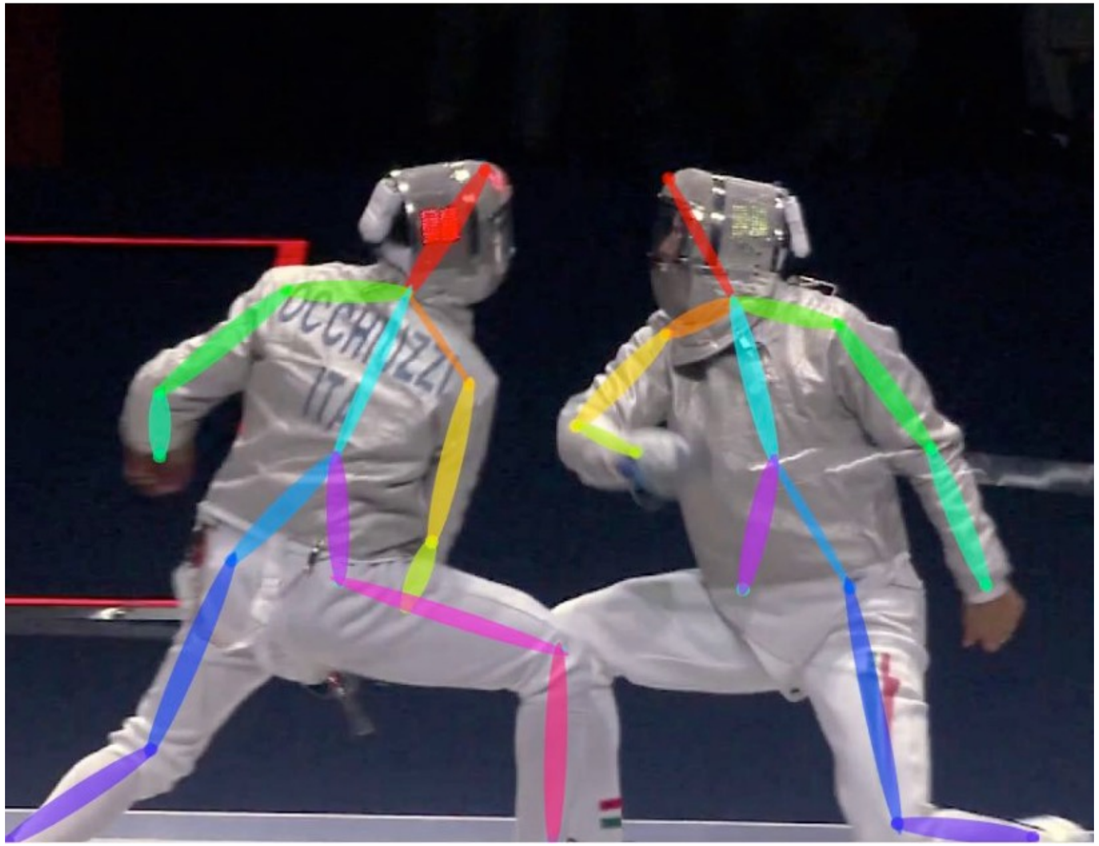
Human pose estimation

The task

- Mark joint locations for person
- Nose
- Right/left shoulder
- Right/left elbow
- Right/left hip
- …



Two versions of task

- Assume people have been detected
- Rough bounding box given
- Key info available:
  - scale
  - only 1 location per joint
- Pros: disentangles detection and pose estimation
- Cons: unrealistic

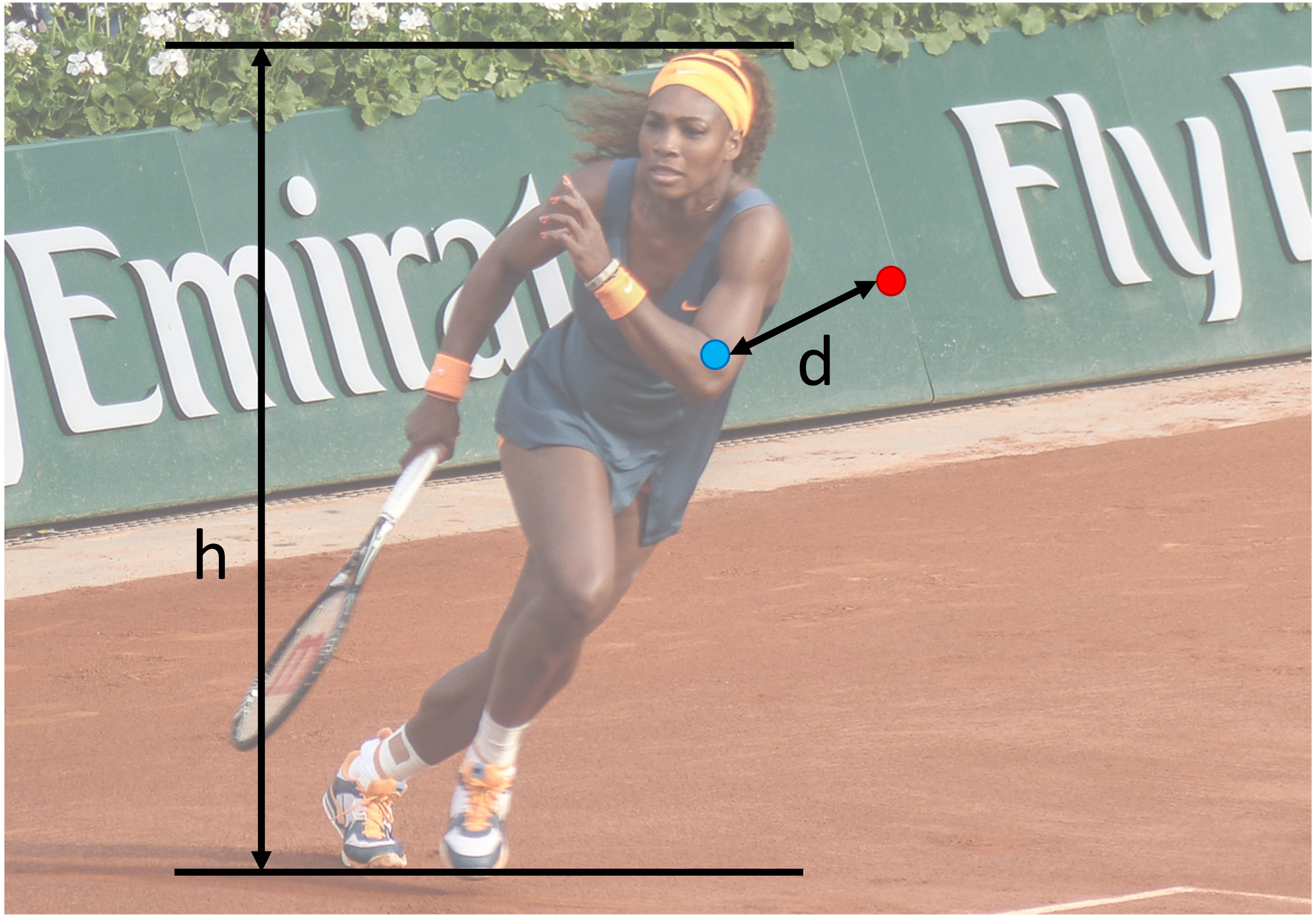
- Tabula rasa without detections
- Challenge: no idea of scale or number
- Possible opportunity: use keypoint estimates to improve detections
- Pros: realistic
- Cons: conflates detection and pose estimation

Pose estimation given detection

Evaluation metric - given detection

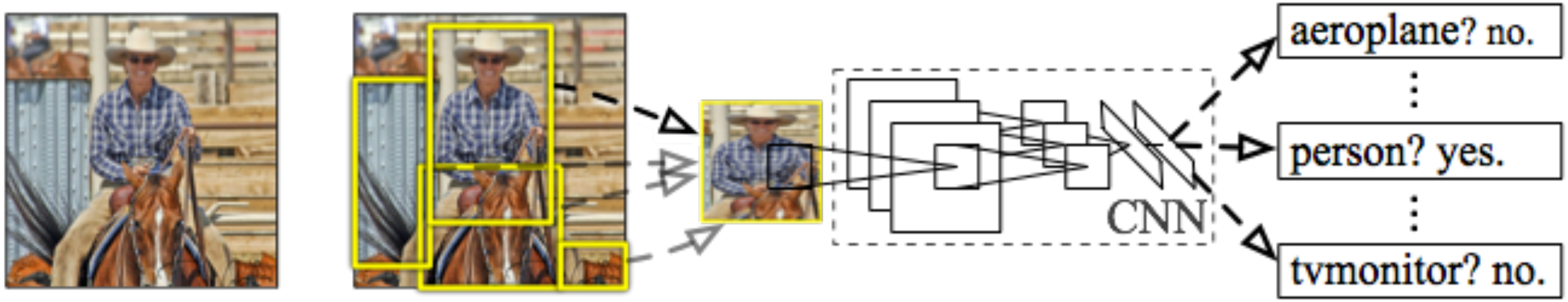
- Evaluate every keypoint separately
- For each person, check if keypoint is correct
- Compure fraction of people for which keypoint is correct: PCK (*Probability of Correct Keypoint*)

Evaluation metric - given detection



$d/h < \alpha$?

