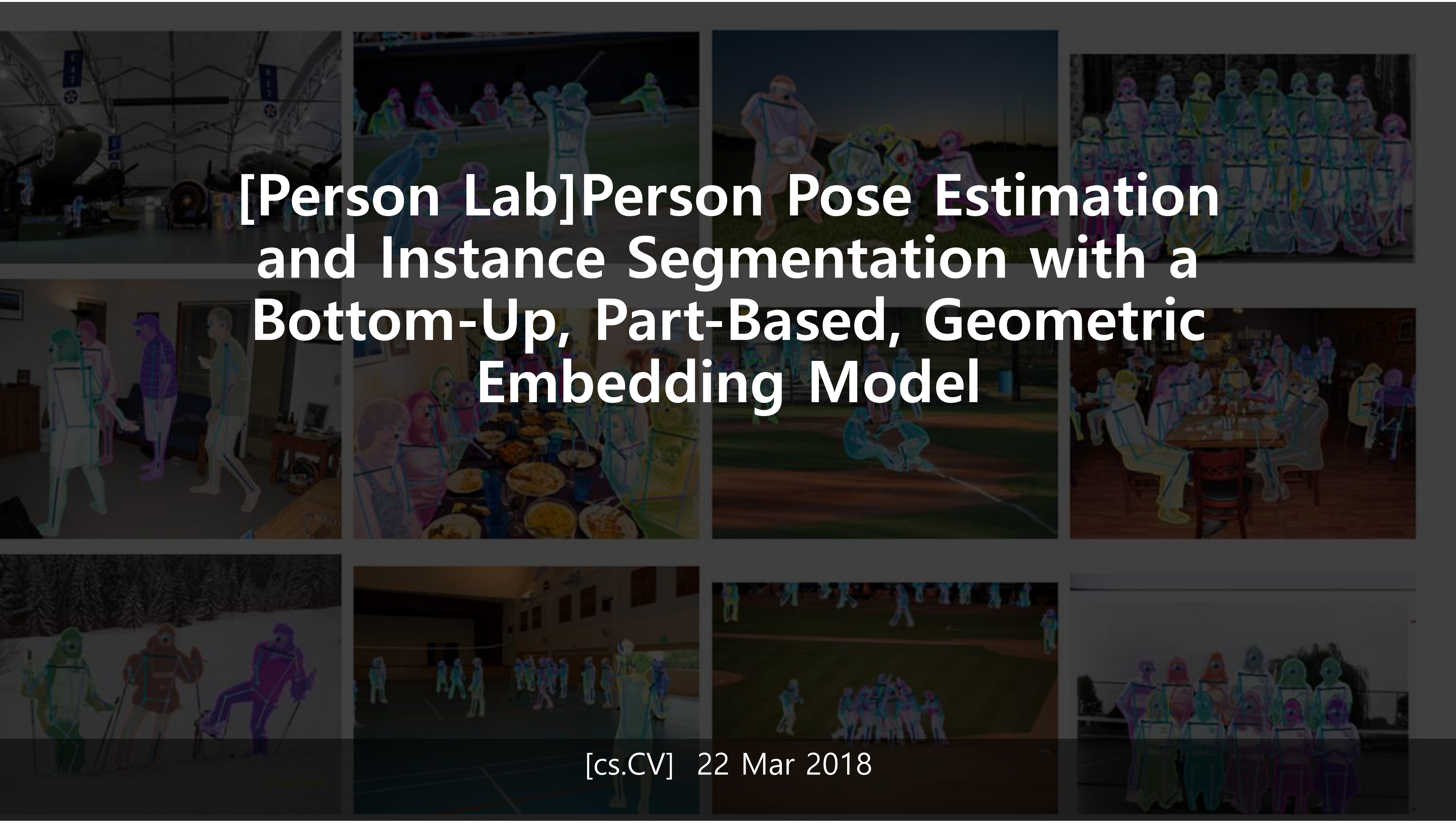
