Human Pose Estimation

Osokin Daniil

IOTG/VMC/VP/ICV

Internet of Things Group

Agenda

Task description

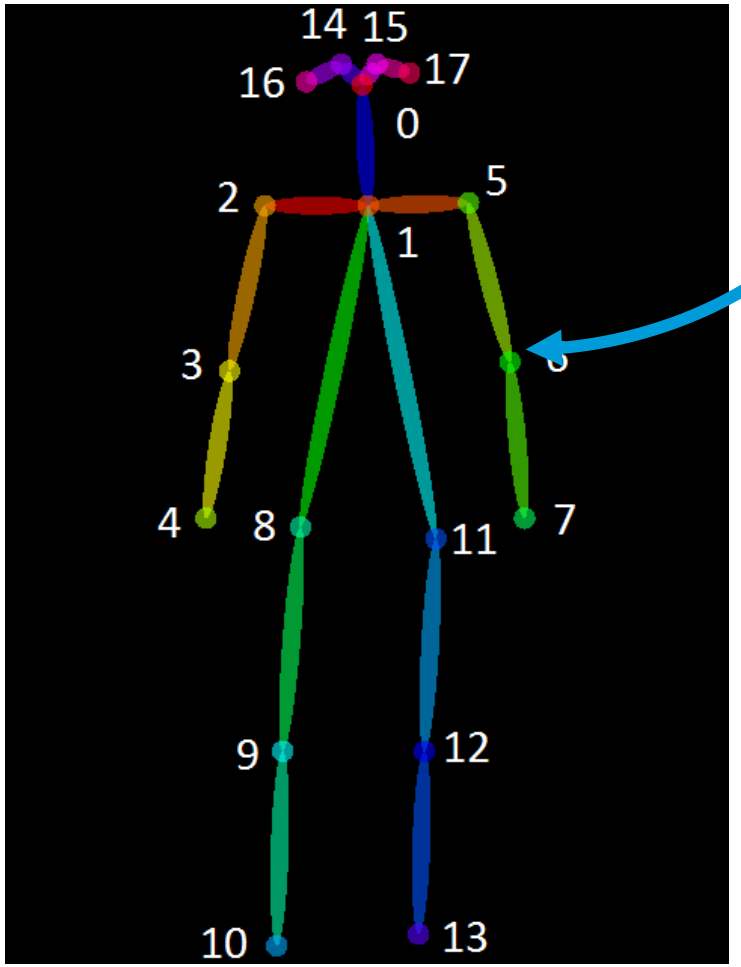
Single person pose estimation

Multi-person pose estimation

Real-time human pose estimation

Results

Human Pose

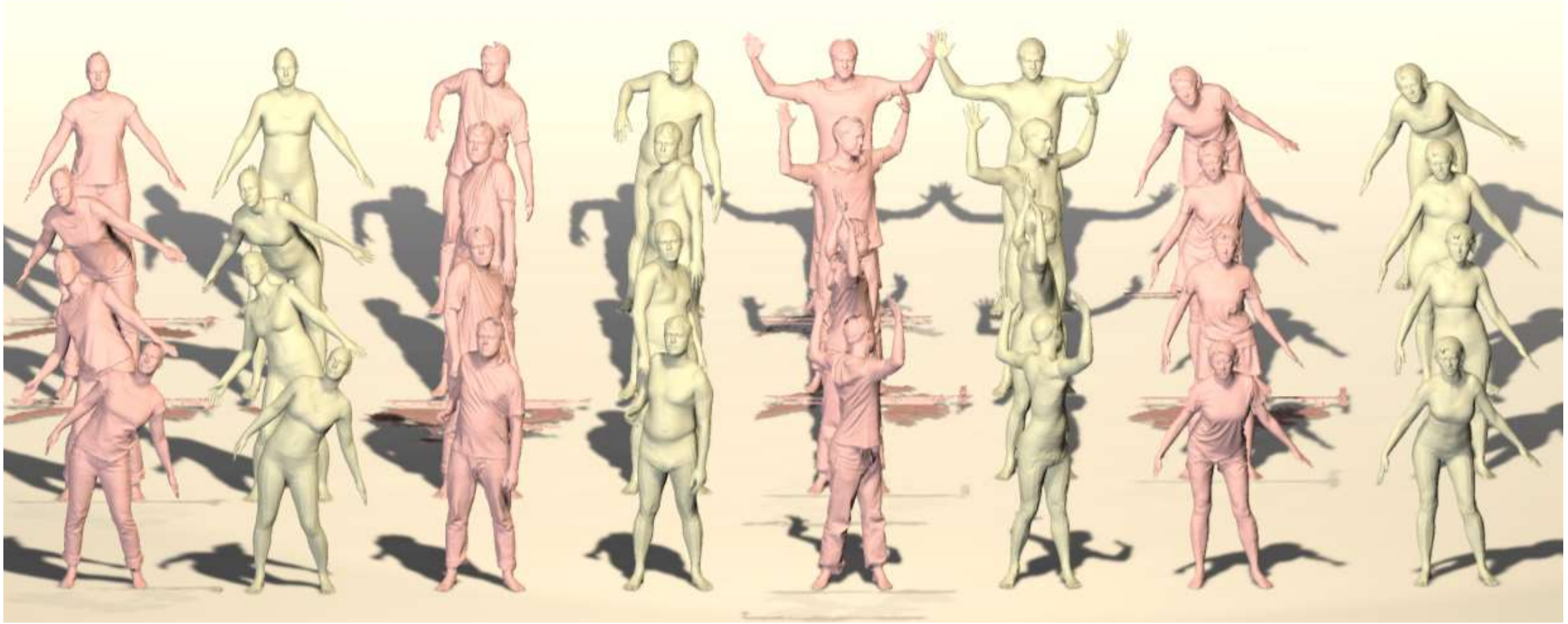


Pose consists of keypoints (or joints): ankles, knees, hips, elbows, etc.