R-CNN: Regions with CNN features



Input image

Extract region proposals (~2k / image)

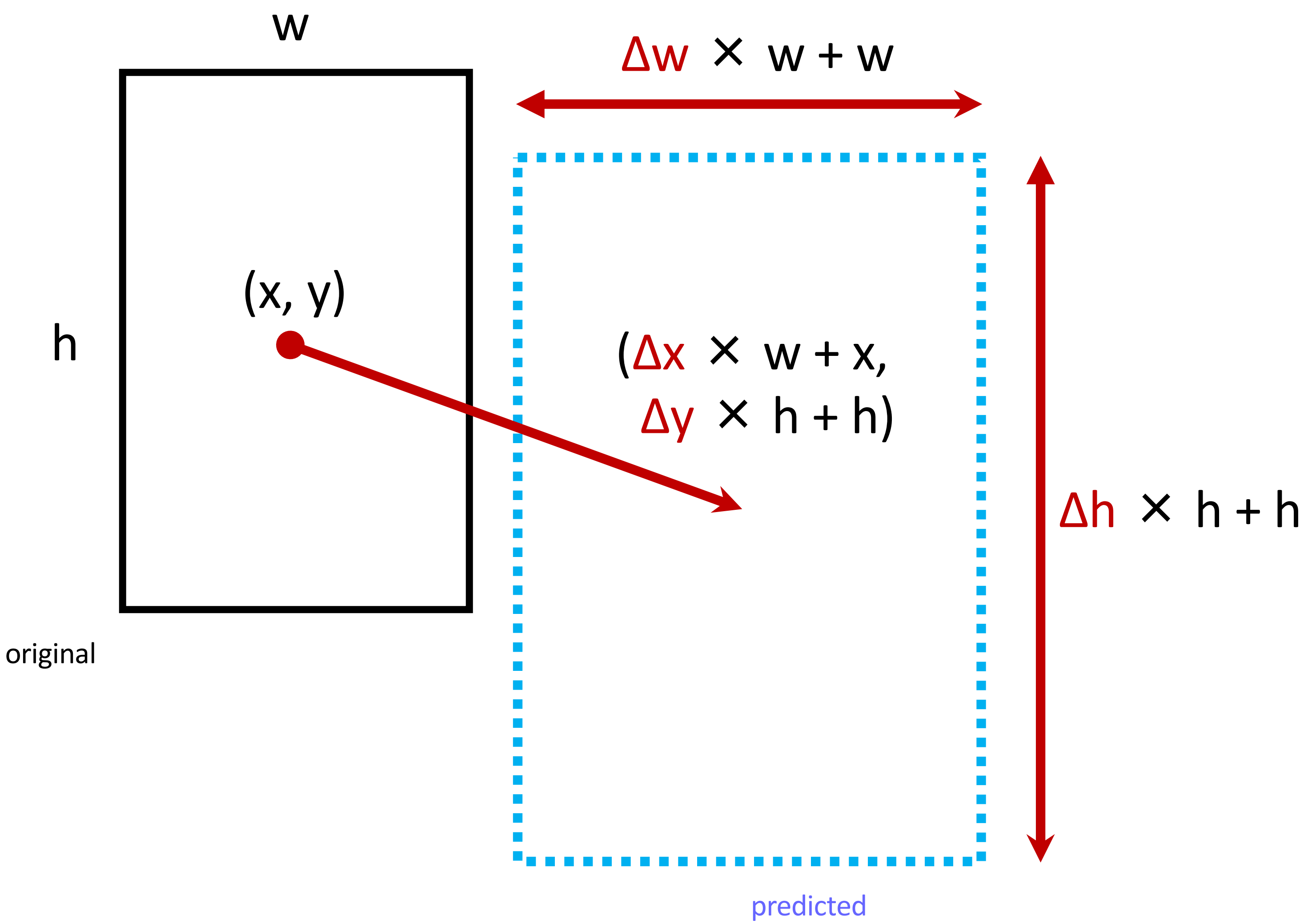
Compute CNN features

Classify regions (linear SVM)

Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation
Ń0ᶇ& çãJl §XŃAᶇ& JììX··Ń0ᶇ5 J·ÄÜ
.((($çãZXìXãNXçã $çä ê§öXì EÄÄçã JãT ; JöëXìã >XNçzãÄöÄçã Ť$E; >tŃ ɔȇ

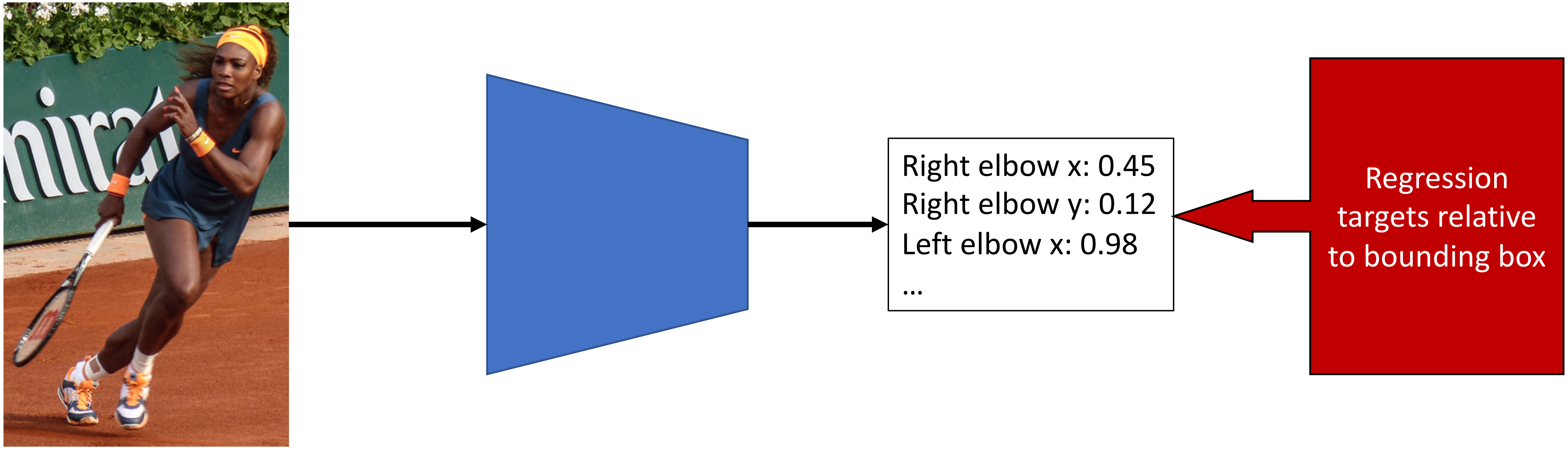
Slide credit : Ross Girshick

Bounding-box regression



Slide credit : Ross Girshick

Strategy 1: Regression



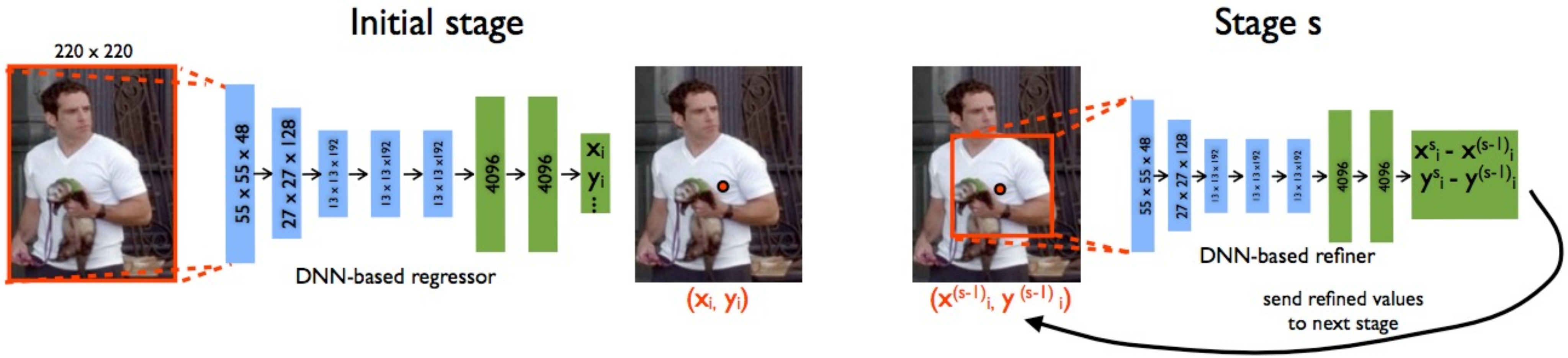
Right elbow x: 0.45
Right elbow y: 0.12
Left elbow x: 0.98
…

Regression targets relative to bounding box

DeepPose: Human Pose Estimation via Deep Neural Networks. Alexander Toshev and Christian Szegedy. In *CVPR,* 2014.

Strategy 1: Regression

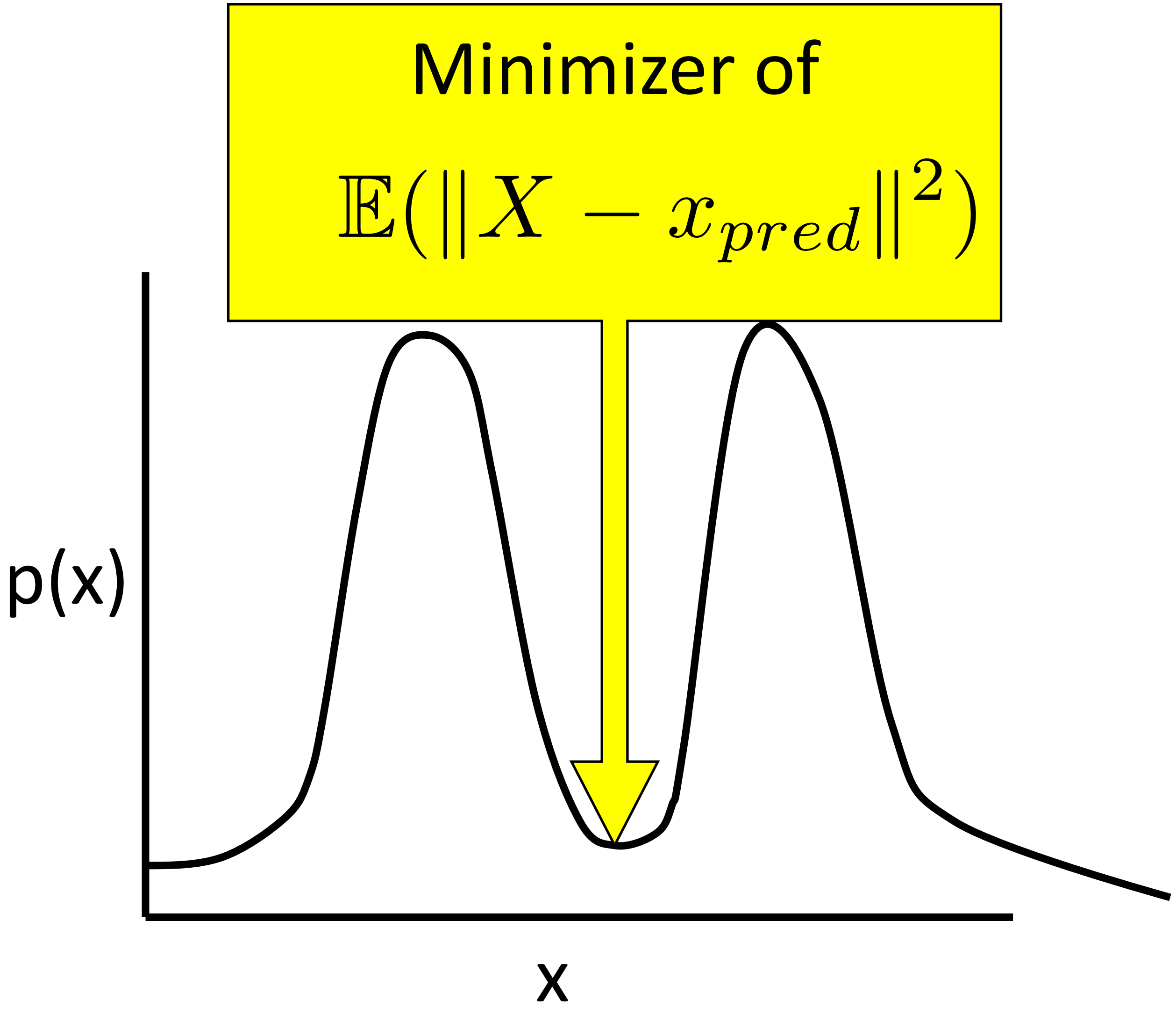
- Assumes global object features has enough information for accurate localization
  - Localization info missing due to subsampling?
- Solution: Refinement!



