Multi-Instance Pose Networks: Rethinking Top-Down Pose Estimation

1. Multi-Instance Modulation Block (MIMB) Code

In this section, we describe the code of MIMB in `PyTorch`. The code in Listing. 1 outlines the details of functions $\mathbf{F}_{sq}$, $\mathbf{F}_{ex}$ and $\mathbf{F}_{em}$. $\mathbf{F}_{sq}$ is a simple global average pool and $\mathbf{F}_{ex}$ and $\mathbf{F}_{em}$ are two-layered neural networks. MIMB can be incorporated in any existing feature extraction backbone, with a relatively simple (< 15 lines) code change.

```
class MIMB(nn.Module):
  def __init__(self, num_channels=c, reduce=r):
    super(MIMB, self).__init__()
    self.F_sqn = nn.AdaptiveAvgPool2d(1)

    self.F_ex = nn.Sequential(
        nn.Linear(c, c // r, bias=False),
        nn.ReLU(inplace=True),
        nn.Linear(c // r, c, bias=False),
        nn.Sigmoid()
    )

    self.F_em = nn.Sequential(
        nn.Linear(2, c // r),
        nn.BatchNorm1d(c // r),
        nn.ReLU(inplace=True),
        nn.Linear(c // r, c),
        nn.Sigmoid()
    )
        return

    def forward(self, x, lambda):
        b, c, _, _ = x.size()
        y = self.F_sqn(x).view(b, c)
        y = self.F_ex(y).view(b, c, 1, 1)

        z = self.F_em(lambda).view(b, c, 1, 1)

        out = x * y.expand_as(x) * z.expand_as(x)
        return out
```

Listing 1: Code for MIMB.

2. Implementation Details

We merge all the instances from $\lambda = 0$ to $N - 1$ and then apply oks-nms. During the merger, we discount the confidence of the instance $\lambda = i$ by γ^i. As the primary instance ($\lambda = 0$) is always centralized in the input, this confidence discounting avoids suppression of a high resolution primary predictions by a low resolution $\lambda > 0$ prediction. We use $\gamma = 0.9$ in all our experiments.

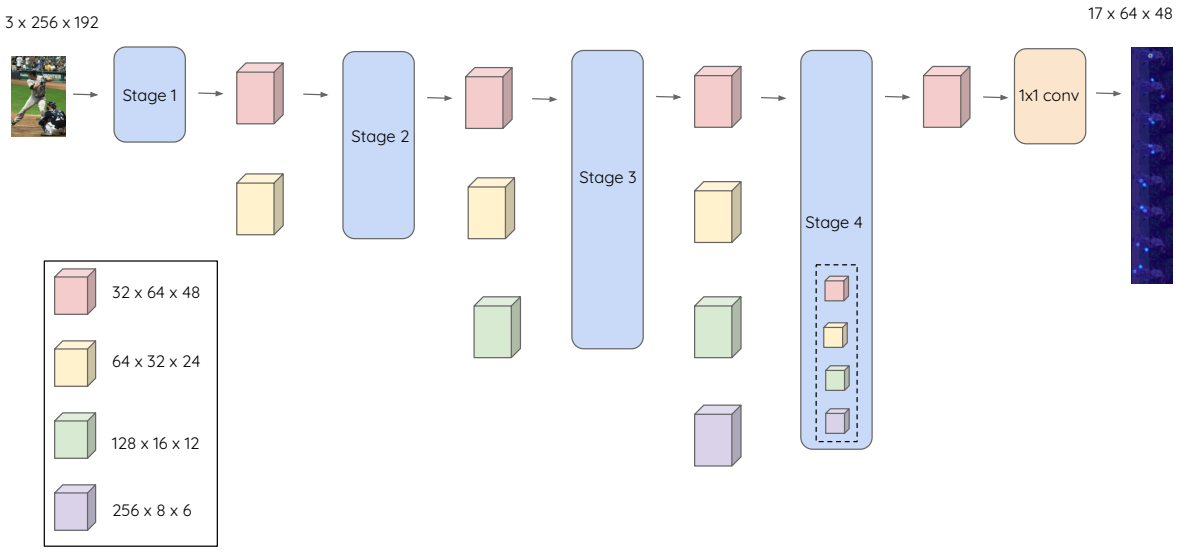

Figure 1: Illustration of HRNet-W32 backbone at input resolution 256×192. The blue blocks depict the four stages in the architecture.

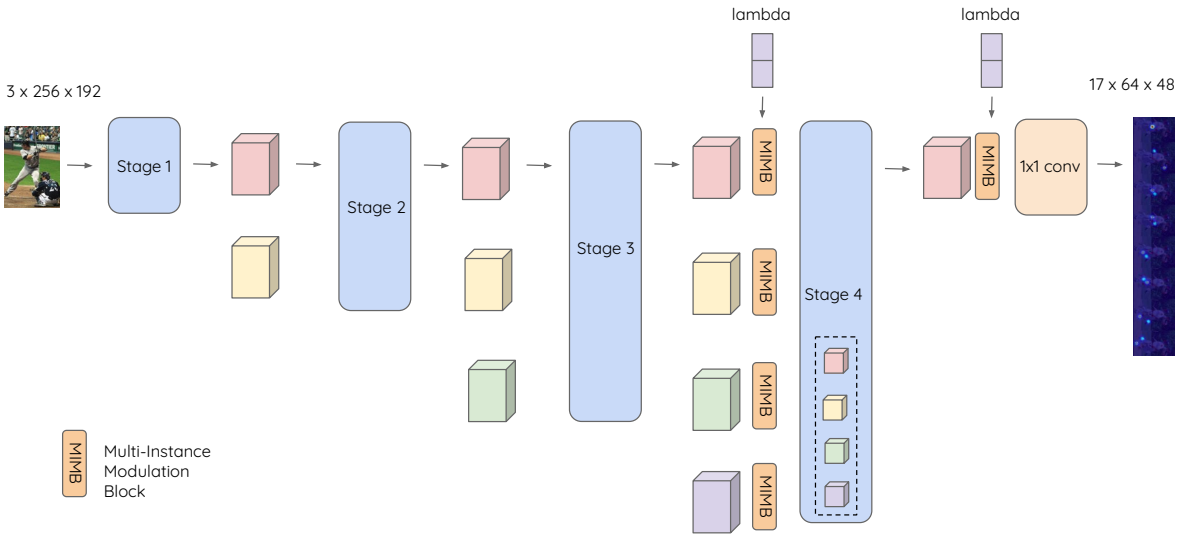

Figure 2: Illustration of MIPNet with HRNet-W32 backbone at input resolution 256×192. We insert 5 MIMBs into the HRNet, 4 MIMBs after Stage 3 and 1 MIMB after Stage 4.