The task is to predict the pose for every person in an image.

Images credit: Z. Cao et al. "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields"

Human Shape

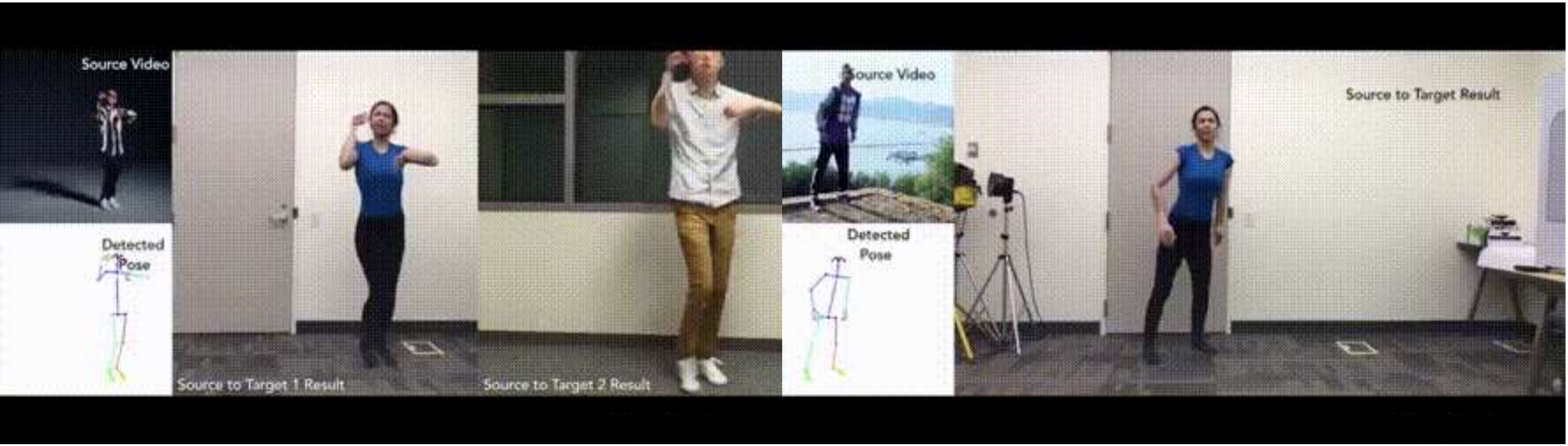


Images credit: C. Zhang et al. “Detailed, accurate, human shape estimation from clothed 3D scan sequences”

Applications: Sport Analytics

Video credit: SPORTLOGiQ, gifs from https://imgflip.com

Applications: Motion Transfer



Video credit: C. Chan et al. "Everybody Dance Now", gifs from http://freegifmaker.me

Applications: Augmented Reality

Video credit: C-Y. Weng et al. “Photo Wake-Up: 3D Character Animation from a Single Photo”, gifs from http://freegifmaker.me

Single Person Pose Estimation

DeepPose, 2014.

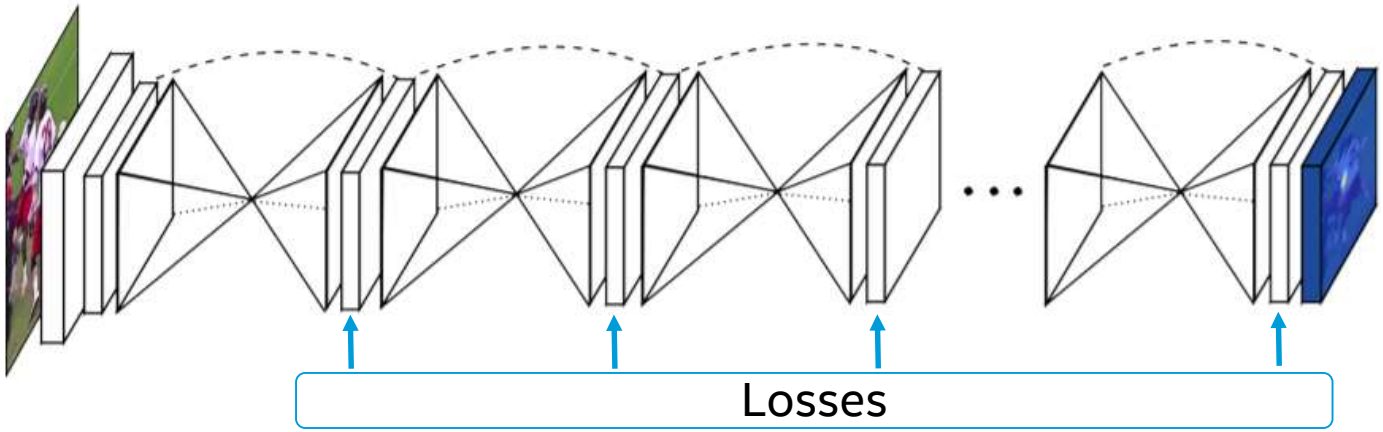
Trained on:

- FLIC - Frames Labeled In Cinema, 4k train, 1k test, from Hollywood movies.
- LSP extended – Leeds Sports *D*ataset, 1k + 10k train, 1k test.

Images credit: A. Toshev et al. "DeepPose: Human Pose Estimation via Deep Neural Networks"

Single Person Pose Estimation

Stacked Hourglass Network, 2016.



Trained on:

- MPII Human Pose Dataset – 28k images.
- FLIC.