DeepPose: Human Pose Estimation via Deep Neural Networks. Alexander Toshev and Christian Szegedy. In *CVPR,* 2014.

Strategy 1: Regression

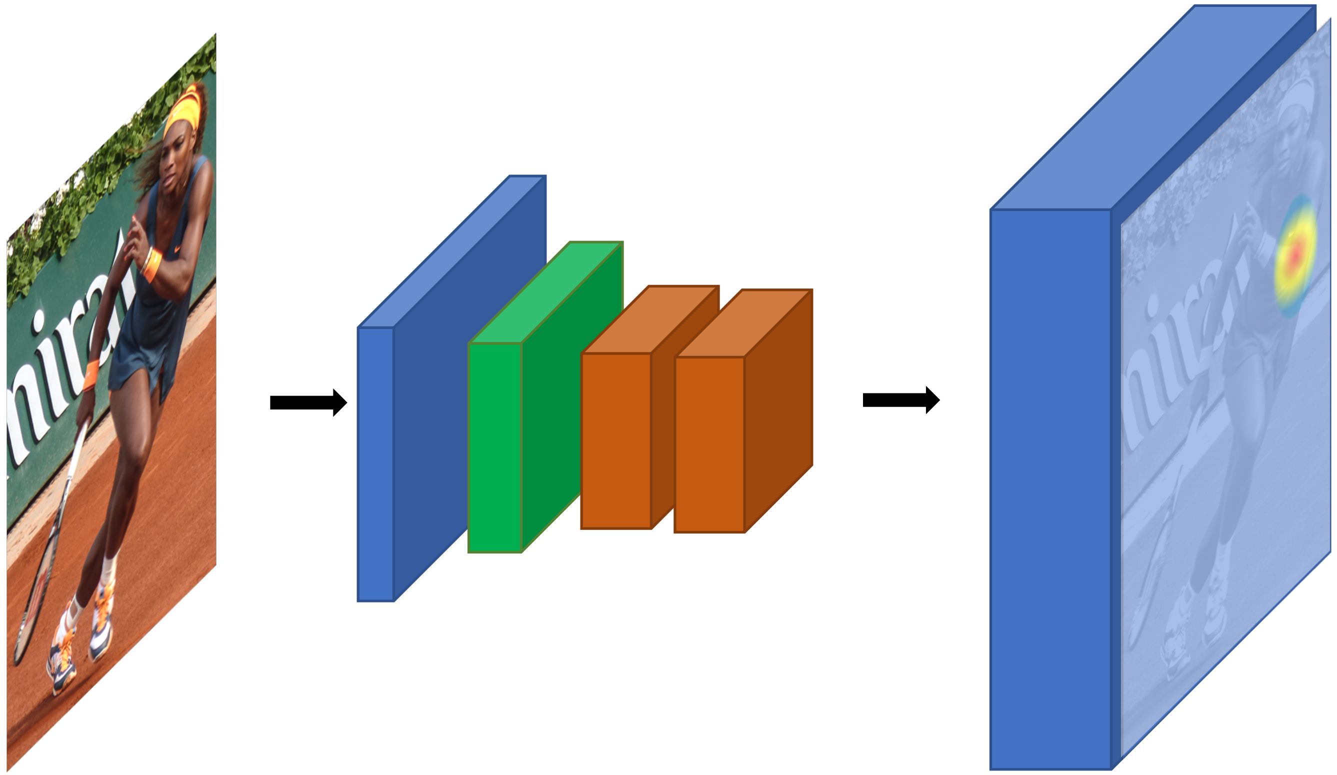
- Multimodal distributions?



Minimizer of

$$\mathbb{E}(\|X - x_{pred}\|^2)$$

Strategy 2: Heatmaps



Strategy 2: Heatmaps - Training

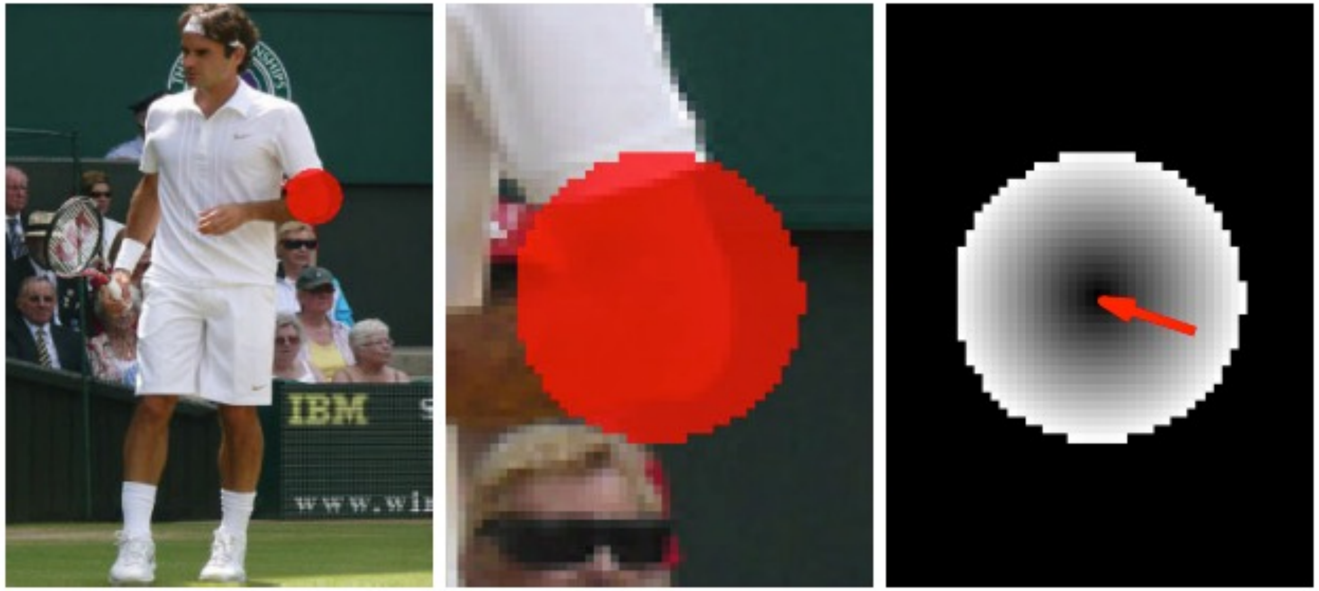
- Each keypoint is a separate binary heatmap
- Keypoint location is positive, all other locations are negative. Options:
    - *Softmax* over all locations in an image
        - $p(x, y) = \frac{e^{s(x,y)}}{\sum_{x',y'} e^{s(x',y')}}$
    - *Sigmoid* at each location
        - $p(x, y) = \frac{1}{1+e^{-s(x,y)}}$

Strategy 2: Heatmaps

- Still have the resolution issue
- Same solutions
  - Dilation?
  - Multiple layers?
  - Multiple image scales?

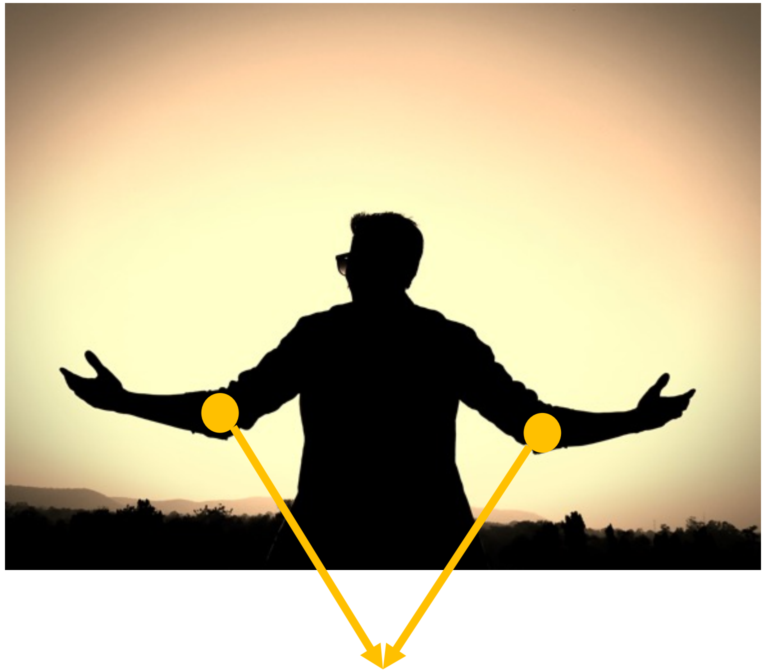
Heatmaps + Regression

- Use heatmap to predict coarse location
- Also predict at each coarse location an *offset* $(\Delta x, \Delta y)$.