MIPNet-HRNet: Figure. 1 shows the architecture details of HRNet [15]. For simplicity, we only show backbone HRNet-W32 at input size 256×192, other HRNet backbones follow similar pipeline. Figure. 2 shows the architecture of MIPNet, where multiple MIMBs are inserted at various stages.

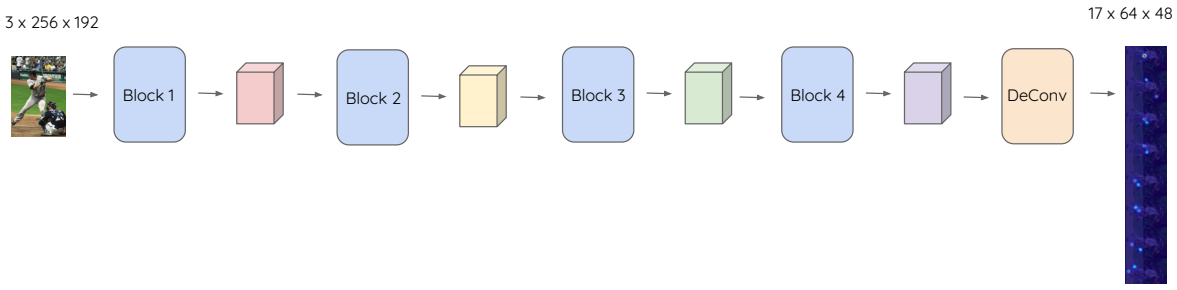

Figure 3: Illustration of SimpleBaseline architecture. The blue blocks represent the four blocks in the encoder of SimpleBaseline.

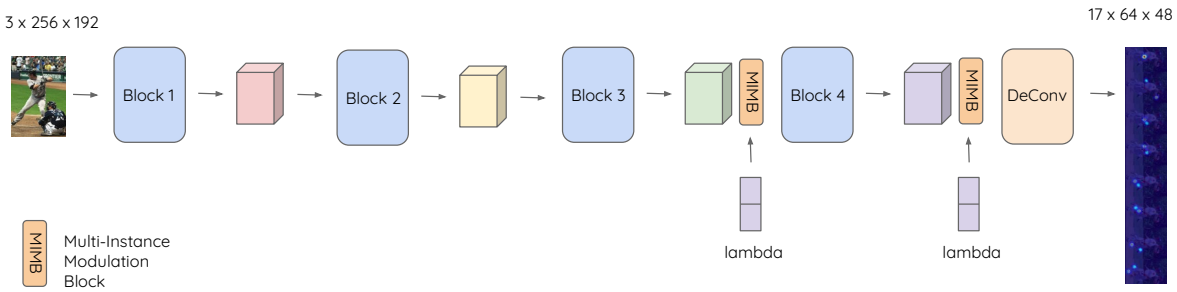

Figure 4: Illustration of MIPNet with SimpleBaseline architecture. We insert 2 MIMBs into the encoder after Block 3 and Block 4.

MIPNet-SimpleBaseline: Figure. 3 shows the architecture details of SimpleBaseline [16]. Figure. 4 shows the architecture of MIPNet, where multiple MIMBs are inserted in the encoder of the pose estimator.

3. Diminishing Returns with $\mathbf{N = 3, 4}$

We observed a small improvement in AP using $N = 3$ and $N = 4$ on top of $N = 2$ respectively on the datasets when evaluated using ground-truth bounding boxes. This is consistent with the fact that most datasets have very few examples with three or more ground-truth pose instances per bounding box (Refer data statistics in the paper). Note, on the more occluded OCHuman dataset, increasing N gives better performance.

Inference	COCO	CrowdPose	OCHuman
HRNet	78.1	72.8	65.0
MIPNet, $N = 2$	**78.8**	73.7	74.1
MIPNet, $N = 3$	78.4	**73.9**	74.3
MIPNet, $N = 4$	78.6	73.7	**74.7**

Table 1: Performance of MIPNet on `val` sets using ground truth bounding boxes with increasing N. We use the backbone W48 with image resolution 384×288, and compare with the same HRNet configuration. By default, HRNet only predicts a single instance.

4. Additional Results on COCO, CrowdPose and OCHuman

4.1. Additional results on COCO

Table 2 shows additional metrics for comparison between MIPNet ($N = 2$) and various baseline architectures on COCO `val` dataset using ground truth bounding boxes for evaluation. We also report GFLOPs for each model. Note that for all baseline evaluations for HRNet and SimpleBaseline, we follow the same protocol as outlined in the respective papers [15, 16].

Table 3 shows additional metrics for comparison between MIPNet and HRNet on COCO `val` and `test` datasets using Faster-RCNN bounding boxes, provided by authors of [15] for evaluation. For HRNet, numbers are reported from their paper [15] (some metrics are not provided).

4.2. Additional results on CrowdPose

Table 5 compares MIPNet to HRNet for various widths and image resolutions on CrowdPose `val` dataset using ground truth bounding boxes. Similarly, Table. 6 compares MIPNet to HRNet on CrowdPose `val` and `test` datasets using Faster-RCNN bounding boxes [14]. Note that, commensurate with increasing percentage of occlusions in the dataset, MIPNet consistently does better than HRNet in most metrics on both datasets.

4.3. Additional results on OCHuman

Table 7 shows our detailed evaluations on the OCHuman `val` dataset using ground truth bounding boxes. As can be seen, MIPNet outpeforms HRNet and SimpleBaseline on **all** metrics, with a maximum improvement of 10.5 AP over SimpleBaseline ($\mathrm{R} - 50, 384 \times 288$) and 9.1 AP over HRNet ($\mathrm{H} - 48, 384 \times 288$).