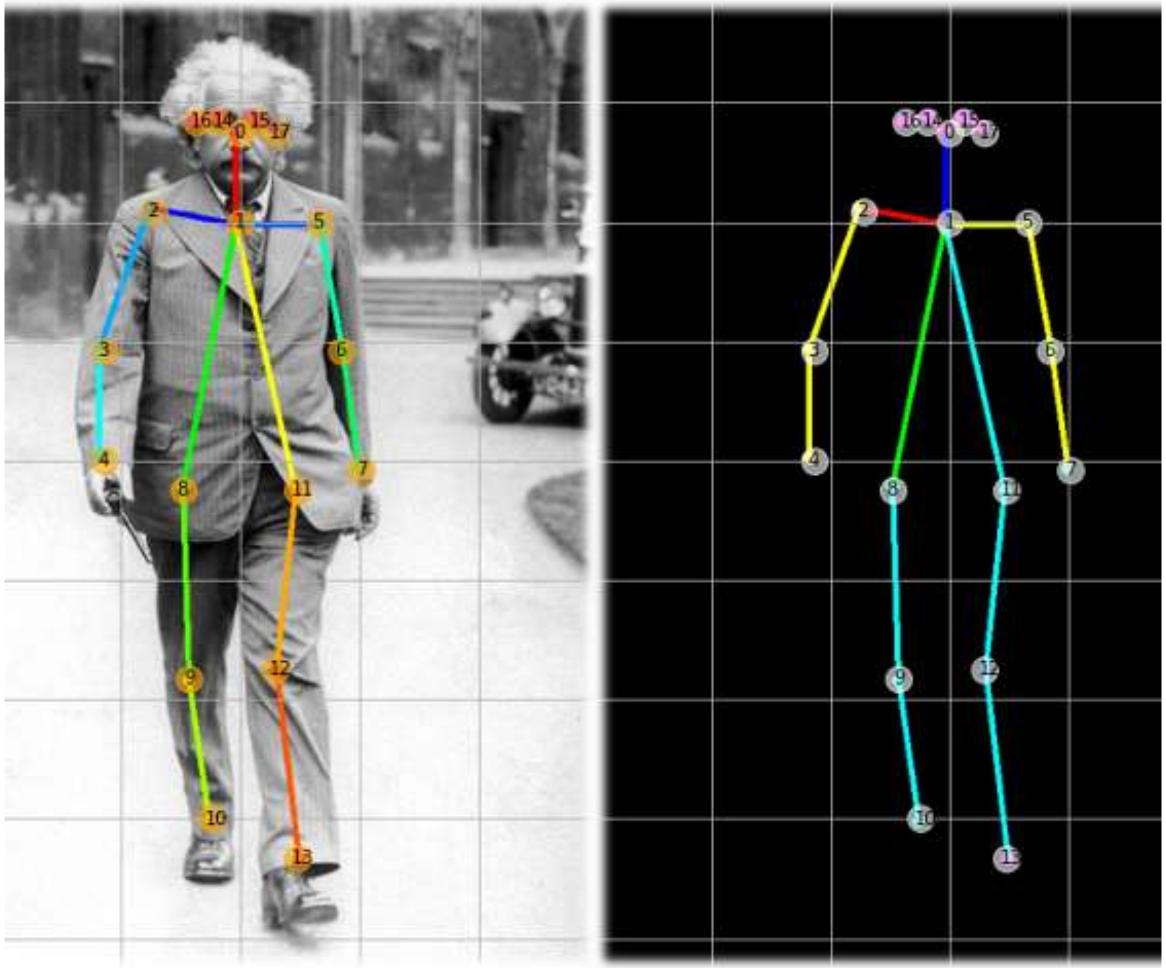


