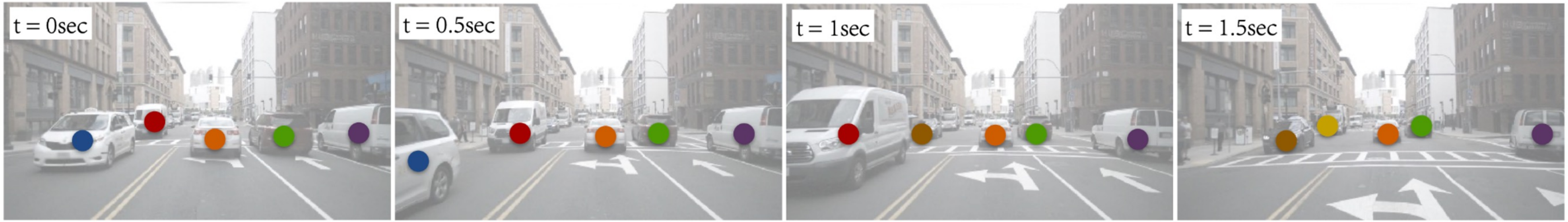
Presented By
Nicholas Almy, Lorry Zou

Tracking as a Video Recognition Task



- One of the primary video recognition tasks
- As of 2020: Tracking-by-Detection is an Dominant Technique in field
    - Detect objects with deep learning models, then track the results
- Problem: Detection networks and algorithms are Inefficient, Complicated, and Costly
- Is it possible to simplify and streamline this process?

Motivation



Simultaneous, online detection of objects as a points is simple, and leads to effective tracking

Related Works

- Tracking-by-detection: SORT, DeepSORT
- Joint detection and tracking: Tracktor, **CenterTrack**
- Motion prediction: Kalman filter, etc.
- Heatmap-conditioned keypoint estimation
- 3D object detection and tracking

CenterNet: *Objects as Points* (Zhou et al., 2019)

- Anchor-free, single-point prediction (center point)
- Run at very high speed

    1. ResNet-18: 142 FPS on MSCOCO
    2. DLA-34: 52 FPS on MSCOCO
    3. Hourglass-104: 1.4 FPS on MSCOCO

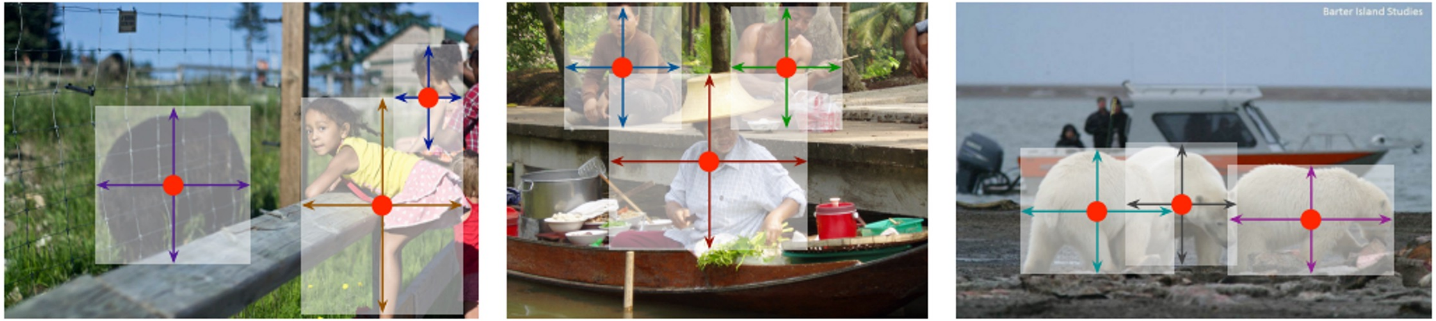


Figure 2: We model an object as the center point of its bounding box. The bounding box size and other object properties are inferred from the keypoint feature at the center. Best viewed in color.

