Detection, Pose Estimation and Segmentation for Multiple Bodies: Closing the Virtuous Circle

Miroslav Purkrabek and Jiri Matas
Visual Recognition Group
Czech Technical University in Prague
purkrmir@fel.cvut.cz

Figure 1. **The BBox-Mask-Pose (BMP) method**. Steps (A) – (D) repeat until no new detections found in step (A). Here, the background player is undetected in the first step (A1). BMP correctly fits the foreground player's pose (B1) which leads to correction of his segmentation and bbox (C1). After masking the foreground player (D1), the background player is detected (A2), his body correctly segemented and pose estimated. Right: BMP output. Note: the loop can be initialised with a bounding box (A), pose (B), or segmentation mask (C).

Abstract

Human pose estimation methods work well on isolated people but struggle with multiple-bodies-in-proximity scenarios. Previous work has addressed this problem by conditioning pose estimation by detected bounding boxes or keypoints, but overlooked instance masks. We propose to iteratively enforce mutual consistency of bounding boxes, instance masks, and poses. The introduced BBox-Mask-Pose (BMP) method uses three specialized models that improve each other's output in a closed loop. All models are adapted for mutual conditioning, which improves robustness in multi-body scenes. MaskPose, a new mask-conditioned pose estimation model, is the best among top-down approaches on OCHuman. BBox-Mask-Pose pushes SOTA on OCHuman dataset in all three tasks – detection, instance segmentation, and pose estimation. It also achieves SOTA performance on COCO pose estimation. The method is especially good in scenes with large instances overlap, where it improves detection by 39% over the baseline detector. With small specialized models and faster runtime, BMP is an effective alternative to large human-centered foundational models. Code and models will be published.

1. Introduction

Human pose estimation (HPE) plays a crucial role in tasks such as action detection and gesture recognition. It is a challenging problem, especially in multi-body scenes where people overlap, leading to issues such as merged bounding boxes or collapsed poses. Results on multi-body datasets are far from saturated, with state-of-the-art below 50% [40].

HPE approaches differ in how they use *conditioning* to guide predictions. Top-down methods [14, 36] are conditioned by bounding boxes, estimating poses within image crops defined by a detector, while single-stage and bottom-up methods [7, 31] are not conditioned at all. Pose-refining methods, such as BUCTD [40], introduce conditioning by prior pose estimates, iteratively refining predictions to improve accuracy.

Bounding boxes, masks, and poses represent different aspects of the human body, at different levels of granularity. Bounding boxes are easy to annotate and effective for detecting small instances but lack detail in crowded scenes. Segmentation masks are more detailed, but are costly to annotate and less common than bboxes. Poses provide anatomical detail, but are less effective for direct detection. Detectors, segmentators, and pose estimators are often trained on different datasets, and their combination increases variance in training data.

The proposed BBox-Mask-Pose (BMP) method extends conditioning to masks and integrates bboxes, masks, and poses into feedback loop (Fig. 1). BMP uses three specialized models that iteratively refine each other's output, allowing (re-)detection, segmentation, and pose estimation to achieve consistent results and performance gains, especially in multi-body scenarios. Specifically, the models are:

- Fine-tuned RTMDet [21]: A detector that could be conditioned by segmentation masks and ignores masked-out instances.
- MaskPose: ViTPose-based [36] pose estimation model conditioned by instance segmentation masks and bounding boxes. Its pose estimation is more robust in dense scenes than the previous top-down SOTA.
- SAM2 (Segment Anything Model) [24], conditioned by suitably selected pose keypoints, which improves segmentation capabilities and facilitates information passing between bounding box locations and pose estimates.

For pose estimation, BMP sets the new state-of-the-art performance on OCHuman, while also achieving SOTA performance on the COCO dataset. For detection and instance segmentation, BMP sets the new SOTA on OCHuman. None of the models in the loop were trained on OCHuman data and the same hyper parameters are used for evaluation on standard dataset (COCO) and multi-body scenes (OCHuman).

Ablations show that mutual conditioning creates a cycle that improves the accuracy of all components. The combination of an object detector with a model that "understands" the object structure could generalize to tasks where specialized models interpret the structure, as HPE models do for human anatomy. Moderately sized models (RTMDet-L [21], ViTPose-B [36], SAM-B+ [24]) are used in all experiments. The modular structure of BMP allows any component to be replaced by a larger or superior alternative to achieve improved performance.

In summary, **the main contribution** is the BBox-Mask-Pose loop, a new method for robust detection, segmentation and pose estimation in multi-body scenes. The core idea of BMP is to enforce mutual consistency between different representations of a human body. Experiments show that three specialized models are an effective alternative to data- and computationally expensive foundational models.

Other **technical contributions** are MaskPose, the first top-down HPE model conditioned by detected masks, the fine-tuned detector ignoring masked-out instances, and the keypoint selection algorithm for automated SAM prompting for pose-to-mask estimation.

2. Related work

Datasets. There are various datasets for 2D human pose estimation. Most notable are: COCO [17], MPII [3] and AIC [34]. Datasets like OCHuman [38] and CrowdPose [16] focus on multibody problems such as occlusion and self-occlusion. OCHuman is too small for large-scale training and is traditionally used only for evaluation. CrowdPose is big enough for training but is unsuitable for evaluation in multi-dataset setup as it mixes train and test sets of COCO, MPII and AIC. For COCO and related datasets, the evaluation metric is Object Keypoint Similarity (OKS), while Percentage of Correct Keypoints (PCKh) is used for MPII. In addition to the pose estimation dataset, CrowdHuman [25] focuses on person detection in crowds.

Human pose estimation. There are two main approaches to 2D human pose estimation: top-down and detector-free. Detector-free can be further divided into single-stage [26, 27, 31, 35], bottom-up [7, 10, 23] and hybrid [40].

Top-down methods [14, 19, 28, 36, 37] use person detector to detect bounding boxes and estimate one skeleton for each bounding box. They leverage big progress in human detection and specialize on understanding of human structure. Top-down methods are the most successful on datasets such as COCO, MPII or AIC but struggle on crowded datasets (OCHuman) due to low-quality detections. Most notably, ViTPose [36] combines multiple datasets into one strong backbone and sets a strong baseline, setting up state-of-the-art performance on most datasets. While conditioning pose estimation on bounding boxes (bbox-to-pose; standard top-down approach) is well researched, conditioning pose on masks (mask-to-pose) was not explored.

On the other hand, detector-free models do not achieve SOTA performance on COCO but are superior to top-down methods on OCHuman as they are specialized on decoupling close-interaction instances. The most successful model, BUCTD [40], conditions top-down pose estimation by previously estimated keypoints from bottom-up methods. It is a pose-refinement method which has state-of-the-art results on OCHuman datasets due to its strong ability to decouple people close interactions.

Foundational models. The latest directions in human body modeling are foundational models [8, 11, 15, 33]. They learn general features describing human body that could be used for all human-related tasks such as segmentation, pose estimation etc. Most notably, Sapiens 2b [15] was trained on staggering 2M images and with 2B parameters is almost four times bigger than ViTPose-h. Even with this size, foundational models perform comparatively or worse than much smaller specialized models.