[Person Lab]Person Pose Estimation and Instance Segmentation with a Bottom-Up, Part-Based, Geometric Embedding Model

[cs.CV] 22 Mar 2018

Pose Estimation?

- 이미지 내의 신체(포즈)의 구조를 추정하는 프로세스



Instance Segmentation with Top-down vs Bottom-Up

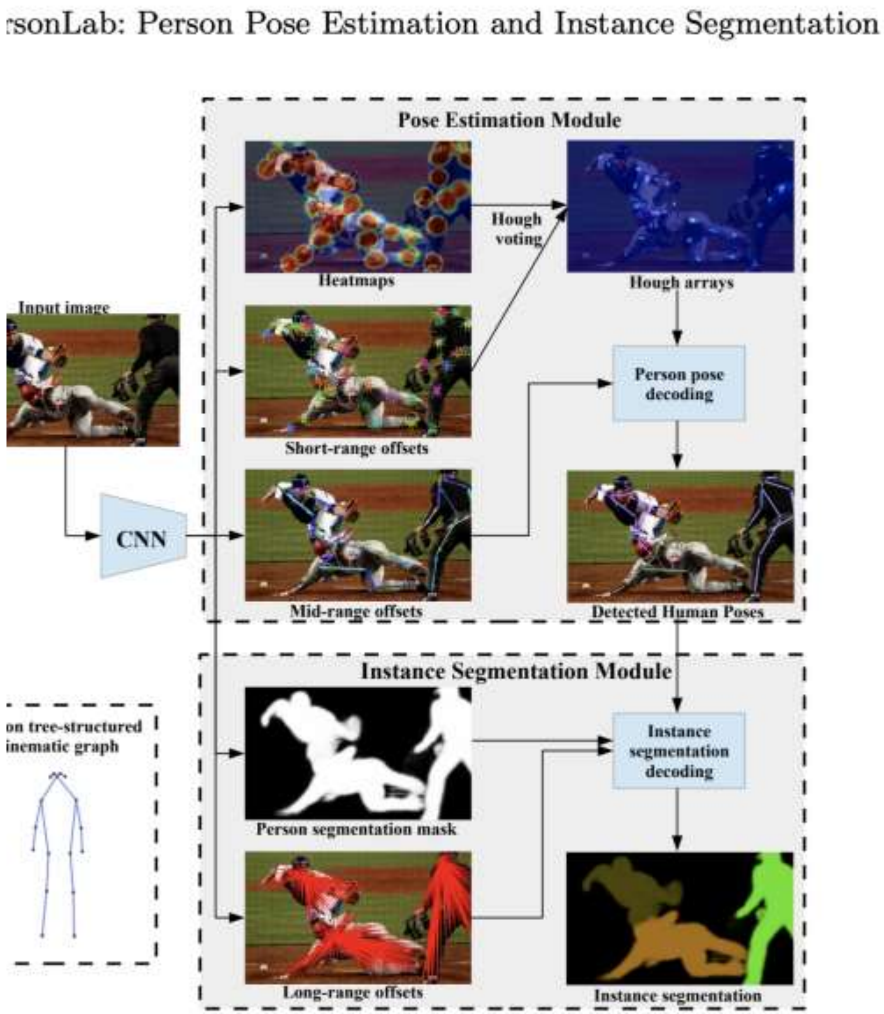
- Box object detector를 이용하여 이미지상에서 사람 Instance를 먼저 찾아낸다.
- 하나의 인물이 있는 사진이거나 Box object detector안에 있는 사람에 대해서만 적용했다.
- 사람의 수에 따라 계산 비용이 달라진다.

- 사람 Instance가 기준이 아닌 개념적으로 신체의 각 keypoint로 각 파트별로 인지하는 것에서부터 시작된다.
- Box object detector 없이 모든 지점이 연결되어있다.
- 사람의 수가 아닌 오로지 학습 신경망에 의해서 추출된 특성으로 계산비용이 정해진다.

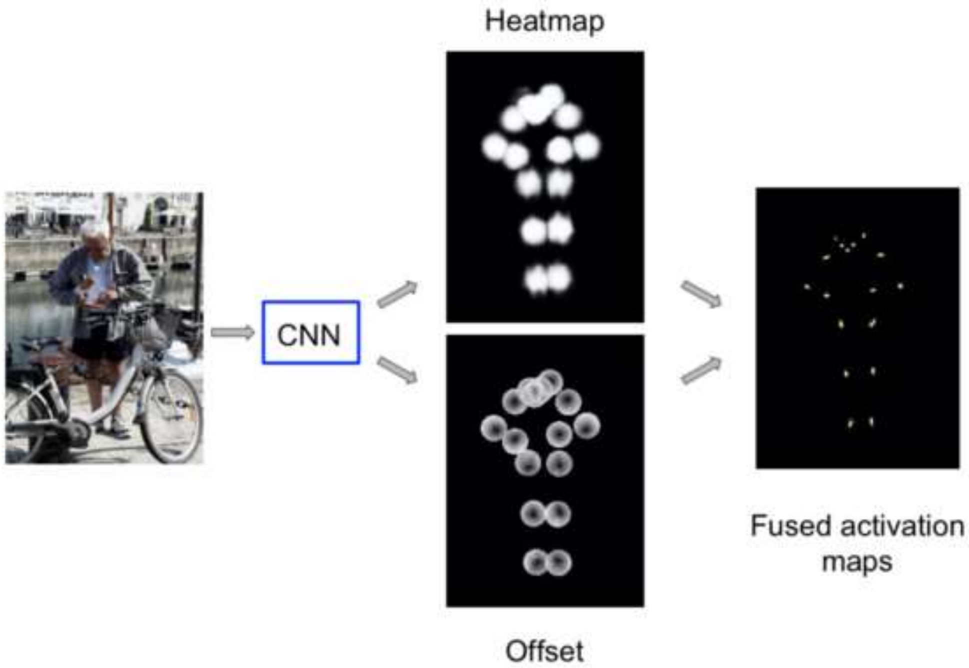
Person detection and Pose estimation (Bottom-up 접근방식)