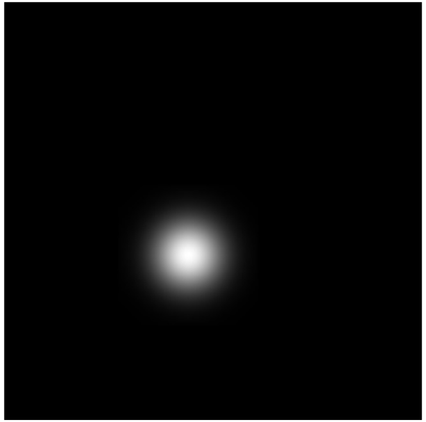
CenterNet: Objects as Points

$$Y \in [0,1]^{\frac{W}{R} \times \frac{H}{R} \times C} \quad => \quad Y_{xyc} = \exp\left(-\frac{(x-\tilde{p}_x)^2+(y-\tilde{p}_y)^2}{2\sigma_p^2}\right)$$

1. C is the number of classes
2. R is the downsampling factor (R=4 in the paper)

(a) Standard anchor based detection. Anchors count as positive with an overlap $IoU > 0.7$ to any object, negative with an overlap $IoU < 0.3$, or are ignored otherwise.

(b) Center point based detection. The center pixel is assigned to the object. Nearby points have a reduced negative loss. Object size is regressed.



A heatmap

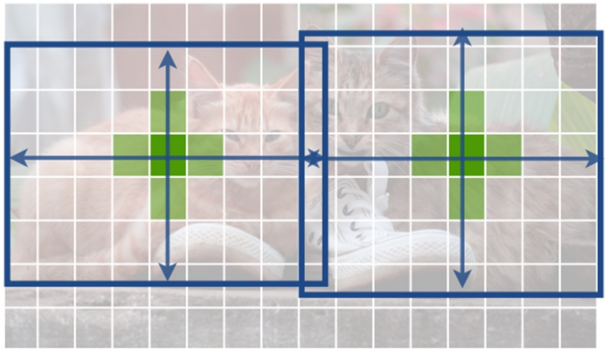
CenterNet: Focal Loss Function

$$L_k = \frac{-1}{N} \sum_{xyc} \begin{cases} (1 - \hat{Y}_{xyc})^{\alpha} \log(\hat{Y}_{xyc}) & \text{if } Y_{xyc} = 1 \\ (1 - Y_{xyc})^{\beta} (\hat{Y}_{xyc})^{\alpha} \\ \log(1 - \hat{Y}_{xyc}) & \text{otherwise} \end{cases}$$

1. N is the number of keypoints
2. α=2, β=4 in the paper

If Y_{xyc} **≠ 1:**

1. $\hat{Y}_{xyc} \rightarrow 0$, $Y_{xyc} \rightarrow 0$, **low * high**
2. $\hat{Y}_{xyc} \rightarrow 0$, $Y_{xyc} \rightarrow 1$, **low * low**
3. $\hat{Y}_{xyc} \rightarrow 1$, $Y_{xyc} \rightarrow 1$, **high * low**
4. $\hat{Y}_{xyc} \rightarrow 1$, $Y_{xyc} \rightarrow 0$, **high * high**



CenterNet: Loss Functions

Offset Loss: $$L_{off} = \frac{1}{N} \sum_{p} \left| \hat{O}_{\tilde{p}} - \left(\frac{p}{R} - \tilde{p} \right) \right|.$$

Size Loss: $$L_{size} = \frac{1}{N} \sum_{k=1}^{N} \left| \hat{S}_{pk} - s_k \right|$$

Overall Loss: $$L_{det} = L_k + \lambda_{size} L_{size} + \lambda_{off} L_{off}.$$

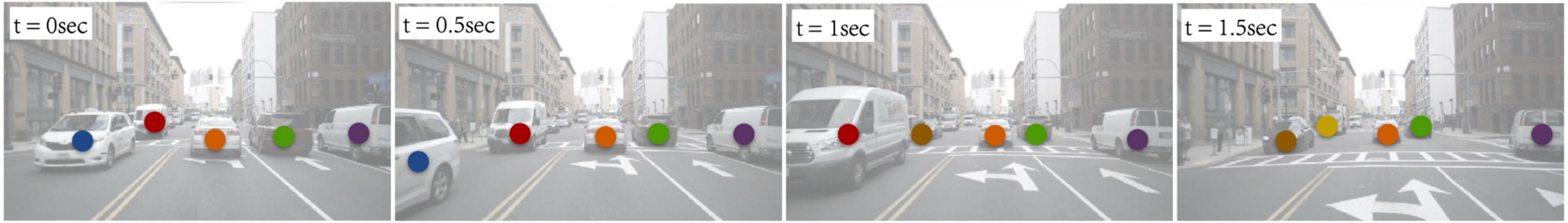
Output: [1,**C**,128,128]: number of classes