Detectors. Object (or person) detection is one of the most researched problems in computer vision. Huge models such as InternImage [32] or Co-DETR [41] holds SOTA performance on multiple datasets. In our comparison, we use smaller almost real-time models RTMDet [21], ConvNeXt [20] and HRNet [28] which have slightly lower performance but run much faster. To the best of our knowl-

edge, the detection of objects in the image given a set of already detected objects was not investigated. In top-down HPE methods, detector guides pose estimation but the information never goes back to the detector. The slight exception is PoseNMS [22], which uses human pose for non-maxima suppression.

Segmentators. The idea of segmentation conditioned by human pose is not new. Many models [1, 2, 30, 38, 39] estimate instance segmentation from either ground truth pose or estimated keypoints. Other methods such as [5] use pose for test-time adaptation in instance segmentation. The latest segmentation foundational model SAM2 [24] is conditioned not only by human pose but by any point(s). Similarly to detection, conditioning mask by pose is well researched, but the other direction (conditioning pose by mask) remains unexplored.

3. Method

The following sections detail the components of the BBox-Mask-Pose (BMP) method. To create an iterative process that involves detection (bboxes $\mathcal{B}_i$), segmentation (binary masks $\mathcal{M}_i$) and pose estimation (keypoints $\mathcal{K}_i$), each component must be conditioned by the others. We adapt the detector ($\mathcal{D}$) and pose estimator ($\mathcal{P}$) for mask conditioning and use Segment Anything Model 2 [24] ($\mathcal{S}$) to condition masks with bounding boxes and keypoints.

The BMP loop starts with the detector. In general, it could start from any of the three representations.

3.1. Detection

The detector $\mathcal{D}$ detects bboxes $\mathcal{B}_i$ and masks $\mathcal{M}_i$ in the image $\mathcal{I}$ (Eq. (1)). The image is masked by previously detected instances $\mathcal{M}_i$ as shown in Fig. 1 (A2).

$$\left(\mathcal{B}_i, \mathcal{M}_i\right) = \mathcal{D}\left(\mathcal{I} \odot \left(1 - \bigcup_i \mathcal{M}_i\right)\right) \tag{1}$$

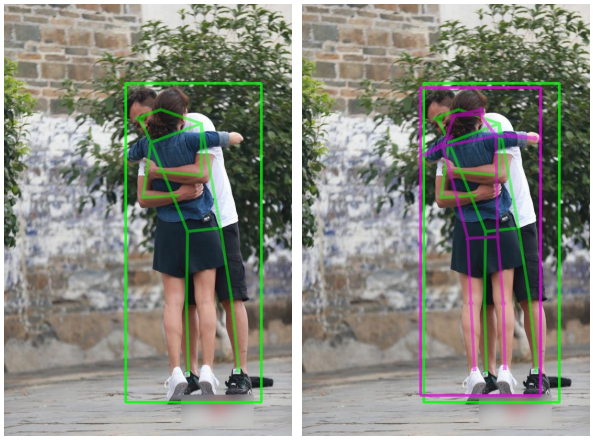
$$\left(\mathcal{B}_i, \mathcal{M}_i\right) = \mathcal{D}^{init}\left(\mathcal{I}\right), \tag{2}$$

where $\odot$ stands for the Hadamard product of two matrices.

During masking, all pixels that belong to at least one mask $\mathcal{M}_i$ are set to black. In the initial stage, there are no detected instances $\mathcal{M}_i$ and the detection phase becomes Eq. (2) – the standard object detection task (Fig. 1 (A1)).

Any standard detector could be used as $\mathcal{D}^{init}$. For detector $\mathcal{D}$ conditioned by $\mathcal{M}_i$, we fine-tuned RTMDet [21] with *instance removal* data augmentation simulating masked-out instances as in Fig. 1 (D1). During training, randomly selected instances in the image are masked out and the model is trained not to predict them. The fine-tuned detector retains its ability to detect instances in unmasked images, and the same model could be used for both $\mathcal{D}$ and $\mathcal{D}^{init}$.

The masked pixels are set to non-transparent black. When the mask is incorrect, non-transparent masking-out



(a) **Missed instance** which is detected in the second iteration of BMP. Left – RTMDet [21]+MaskPose, right – BMP.

(b) **Two instances in one detection** are resolved by refining segmentation masks with SAM [24] prompted by the detected pose. Left – RTMDet [21], right – BMP. Note that the detection of the woman is improved, but the right leg is still wrong.

(c) **Collapse of pose estimates for two instances** with correctly detected overlapping bboxes onto one body. Left – ViTPose-B [36] conditioned by bounding box, right – MaskPose-b conditioned by masks.

Figure 2. **BMP resolves detection errors** (top and middle) and pose errors (bottom) on OCHuman. Quantitative results in Tab. 3.

leads to information loss. In the next section, MaskPose uses semi-transparent masking to improve robustness to incorrectly estimated masks. Training the detector with semi-transparent masking led to much worse performance as the detector kept detecting masked-out instances.

RTMDet estimates both bounding boxes and segmentation masks. MaskPose leverages estimated masks to predict more accurate poses in the next section. Alternatively, a bbox-conditioned pose estimator could be used with the detector that estimates only bboxes.

3.2. Pose Estimation

Traditional top-down methods (Eq. (3)) rely solely on bounding boxes, cropping an image patch centered on the bounding box. If multiple people appear in the same crop, the model estimates the pose of the central person but often merges body parts from others into a single skeleton. We introduce MaskPose (Eq. (4)), which builds on ViTPose [36] and adapts it to use segmentation masks for conditioning.

$$\mathcal{K}_i = \mathcal{P}\big(\mathcal{I}, \mathcal{B}_i\big) \tag{3}$$

$$\mathcal{K}_i = \mathcal{P}_\alpha\big(\alpha\mathcal{I} + (1-\alpha)(\mathcal{I}\odot\mathcal{M}_i), \mathcal{B}_i\big) \tag{4}$$

In Eq. (4), pose estimator $\mathcal{P}_\alpha$ is trained to predict pose in semi-transparently masked image $\mathcal{I}\odot\mathcal{M}_i$. Pixels not belonging to mask $\mathcal{M}_i$ are darkened as shown in Fig. 1 (B1).

The model $\mathcal{P}_\alpha$ needs to be re-trained for a given α. Fully masking the background (α=0) causes loss of contextual information, impairing MaskPose's recovery from inaccurate masks. No masking (α=1) reverts to a traditional bounding-box-based approach (Eq. (3)). All preliminary experiments with $\alpha \in (0, 1)$ had the same performance and we settled with $\alpha = 0.8$. To enhance robustness to inaccurate masks, we randomly deform ground truth masks during training, allowing the model to predict keypoints outside the mask.

ViTPose trained in multi-dataset setup generalizes well across datasets, leveraging the strength of the ViT [9] backbone. ViTPose uses specialized head for each dataset with shared backbone. MaskPose is also trained on the COCO, MPII, and AIC datasets but has a single head for all datasets. The head predicts all 22 keypoints defined across COCO, AIC, and MPII, resulting in negligible performance loss compared to using separate heads. MaskPose can thus be evaluated directly on any dataset without switching heads.