Similarly, Table. 8 shows detailed results on the OCHuman `val` and `test` datasets using Faster-RCNN bounding boxes. MIPNet achieves a **state-of-the-art** 42.5AP across *both* top-down and bottom-up pose estimation networks, to the best of our knowledge. We show a 4.2 AP improvement over HRNet on `val` dataset and a 5.3 AP improvement over HRNet on `test` dataset in this case.

4.4. Robustness to Bounding Box Confidence

Table 9 illustrates the number of Faster-RCNN bounding boxes as a function of minimum bounding box confidence. Notice that a majority of all available bounding boxes (min. confidence = 0.0) have confidence < 0.4.

Method	Arch	Input Size	GFLOPS	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
SBL	R-50	256×192	8.90	72.4	91.5	80.4	69.7	76.5	75.6	93.0	82.3	72.3	80.4
MIPNet	R-50	256×192	16.3	**73.3 (+0.9)**	**93.3**	**81.2**	**70.6**	**77.6**	**76.7**	**94.2**	**83.4**	**73.4**	**81.6**
SBL	R-101	256×192	12.4	73.4	92.6	81.4	70.7	77.7	76.5	93.4	83.1	73.3	81.2
MIPNet	R-101	256×192	23.1	**74.1 (+0.7)**	**93.3**	**82.3**	**71.3**	**78.6**	**77.4**	**94.4**	**84.4**	**74.1**	**82.3**
SBL	R-152	256×192	15.7	74.3	92.6	**82.5**	71.6	78.7	77.4	93.8	84.2	74.4	82.0
MIPNet	R-152	256×192	29.1	**74.8 (+0.5)**	**93.3**	82.4	**71.7**	**79.4**	**78.2**	**94.6**	**84.9**	**74.7**	**83.2**
SBL	R-50	384×288	20.2	74.1	92.6	80.5	70.5	79.6	76.9	93.2	82.7	73.0	82.6
MIPNet	R-50	384×288	36.7	**75.3 (+1.2)**	**93.4**	**82.4**	**72.0**	**80.4**	**78.4**	**94.6**	**84.7**	**74.6**	**83.8**
SBL	R-101	384×288	27.8	75.5	92.5	82.6	72.4	80.8	78.4	93.6	84.5	74.9	83.8
MIPNet	R-101	384×288	51.9	**76.0 (+0.5)**	**93.4**	**83.5**	**72.6**	**81.1**	**79.1**	**94.8**	**85.6**	**75.5**	**84.5**
SBL	R-152	384×288	35.5	76.6	92.6	83.6	**73.7**	81.3	79.3	94.0	85.3	75.9	84.5
MIPNet	R-152	384×288	65.4	**77.0 (+0.4)**	**93.5**	**84.3**	**73.7**	**81.9**	**80.0**	**94.9**	**86.1**	**76.4**	**85.3**
HRNet	H-32	256×192	7.10	76.5	93.5	83.7	73.9	80.8	79.3	94.5	85.8	76.2	84.1
MIPNet	H-32	256×192	9.80	**77.6 (+1.1)**	**94.4**	**85.3**	**74.7**	**81.9**	**80.6**	**95.6**	**87.1**	**77.3**	**85.4**
HRNet	H-48	256×192	14.6	77.1	93.6	84.7	74.1	81.9	79.9	94.5	86.3	76.5	85.1
MIPNet	H-48	256×192	20.7	**77.6 (+0.5)**	**94.4**	**85.4**	**74.6**	**82.1**	**80.6**	**95.6**	**87.0**	**77.3**	**85.5**
HRNet	H-32	384×288	16.0	77.7	93.6	84.7	74.8	82.5	80.4	94.4	86.4	77.0	85.6
MIPNet	H-32	384×288	22.1	**78.5 (+0.8)**	**94.4**	**85.7**	**75.6**	**83.0**	**81.4**	**95.6**	**87.4**	**78.0**	**86.3**
HRNet	H-48	384×288	32.9	78.1	93.6	84.9	75.3	83.1	80.9	94.7	86.7	77.5	86.0
MIPNet	H-48	384×288	46.5	**78.8 (+0.7)**	**94.4**	**85.7**	**75.5**	**83.7**	**81.6**	**95.5**	**87.5**	**78.0**	**86.8**

Table 2: Additional metrics for comparison between MIPNet and various architectures on COCO `val` set using ground-truth bounding boxes for evaluation.

Method	Arch	Input Size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
`val`												
HRNet	H-48	384×288	**76.3**	**90.8**	82.9	**72.3**	**83.4**	81.2	-	-	-	-
MIPNet	H-48	384×288	**76.3 (+0.0)**	90.6	**83.0**	72.1	83.3	**81.4**	94.2	87.6	76.	88.2
`test`												
Bottom-Up												
OpenPose [1]	-	-	61.8	84.9	67.5	57.1	68.2	66.5	-	-	-	-
AE [10]	-	-	65.5	86.8	72.3	60.6	72.6	70.2	-	-	-	-
PersonLab [11]	-	-	68.7	89.0	75.4	64.1	75.5	75.4	-	-	-	-
MultiPoseNet [9]	-	-	69.6	86.3	76.6	65.0	76.3	73.5	-	-	-	-
Top-Down												
MaskRCNN [5]	R-50	-	63.1	87.3	68.7	57.8	71.4	-	-	-	-	-
G-RMI [12]	R-101	353×257	64.9	85.5	71.3	62.3	70.0	69.7	-	-	-	-
CPN [2]	R-Incep	384×288	72.1	91.4	80.0	68.7	77.2	78.5	-	-	-	-
RMPE [4]	PyraNet	320×256	72.3	89.2	79.1	68.0	78.6	-	-	-	-	-
HRNet	H-48	384×288	75.5	**92.5**	**83.3**	**71.9**	**81.5**	**80.5**	-	-	-	-
MIPNet	H-48	384×288	**75.7 (+0.2)**	92.4	83.3	71.4	81.2	**80.5**	95.5	87.4	76.1	86.5

Table 3: Additional metrics for comparison between MIPNet and various architectures on COCO `val` and `test` set using Faster-RCNN bounding boxes for evaluation.