1. (Pose estimation) 이미지내의 K keypoints를 탐지
    - Short – range Offset
    - Mid – range Offset

2. (Person detection) 인물별 17개 keypoints(얼굴 5, 몸 12)를 통해 Human Instance로 Grouping
    - Long – range Offset



KEYPOINT – DETECTION (Short-range Offset)



Heatmap과 Short-range Offset을 voting하여 명확한 keypoints를 도출해낸다.

$$x \in \mathcal{D}_R(y_{j,k})$$

$$S_k(x) = y_{j,k} - x$$

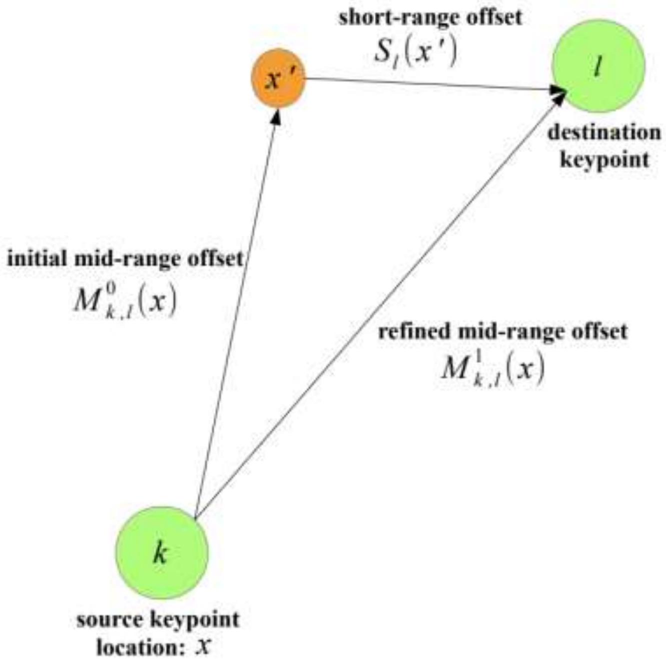
$$h_k(x) = \frac{1}{\pi R^2} \sum_{i=1:N} p_k(x_i) B(x_i + S_k(x_i) - x),$$

KEYPOINT – DETECTION (Mid-range Offset)

학습에 의하여 두 Keypoints의 거리가 한 Instance안에 포함될 기준 내에 있으면 두 Keypoints를 이어준다.

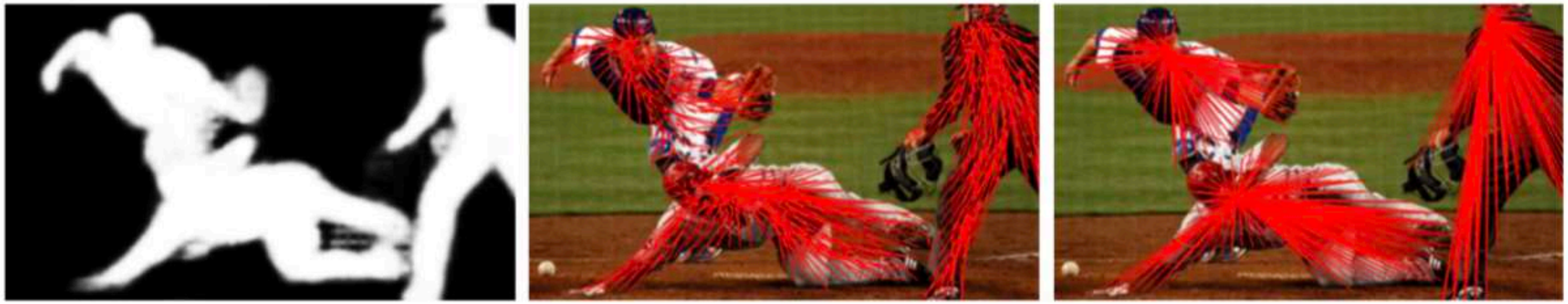
$$M_{k,l}(x) = (y_{j,l} - x)[x \in \mathcal{D}_R(y_{j,k})]$$