MaskPose has approximately the same number of parameters as ViTPose, differing only in head architecture and preprocessing. These small changes enable MaskPose to perform similarly on standard datasets (COCO, AIC, MPII) while improving performance in multi-body scenarios. Mask conditioning adapts the top-down method for multi-body cases, allowing detailed instance specification in densely overlapping scenes.

(a) **Number of keypoints**. Too many points hinders performance. Left – 6 keypoint prompts, right – 13 correct prompts.

(b) Prompting **with and without a bounding box**. Prompting with bbox prevents SAM from correcting body masks outside of the bounding box. Left – RTMDet [21], middle – SAM with bbox, right – SAM without bbox.

Figure 3. **SAM**: influence of prompting parameters.

3.3. Segmentation

We use Segment Anything Model v2 (SAM) [24] ($\mathcal{S}$) for mask generation, conditioned by estimated bounding boxes ($\mathcal{B}_i$) and keypoints ($\mathcal{K}_i$).

$$\big(\mathcal{B}_i, \mathcal{M}_i\big) = \mathcal{S}\big(\mathcal{I}, f(\mathcal{K}_i), g(\mathcal{B}_i)\big) \tag{5}$$

SAM is inherently a conditioned segmentator, so no architecture adaptations are needed. The key challenge is prompting – how to select keypoint prompts ($f(\mathcal{K}_i)$) and whether to prompt with bounding box ($g(\mathcal{B}_i)$).

SAM was trained with a maximum of 8 point prompts and fails with more, such as all 17 keypoints from COCO pose (Fig. 3a). The challenge is twofold: determining the number of keypoints and selecting them. This chapter outlines our prompting method for a successful BBox-MaskPose loop and analyzes hyper-parameter effects on the loop. Extensive ablation study on segmentation conditioned by pose with SAM is in the supplementary material.

Visibility. Ideally, SAM should be prompted only by visible keypoints. However, pose models estimate both visible and occluded keypoints and do not distinguish between

them (with some exceptions, such as [29]). SAM can handle occluded keypoints if they are on the instance border but struggles if they are within another instance. We approximate visibility by confidence and prompt only with keypoints above a confidence threshold; we have not been able to train, following [29], a reliable visibility predictor.

Spread. To segment disconnected parts of an instance (for example, the legs of the background player in Fig. 1), we maximize keypoints spread. Selecting keypoints along the bounding box border provides a good spread, but SAM still needs at least one unambiguous keypoint to specify the instance. We mimic human annotation by first choosing the most confident keypoint (analogous to a human's initial click in the center) and then selecting keypoints to maximize spread. To avoid redundancy, we select at most one facial keypoint (an eye or the nose).

Bounding box. Another question is whether to use bounding boxes (Fig. 3b). Experiments with ground truth boxes show that bounding boxes improve mask quality, but the situation changes with detected bounding boxes, especially in multi-body scenarios. The detector may only capture part of an instance or merge two instances. Prompting SAM with detected boxes restricts it to the detected area, limiting its ability to correct detection errors. Conversely, SAM without a bounding box can "explore" undetected areas but loses precision within the bounding box. Since detection correction is critical for BMP success, we prompt SAM without a detected bounding box. Prompting with bounding box would be useful for final mask refinement after the BMP loop when bounding boxes are stable.

The keypoint selection algorithm is summarized in Alg. 1. It maximizes keypoint spread similar to KMeans++ initialization [4], factoring in keypoint confidence. We used 6 positive keypoints for each instance (N_{max}) and confidence threshold $T_c = 0.5$.

Our experiments suggest that automatically selected keypoints have a different distribution from human-annotated prompts. Human annotators intuitively understand the scene, and SAM generally performs better with human prompts than with automated keypoint selection. Pose keypoints tend to lie on the borders and extremes of the instance, whereas humans often click in the middle of the instance. By choosing visible, high-confidence, and spread keypoints, we partially simulate human prompting. Although automated prompts do not match human effectiveness, the BBox-Mask-Pose loop still improves segmentation, and pose-prompted SAM outperforms bounding-box-prompted SAM.

Pose-Mask consistency. When incorrect keypoints are selected during prompting, SAM's segmentation mask may be worse than the original detector mask. After mask generation, we measure the *pose-mask consistency* of both the original detector mask and the mask refined by SAM. Pose-

Algorithm 1: SAM prompts selection $f(\mathcal{K}_i)$

Inputs : Set ot detected keypoints K,
Confidence threshold T_c,
Max number of keypoint N_{max}

Output: Set of selected keypoints K_s

```
Select keypoints from K with confidence ≥ T_c
Sort keypoints in K by confidence
K_s ← ∅
Select the most confident keypoint into K_s
while len(K_s) < N_max do
    k ← keypoint from K furthest to K_s
    Add k to K_s
end
return K_s
```

mask consistency (*P-Mc*) is defined as:

$$\text{P-Mc} = \frac{\sum k_p^+}{\sum k_p} + \frac{\sum k_n^-}{\sum k_n} \tag{6}$$

where k_p represents the positive keypoints of the instance, and k_n represents negative keypoints (those from other instances in the image). k_p^+ are positive keypoints inside the mask, while k_p^- are negative keypoints outside the mask. Thus, pose-mask consistency measures the proportion of keypoints (both positive and negative) that are consistent with the mask. If the refined mask has a lower *P-Mc* than the previous mask, we discard it. BBox-Mask-Pose discard approximately 15% of SAM-refined masks.

Prompting with ground truth data behaves differently than with noisy estimated data. As mentioned, the ground truth bounding box consistently improves the predicted mask. Similarly, ground truth data includes annotated visibility, allowing us to use only visible keypoints. We prompted SAM with ground truth bounding boxes and poses when generating pseudo ground truth for AIC and MPII to train MaskPose. For an extensive ablation study on prompting with ground truth or detections, see the supplementary material.

3.4. Closing the "circle"

With all three models adapted for mutual conditioning, we establish a closed iterative loop.

$$(\mathcal{B}_i, \mathcal{M}_i) = \mathcal{D}\big(\mathcal{I} \odot (1 - \bigcup_i \mathcal{M}_i)\big) \tag{7}$$

$$\mathcal{K}_i = \mathcal{P}_\alpha\big(\alpha\mathcal{I} + (1-\alpha)(\mathcal{I} \odot \mathcal{M}_i), \mathcal{B}_i\big) \tag{8}$$

$$(\mathcal{B}_i, \mathcal{M}_i) = \mathcal{S}\big(\mathcal{I}, f(\mathcal{K}_i), g(\mathcal{B}_i)\big) \tag{9}$$

As shown in Fig. 1, the detector conditions MaskPose (Eq. (8)), which in turn conditions SAM2 segmentation

(Eq. (9)). The loop completes by masking out processed instances and rerunning the detector (Eq. (7)).

Each BBox-Mask-Pose iteration masks out more of the image, and when all instances are masked, the detector no longer identifies new instances, ending the loop. In practice, the user can manually set the number of iterations, as later iterations yield diminishing performance gains.

To minimize duplicate detections, we use two forms of non-maximum suppression (NMS): bounding box NMS in the detector and pose NMS in the pose estimator. We apply both with standard settings. Bounding box NMS with intersection-over-union (IoU) at 0.3 and pose NMS with object-keypoint-similarity (OKS) at 0.9. If valid detections are mistakenly suppressed, they are re-detected in the next BMP loop iteration.