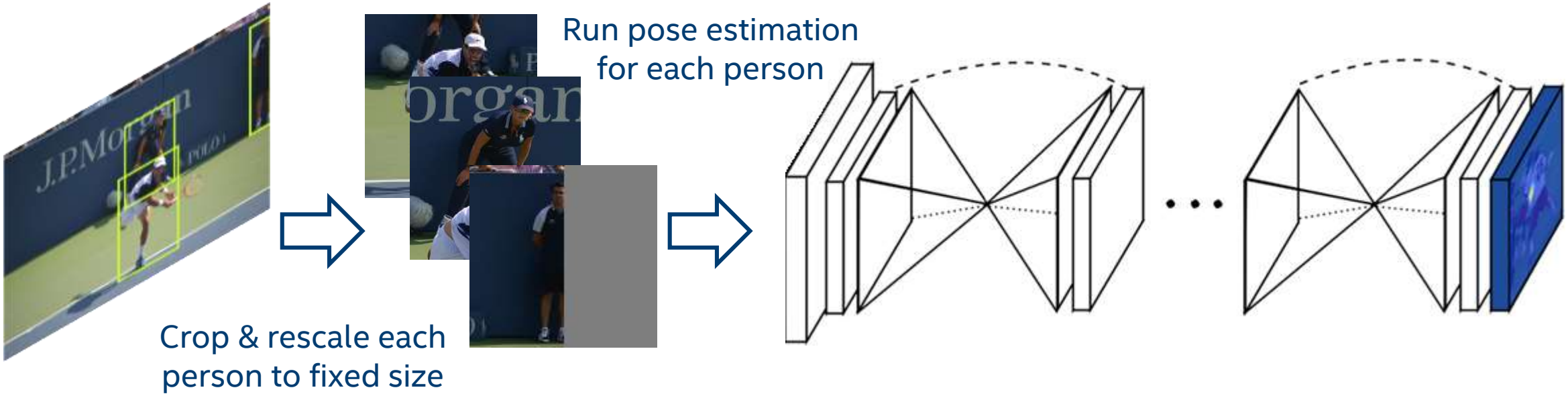
Keypoint heatmaps

Images credit: A. Newell et al. “Stacked Hourglass Networks for Human Pose Estimation”

Multi-person Pose Estimation

Two major approaches:

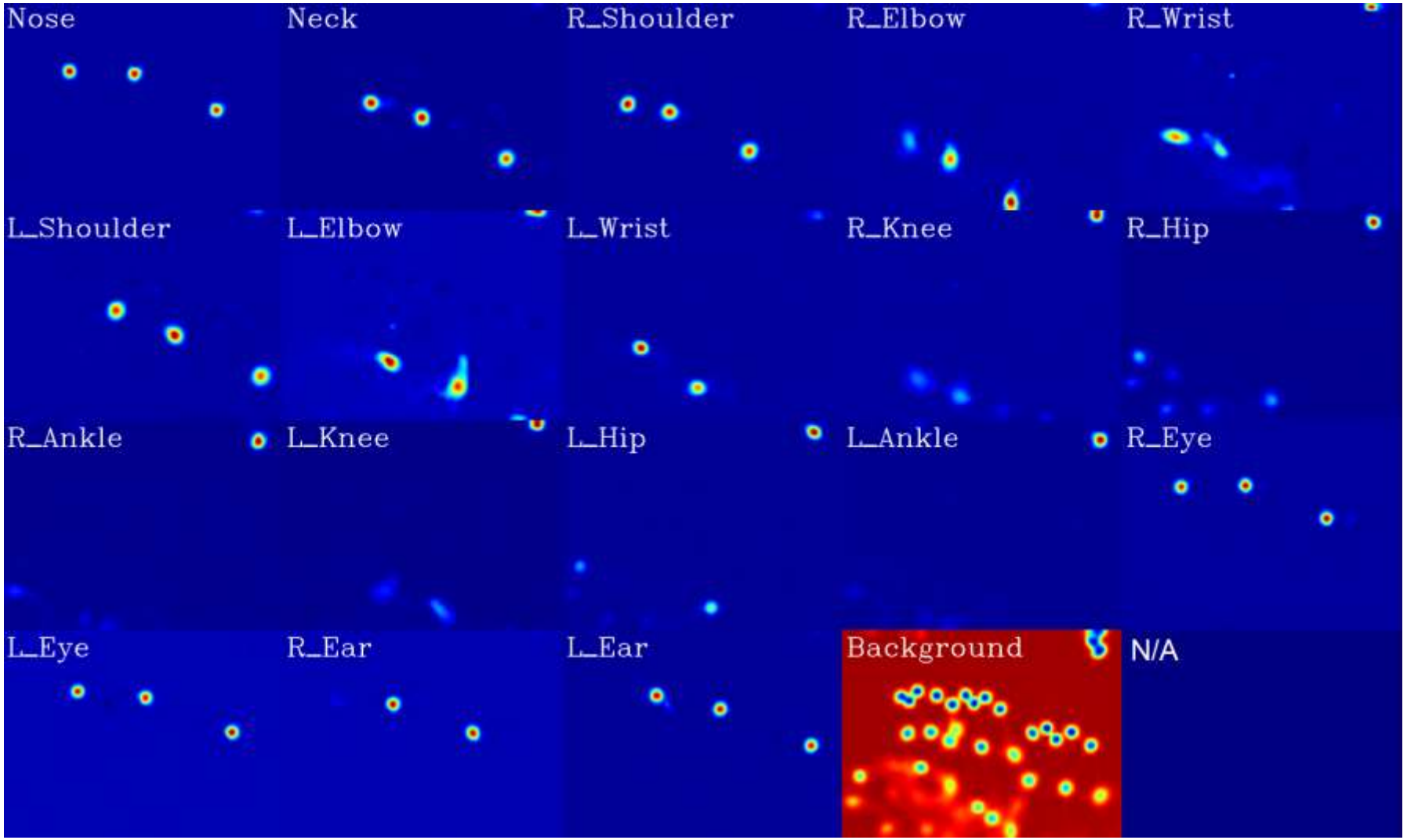
- Top-Down: detect all people with object detector in entire frame, for each people run single person pose estimation.