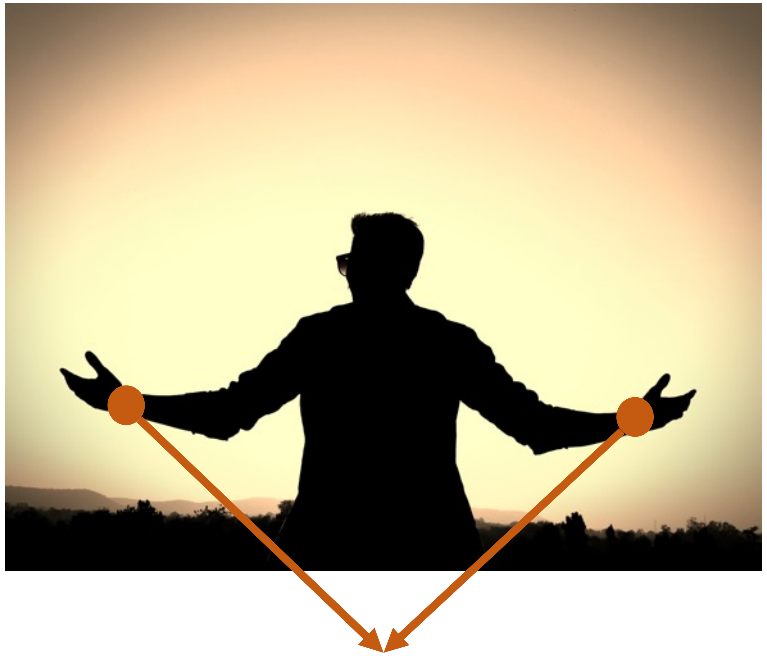


Papandreou, George, et al. "Towards accurate multi-person pose estimation in the wild." *CVPR*. Vol. 3. No. 4. 2017.

Are all keypoints independent given the image?

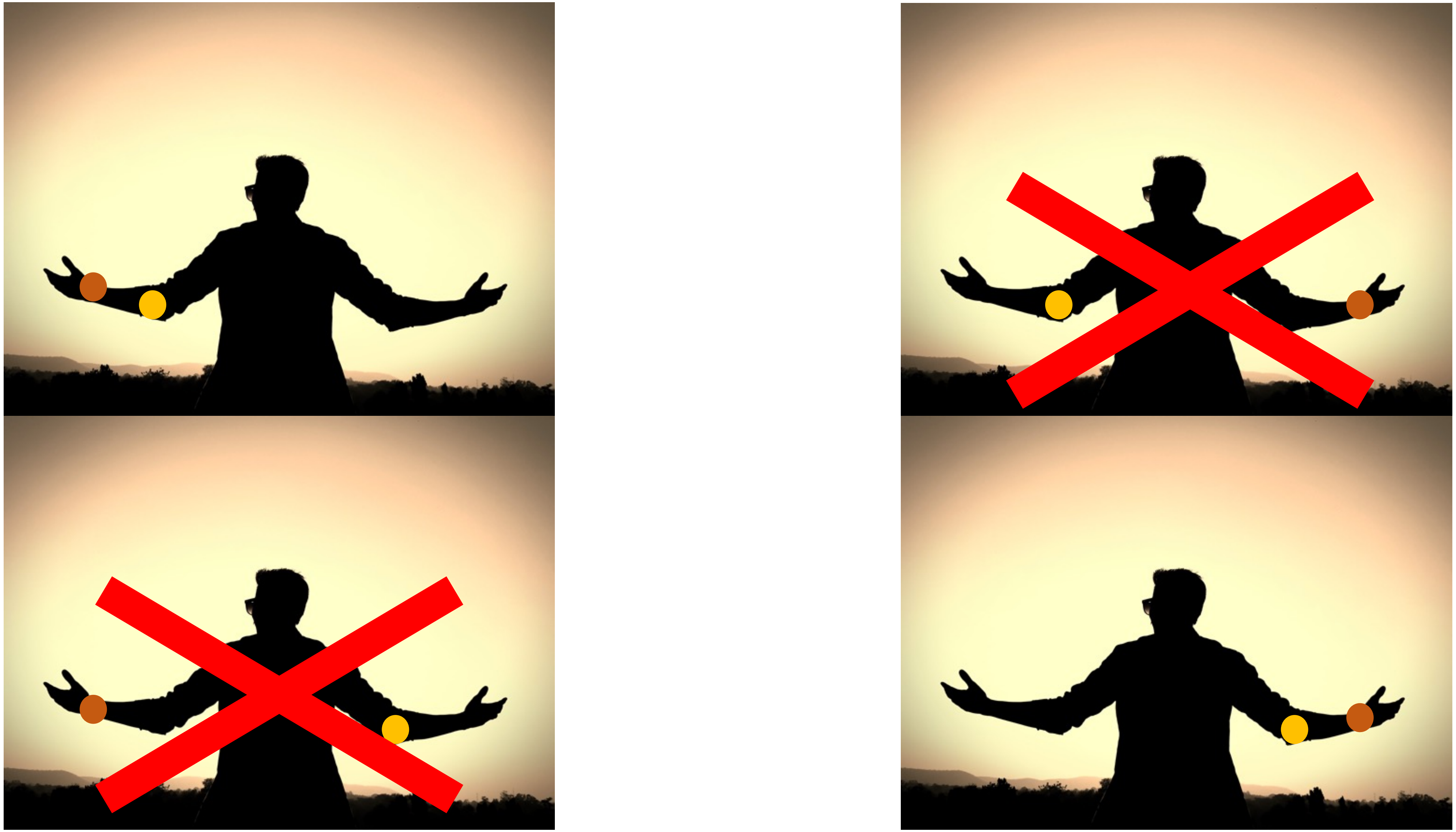


Equally likely locations for
right elbow



Equally likely locations for
right wrist

Are all keypoints independent given the image?



Capturing keypoint dependence!

- **Structured prediction**
- **l** is a candidate location for each keypoint

$$E(\mathbf{l}) = \underbrace{\sum_i s_i(l_i)}_{\text{Score from heatmap}} + \overbrace{\phi(\mathbf{l})}^{\text{Consistency between joints}}$$

Are all keypoints independent?

- **l** is a candidate location for each keypoint

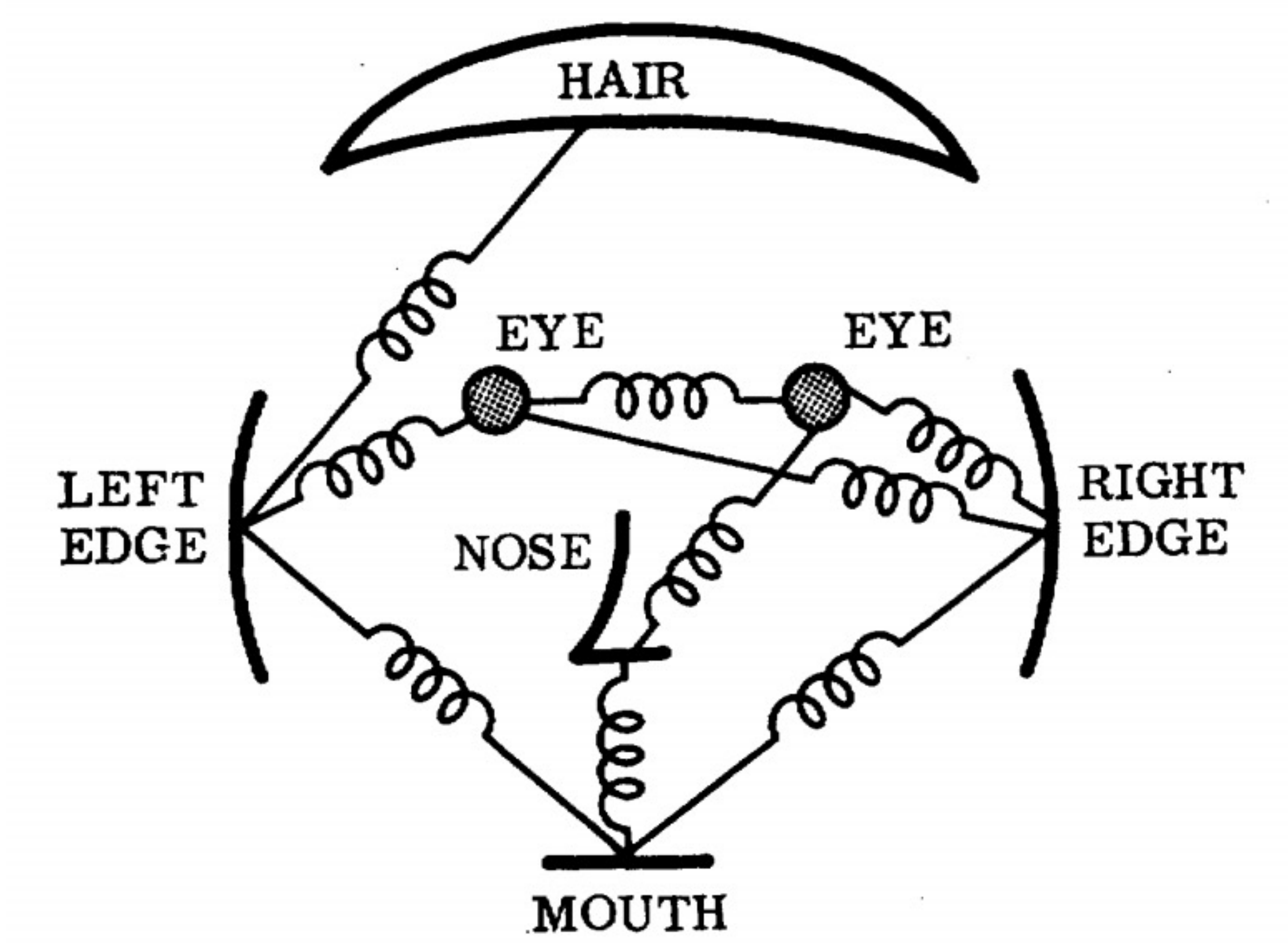
$$E(\mathbf{l}) = \sum_i \underbrace{s_i(l_i)}_{\text{Score from heatmap}} + \sum_{ij} \overbrace{\phi_{ij}(l_i, l_j)}^{\text{Consistency between joints}}$$