4. Results

4.1. Implementation details

RTMDet-L [21] is used in the BMP loop. We fine-tuned RTMDet with instance-removal augmentation for 10 epochs on COCO-human, to enable it to ignore already-processed instances. The same detector was used in top-down model experiments for a fair comparison.

MaskPose builds on ViTPose [36], so we use the same training setup: 210 epochs on COCO, AIC and MPII with three learning rate steps. Since MPII and AIC lack ground truth segmentation, we generate pseudo ground truth using SAM2, prompted with ground truth bounding boxes and visible keypoints.

The Segment Anything Model (SAM) is used without fine-tuning. We use version *sam2-hiera-base+* with post-processing settings: max_hole_area at 10 and max_sprinkle_area at 50. Each instance is processed independently, which yields slightly better results than batch processing.

4.2. Comparison with SOTA

Pose estimation. Tab. 1 compares pose estimation performance on the OCHuman and COCO datasets. MaskPose improves the ViTPose [36] baseline from 42.6 to 45.0 AP by mask conditioning, making it a new SOTA among top-down methods. BMP 1× yields identical results as MaskPose, since BMP 1× is MaskPose with an additional mask refinement step, which does not affect pose. BBox-Mask-Pose 2× further increases MaskPose performance from 45.0 to 49.3 AP through iterative conditioning between masks and poses. BMP sets the new SOTA performance on OCHuman, beating BUTCD [40]. BMP and MaskPose perform similarly on COCO, as the detector captures nearly all instances in the first pass, with only a few additional detections in the second iteration.

Additionally, the numbers could improve with bigger specialized models (ViTPose-h, RTMDet-x, SAM2.1-large) and additional bells and whistles (e.g. BUCTD).

[1] [13] also reports version with ViT-L backbone with better results. Its results could not be replicated as the authors do not provide weights.

Model	OCHuman test AP	COCO val AP
DEKR [10]	36.5	71.0
HQNet R-50¶ [13]	40.0	69.5
CID-w48 [31]	45.0	68.9
BUCTD [40]	47.4	74.8
Sapiens 0.3b [15]	41.3	66.1
MIPNet† [14]	42.5	76.3
ViTPose-B [36]	42.6	<u>76.4</u>
MaskPose-b	45.0	**76.5**
BUCTD 2× [40]	<u>48.3</u>	—‡
BBox-Mask-Pose 1×	46.6	**76.5**
BBox-Mask-Pose 2×	**49.2**	**76.5**

Table 1. **Pose estimation – comparison with state-of-the-art**. Best results in bold, second best underlined. Results of detection-free (top), top-down (middle) and iterative (bottom) methods. Top-down methods used detections from RTMDet-L [21], except MIPNet† which reports results from [14]. ‡ BUCTD 2× result on COCO not reported [40]. ¶[13] ignores *small* instances in COCO. Summary: MaskPose improves ViTPose and it sets the new SOTA for top-down approaches. BMP further improves on MaskPose and set the SOTA for OCHuman while keeping SOTA on COCO.

Model	OCHuman test	
	bbox AP	mask AP
HRNet [28]	27.1	19.4
ConvNeXt [20]	29.4	20.4
HQNet R-50[1] [13]	29.5	31.1
CoDETR SWIN-L‡ [41]	29.6	—
RTMDet-L [21]	30.0	26.5
Occlusion C&P‡ [18]	—	28.3
ExPoSeg‡ [39]	—	26.8
Crowd-SAM‡ [6]	—	<u>31.4</u>
BBox-Mask-Pose 1×	<u>32.4</u>	30.2
BBox-Mask-Pose 2×	**35.9**	**34.0**

Table 2. **Detection and instance segmentation – comparison with state-of-the-art**. Best results in bold, second best underlined. Results of COCO-trained detectors (top), segmentation models relyiong on previous detections or poses (middle). Models with ‡ estimate either masks or report detection AP. Note that even CoDETR, a huge COCO SOTA model, struggles with multi-body scenes. BMP 2× improves detection of RTMDet [21] setting a new SOTA on OCHuman dataset. Qualitative results are in Fig. 2.

bbox AP @ max_IoU	0.0 – 0.2	0.2 – 0.4	0.4 – 0.6	0.6 – 0.8	0.8 – 1.0	mAP
RTMDet-L	16.9	0.1	20.4	15.7	8.7	31.1
BBox-Mask-Pose 2×	18.1 (+1.2)	0.2 (+0.1)	21.4 (+1.0)	21.5 (+5.8)	10.7 (+2.0)	35.7 (+4.6)

Table 3. **BMP's effectiveness for people with high overlap** on OCHuman-val. BMP improves detection especially in multi-body scenarios with big bbox overlap. Traditional detectors like RTMDet often merge two individuals into one instance or ignore the background individual. BMP resolves the issues with instance understanding through pose estimation. See e.g. the detection errors in Fig. 2.

det	pose	SAM	pose	loops	bbox	pose
✓	✓	✓	✗	1×	31.1	45.3
✓	✓	✓	✗	2×	**32.1**	**48.6**
✓	✓	✓	✓	1×	31.1	46.4
✓	✗	✓	✗	2×	31.9	47.3
✓	✓	✗	✗	2×	30.8	47.0

Table 4. **Ablation study** of BBox-Mask-Pose components evaluated on OCHuman-val. Bbox and pose evaluated with AP. The sum of trainable parameters approximates computational complexity. First row corresponds to BMP 1×, second to BMP 2×.

BUCTD could either refine MaskPose's keypoints or replace MaskPose within the BMP loop as it conditions pose estimation on bottom-up poses while MaskPose is conitioned on masks.

Experiments show that performance plateaus after two iterations, similar to BUCTD. Further iterations add computational cost without notable performance gains.

Detection and segmentation. Tab. 2 shows BMP detection and segmentation performance on the OCHuman dataset. BMP 1× improves the RTMDet pipeline by refining bounding boxes and segmentation masks using pose-prompted SAM, as illustrated in Fig. 1. BMP 2× further improves detection and segmentation through re-detection of background instances in images with masked-out instances, as shown in Fig. 2. BBox-Mask-Pose sets a new SOTA on OCHuman detection and segmentation beating both object detectors and pose-conditioned segmentators such as ExPoSeg [39].

Detection accuracy in multi-body scenarios. Tab. 3 shows that the detection performance is improved most in scenarios with a high bbox overlap. For each GT instance, we calculate its highest IoU with other GT instances (max_IoU) and split the OCHuman dataset accordingly. Detections cannot be split accordingly as high inter-detection overlap could be both multi-body scenarios and false positives. Therefore, AP numbers are generally lower than for standard mAP metric but the comparison between models is fair. Qualitative examples of improvement are in Fig. 2.

4.3. Ablation study

Looping SAM and pose estimation. The third row of Tab. 4 shows a slight improvement in pose estimation when re-running pose on SAM-refined masks. This pipeline, detect-pose-SAM-pose, is comparable to one BMP iteration as it cannot re-detect previously missed instances. Formally, the experiment is chaining Eqs. (8) and (9) without Eq. (7). SAM mask refinement improves MaskPose keypoint predictions, suggesting that an SAM-pose-SAM loop could further enhance the results. However, the additional computational cost outweighs the gains, so we exclude it to keep BMP efficient.