- Bottom-Up: predict all keypoints in entire frame, group them by persons.

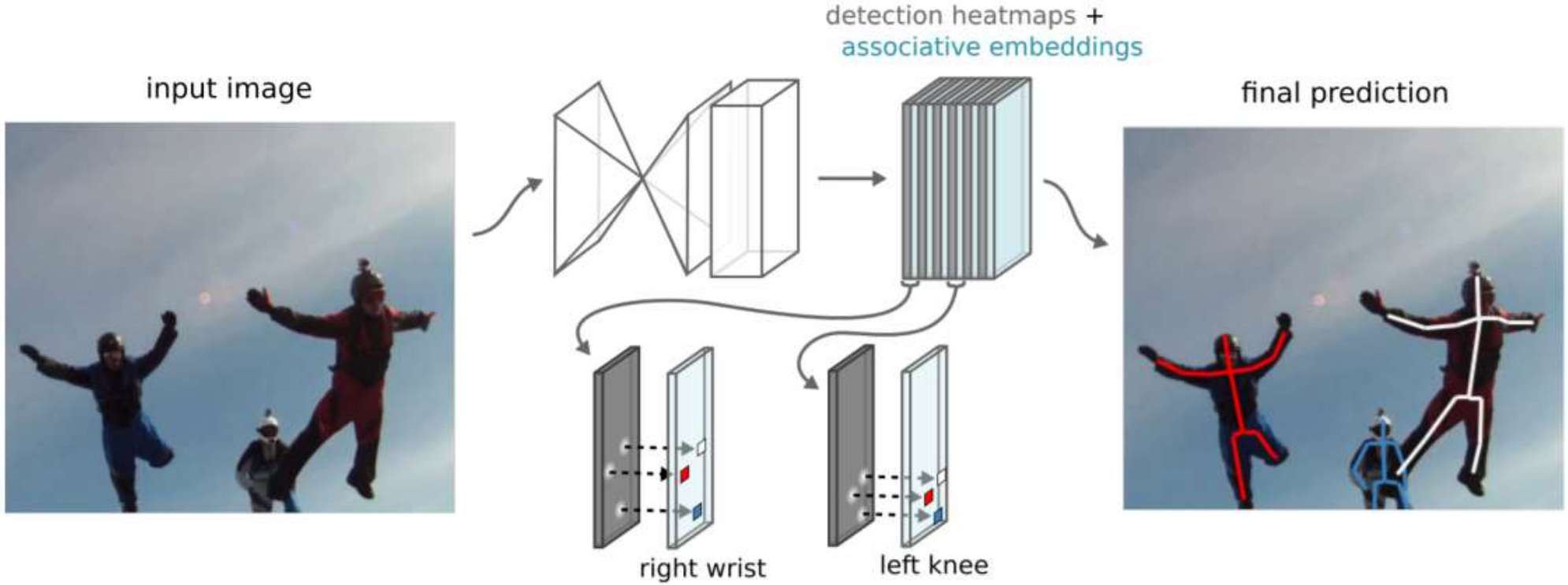
Multi-person Pose Estimation

Bottom-up heatmaps:



Multi-person Pose Estimation

Associative Embedding:
End-to-End Learning for Joint Detection and Grouping, 2017.

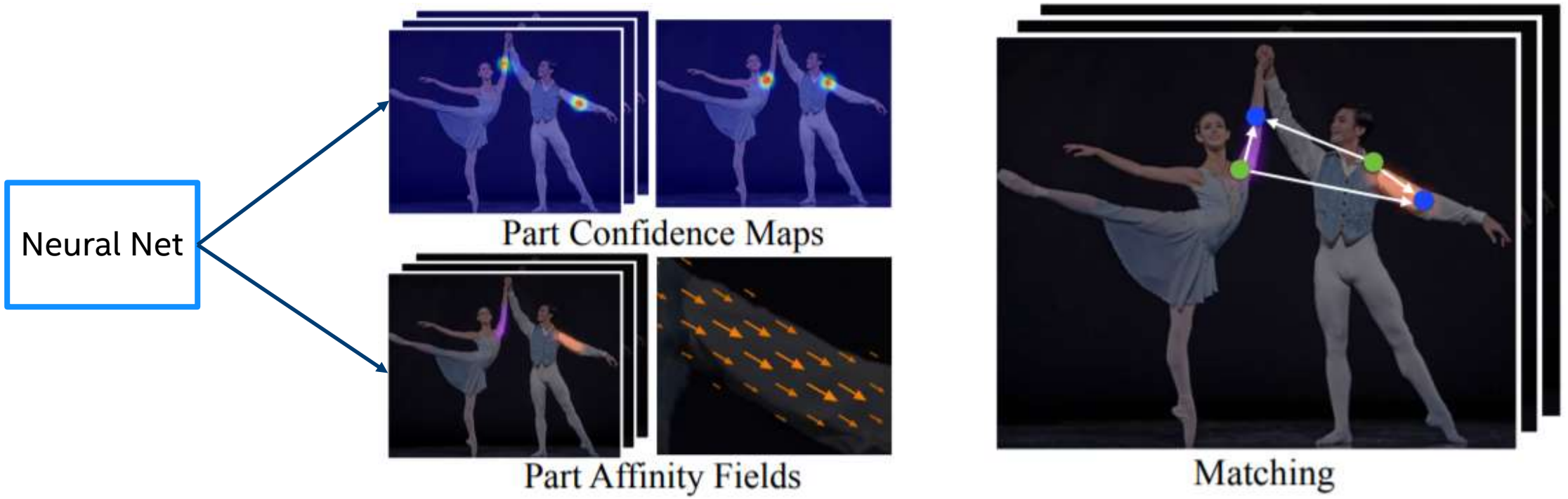


Predict *tag* maps along with keypoint maps.