Joint prediction of keypoints

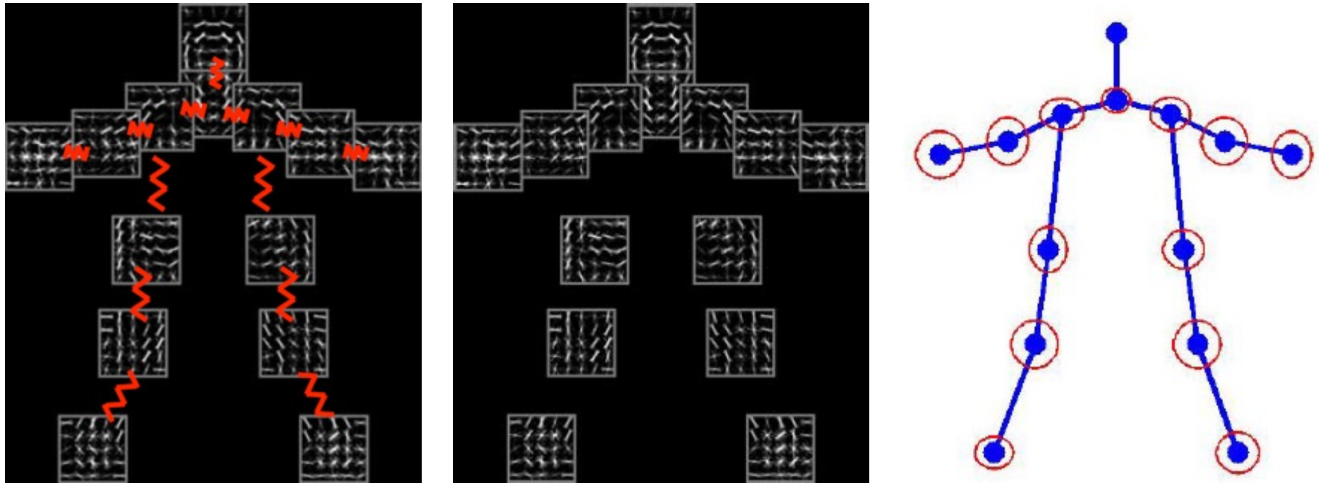
$$\mathbf{l}^* = \arg\min E(\mathbf{l})$$

- Conditional Random Field
- But not just smoothness: ϕ is unknown!
- Needs to be learnt

Pictorial structures



Flexible Mixture of Parts



$$S(I,L) = \sum_{i\in V} \alpha_i \cdot \phi(I,l_i) + \sum_{ij\in E} \beta_{ij} \cdot \psi(l_i,l_j)$$

- $\psi(l_i, l_j)$: Spatial features between l_i and l_j
- β_{ij}: Pairwise springs between part i and part j

Articulated Human Pose Estimation with Flexible Mixtures of Parts. Yi Yang and Deva Ramanan. *TPAMI* 2013.

Slide credit: Yang et al

Flexible Mixture of Parts

- Learning?
- Structured SVMs
  - Very large output spaces
  - A scoring function that scores input-output pairs $h_{\mathbf{w}}(x, \mathbf{y})$
  - Predicted output is arg max of scoring function
  - Loss is margin rescaled loss

Support vector machine learning for interdependent and structured output spaces. Tsochantaridis I, Hofmann T, Joachims T, Altun Y. In *ICML,* 2004

Inference?

$$E(\mathbf{l}) = \sum_i s_i(l_i) + \sum_{ij} \phi_{ij}(l_i, l_j)$$

$$\mathbf{l}^* = \arg\min E(\mathbf{l})$$

$$\min_{\mathbf{l}} \sum_i s_i(l_i) + \sum_{ij} \phi_{ij}(l_i, l_j)$$

$$l_i^* = \arg\min_{l_i} s_i(l_i) + \sum_j \phi_{ij}(l_i, l_j^*)$$

$$l_i^{(t+1)} \leftarrow \arg\min_{l_i} s_i(l_i) + \sum_j \phi_{ij}(l_i, l_j^{(t)})$$

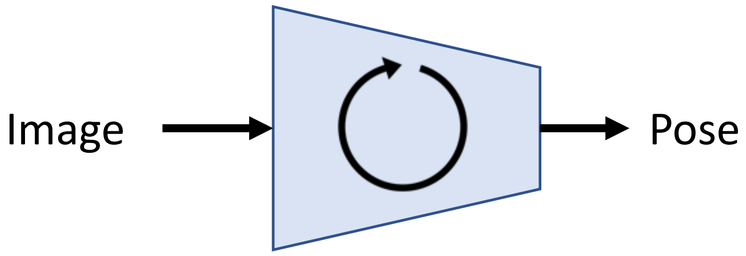
Iterative models

- Inference in MRFs and CRFs usually iterative and approximate

$$l_i^{(t+1)} \leftarrow \arg\min_{l_i} s_i(l_i) + \sum_j \phi_{ij}(l_i, l_j^{(t)})$$

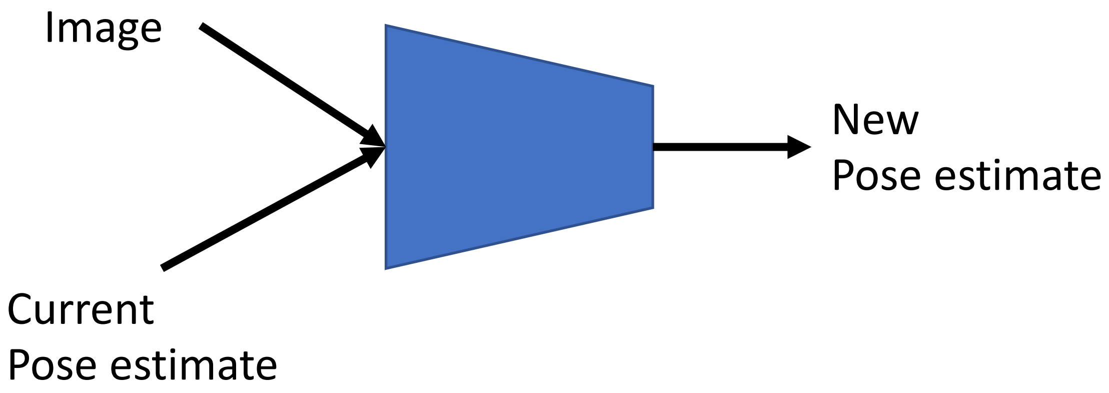
  - Except trees: FMP
- Instead of learning scoring function, then *approximately* minimizing it
- Learn iterative inference procedure?
  - Similar to autocontext/inference machines

Autocontext and Inference Machines



Image

Pose

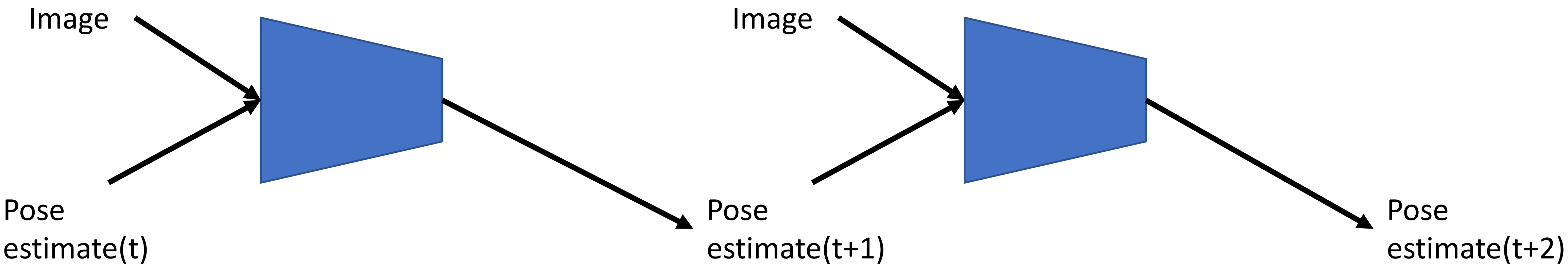


Image

Current
Pose estimate

New
Pose estimate

Autocontext and Inference Machines



- Shared parameters: *Inference Machines*
- Unshared parameters: *Autocontext*

Auto-context and Its Application to High-level Vision Tasks. Zhuowen Tu. In *CVPR* 2008.
Learning Message-Passing Inference Machines for Structured Prediction. Stephane Ross, Daniel Munoz, Martial Hebert, J. Andrew Bagnell. In *CVPR* 2011.

Iterative models

- In each iteration, beliefs of one variable are updated using current beliefs of the others

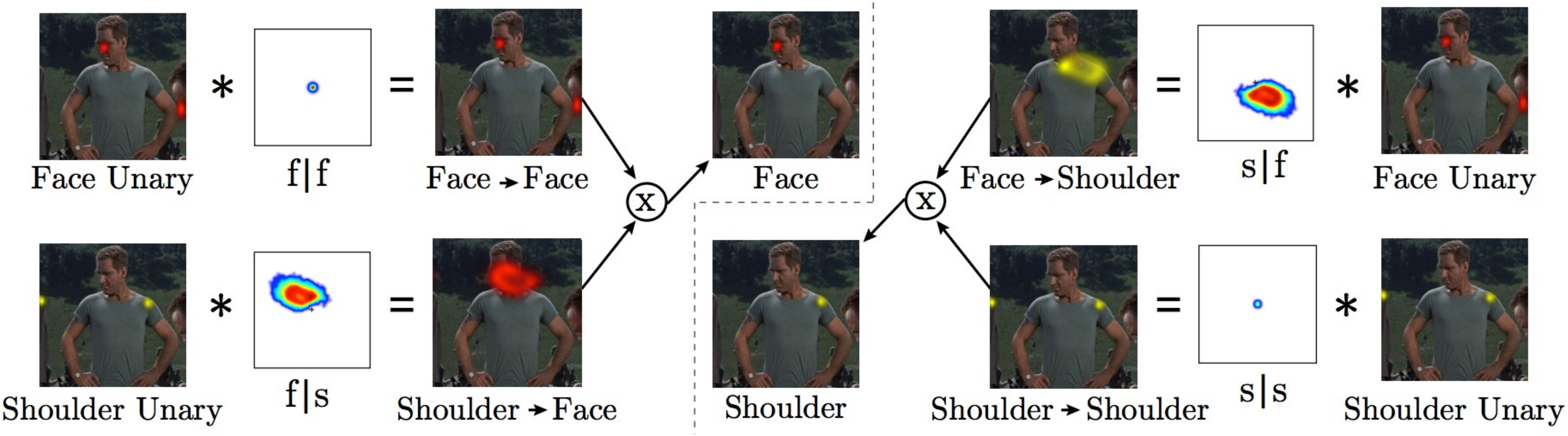
$$l_i^{(t+1)} \leftarrow \arg\min_{l_i} s_i(l_i) + \sum_j \phi_{ij}(l_i, l_j^{(t)})$$

- Frame each iteration of inference as a differentiable function
- Write inference as a convolutional network

Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. Jonathan Tompson, Arjun Jain, Yann LeCun, Christoph Bregler. In *NIPS,* 2014.