Recurrent한 방식을 이용해서 이를 개선



$$M_{k,l}(x) \leftarrow x' + S_l(x')\,, \text{where } x' = M_{k,l}(x)\,,$$

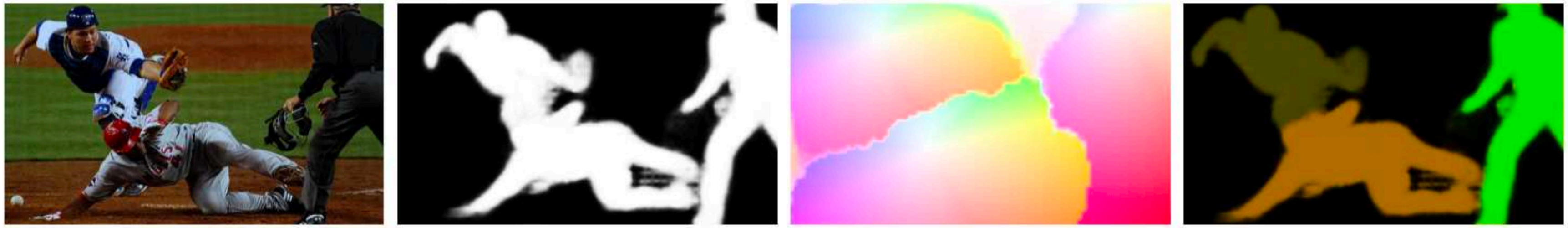
Long-range Offset



(a) (b) (c)

Fig. 3. Long-range offsets defined in the person segmentation mask. (a) Estimated person segmentation map. (b) Initial long range offsets for the *Nose* destination keypoint: each pixel in the foreground of the person segmentation mask points towards the *Nose* keypoint of the instance that it belongs to. (c) Long-range offsets after their refinements with the short-range offsets.

Instance-level person segmentation



(a) (b) (c) (d)

Fig. 4. From semantic to instance segmentation: (a) Image; (b) person segmentation; (c) basins of attraction defined by the long-range offsets to the *Nose* keypoint; (d) instance segmentation masks.

PersonLab: Person Pose Estimation and Instance Segmentation

	AP	$AP^{.50}$	$AP^{.75}$	AP^{M}	AP^{L}	AR	$AR^{.50}$	$AR^{.75}$	AR^{M}	AR^{L}
Bottom-up methods:										
CMU-Pose [32] (+refine)	0.618	0.849	0.675	0.571	0.682	0.665	0.872	0.718	0.606	0.746
Assoc. Embed. [2] (multi-scale)	0.630	0.857	0.689	0.580	0.704	-	-	-	-	-
Assoc. Embed. [2] (mscale, refine)	0.655	0.879	0.777	0.690	0.752	0.758	0.912	0.819	0.714	0.820
Top-down methods:										
Mask-RCNN [34]	0.631	0.873	0.687	0.578	0.714	0.697	0.916	0.749	0.637	0.778
G-RMI *COCO-only* [33]	0.649	0.855	0.713	0.623	0.700	0.697	0.887	0.755	0.644	0.771
PersonLab (ours):										
ResNet101 (single-scale)	0.655	0.871	0.714	0.613	0.715	0.701	0.897	0.757	0.650	0.771
ResNet152 (single-scale)	**0.665**	0.880	0.726	0.624	0.723	0.710	0.903	0.766	0.661	0.777
ResNet101 (multi-scale)	0.678	0.886	0.744	0.630	0.748	0.745	0.922	0.804	0.686	0.825
ResNet152 (multi-scale)	**0.687**	0.890	0.754	0.641	0.755	0.754	0.927	0.812	0.697	0.830

기존 Openpose 등 대비해서 훨씬 높은 성능.

	AP	AP^{50}	AP^{75}	AP^{S}	AP^{M}	AP^{L}	AR^{1}	AR^{10}	AR^{100}	AR^{S}	AR^{M}	AR^{L}
Mask-RCNN [34]	0.455	0.798	0.472	0.239	0.511	0.611	0.169	0.477	0.530	0.350	0.596	0.721
PersonLab (ours):												
ResNet101 (1-scale, 20 prop)	0.382	0.661	0.397	0.164	0.476	0.592	0.162	0.416	0.439	0.204	0.532	0.681
ResNet152 (1-scale, 20 prop)	0.387	0.667	0.406	0.169	0.483	0.595	0.163	0.423	0.446	0.213	0.539	0.686
ResNet101 (mscale, 20 prop)	0.414	0.684	0.447	0.213	0.492	0.621	0.170	0.454	0.492	0.278	0.566	0.728
ResNet152 (mscale, 20 prop)	0.418	0.688	0.455	0.219	0.497	0.621	0.170	0.460	0.497	0.284	0.573	0.730
ResNet152 (mscale, 100 prop)	0.429	0.711	0.467	0.235	0.511	0.623	0.170	0.460	0.539	0.346	0.612	0.741

Human category Segmentation
결과에서도 높은 성능