Images credit: A. Newell et al. "Associative Embedding: End-to-End Learning for Joint Detection and Grouping"

Multi-person Pose Estimation

Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields, 2017.



Images credit: Z. Cao et al. "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields"

Multi-person Pose Estimation: The Trade-off

		FPS	AP	AP_{50}	AP_{75}
BU	MultiPoseNet	23	**69.6**	86.3	**76.6**
BU	Newell et al.	6	65.5	**86.8**	72.3
BU	CMU-Pose	10	61.8	84.9	67.5
TD	Megvii	-	73.0	91.7	80.9
TD	CFN	3	72.6	86.7	69.7
TD	Mask R-CNN	5	69.2	90.4	76.0
TD	SJTU	0.4	68.8	87.5	75.9
TD	GRMI-2017	-	66.9	86.4	73.6
TD	G-RMI-2016	-	60.5	82.2	66.2

Images credit: M. Kocabas et al. "MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network"

Multi-person Pose Estimation: The Trade-off

		FPS	AP	AP_{50}	AP_{75}
BU	MultiPoseNet	23	**69.6**	86.3	**76.6**
BU	Newell et al.	6	65.5	**86.8**	72.3
BU	CMU-Pose	10	61.8	84.9	67.5
TD	Megvii	-	73.0	91.7	80.9
TD	CFN	3	72.6	86.7	69.7
TD	Mask R-CNN	5	69.2	90.4	76.0
TD	SJTU	0.4	68.8	87.5	75.9
TD	GRMI-2017	-	66.9	86.4	73.6
TD	G-RMI-2016	-	60.5	82.2	66.2

FPS results obtained on a *GTX1080Ti* GPU.

Images credit: M. Kocabas et al. "MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network"

Multi-person Pose Estimation: The Trade-off

		FPS	AP	AP_{50}	AP_{75}
BU	MultiPoseNet	23	**69.6**	86.3	**76.6**
BU	Newell et al.	6	65.5	**86.8**	72.3
BU	CMU-Pose	10	61.8	84.9	67.5
TD	Megvii	-	73.0	91.7	80.9
TD	CFN	3	72.6	86.7	69.7
TD	Mask R-CNN	5	69.2	90.4	76.0
TD	SJTU	0.4	68.8	87.5	75.9
TD	GRMI-2017	-	66.9	86.4	73.6
TD	G-RMI-2016	-	60.5	82.2	66.2

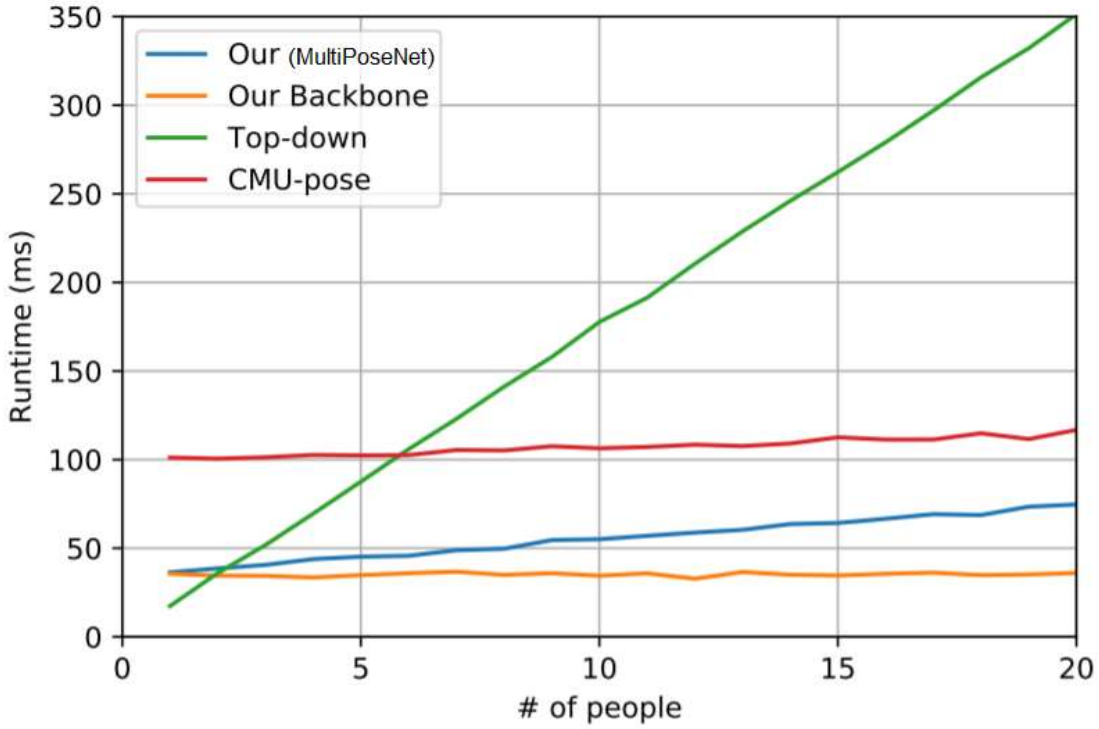
FPS results obtained on a *GTX1080Ti* GPU, images in average contain 3 people.