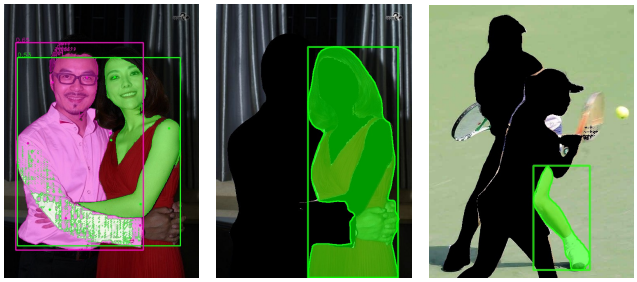
Prompting SAM only with bounding box. This approach effectively omits the pose estimation model (Eq. (8)) from the loop, as SAM is prompted solely by the bounding box detected in the first step. SAM refines the segmentation mask and updates the bounding box accordingly. Tab. 4 shows that SAM alone improves performance over omitting SAM entirely (second-last and last rows). Adding keypoints as prompts further boosts detection from 31.9 to 32.1 AP and pose estimation from 47.3 to 48.6 AP.

Omitting SAM. When SAM (Eq. (9)) is omitted from BMP, segmentation masks are provided only by the detector from Eq. (7). This causes the detector to loop with itself without conditioning from masks or poses, often resulting in un-segmented body parts, such as missed limbs. For example, in Fig. 1, un-segmented legs of a background player could be detected as separate instances, as shown in Fig. 4. In practice, omitting SAM resembles running a detector with a low non-maxima suppression (NMS) threshold, resulting in many false-positive bounding boxes. This hinders detection performance, but slightly boosts pose accuracy. Low-confidence poses minimally impact the COCO evaluation, as they do not deform the precision-recall curve in the AP computation. That explain why looping the detector with itself still improves the pose. However, using SAM improves detection from 30.8 to 32.1 AP and pose estimation from 47.0 to 48.6 AP, as shown in Tab. 4.

Computational complexity estimation. Tab. 5 compares BMP runtime to Sapiens 0.3b [15], a recent foundational model with 336M parameters. Combined with RTMDet-L, it totals 393M parameters, surpassing the 369M of two BMP iterations. Similarly, runtime analysis shows that BMP 2× runs half the time while outperforming Sapi-

model	s/img	params	pose mAP
RTMDet-L	0.03	57 M	—
Sapiens 0.3b	1.95	336 M	—
MaskPose-b	0.06	87 M	—
SAM2-hiera-base+	0.47	81 M	—
RTMDet-L + Sapiens 0.3b	2.03	393 M	41.3
BBox-Mask-Pose 1×	**0.56**	225 M	<u>46.6</u>
BBox-Mask-Pose 2×	<u>1.12</u>	369 M	**49.2**

Table 5. **Runtime analysis** on OCHuman; s/img – seconds per image. Measured on a A-100 GPU with 40 GB. BMP 2× is almost two times faster than Sapiens while having better performance.



(a) Segmenting only skin (green) (b) Re-detection of unsegmented clothes. (c) Missed limb is re-detected

Figure 4. **Characteristic errors** in the BMP loop. The weakest part is SAM and its prompting with correct keypoints.

ens on both COCO and OCHuman datasets. The runtime analysis proves that multiple small specialized models are faster and achieve better performance than huge foundational models. For complexity analysis of various components of the BMP loop, see the supplementary material.

5. Conclusions

We present BBox-Mask-Pose (BMP), a method for detection, segmentation, and pose estimation in multi-body scenarios. Part of the BMP loop, a new top-down model MaskPose, conditions pose estimation on predicted instance masks unlike prior approaches. BMP integrates detector, MaskPose and (SAM) into a self-improving loop. By conditioning each model on outputs from the others, BMP simultaneously improves detection, segmentation, and pose estimation and set a new SOTA on the OCHuman dataset in all three tasks. Key findings are:

1. Conditioning the top-down pose model with masks and bounding boxes improves performance, especially in crowded scenes.
2. BMP demonstrates that explicit mutual conditioning between the detector, segmentator, and pose estimation models improve performance in all tasks. Small specialized models give better results than large foundational models with shared features. However, adapting these models for mutual conditioning is non-trivial.
3. BMP's effectiveness diminishes after two iterations, with additional iterations offering little performance gain while increasing computational cost.
4. BMP sets the new SOTA on OCHuman while also matching the SOTA performance of top-down models on COCO.
5. Surprisingly, the Segment Anything Model proved the least effective component in BMP. Even though BMP segmentation is the new SOTA, automated SAM prompting falls short compared to human interaction and most of the errors come from incorrect masks.
6. The modular structure of BMP enables further performance gains by integrating improved models or adding BUCTD [40] to the loop.

Limitations of BMP primarily involve imperfect SAM mask refinement. When SAM is prompted with inaccurate keypoints (e.g., occluded or mislocalized), it has limited recovery ability, which can lead to masking out the wrong instances, preventing the detector from retrieving them. We experimented with semi-transparent masking, as used in MaskPose, but found it ineffective.

A second limitation occurs when detecting in masked-out images. If a foreground instance divides a background instance into disconnected parts, the detector often fails to connect these, generating multiple small bounding boxes for each segment. Although pose NMS suppresses redundant detections, disconnected body parts remain separate. Attempts to use data augmentation to improve detector robustness in such cases were unsuccessful. Examples of these errors are included in Fig. 4. More detailed analysis of SAM errors is provided in the supplementary material.

Future work. MaskPose has demonstrated robustness to incorrect masks due to training augmentations; extending this robustness to the detector and pose-to-seg models could significantly enhance BMP performance. Beyond robustness, improving the efficiency of interactions between bounding boxes, masks, and poses is an area for exploration. Foundational models aim to unify body representations at a feature level but lack the explicit constraints offered by different representations. Although foundational models are non-iterative, their large size often results in longer inference times compared to smaller, specialized models. Our findings indicate that explicit constraints within specialized models could improve performance while keeping the models smaller and faster.

References

[1] Niaz Ahmad, Jawad Khan, Jeremy Yuhyun Kim, and Youngmoon Lee. Multiposeseg: Feedback knowledge transfer for multi-person pose estimation and instance segmentation.