Method	Arch	AP	AP^{50}	AP^{75}
HRNet	H-32	76.5	93.5	83.7
MIPNet	H-32	$\mathbf{77.44 \pm 0.185}$	$\mathbf{94.42 \pm 0.039}$	$\mathbf{85.32 \pm 0.025}$
HRNet	H-48	77.1	93.6	84.7
MIPNet	H-48	$\mathbf{77.84 \pm 0.162}$	$\mathbf{94.44 \pm 0.079}$	$\mathbf{85.4 \pm 0.012}$

Table 4: We report mean $\pm$ std-dev of MIPNet over five runs on the COCO `val` set with ground-truth bounding boxes using 256×192 input resolution. H-@ stands for HRNet-W@ backbone.

Method	Arch	Input Size	AP	AP^{50}	AP^{75}	AR	AR^{50}	AR^{75}	AP^{easy}	AP^{med}	AP^{hard}
HRNet	H-32	256×192	70.0	91.0	76.3	73.9	92.6	79.4	**78.8**	70.3	61.7
MIPNet	H-32	256×192	**71.2 (+1.2)**	**91.9**	**77.4**	**76.1**	**94.4**	**81.7**	**78.8**	**71.5**	**63.8**
HRNet	H-48	256×192	71.3	91.1	77.5	74.8	92.4	80.5	80.5	71.4	62.5
MIPNet	H-48	256×192	**72.8 (+1.5)**	**92.0**	**79.2**	**77.4**	**94.8**	**83.0**	**80.6**	**73.1**	**65.2**
HRNet	H-32	384×288	71.6	91.1	77.7	75.0	92.6	80.4	80.4	72.1	62.6
MIPNet	H-32	384×288	**73.0 (+1.4)**	**91.8**	**79.3**	**77.9**	**94.8**	**83.4**	**80.7**	**73.3**	**65.5**
HRNet	H-48	384×288	72.8	**92.1**	78.7	76.3	93.3	81.4	**81.3**	73.3	64.0
MIPNet	H-48	384×288	**73.7 (+0.9)**	91.9	**80.0**	**78.4**	**94.8**	**84.0**	80.7	**74.1**	**66.5**

Table 5: Additional metrics for comparison between MIPNet and various architectures on CrowdPose `val` set using ground-truth bounding boxes for evaluation.

We compare the performance of MIPNet to HRNet as a function of varying minimum confidence on OCHuman `test` dataset in Fig. 6 and `val` dataset in Fig. 5 (also shown in the paper). MIPNet is much more stable w.r.t bounding box confidence thresholding, as compared to baseline networks like HRNet. We note that while MIPNet AP drops from 42.5 to 41.4 (1.1 AP drop) on `test` set at minimum confidence of 0.9, HRNet drops by more than 6 AP. This performance is consistent with the performance on the `val` dataset (Fig. 4 in the paper).

5. Individual Instance Performance

It is interesting to compare the performance of each individual instances predicted by MIPNet in isolation. Since $\lambda = 0$ correspond to the primary instance (centered on the person), only using the primary instance for inference is expected to give better results compared to only using $\lambda = 1$ instance during inference. In addition, we also expect $\lambda = 0$ instance to provide similar performance as baseline top-down network, if used in isolation. Table 10 shows the performance of each

Method	Arch	Input Size	AP	AP^{50}	AP^{75}	AR	AR^{50}	AR^{75}	AP^{easy}	AP^{med}	AP^{hard}
				`val`							
HRNet	H-48	384 × 288	68.0	85.5	73.4	76.6	93.8	81.9	**77.4**	68.8	57.8
MIPNet	H-48	384 × 288	**68.8 (+0.8)**	**85.9**	**74.5**	**78.1**	**94.5**	**83.6**	77.1	**69.4**	**59.8**
				`test`							
				Bottom-Up							
OpenPose [1]	-	-	-	-	-	-	-	-	62.7	48.7	32.3
HigherHRNet [3]	HH-48	640 × 640	67.6	87.4	72.6	-	-	-	75.8	68.1	58.9
HghHRNet + UDP [7]	HH-48	640 × 640	68.2	88.0	72.9	-	-	-	76.6	68.7	59.9
				Top-Down, YOLO-v3							
MaskRCNN [5]	R-101	-	57.2	83.5	60.3	-	-	-	-	-	-
SimpleBaseline [16]	R-101	-	60.8	81.4	65.7	-	-	-	-	-	-
AlphaPose+ [13]	R-101	-	27.5	40.8	29.9	-	-	-	-	-	-
OPEC-Net [13]	R-101	-	**70.6**	86.8	75.6	-	-	-	-	-	-
MIPNet	R-101	384 × 288	68.1	85.2	73.8	75.1	92.3	79.2	74.6	69.2	53.4
				Top-Down, Faster-RCNN							
HRNet	H-48	384 × 288	69.3	**86.9**	74.7	77.3	94.2	82.5	77.7	70.6	57.8
MIPNet	H-48	384 × 288	70.0 (+0.7)	86.8	**75.7**	**78.8**	**94.9**	**84.3**	**78.1**	**71.1**	**59.4**

Table 6: Additional metrics for comparison between MIPNet and various architectures on CrowdPose `val` and `test` set using Faster-RCNN and YOLO-v3 bounding boxes for evaluation.