Images credit: M. Kocabas et al. "MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network"

Multi-person Pose Estimation: The Trade-off

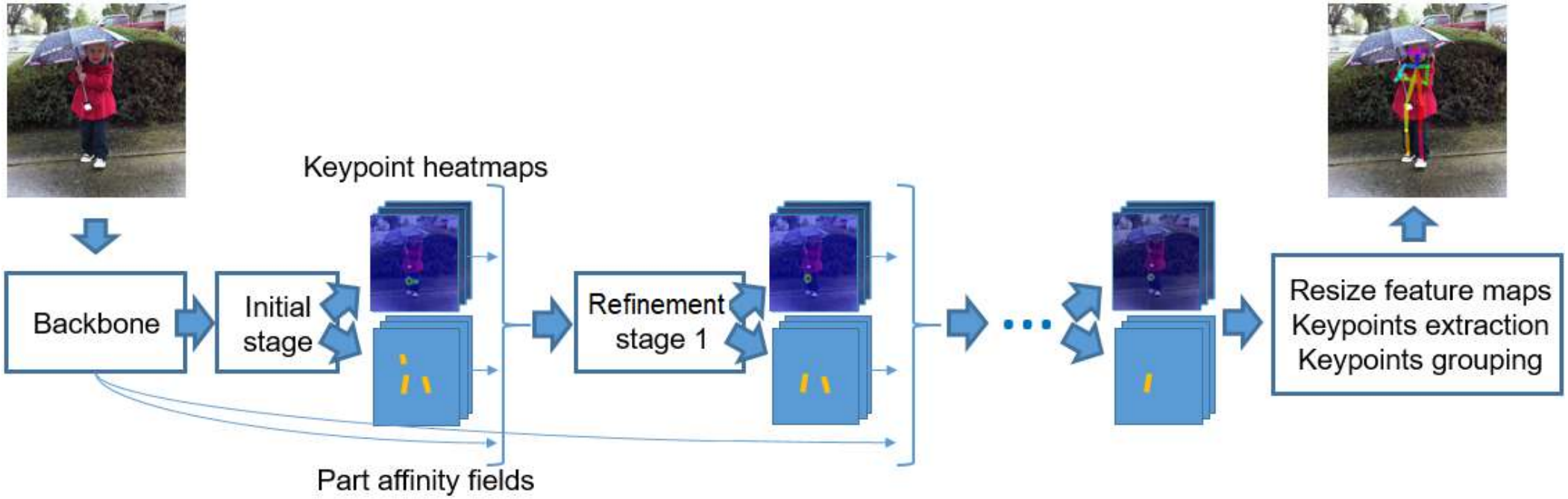
		FPS	AP	AP_{50}	AP_{75}
BU	MultiPoseNet	23	**69.6**	86.3	**76.6**
BU	Newell et al.	6	65.5	**86.8**	72.3
BU	CMU-Pose	10	61.8	84.9	67.5
TD	Megvii	-	73.0	91.7	80.9
TD	CFN	3	72.6	86.7	69.7
TD	Mask R-CNN	5	69.2	90.4	76.0
TD	SJTU	0.4	68.8	87.5	75.9
TD	GRMI-2017	-	66.9	86.4	73.6
TD	G-RMI-2016	-	60.5	82.2	66.2

FPS results obtained on a *GTX1080Ti* GPU, images in average contain 3 people.



Images credit: M. Kocabas et al. "MultiPoseNet: Fast Multi-Person Pose Estimation using Pose Residual Network"

Optimization: Analysis of the OpenPose



Post-processing profile:

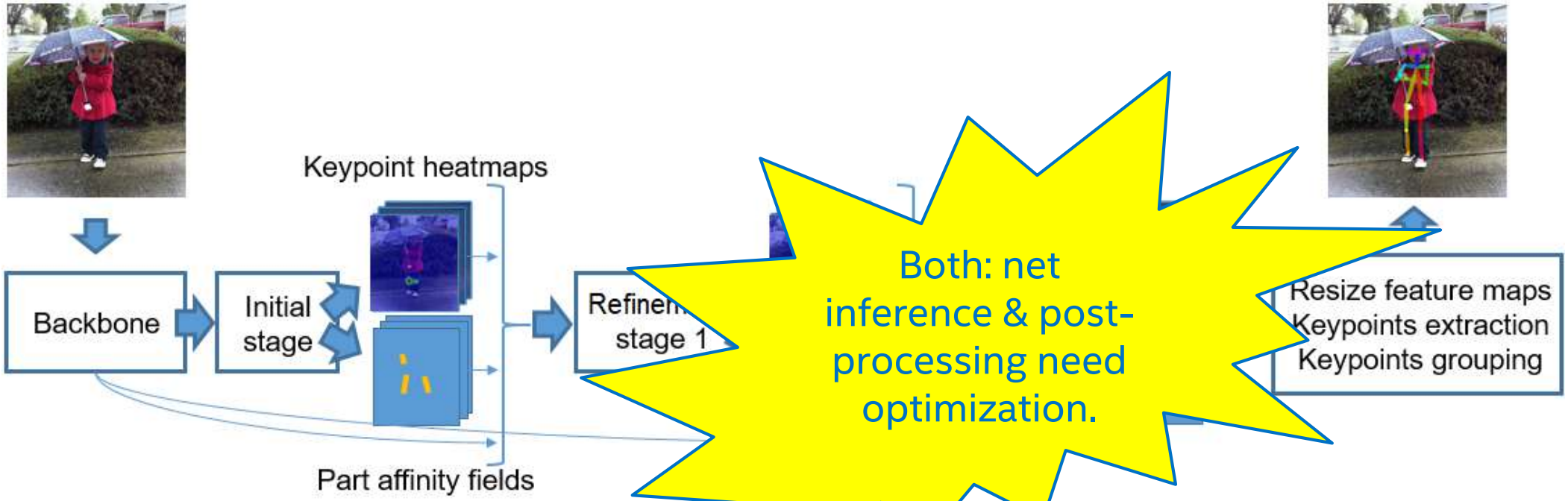
	Resize feature maps	Extract keypoints	Group keypoints	Total
Fps	10.5	1.81	454	**1.54**