[1,**2**,128,128]: offsets in x and y directions

[1,**2**,128,128]: height and width

CenterTrack

- Utilizing CenterNet on video, information about objects' movements in space emerge
- Find an object in space and find it's previous position online



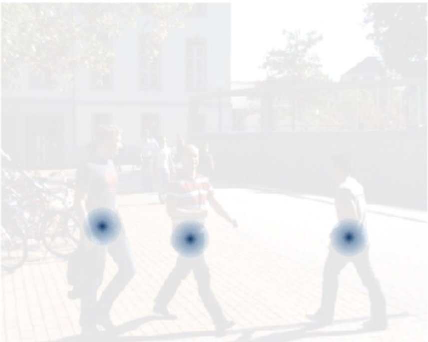
CenterTrack: Model

For a Current Timestep *t*

Frame *t*

Frame *t-1*

Detections (Frame *t-1)*

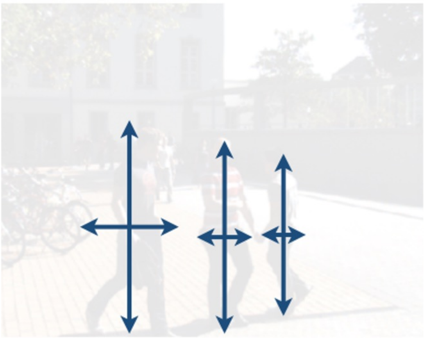


CenterTrack: Model

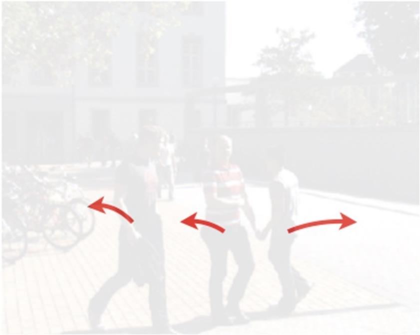
For a Current Timestep *t*

Detections *t*

Sizes *t*



Detections (Frame *t-1)*



CenterTrack: Model

In time t, Each Object, b, has 4 Traits:

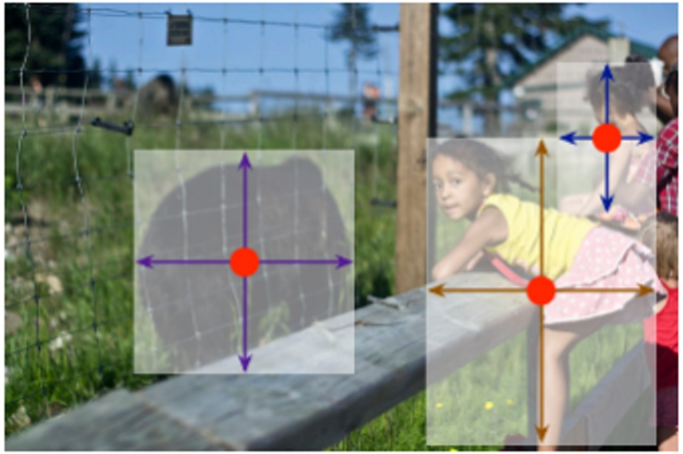
b = (**p**, **s**, w, *id*)