Iterative models

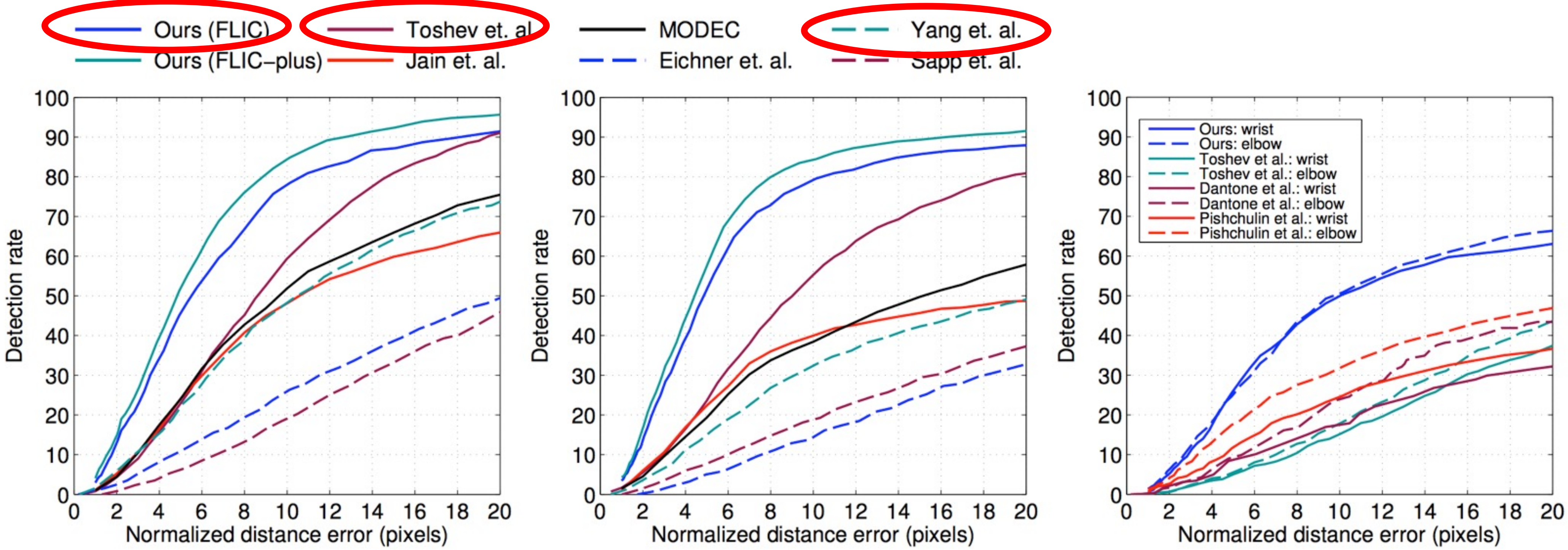
- P(eye at p) = $\sum_q$ P(eye at p | nose at q) P (nose at q)
- P(eye at p | nose at q) only depends on relative location of p and q
- $f(p) = \sum_q w(p - q)\, g(q)$: convolution!
- $f = w * g$



Iterative models

Regression

Mixture of parts



(a) FLIC: Elbow

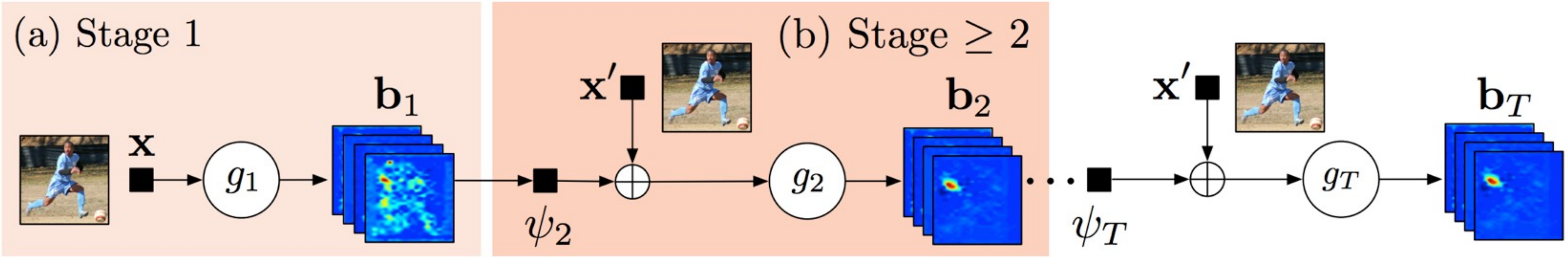
(b) FLIC: Wrist

(c) LSP: Wrist and Elbow

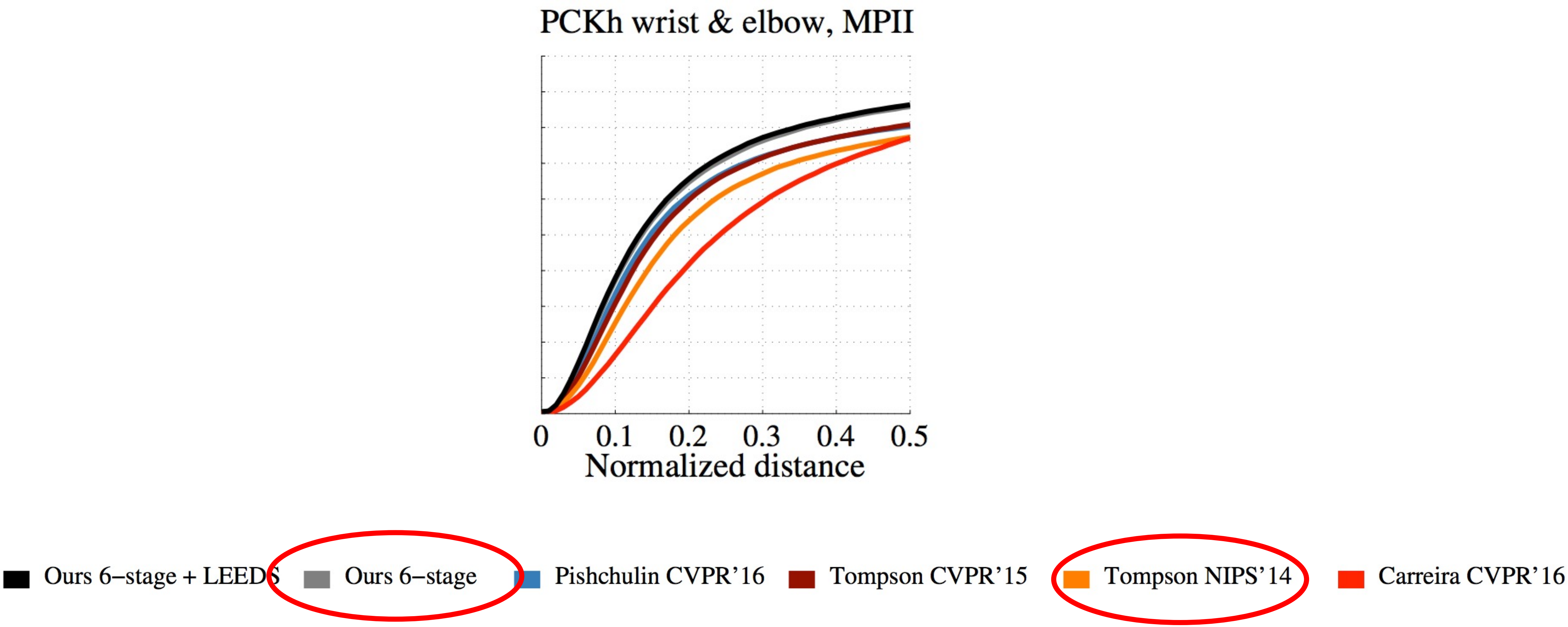
Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation. Jonathan Tompson, Arjun Jain, Yann LeCun, Christoph Bregler. In *NIPS,* 2014.