2022 26th International Conference on Pattern Recognition (ICPR), pages 2086–2092, 2022. 3

[2] Niaz Ahmad, Jawad Khan, Jeremy Yuhyun Kim, and Youngmoon Lee. Joint human pose estimation and instance segmentation with poseplusseg. In *AAAI Conference on Artificial Intelligence*, 2022. 3

[3] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2

[4] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. Technical report, Stanford, 2006. 5

[5] Kambiz Azarian, Debasmit Das, Hyojin Park, and Fatih Murat Porikli. Test-time adaptation vs. training-time generalization: A case study in human instance segmentation using keypoints estimation. *2023 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, pages 411–420, 2022. 3

[6] Zhi Cai, Yingjie Gao, Yaoyan Zheng, Nan Zhou, and Di Huang. Crowd-sam: Sam as a smart annotator for object detection in crowded scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024. 6

[7] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 1, 2

[8] Yuanzheng Ci, Yizhou Wang, Meilin Chen, Shixiang Tang, Lei Bai, Feng Zhu, Rui Zhao, Fengwei Yu, Donglian Qi, and Wanli Ouyang. Unihcp: A unified model for human-centric perceptions. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17840–17852, 2023. 2

[9] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 4

[10] Zigang Geng, Ke Sun, Bin Xiao, Zhaoxiang Zhang, and Jingdong Wang. Bottom-up human pose estimation via disentangled keypoint regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14676–14686, 2021. 2, 6

[11] Seyed Abolfazl Ghasemzadeh, Gabriel Van Zandycke, Maxime Istasse, Niels Sayez, Amirafshar Moshtaghpour, and Christophe De Vleeschouwer. Deepsportlab: a unified framework for ball detection, player instance segmentation and pose estimation in team sports scenes. *ArXiv*, abs/2112.00627, 2021. 2

[12] Kerui Gu, Rongyu Chen, and Angela Yao. On the calibration of human pose estimation. *arXiv preprint arXiv:2311.17105*, 2023. 13

[13] Sheng Jin, Shuhuai Li, Tong Li, Wentao Liu, Chen Qian, and Ping Luo. You only learn one query: learning unified human query for single-stage multi-person multi-task human-centric perception. In *European Conference on Computer Vision*, pages 126–146. Springer, 2024. 6

[14] Rawal Khirodkar, Visesh Chari, Amit Agrawal, and Ambrish Tyagi. Multi-instance pose networks: Rethinking top-down pose estimation. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3102–3111, 2021. 1, 2, 6

[15] Rawal Khirodkar, Timur Bagautdinov, Julieta Martinez, Zhaoen Su, Austin James, Peter Selednik, Stuart Anderson, and Shunsuke Saito. Sapiens: Foundation for human vision models. In *European Conference on Computer Vision*, 2024. 2, 6, 7

[16] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. *arXiv preprint arXiv:1812.00324*, 2018. 2

[17] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. 2

[18] Evan Ling, De-Kai Huang, and Minhoe Hur. Humans need not label more humans: Occlusion copy & paste for occluded human instance segmentation. In *British Machine Vision Conference*, 2022. 6

[19] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021. 2

[20] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 2, 6

[21] Chengqi Lyu, Wenwei Zhang, Haian Huang, Yue Zhou, Yudong Wang, Yanyi Liu, Shilong Zhang, and Kai Chen. Rtmdet: An empirical study of designing real-time object detectors. *ArXiv*, abs/2212.07784, 2022. 2, 3, 4, 6, 11, 12, 13, 14, 16, 17

[22] George Papandreou, Tyler Lixuan Zhu, Nori Kanazawa, Alexander Toshev, Jonathan Tompson, Christoph Bregler, and Kevin P. Murphy. Towards accurate multi-person pose estimation in the wild. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3711–3719, 2017. 3

[23] George Papandreou, Tyler Lixuan Zhu, Liang-Chieh Chen, Spyros Gidaris, Jonathan Tompson, and Kevin P. Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *European Conference on Computer Vision*, 2018. 2

[24] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 2, 3, 4, 11, 12, 13, 14, 15

[25] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *ArXiv*, abs/1805.00123, 2018. 2

[26] Dahu Shi, Xing Wei, Liangqi Li, Ye Ren, and Wenming Tan. End-to-end multi-person pose estimation with transformers.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11069–11078, 2022. 2

[27] Lucas Stoffl, Maxime Vidal, and Alexander Mathis. End-to-end trainable multi-instance pose estimation with transformers. *arXiv preprint arXiv:2103.12115*, 2021. 2

[28] 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, 6

[29] Pengzhan Sun, Kerui Gu, Yunsong Wang, Linlin Yang, and Angela Yao. Rethinking visibility in human pose estimation: Occluded pose reasoning via transformers. *2024 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5891–5900, 2024. 5

[30] Subarna Tripathi, Maxwell D. Collins, Matthew A. Brown, and Serge J. Belongie. Pose2instance: Harnessing keypoints for person instance segmentation. *ArXiv*, abs/1704.01152, 2017. 3

[31] Dongkai Wang and Shiliang Zhang. Contextual instance decoupling for robust multi-person pose estimation. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11050–11058, 2022. 1, 2, 6

[32] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14408–14419, 2023. 2

[33] Yizhou Wang, Yixuan Wu, Shixiang Tang, Weizhen He, Xun Guo, Feng Zhu, Lei Bai, Rui Zhao, Jian Wu, Tong He, and Wanli Ouyang. Hulk: A universal knowledge translator for human-centric tasks. *ArXiv*, abs/2312.01697, 2023. 2

[34] Jiahong Wu, He Zheng, Bo Zhao, Yixin Li, Baoming Yan, Rui Liang, Wenjia Wang, Shipei Zhou, Guosen Lin, Yanwei Fu, Yizhou Wang, and Yonggang Wang. Ai challenger : A large-scale dataset for going deeper in image understanding. *ArXiv*, abs/1711.06475, 2017. 2

[35] Yabo Xiao, Xiaojuan Wang, Dongdong Yu, Kai Su, Lei Jin, Mei Song, Shuicheng Yan, and Jian Zhao. Adaptivepose++: A powerful single-stage network for multi-person pose regression. *arXiv preprint arXiv:2210.04014*, 2022. 2

[36] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. ViTPose: Simple vision transformer baselines for human pose estimation. In *Advances in Neural Information Processing Systems*, 2022. 1, 2, 3, 4, 6, 14

[37] Yuhui Yuan, Rao Fu, Lang Huang, Weihong Lin, Chao Zhang, Xilin Chen, and Jingdong Wang. Hrformer: High-resolution vision transformer for dense predict. *Advances in neural information processing systems*, 34:7281–7293, 2021. 2

[38] Song-Hai Zhang, Ruilong Li, Xin Dong, Paul Rosin, Zixi Cai, Xi Han, Dingcheng Yang, Haozhi Huang, and Shi-Min Hu. Pose2seg: Detection free human instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 889–898, 2019. 2, 3

[39] Desen Zhou and Qian He. Poseg: Pose-aware refinement network for human instance segmentation. *IEEE Access*, 8: 15007–15016, 2020. 3, 6, 7

[40] Mu Zhou, Lucas Stoffl, Mackenzie W. Mathis, and Alexander Mathis. Rethinking pose estimation in crowds: overcoming the detection information bottleneck and ambiguity. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14643–14653, 2023. 1, 2, 6, 8

[41] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6748–6758, 2023. 2, 6

Detection, Pose Estimation and Segmentation for Multiple Bodies: Closing the Virtuous Circle

Supplementary Material

A. Prompting SAM ablation study

A.1. Setup

Here, we describe the ablation study on prompting SAM. The study evaluates three metrics: detection improvement (bounding box; bbox), segmentation improvement (segm), and pose improvement (pose). For all experiments, we use bounding boxes and segmentation masks from RTMDet-l and pose estimates from MaskPose as the baseline pipeline. The experimental pipeline remains consistent throughout.