p: Position of Heatmap center

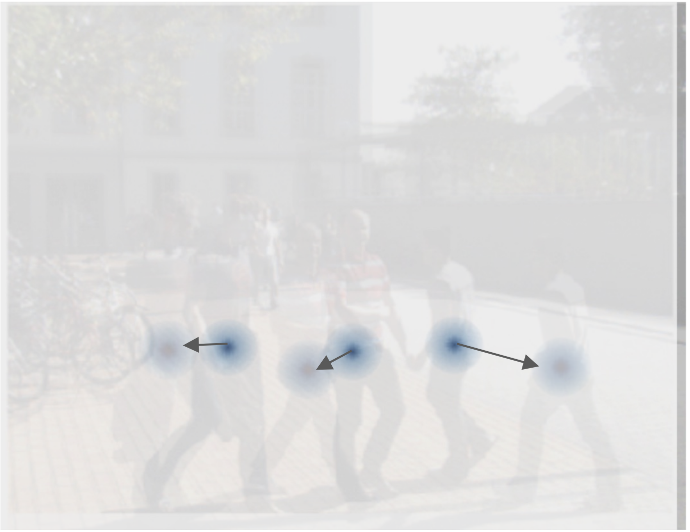
s: Size vector, for boundary box calculation

w: Confidence Interval

id: Identification Integer. If an object is found to be the same in both frames t and t-1, id shall remain the same



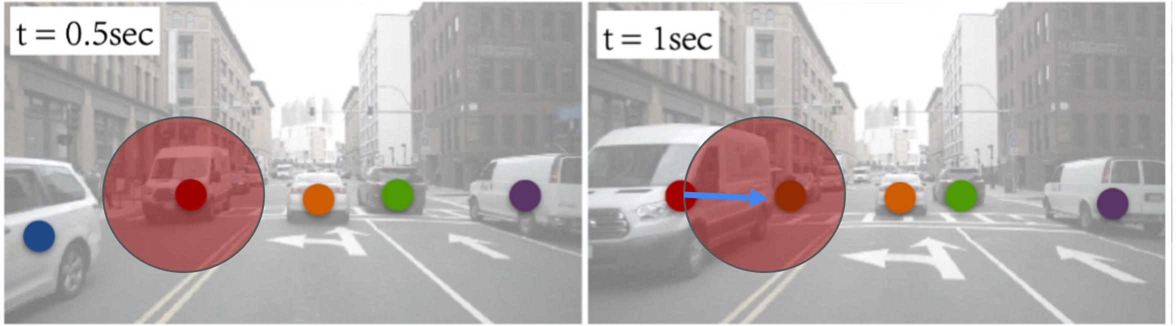
Tracking and Offset



- Point, size detection handled by CenterNet base model
- For Tracking, regress two new channels for Displacement, Represented by D-hat

$$L_{off} = \frac{1}{N} \sum_{i=1}^{N} \left| \hat{D}_{\mathbf{p}_i^{(t)}} - (\mathbf{p}_i^{(t-1)} - \mathbf{p}_i^{(t)}) \right|$$

Associating Two Objects



Greedy Approach to linking Objects:

- Using w-hat (object confidence interval) to determine order, Find closest p_{t-1} to p_t - D-hat
- If No p is in a radius *k*, a new tracklet is formed

Training

At inference time on video, we must be careful with Dropped tracklets, False Positive Detections, and Incorrectly localized objects

When Training on video, measures are taken to mitigate these issues:

- Jittering Applied to heatmap for resilient localization
- Adding unexpected hotspots in Image t (with distribution λ_{fp})
- Removing expected tracklets in Image t (with distribution λ_{fn})

On Image data: Images are randomly translated so that objects have known offsets and positions

On 3D Data: More D-hat channels to predict depth and rotation, addition of a 2D-3D Offset

Experiments

Testing and Experimentation done on MOT17, KITTI, and nuScenes

Training Specifications:

- Learning Rate 1.25e - 4
- Batch Size 32
- 70 Epochs: 60 at LR above, last 10 have LR dropped by a factor of 10
- Intel i7-8086k CPU, Titan Xp GPU

Evaluation Metrics Used: MOTA, AMOTA

$$MOTA = 1 - \frac{\sum_t (FP_t + FN_t + IDSW_t)}{\sum_t GT_t}$$

$$AMOTA = \frac{1}{n-1} \sum_{r \in \{\frac{1}{n-1}, \frac{2}{n-1}, \cdots, 1\}} MOTA_r$$

$$MOTA_r = max(0, 1 - \alpha \frac{IDSW_r + FP_r + FN_r - (1-r) \times P}{r \times P})$$