Method	Arch	Input Size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
SimpleBaseline	R-50	256 × 192	56.3	76.1	61.2	66.4	56.3	61.0	78.0	65.9	70.0	61.0
MIPNet	R-50	256 × 192	**64.4 (+8.1)**	**86.0**	**70.4**	**66.8**	**64.6**	**72.3**	**91.5**	**78.5**	**71.4**	**72.3**
SimpleBaseline	R-101	256 × 192	60.5	77.2	66.6	68.3	60.5	64.7	79.6	70.1	72.9	64.7
MIPNet	R-101	256 × 192	**68.2 (+7.7)**	**87.4**	**75.1**	**67.0**	**68.2**	**75.5**	**92.9**	**82.1**	**74.3**	**75.5**
SimpleBaseline	R-152	256 × 192	62.4	78.3	68.1	**68.3**	62.4	66.5	80.2	71.8	**74.3**	66.5
MIPNet	R-152	256 × 192	**70.3 (+7.9)**	**88.6**	**77.9**	66.9	**70.2**	**77.0**	**93.0**	**84.1**	72.9	**77.0**
SimpleBaseline	R-50	384 × 288	55.8	74.8	60.4	64.7	55.9	60.7	78.0	65.2	**71.4**	60.7
MIPNet	R-50	384 × 288	**66.3 (+10.5)**	**87.5**	**72.2**	**66.0**	**66.3**	**74.1**	**92.7**	**80.3**	**71.4**	**74.1**
SimpleBaseline	R-101	384 × 288	61.6	77.2	66.6	62.1	61.6	65.8	79.4	70.5	**72.9**	65.8
MIPNet	R-101	384 × 288	**70.3 (+8.7)**	**88.4**	**77.1**	**64.1**	**70.4**	**77.7**	**93.4**	**84.0**	**72.9**	**77.7**
SimpleBaseline	R-152	384 × 288	64.2	78.3	69.1	66.5	64.2	68.1	80.4	73.0	**74.3**	68.1
MIPNet	R-152	384 × 288	**72.4 (+8.2)**	**89.5**	**79.5**	**67.7**	**72.5**	**79.6**	**94.1**	**86.2**	71.4	**79.6**
HRNet	H-32	256 × 192	63.1	79.4	69.0	64.2	63.1	67.3	81.9	72.4	68.6	67.3
MIPNet	H-32	256 × 192	**72.5 (+9.4)**	**89.2**	**79.4**	**65.1**	**72.6**	**79.1**	**93.6**	**85.2**	**71.4**	**79.1**
HRNet	H-48	256 × 192	64.5	79.4	70.1	65.1	64.5	68.5	81.6	73.7	68.6	68.5
MIPNet	H-48	256 × 192	**72.2 (+7.7)**	**89.5**	**78.7**	**66.5**	**72.3**	**79.2**	**94.2**	**85.4**	**70.0**	**79.2**
HRNet	H-32	384 × 288	63.7	78.4	69.0	64.3	63.7	67.6	80.8	72.6	**70.0**	67.6
MIPNet	H-32	384 × 288	**72.7 (+9.0)**	**89.6**	**79.6**	**66.5**	**72.7**	**79.7**	**94.3**	**86.1**	**70.0**	**79.7**
HRNet	H-48	384 × 288	65.0	78.4	70.3	68.4	65.0	68.8	80.6	73.4	71.4	68.8
MIPNet	H-48	384 × 288	**74.1 (+9.1)**	**89.7**	**80.1**	**68.4**	**74.1**	**81.0**	**94.4**	**87.0**	**72.9**	**81.0**

Table 7: Additional metrics for comparison between MIPNet and various architectures on OCHuman `val` set using ground-truth bounding boxes for evaluation.

individual instance mode of MIPNet with HRNet-W48 backbone at input size 384×288 on various datasets, using ground truth bounding boxes. Note that when using only a single hypothesis from MIPNet for inference, performance of primary instance ($\lambda = 0$) is similar to HRNet. When using multiple instances during inference, we get an improvement of 8.4 AP (65.7 to 74.1 AP) on the OCHuman dataset.

6. Ablation: MIMB

In this section, we study the effect of ablation for MIMB. As outlined in the paper, MIMB consists of three operations *squeeze* $\mathbf{F}_{sq}$, *excite* $\mathbf{F}_{ex}$ and *embed* $\mathbf{F}_{em}$. Of the three operations, the *embed* operation $\mathbf{F}_{em}$ consumes the λ parameter that we

Method	Arch	Input Size	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
`val`												
HRNet	H-48	384 × 288	37.8	50.6	40.5	**3.8**	40.4	69.9	89.0	73.9	67.1	69.9
MIPNet	H-48	384 × 288	**42.0 (+4.2)**	**51.2**	**45.6**	3.2	**43.5**	**82.5**	**96.7**	**88.5**	**71.4**	**82.5**
`test`												
Bottom-Up												
AE [10]	Hourglass	-	29.5	-	-	-	-	-	-	-	-	-
AE-multiscale [10]	Hourglass	-	32.8	-	-	-	-	-	-	-	-	-
HGG [8]	Hourglass	-	34.8	-	-	-	-	-	-	-	-	-
HGG-multiscale [8]	Hourglass	-	36.0	-	-	-	-	-	-	-	-	-
Top-Down, YOLO-v3												
MaskRCNN [5]	R-101	-	20.2	33.2	24.5	-	-	-	-	-	-	-
SimpleBaseline	R-101	-	24.1	37.4	26.8	-	-	-	-	-	-	-
AlphaPose+ [13]	R-101	-	27.5	40.8	29.9	-	-	-	-	-	-	-
OPEC-Net [13]	R-101	-	29.1	41.3	31.4	-	-	-	-	-	-	-
MIPNet	R-101	384 × 288	35.0	44.1	36.1	-	35.1	74.5	88.6	79.1	-	72.8
Top-Down, FasterRCNN												
HRNet	H-48	384 × 288	37.2	46.7	40.0	-	39.8	78.0	93.5	83.7	-	78.0
MIPNet	H-48	384 × 288	**42.5 (+5.3)**	**51.8**	**46.3**	-	**44.1**	**83.0**	**97.1**	**89.2**	-	**83.0**

Table 8: Additional metrics for comparison between MIPNet and various architectures on OCHuman `val` and `test` set using Faster-RCNN and YOLO-v3 bounding boxes for evaluation.

Min. BB Confid.	OCHuman	
	`val`	`test`
0.0	30637	26992
0.1	22247	19704
0.2	16273	14613
0.3	13603	12216
0.4	11944	10767
0.5	10654	9645
0.6	9626	8697
0.7	8699	7880
0.8	7768	7018
0.9	6644	5989
0.99	4416	3883

Table 9: Number of Faster-RCNN bounding boxes greater than a given confidence score.