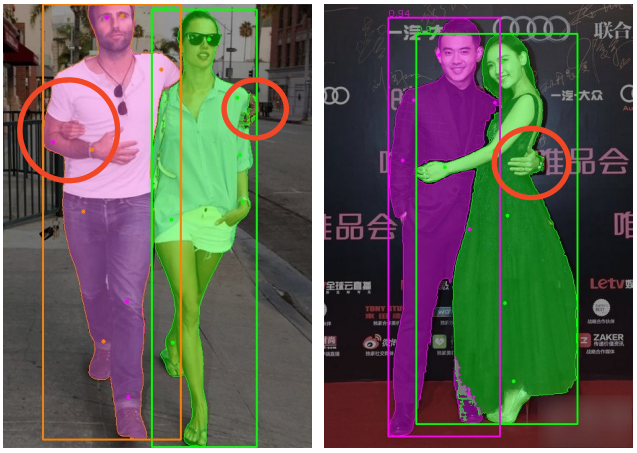
Detection and segmentation changes are evaluated on bounding boxes and segmentation masks refined by SAM, following the det-pose-SAM pipeline. Pose estimation is assessed by re-running MaskPose on refined masks, forming a det-pose-SAM-pose pipeline, similar to the setup in Tab. 4.

All experiments use *RTMDet-l* [21] as the detector, *MaskPose-b* as the pose estimator, and *sam2-hiera-base+* as the SAM2 [24] model. Each experiment is assigned a specific name, listed in the leftmost column of the tables, for clear referencing. When experiments appear in multiple tables for comparison, their names remain consistent for easier cross-referencing. Each result is highlighted in green or red depending on whether it improves or hinders performance compared to the RTMDet+MaskPose baseline.

Detection vs. segmentation. Before analyzing the results of the ablation study, we address a counterintuitive observation. When refining masks on OCHuman, segmentation and detection often conflict; improvement in one can lead to a decrease in the other. This is due to the focus on people with high overlap in the OCHuman dataset. Many examples consist of a large area representing the main body and smaller, disconnected body parts. Examples are shown in Fig. 5.

When mask refinement focuses heavily on the main segment, segmentation scores improve, as missing disconnected parts has little impact on mask IoU. Conversely, overly general prompting can cause SAM to merge both instances into one mask, creating a bounding box that may be more accurate than the original. Large masks merge instances, while small masks often miss disconnected body parts.

We prioritize detection, even though the goal is to improve all three metrics. The mask refinement step in BBox-Mask-Pose must ensure that segmented masks adequately remove limbs during the mask-out step, as shown in Figs. 4c and 9. However, excessively large masks prevent decoupling of merged instances, as seen in Fig. 2b. Thus, our aim is to improve detection without significantly hindering segmentation performance.



(a) (b)

Figure 5. Segmentation error involving a small number of pixels, like the circled hands, may have a large impact on detection accuracy measured by bounding box IoU. A detector returning correct bounding boxes, which would be nearly identical for both persons especially in (a), can make segmentation of the two people very challenging. Improving detection may thus lead to decrease in segmentation performance. Keypoints used for SAM prompting are marked (best viewed in zoom).

A.2. Results

Bounding box. The question of whether to prompt SAM with a bounding box is addressed in Tab. 6, with examples provided in Fig. 3b. When the bounding box is accurate, or nearly so, it significantly improves segmentation quality. However, when the bounding box is incorrect, such as missing parts of an occluded person (Fig. 4c), prompting restricts mask refinement to the given bounding box, reducing the chance of recovery.

In the final version of BBox-MaskPose, we do not use bounding box prompting, as we prioritize SAM's ability to explore and detect previously missed body parts (Fig. 10). However, when bounding boxes are reliable, prompting with them can further refine segmentation and pose estimation, yielding improved results, as shown in Tab. 4 in Sec. 4.3. Bounding box prompting is also advantageous when ground truth bounding boxes are available.

Number of positive keypoints (⊕). Tab. 6 evaluates the effect of using different numbers of keypoints for prompt-

name	batch	bbox	⊕	⊖	bbox	segm	pose
RTMDet [21] + MaskPose					31.1	27.1	45.3
A1	✗	✓	0	0	27.5	**31.6**	44.2
A2	✗	✓	2	0	28.5	**31.6**	**44.3**
A3	✗	✓	4	0	29.3	30.9	44.0
A4	✗	✓	6	0	30.4	29.0	43.6
A5	✗	✓	8	0	**31.4**	26.9	43.5
B1	✗	✗	1	0	2.5	2.8	12.6
B2	✗	✗	2	0	20.5	20.6	39.8
B3	✗	✗	4	0	31.6	**29.1**	**43.5**
B4	✗	✗	6	0	32.2	27.3	42.7
B5	✗	✗	8	0	**32.5**	26.0	42.1
B6	✗	✗	10	0	32.2	24.2	41.4

Table 6. Ablation study on prompting SAM [24] with varying positive keypoints (⊕) on OCHuman-val. Best results for each metric highlighted in **bold**; best method for BMP highlighted in blue. Green text indicates improvement over the baseline, red text indicates a decline. Detection and segmentation often conflict (Fig. 5). More keypoints improve segmentation (including incorrect masks) and bounding box detection, but increase segmentation errors. Pose remains stable but suffers from both wrong segmentation (guidance errors) and wrong detection (crop errors).

name	batch	bbox	⊕	⊖	bbox	segm	pose
RTMDet [21] + MaskPose					31.1	27.1	45.3
A3	✗	✓	4	0	29.3	**30.9**	44.0
C1	✗	✓	4	1	29.5	30.5	44.3
C2	✗	✓	4	3	**29.8**	28.2	**44.2**
C3	✓	✓	4	–	29.3	**30.9**	44.0
B4	✗	✗	6	0	**32.2**	**27.3**	42.7
C4	✗	✗	6	1	29.9	23.8	43.6
C5	✗	✗	6	3	27.5	19.2	**44.1**
C6	✓	✗	6	–	**32.2**	**27.3**	42.7

Table 7. Ablation study on prompting SAM [24] with varying negative keypoints (⊖) on OCHuman-val. Best results for each metric in **bold**; best method for BMP highlighted in blue. Green text indicates improvement over the baseline, red text indicates a decline. Adding negative keypoints to bounding boxes hinders segmentation but slightly improves detection. Without bounding boxes, negative keypoints degrade both detection and segmentation. Processing all image instances simultaneously (batch) gives the same or worse results.

ing.

In the top section, which includes bounding box prompts, using more keypoints increases the likelihood of confusing the model, leading to a drop in segmentation quality. However, more keypoints also increase the chance of expanding the mask beyond the bounding box, which improves detection. In particular, using 8 keypoints as positive prompts slightly outperforms the original baseline in detection.

The second section, without bounding box prompts, highlights that too few keypoints fail to define the instance adequately, causing both detection and segmentation to fail catastrophically. The best segmentation results occur with 4 keypoints, while detection performs best with 8. We chose 6 keypoints as a middle ground, balancing strong detection performance with slightly improved segmentation.

Number of negative keypoints (⊖). SAM2 provides two methods for negative prompting: explicit negative prompts and batch processing of all instances in the image. For explicit negative prompts, we identify the closest keypoint from other instances in the same image, provided it has confidence above a specified threshold.