More iterative models

- Why only one convolution?
- Each iteration can involve multiple convolution/subsampling layers over beliefs from previous iteration

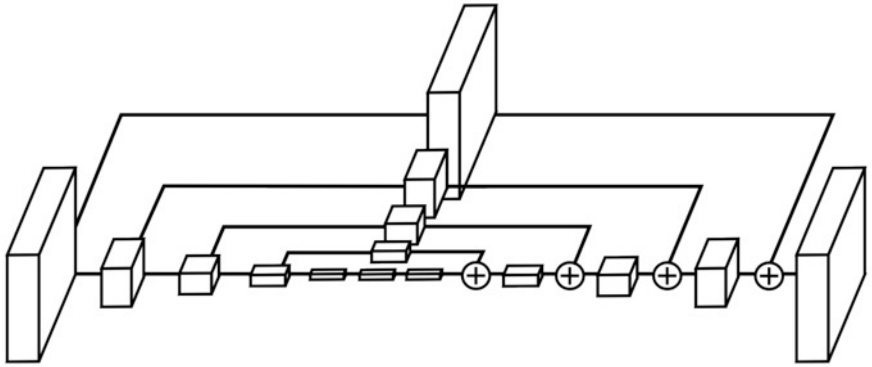


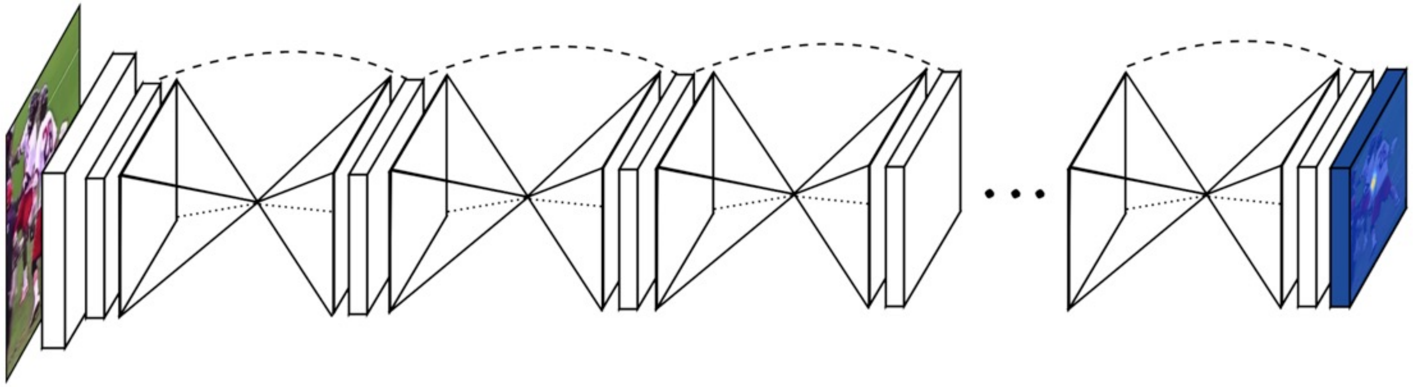
Convolutional Pose Machines. Shih-En Wei, Varun Ramakrishna, Takeo Kanade, Yaser Sheikh. In *CVPR,* 2016


PCKh wrist & elbow, MPII
0
0.1
0.2
0.3
0.4
0.5
Normalized distance
Ours 6–stage + LEEDS
Ours 6–stage
Pishchulin CVPR'16
Tompson CVPR'15
Tompson NIPS'14
Carreira CVPR'16

Stacked Hourglass Networks

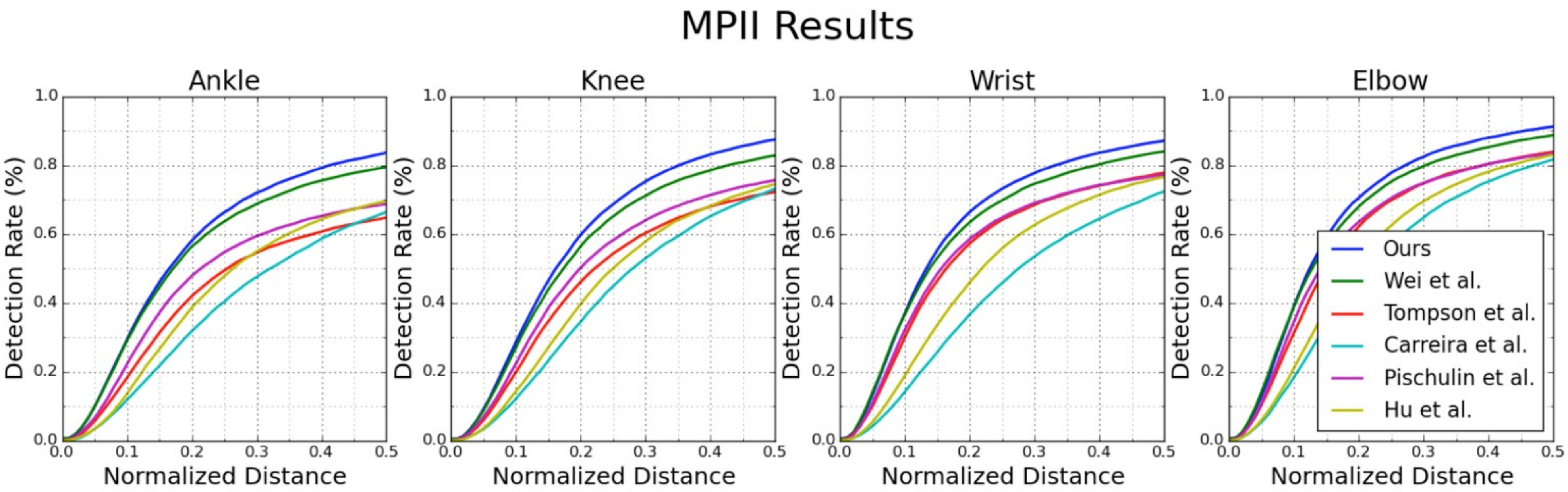
- Each refinement round has to
  - Combine global information about pose
  - Use global pose information to produce new precise pose estimate
- Rounds need not share parameters
- ”Hourglass structure”





Stacked Hourglass Networks for Human Pose Estimation. Alejandro Newell, Kaiyu Yang, and Jia Deng. In *ECCV,* 2016.

Stacked hourglass networks



Pose estimation without detection

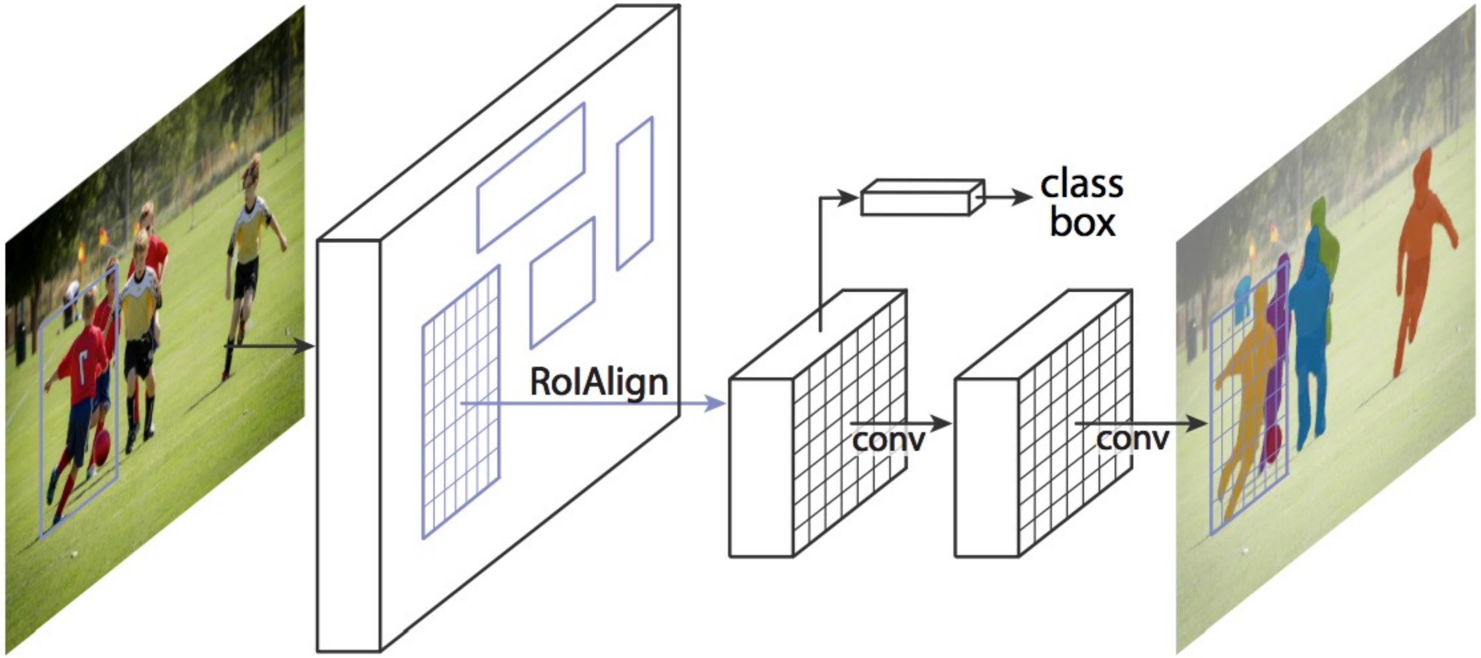
Evaluation metric - tabula rasa

- Algorithm detects keypoints + scores
- Match keypoint to a ground truth keypoint if d/h is less than threshold
- Compute precision-recall curve
- Compute AP (called APK : AP Keypoint)

Two strategies

- First detect, then estimate keypoints
    - Can use any of previous techniques
    - Similar to instance segmentation
    - Easy to get object level information
    - Hard to recover from bad detections
    - e.g. Mask R-CNN
- Detect keypoints, then group into people
    - Need a way to group keypoints: hard problem, requires heuristics
    - No simple way to have object level information