Inference	COCO	CrowdPose	OCHuman
HRNet	78.1	72.8	65.0
MIPNet (SIP, $\lambda = 1$)	55.8	42.2	41.4
MIPNet (SIP, $\lambda = 0$)	78.3	72.7	65.7
MIPNet (MIP)	**78.8**	**73.7**	**74.1**

Table 10: Performance of each individual instances of MIPNet on `val` sets using ground truth bounding boxes. We use the backbone W48 with image resolution 384 × 288, and compare with the same HRNet configuration. By default, HRNet only predicts a single instance.

pass as additional input to MIMB. In Tab. 11, we show the effect of only using the embed block by disabling $\mathbf{F}_{sq}$ and $\mathbf{F}_{ex}$, in the first row for both COCO and OCHuman `val` datasets. Note that these numbers are lower than corresponding experiments that use $\mathbf{F}_{sq}$ and $\mathbf{F}_{ex}$ operations, by 0.3 AP for COCO (Tab. 2, last row in paper) and 3.3 AP (Tab. 4, last row in paper) for OCHuman `val` datasets. This confirms that all three operations contribute to MIMB, and therefore to MIPNet. We further study the effect of varying the intermediate linear layer within $\mathbf{F}_{sq}$ and $\mathbf{F}_{ex}$, which is controlled by the `reduce` parameter [6]

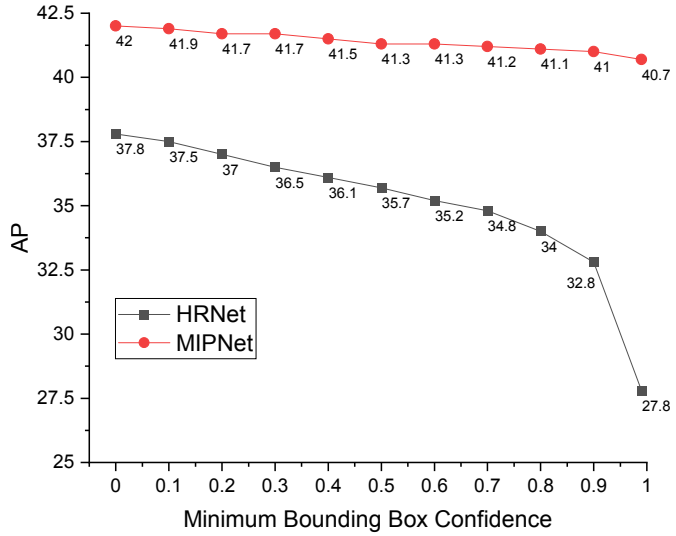

Figure 5: Unlike HRNet, MIPNet maintains a stable performance as a function of detector confidence for selecting input bounding boxes. Results are shown using HRNet-W48-384×288 evaluated on the `val` set of OCHuman.

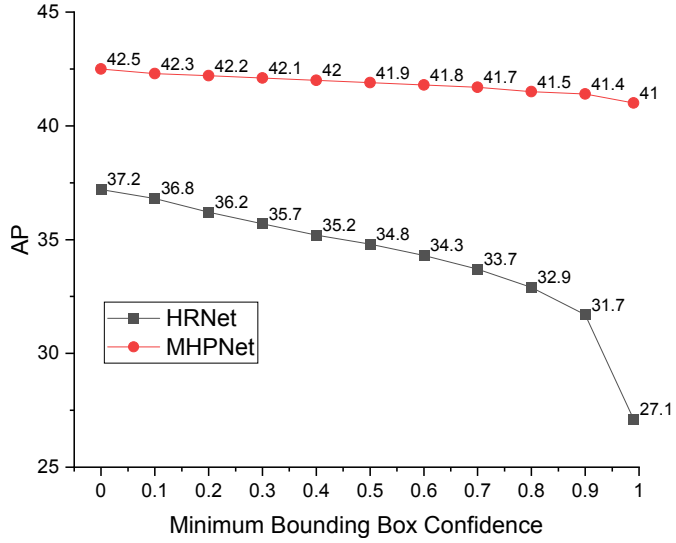

Figure 6: Similar to Figure 5 we show results on the `test` set of OCHuman.

in Listing 1. While all the results reported in the paper use the default value of `reduce=4`, we show that `reduce=2` and `reduce=1` show comparable results.

7. OCPose Dataset

For completeness, we also benchmark MIPNet on another occlusion specific OCPose dataset [13]. OCPose is a larger dataset than OCHuman with pose annotations of occluded humans. It contains $9K$ images and 18000 persons labeled with 12 keypoints. The number of examples with occlusion IoU > 0.5 is 78% for OCPose [13]. Each image in the dataset, is annotated with exactly *two* person keypoints. Further, both the persons have the same bounding box, this is in contrast to tight fitting bounding box annotations in the datasets like COCO, Crowdpose and OCHuman. This results in inflated occlusion levels for the OCPose dataset in comparison to the OCHuman dataset reported in [13] (refer its Table 1).

Table. 12 reports the MIPNet's results on the OCPose dataset [13] with custom `train:test` splits as the OPEC-Net [13] splits are not released. All the models are trained on the COCO dataset and evaluated on the `test` set of OCPose. We evaluate

Method	Arch	Ablation	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
			COCO									
MIPNet	H-48	only embed	78.5	94.4	85.5	75.3	83.5	81.4	95.8	87.5	77.8	86.7
MIPNet	H-48	reduce=1	78.8	94.4	85.8	75.5	83.6	81.5	95.4	87.8	78.0	86.6
MIPNet	H-48	reduce=2	78.8	94.4	85.6	75.8	83.6	81.7	95.7	87.7	78.3	86.8
MIPNet	H-48	reduce=4	**78.8**	**94.4**	**85.7**	**75.5**	**83.7**	**81.6**	**95.5**	**87.5**	**78.0**	**86.8**
			OCHuman									
MIPNet	H-48	only embed	70.8	89.8	77.5	65.7	70.9	77.9	94.2	84.2	68.6	77.9
MIPNet	H-48	reduce=1	74.4	90.7	80.9	66.9	74.4	81.2	95.1	87.2	70.0	81.2
MIPNet	H-48	reduce=2	74.0	90.1	80.3	63.6	74.0	80.7	94.5	86.7	68.6	80.7
MIPNet	H-48	reduce=4	**74.1**	**89.7**	**80.1**	**68.4**	**74.1**	**81.0**	**94.4**	**87.0**	**72.9**	**81.0**

Table 11: We illustrate different ablations of MIMB. For MIPNet with backbone W48 on resolution 384×288, we train models with varying capacity for *squeeze* $\mathbf{F}_{sq}$ and *excite* $\mathbf{F}_{ex}$ operations. When both operations are disabled, and only *embed* operation $\mathbf{F}_{embed}$ is used within MIMB, we get sub-optimal results on both COCO `val` (0.3 AP drop) and OCHuman `val` (3.6 AP drop) datasets (first row of each dataset). When *squeeze* and *excite* operations are employed, we get a good performance boost, especially on the OCHuman `val` dataset. All results in the paper employ `reduce=4` (bold).

