# Cao, Simon, Wei, Sheikh

Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields

## 1 Summary

The authors present a method for multi-person 2D pose estimation by learning to detect and associate parts in an image. Detection is done via *confidence maps*; association is done via *part affinity fields*. Part affinity fields (PAFs) are 2D vector fields denoting the direction from some part to the next part in the tree, and thereby encode scores for part-to-part connections (*affinity scores*).

i.e. are these two parts connected on one person?

To what degree do the vectors at pixels in the "elbow" → "wrist" area agree with the vector from "elbow" to "wrist?"

Confidence maps and PAFs are iteratively and jointly refined using a convolutional pose machine framework. At the final stage, we can construct full skeletons via a greedy strategy: connect the parts in the tree with the highest affinity scores, repeat until all parts are connected, and then link the connections that share parts.

## 2 Introduction

- Human 2D pose estimation is hard (unknown number of people at arbitrary position and scale, complex interactions, runtime often grows with number of people in scene).
- Previous *top-down* (working from *person down*, as opposed to from *parts up*) approaches detect people and then do single-person pose estimation for each person.
  - but the first stage (person detector) can easily fail, ruining everything else's day, and runtime definitely scales with headcount
- Previous *bottom-down* approaches do not directly use global contextual cues and are often inefficient.
- **This paper's contribution**: a real-time bottom-up approach which encodes global context at a level of quality that allows greedy skeleton construction to achieve state-of-the-art results.
  - represent part association scores via PAFs (*set* of 2D vector fields encoding location and orientation of limbs in image space)
  - simultaneously infer representations for detection and association, thus encoding global context

## 3 Method

- Overall pipeline:
  - input: color image of size $w \times h$
  - the image is first fed to a CNN to generate features $\mathbf{F}$
  - two feedforward CNN branches, which each proceed in stages: $\rho^i$, which takes $\mathbf{F}^*$ and produces $\mathbf{S}^i$ (the $i$th set of confidence maps, one for each part), and $\phi^i$, which takes $\mathbf{F}^*$ and produces $\mathbf{L}^i$ (the $i$th set of part affinity fields, one for each limb)

    * and $\mathbf{S}^{i-1}$, $\mathbf{L}^{i-1}$ if not the first stage
  - after stage $i$, $\mathbf{S}^i$, $\mathbf{L}^i$, and $\mathbf{F}$ are sent to stage $i+1$ for refinement
  - final $\mathbf{S}$ and $\mathbf{L}$ are greedily parsed to produce 2D locations of keypoints for each person in image

## 3.1 Simultaneous Detection and Association

- The network simultaneously predicts *confidence maps* for detection and *affinity fields* for part-to-part association.
- The predictions are refined over multiple stages, with intermediate supervision at each stage.
  - $L_2$ loss between predictions and ground truth confidence maps/PAFs
  - loss functions weighted spatially by a binary mask (only 1 where annotations exist)
  - intermediate supervision replenishes the gradient (avoiding vanishing gradients)

## 3.2 Confidence Maps for Part Detection

- Ground truth confidence map: place a 2D Gaussian around each body part for each person, take the max over all people to get a single map for each body part.
  - max preserves precision around close-by peaks

## 3.3 Part Affinity Fields for Part Association

- A PAF is a 2D vector field which encodes location and **orientation** information across the region of support for a limb; there is a PAF for each type of limb.
  - for each pixel in the area belonging to a particular limb, a 2D vector gives the direction pointing from one part of the limb to the other
- Ground truth affinity vector for limb $c$ at point $\mathbf{p}$: average of unit vectors for each person's limb $c$, where a person is only included if point $\mathbf{p}$ is part of his/her limb $c$.
  - to clarify, each unit vector is in the direction of some person's limb $c$
  - point $\mathbf{p}$ is part of limb $c$ if it is less than some distance away from $c$'s line segment
- Affinity score:

$$E = \int_{u=0}^{u=1} \underbrace{\mathbf{L}_c(\mathbf{p}(u))}_{\substack{\text{affinity vector} \\ \text{at point on} \\ \text{candidate limb}}} \cdot \underbrace{\frac{\mathbf{d}_{j_1} - \mathbf{d}_{j_2}}{\|\mathbf{d}_{j_2} - \mathbf{d}_{j_1}\|_2}}_{\substack{\text{unit vector} \\ \text{from } d_{j_1} \text{ to } d_{j_2}}} du$$

  where $j_1$ and $j_2$ are parts, $\mathbf{d}_{j_i}$ is the candidate location of part $j_i$, $\mathbf{L}_c$ is the affinity field for limb $c$ which joins $j_1$ and $j_2$, $u$ is an interpolation parameter, and $\mathbf{p}(u) = (1-u)\mathbf{d}_{j_1} + u\mathbf{d}_{j_2}$ is an interpolated position between $\mathbf{d}_{j_1}$ and $\mathbf{d}_{j_2}$.
  - interpretation: how well do the orientations of affinity vectors on this proposed limb match the actual orientation of the limb (computed according to the vector from part $j_1$ to part $j_2$)?
  - evaluation: sample uniformly-spaced values of $u$ and sum the corresponding integrand values

## 3.4 Multi-Person Parsing Using PAFs

- First we perform non-maximum suppression on the $\mathbf{S}$'s to get a set of candidate locations for each part.
  - note: there can be multiple people or false positives
  - by extension, we now have a bunch of candidate limbs
- Then we compute the affinity score $E$ for each candidate limb.
  - now, if we were to define nodes for all $j_1$ and $j_2$ detection candidates [*1] and draw an edge of weight ${E_{mn}}^{*2}$ between every $j_1$ candidate and every $j_2$ candidate, finding the full optimal association

would be a maximum weight bipartite graph matching problem

[*1] where $j_1$ and $j_2$ are parts that should be connected
[*2] for $m$ a $j_1$ candidate and $n$ a $j_2$ candidate

  - i.e. we want to make connections such that (a) no two limbs share one part and (b) the sum of all final limbs' $E_{mn}$ values is maximized
- A relaxation: matchings for adjacent parts are computed independently of each other.
- After we have all of the limbs, we can join them all together based on matching detection candidates.

# 4 Results

- Method evaluated on two multi-person pose estimation benchmarks: MPII and COCO 2016.
- Faster and more accurate than previous approaches.

# 5 Discussion

- Takeaways:
  - real-time algorithm to detect 2D pose of multiple people in an image
  - new non-parametric representation of association between keypoints (captures orientation)
  - greedy parsing is sufficient to produce high-quality skeletons; doesn't scale with number of people

# 6 Clarifications

- Sometimes, the predicted skeletons will be incomplete, because a part might not be visible or detected. There might not be a match for everything.
- The same supervision is used at every stage of the network.
- The network architecture equates to global inference because the effective receptive field gets larger over stages, and each stage gets to see all of the detections and associations from the previous stage.
- **S** finds *all* occurrences of a part in an image.
- **L** represents the degree/direction of association at every pixel between part A and part B.
  - If there is an association (i.e. a limb) at a pixel, the 2D vector there is trained to be the average direction (across people) from part $A$ to part $B$. If there is no association (/limb) at a pixel, the vector should be the zero vector.
  - If not normalized, the magnitude of an output vector should represent the confidence that the pixel is on a limb. However, the network is trained to output unit vectors, and they might be normalized anyway, since we are mainly supposed to care about the *direction* at each location.

# References

[1] Zhe Cao and Tomas Simon and Shih-En Wei and Yaser Sheikh. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *arXiv preprint arXiv:1611.08050 (2017).*