Network inference/post-processing fps:

	CPU
Baseline	0.95 (2.47/1.54)

Measured on Intel Core i7-6850K @ 3.6GHz, 6 cores

Optimization: Analysis of the OpenPose



Post-processing profile:

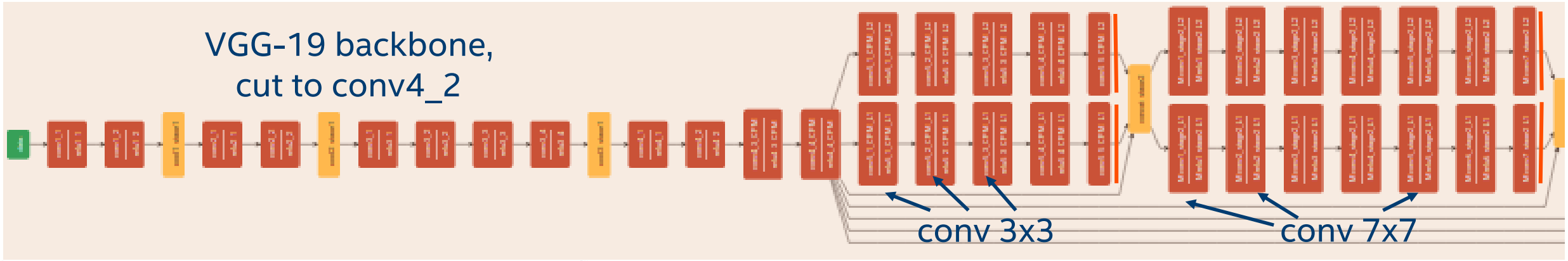
	Resize feature maps	Extract keypoints	Group keypoints	Total
Fps	10.5	1.81	454	**1.54**

Network inference/post-processing fps:

	CPU
Baseline	0.95 (2.47/1.54)