Method	AP	AP^{50}	AP^{75}	AP^{M}	AP^{L}	AR	AR^{50}	AR^{75}	AR^{M}	AR^{L}
ground truth bounding box										
HRNet	34.2	48.2	36.7	36.6	34.1	36.8	48.9	39.5	38.3	36.8
MIPNet ($\lambda = 0$)	34.6	49.2	36.7	37.0	34.6	37.3	49.2	39.9	36.7	37.3
MIPNet ($\lambda = 1$)	23.8	34.9	25.2	30.6	24.0	28.6	39.7	30.0	45.0	28.6
MIPNet	**49.7 (+15.5)**	**72.3**	**53.0**	**59.4**	**49.7**	**56.4**	**74.8**	**60.1**	**70.0**	**56.4**
ground truth bounding box - tight fitting										
HRNet	47.7	74.6	50.1	35.8	47.7	53.0	77.0	56.4	41.3	53.1
MIPNet ($\lambda = 0$)	46.6	73.2	49.1	33.7	46.9	52.5	76.2	55.8	37.0	52.6
MIPNet ($\lambda = 1$)	26.5	51.2	23.9	10.7	26.9	36.4	61.4	35.7	32.3	36.4
MIPNet	**49.3 (+1.6)**	**77.3**	**51.9**	**33.6**	**49.5**	**56.9**	**82.8**	**60.7**	**37.3**	**57.1**

Table 12: Results on the OCPose `val` set. All the evaluations use the HRNet-W48 backbone at 348×288 image resolution. We provide both evaluations, using the relaxed gt bounding boxes provided by the OCPose and the tight fitting gt bounding box. The tight fitting bounding box is using the keypoint annotations.

only on the common keypoints between the both datasets.

8. Qualitative Results

Figure 7 and Figure 8 shows additional results on the OCHuman dataset, comparing MIPNet to HRNet. Note that in all of these cases, HRNet faces the problem of having highly overlapping bounding boxes because of the spatial proximity of humans in these images. Consequently, HRNet picks one dominant person and detects key-points on the same person within both bounding box instances. In contrast, MIPNet can clearly identify the correct set of key-points and associate them to the correct human(s) in each example. We especially want to point attention to the cases where people are dancing in tandem, or tackling each other while playing sports. Such situations produce extremely complicated occlusions. However, MIPNet is able to successfully attribute the correct key-points to each human in the input bounding boxes in such situations, highlighting its usefulness in occlusion scenarios.

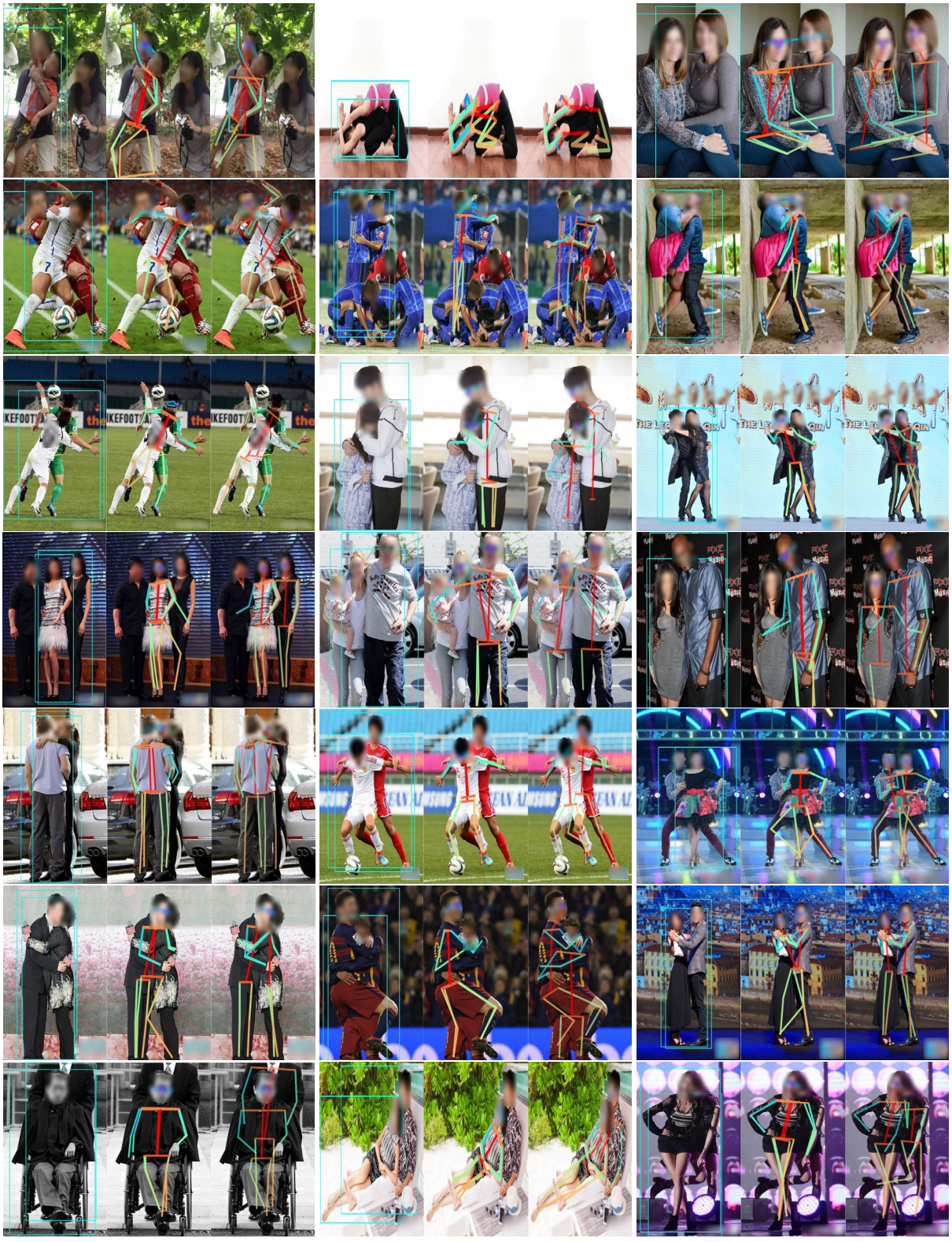

Figure 7: Qualitative results of MIPNet. Each image (left to right) shows input bounding boxes, HRNet predictions and MIPNet predictions.