MOT17

- Comparison with both Public Detection (objects tracked are already found) and private (CenterTrack finds objects)
- Trained on CrowdHumans dataset
- Output tracklets with Confidence θ=0.4
- Added Hyperparameter K=32 for Tracking Rebirth
- Image input size downscaled from 1920x1080 → 960x544
- λ_{fp}: 0.1
- λ_{fn}:0.4

KITTI

- Original Resolution is used for Image input: 1280x384
- Finetuned from a nuScenes trained Tracking model
- λ_{fp}: 0.1
- λ_{fn}:0.2
- θ: 0.4

nuScenes

- Original Resolution is used for Image input: 800 x 448
- For 3D Tracking:
  - Trained for 140 Epochs
- Image/Video data is 360 degree Panorama
  - To deal with this, detect each composite image independently and naively fuse all detections.
  - Ignores Cases when objects are between two images
- λ_{fp}: 0.1
- λ_{fn}:0.4
- θ: 0.1

Results on MOT17

	Time(ms)	MOTA ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDSW ↓
Tracktor17 [1]	666+D	53.5	52.3	19.5	36.6	12201	248047	2072
LSST17 [10]	666+D	54.7	**62.3**	20.4	40.1	26091	228434	**1243**
Tracktor v2 [1]	666+D	56.5	55.1	21.1	35.3	**8866**	235449	3763
GMOT	167+D	55.4	57.9	22.7	34.7	20608	229511	1403
Ours (Public)	**57+D**	**61.5**	59.6	**26.4**	**31.9**	14076	**200672**	2583
Ours (Private)	57	67.8	64.7	34.6	24.6	18498	160332	3039

Table 1: Evaluation on the MOT17 test sets (top: public detection; bottom: private detection). We compare to published entries on the leaderboard. The runtime is calculated from the HZ column on the leaderboard. +D means detection time, which is usually $>$ 100ms [31].

Results on KITTI

	Time(ms)	MOTA ↑	MOTP ↑	MT ↑	ML ↓	IDSW ↓	FRAG ↓
AB3D [46]	4+D	83.84	85.24	66.92	11.38	**9**	**224**
BeyondPixel [35]	300+D	84.24	85.73	73.23	2.77	468	944
3DT [14]	30+D	84.52	85.64	73.38	2.77	377	847
mmMOT [54]	10+D	84.77	85.21	73.23	2.77	284	753
MOTSFusion [27]	440+D	84.83	85.21	3.08	2.77	275	759
MASS [18]	10+D	85.04	**85.53**	74.31	2.77	301	744
Ours	82	**89.44**	85.05	**82.31**	**2.31**	116	334

Table 2: Evaluation on the KITTI test set. We compare to all published entries on the leaderboard. Runtimes are from the leaderboard. +D means detection time.

Results: Ablation Studies

	MOT17				KITTI				nuScenes	
	MOTA↑	FP↓	FN↓	IDSW↓	MOTA↑	FP↓	FN↓	IDSW↓	AMOTA@0.2↑	AMOTA@1↑
detection only	63.6	3.5%	30.3%	2.5 %	84.3	4.3%	9.8%	1.5%	18.1	3.4
w/o offset	65.8	**4.5%**	**28.4%**	1.3%	87.1	**5.4%**	**5.8%**	1.6%	17.8	3.6
w/o heatmap	63.9	3.5%	30.3%	2.3%	85.4	4.3%	9.8%	0.4%	26.5	5.9
Ours	**66.1**	**4.5%**	**28.4%**	**1.0%**	**88.7**	**5.4%**	**5.8%**	**0.1%**	**28.3**	**6.8**

Table 4: Ablation study on MOT17, KITTI, and nuScenes. All results are on validation sets (Section 5.1). For each dataset, we report the corresponding official metrics. ↑ indicates that higher is better, ↓ indicates that lower is better.

Summary

- An end-to-end simultaneous object detection and tracking framework
- Largely based on CenterNet
- It outperforms state-of-the-arts in both run time and MOTA on MOT17, KITTI, and nuScenes benchmarks