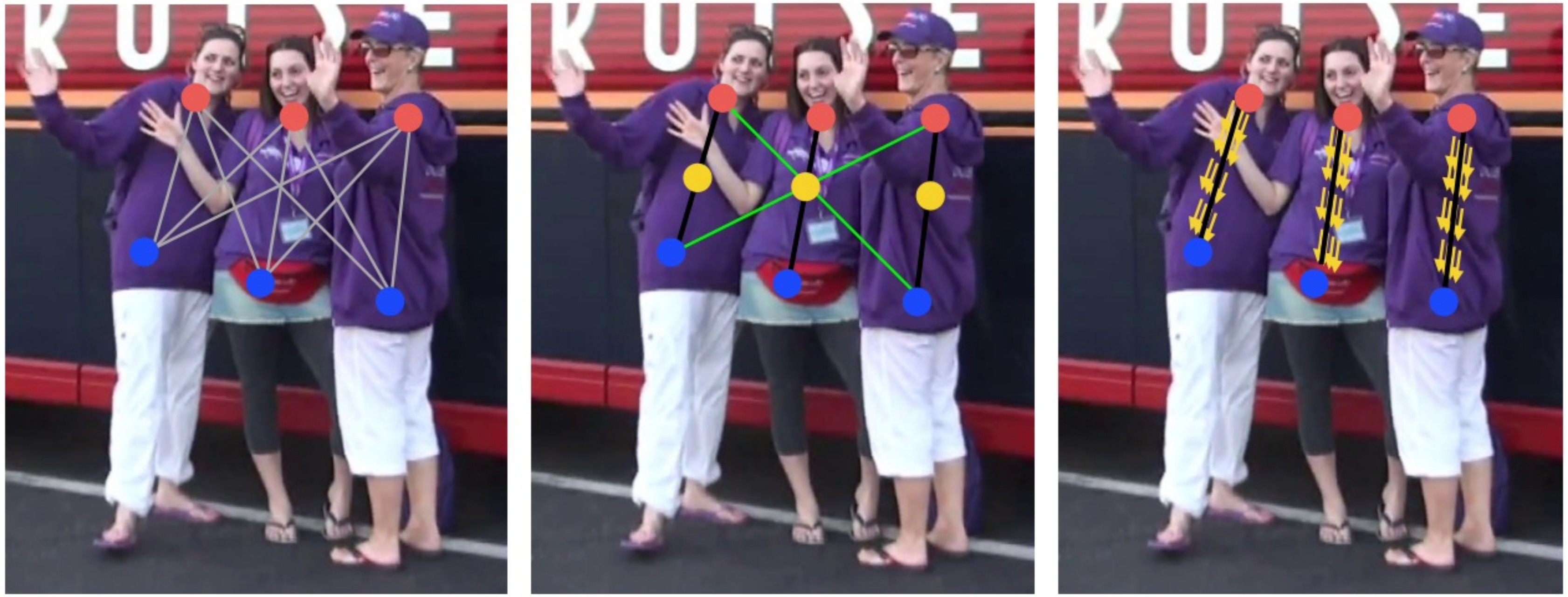
Top-down keypoint detection



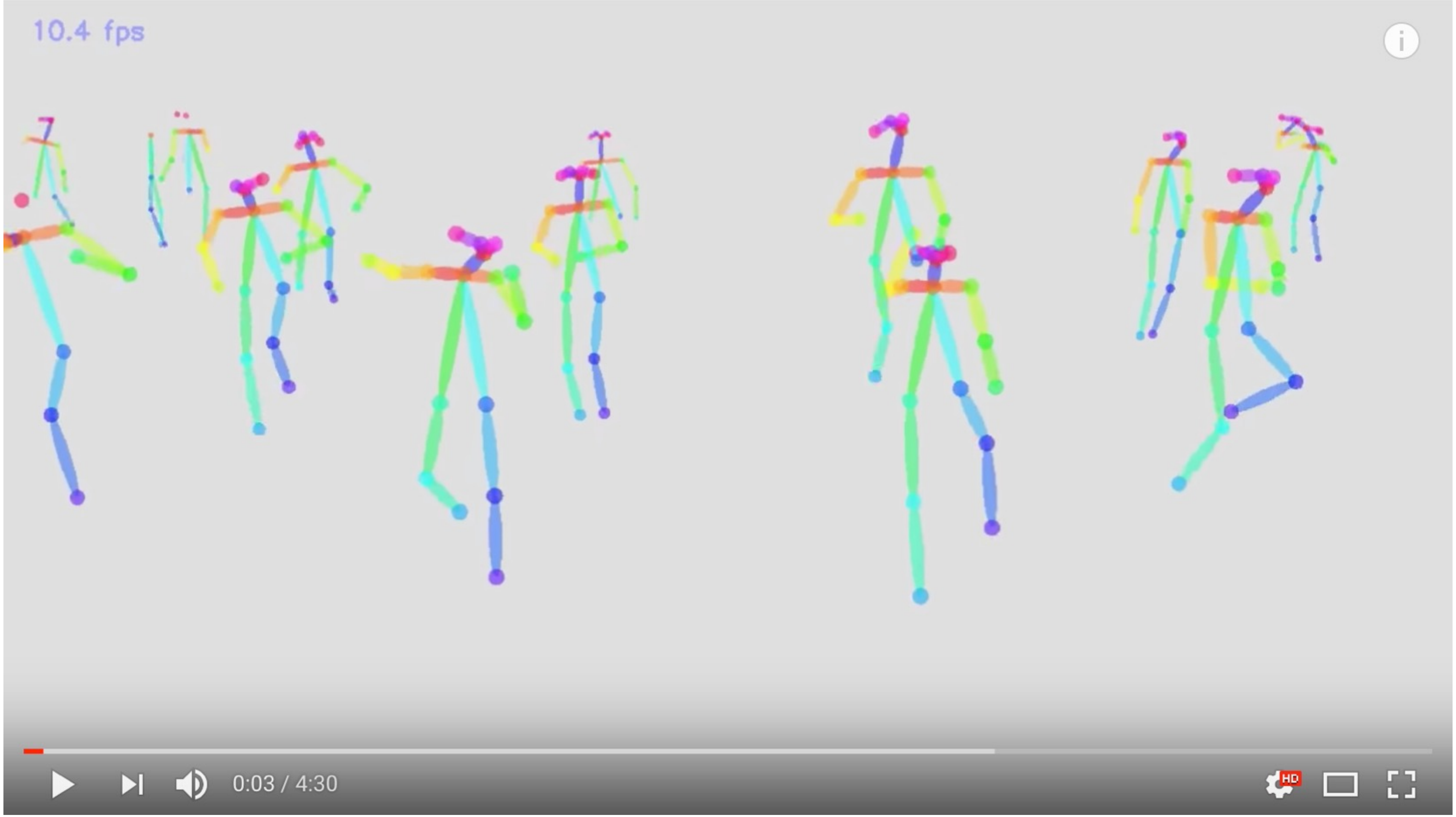
He, Kaiming, et al. "Mask r-cnn." *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017.

Bottom-up keypoint detection

- Need to group keypoints. Can be really ambiguous
- Idea: detect not just keypoints but also limbs + limb orientation

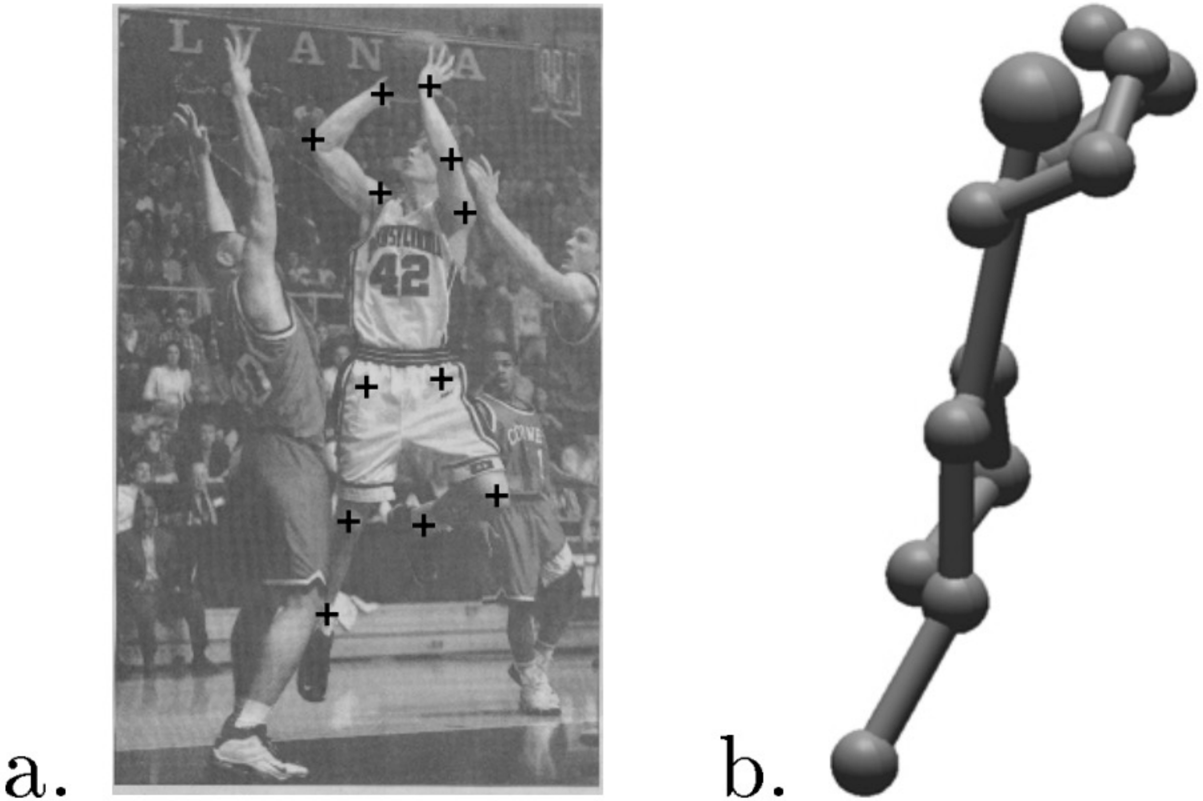


Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh. In *CVPR,* 2017.



Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh. In *CVPR,* 2017.

Pose estimation in 3D



Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image. C. J. Taylor. In *CVPR,* 2000

Pose estimation in 3D

- Key idea: know relative lengths of each limb4
- Assume *scaled orthographic projection*
  - Valid when variation in depth much smaller than depth

$$x = \frac{X}{Z} \approx \frac{X}{Z_0}$$

$$y = \frac{Y}{Z} \approx \frac{Y}{Z_0}$$

constant

Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image. C. J. Taylor. In *CVPR,* 2000

Pose estimation in 3D

$$
\begin{aligned}
l^2 &= (X_1 - X_2)^2 + (Y_1 - Y_2)^2 + (Z_1 - Z_2)^2 \\
(u_1 - u_2) &= s(X_1 - X_2) \\
(v_1 - v_2) &= s(Y_1 - Y_2) \\
dZ &= (Z_1 - Z_2) \\
\Rightarrow dZ &= \sqrt{l^2 - ((u_1 - u_2)^2 + (v_1 - v_2)^2)/s^2}
\end{aligned}
$$

Reconstruction of Articulated Objects from Point Correspondences in a Single Uncalibrated Image. C. J. Taylor. In *CVPR,* 2000