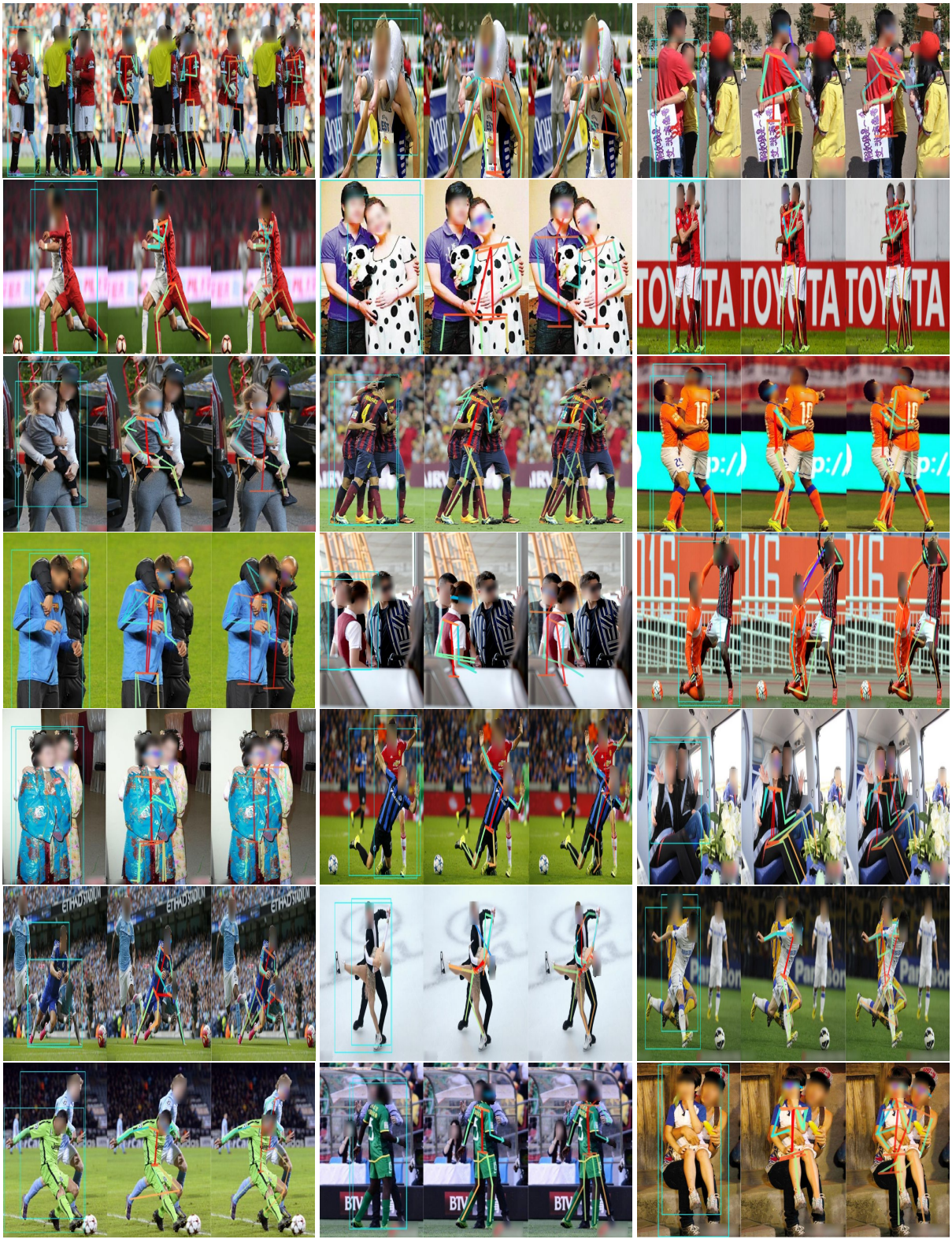

Figure 8: Qualitative results of MIPNet. Each image (left to right) shows input bounding boxes, HRNet predictions and MIPNet predictions.

References

[1] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. *arXiv preprint arXiv:1812.08008*, 2018. 5, 6

[2] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018. 5

[3] Bowen Cheng, Bin Xiao, Jingdong Wang, Honghui Shi, Thomas S Huang, and Lei Zhang. Higherhrnet: Scale-aware representation learning for bottom-up human pose estimation. *arXiv preprint arXiv:1908.10357*, 2019. 6

[4] Hao-Shu Fang, Shuqin Xie, Yu-Wing Tai, and Cewu Lu. Rmpe: Regional multi-person pose estimation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2334–2343, 2017. 5

[5] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B Girshick. Mask r-cnn. corr abs/1703.06870 (2017). *arXiv preprint arXiv:1703.06870*, 2017. 5, 6, 7

[6] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 7

[7] Junjie Huang, Zheng Zhu, Feng Guo, and Guan Huang. The devil is in the details: Delving into unbiased data processing for human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5700–5709, 2020. 6

[8] Sheng Jin, Wentao Liu, Enze Xie, Wenhai Wang, Chen Qian, Wanli Ouyang, and Ping Luo. Differentiable hierarchical graph grouping for multi-person pose estimation. In *European Conference on Computer Vision*, pages 718–734. Springer, 2020. 7

[9] Muhammed Kocabas, Salih Karagoz, and Emre Akbas. Multiposenet: Fast multi-person pose estimation using pose residual network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 417–433, 2018. 5

[10] Alejandro Newell, Zhiao Huang, and Jia Deng. Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in neural information processing systems*, pages 2277–2287, 2017. 5, 7

[11] George Papandreou, Tyler Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 269–286, 2018. 5

[12] George Papandreou, Tyler Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Chris Bregler, and Kevin Murphy. Towards accurate multi-person pose estimation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4903–4911, 2017. 5

[13] Lingteng Qiu, Xuanye Zhang, Yanran Li, Guanbin Li, Xiaojun Wu, Zixiang Xiong, Xiaoguang Han, and Shuguang Cui. Peeking into occluded joints: A novel framework for crowd pose estimation. *arXiv preprint arXiv:2003.10506*, 2020. 6, 7, 8

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 4

[15] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 2, 4

[16] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *Proceedings of the European conference on computer vision (ECCV)*, pages 466–481, 2018. 3, 4, 6