Measured on Intel Core i7-6850K @ 3.6GHz, 6 cores

Optimization: Analysis of the OpenPose

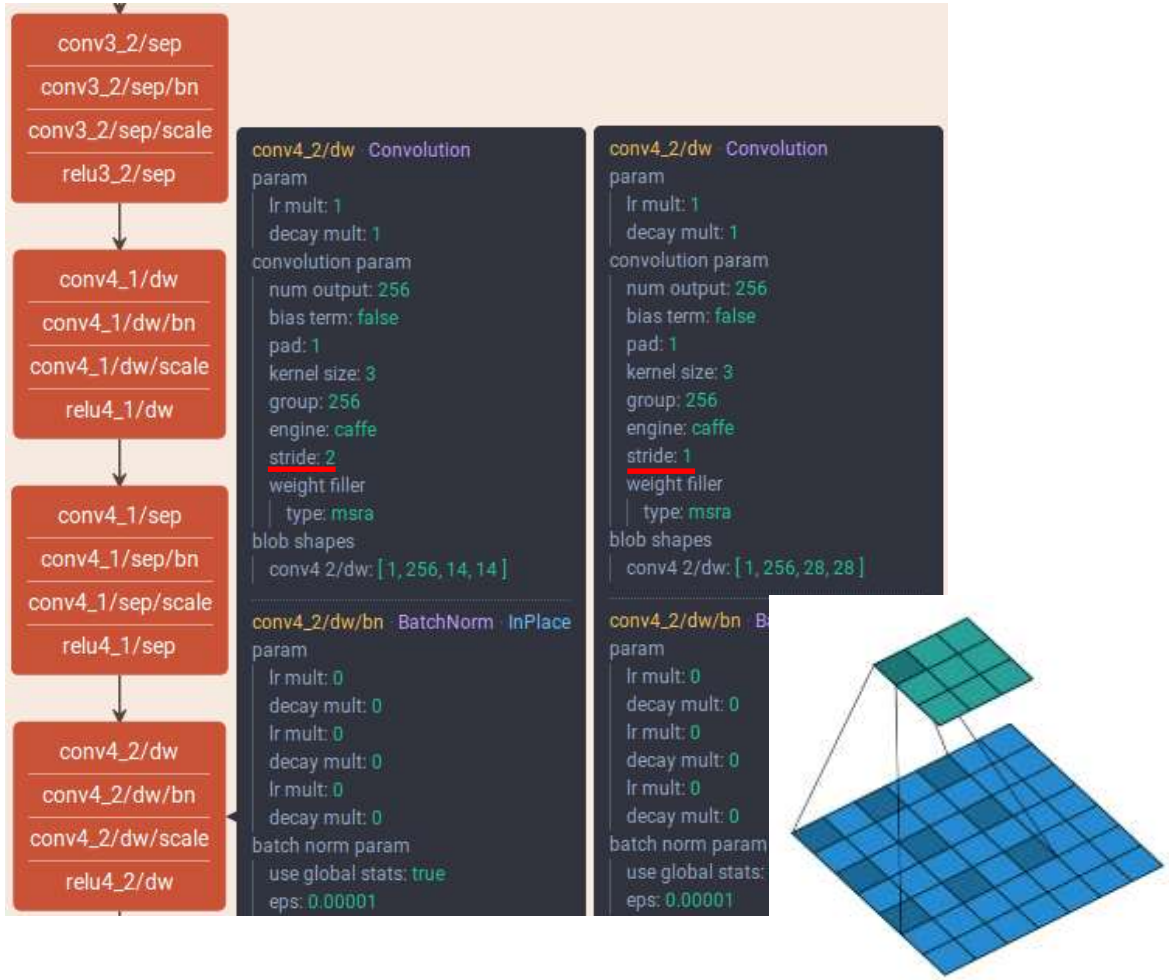


	AP, %	GFLOPs	GFLOPs total
Backbone	n/a	37.8	37.8
conv4_3	n/a	2.5	40.3
conv4_4	n/a	0.6	40.9
Initial stage	35.5	2.2	43.1
Refinement stage 1	43.4	18.6	61.7
Refinement stage 2	46.2	18.6	80.3
Refinement stage 3	47.4	18.6	98.9
Refinement stage 4	48.1	18.6	117.5
Refinement stage 5	48.6	18.6	136.1

Optimization: Lightweight Backbone

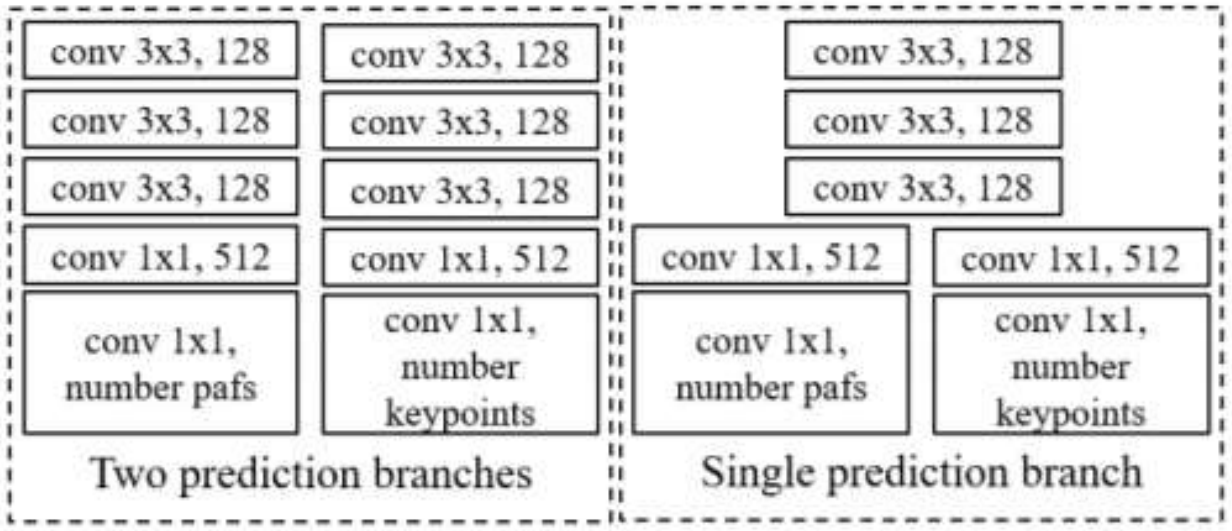
Evaluated MobileNet v1 and v2:

	AP, %	GFLOPs
MobileNet v1 *(cut to conv4_1)*	37.9	23.3
Dilated MobileNet v1 *(cut to conv5_5)*	42.8	27.7
Dilated MobileNet v1 *(cut to conv5_6)*	43.2	31.3
Dilated MobileNet v2 *(cut to conv6_3)*	39.6	27.2

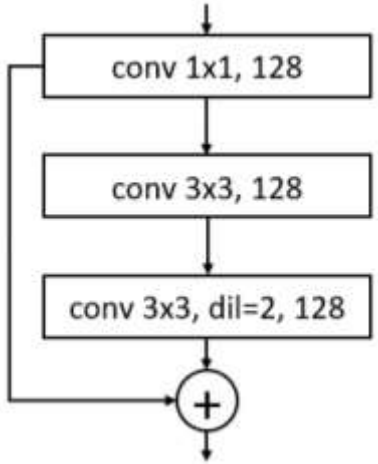


Optimization: Lightweight Stages

Single prediction branch



Lightweight block instead of convolutions with 7x7 kernel



Optimization: Result of Lightweight Network Design

	AP, %	GFLOPs	GFLOPs total
Dilated MobileNet v1	n/a	3.7	3.7
conv4_3	n/a	0.3	4
conv4_4	n/a	0.3	4.3
Initial stage	35	1.3	5.6
Refinement stage 1	**42.8**	3.4	**9**

Optimization: Fast Post-processing

Original steps:

1. Resize output feature maps ~~to image size~~ ← Bottleneck (10.5 fps)
2. Extract keypoints ← Cleaned code, parallelized with cv::parallel_for_ (made also fast)
3. Group keypoints into poses ← Already fast (454 fps)

Final network inference/post-processing fps on a video with more than 20 estimated poses:

	NUC (FP16)	CPU (FP32)
Baseline	1.17 (3.92/1.66)	0.95 (2.47/1.54)
Proposed	**28** (33/160)	**26** (33/125)

Network input is set to 456x256
CPU: Intel Core i7-6850K @ 3.6GHz, 6 cores
NUC: Intel NUC6i7KYB, GPU Iris Pro Graphics P580 (GT4e)

Q&A

- Available in OpenVINO Toolkit: `inference_engine/samples/human_pose_estimation_demo`.
- Paper on arXiv.org: “Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose”, https://arxiv.org/pdf/1811.12004.pdf.
- Training code in PyTorch: released.