Tab. 7 evaluates the impact of negative keypoint prompts. The top section examines adding negative prompts to 4 positive prompts and a bounding box. Negative prompts slightly improve detection quality, but significantly reduce segmentation quality. Given the trade-off, the decrease in segmentation outweighs the minor improvement in detection, so we avoid using negative keypoints in this setup.

The bottom section evaluates the effect of negative prompts without a bounding box prompting. Here, adding negative keypoints decreases both detection and segmentation performance, making it ineffective for this configuration.

Batch processing. Tab. 7 also evaluates the impact of batch processing, where SAM is prompted with multiple instances simultaneously. In this approach, SAM outputs non-overlapping masks for each prompted instance, ensuring that no mask is a subset of another. Although this behavior is logical, batch processing consistently produced the same or slightly lower results compared to single-instance processing in all our experiments.

We chose to stick with single-instance processing, as it likely allows the model to optimize better for one instance at a time, even if the resulting masks may overlap. Overlaps could be resolved in a post-processing step using pose information.

Confidence threshold (T_c). The top part of Tab. 8 examines the effect of varying the confidence threshold T_c for selecting keypoints as prompts. Lower thresholds select keypoints with greater variability but increase the risk of using incorrectly estimated keypoints. The best results are achieved with a threshold of $T_c = 0.3$, which aligns with its common use in heatmap-based pose estimation models.

Interestingly, a lower threshold ($T_c = 0.1$) outperforms a higher threshold ($T_c = 0.8$), suggesting that variability is more important than strictly ensuring keypoint correctness. This may indicate that SAM is either robust to incor-

name	batch	bbox	$\oplus$	$\ominus$	T_c	sel.	ext. bbox	P-Mc	bbox by IoU	bbox	segm	pose
RTMDet [21] + MaskPose										31.1	27.1	45.3
Confidence threshold T_c												
D1	✗	✗	6	0	0.8	c+d	—	✗	✗	29.9	27.2	42.1
B4	✗	✗	6	0	0.5	c+d	—	✗	✗	32.2	27.3	42.7
D2	✗	✗	6	0	0.4	c+d	—	✗	✗	32.4	27.6	43.1
D3	✗	✗	6	0	0.3	c+d	—	✗	✗	**32.7**	27.9	43.3
D4	✗	✗	6	0	0.2	c+d	—	✗	✗	32.5	**28.3**	**43.6**
D5	✗	✗	6	0	0.1	c+d	—	✗	✗	32.5	28.2	**43.6**
Selection method												
D3	✗	✗	6	0	0.3	c+d	—	✗	✗	**32.7**	**27.9**	43.3
E1	✗	✗	6	0	0.3	c	—	✗	✗	29.7	26.2	**45.0**
E2	✗	✗	6	0	0.3	d	—	✗	✗	34.6	20.6	36.8
Extended bounding box												
F1	✗	✓	4	0	0.3	c+d	✗	✗	✗	29.3	**31.1**	44.1
F2	✗	✓	4	0	0.3	c+d	✓	✗	✗	29.7	31.0	44.1
Pose-Mask consistency												
D3	✗	✗	6	0	0.3	c+d	—	✗	✗	**32.7**	27.9	43.3
G1	✗	✗	6	0	0.3	c+d	—	✓	✗	30.9	**31.1**	**45.0**
Bounding box by max_IoU												
D3	✗	✗	6	0	0.3	c+d	—	✗	✗	**32.7**	27.9	43.3
F1	✗	✓	4	0	0.3	c+d	✗	✗	✗	29.3	**31.1**	**44.1**
H1	✗	✗/✓	6/4	0	0.3	c+d	✗	✗	✓	29.7	30.1	43.9
Final methods												
D3	✗	✗	6	0	0.3	c+d	—	✗	✗	**32.7**	27.9	43.3
J1	✗	✗/✓	6/4	0	0.5	c+d	✓	✓	✓	29.2	31.1	46.3

Table 8. Ablation study on prompting SAM [24] with varying confidence thresholds (T_c), keypoint selection methods (sel.), and additional techniques on OCHuman-val. Best results for each metric in **bold**; best method for BMP highlighted in blue. Green text indicates improvement over the baseline, red text indicates a decline. Final methods used in BBox-Mask-Pose are highlighted in green. Two different methods used: one for the BMP loop, another for mask and pose refinement.

rect prompts (which we find unlikely) or that confidence is not a reliable metric for evaluating keypoint accuracy. As human pose estimation models are often overconfident, using self-estimated OKS from [12] could likely yield better results than relying on confidence.

Selection method (sel.). We compare three methods for selecting keypoints as prompts. The first method, confidence-only (c), sorts keypoints by confidence and selects the top N most confident ones. The second, distance-only (d), selects the N keypoints farthest from the center of the bounding box. The third method, described in Sec. 3.3, combines confidence and distance (c+d).

The second part of Tab. 8 shows that combining confidence and distance (c+d) outperforms either approach alone, providing superior results.

Extending bounding box. Experiment F2 in Tab. 8 explores the idea of extending the bounding box when using it for prompting. If selected keypoints fall outside the bounding box, it is extended to include all prompt keypoints. This ensures that no positive prompt lies outside the bounding box.

The results show that extending the bounding box slightly improves the detection accuracy while maintaining segmentation and pose estimation performance when using the bounding box. This approach is not applicable when prompting without a bounding box.

Pose-Mask consistency (P-Mc). Experiment G1 in Tab. 8 evaluates the effect of Pose-Mask Consistency (P-Mc), as described in Sec. 3.3. P-Mc significantly improves segmentation and pose estimation, but reduces detection performance. As a result, it is highly effective for refining masks and poses when the bounding box is approximately correct but not suitable for use in the iterative BBox-Mask-Pose loop.

Bounding box depending on max_IoU. The last experiment (H1) involves prompting with a bounding box only

Figure 6. Multiple background instances may merge into a single mask when no bounding box is provided as a prompt. The yellow mask was refined and covers all spectators. Foreground instances are omitted in the left image for clarity.
Left – RTMDet [21], right – BMP.

for instances with $max\_IoU > 0.5$. The rationale is that bounding boxes are typically accurate for isolated instances, where bounding box prompting improves results. However, for highly overlapping instances, the bounding box is often inaccurate and degrades detection performance. The results of this experiment are in Tab. 8.

As expected, the results fall between always prompting with bounding boxes and never using them. While this approach significantly improves segmentation compared to prompting without bounding boxes, the improvement in detection over always prompting with bounding boxes is minor. A qualitative analysis reveals that this method is primarily beneficial for low-resolution background instances, such as spectators in sports images. Without bounding box prompting, SAM often segments the entire background, leading to inaccuracies. This phenomenon is not well captured in the evaluation, as background instances rarely have pose annotations and have limited detection and segmentation labels. An example is shown in Fig. 6.

A.3. Summary

The ablation study on automated SAM prompting is extensive and may seem overwhelming. To provide a clear summary, the last rows of Tab. 8 present two prompting methods used in BBox-Mask-Pose (BMP).

D3: This method is used in the BMP loop to balance refined masks with improved detection. It primarily enhances detection accuracy while slightly improving segmentation. Although it does not achieve the best standalone results, it performs best when used within the closed BMP loop with re-detections.

J1: This method is designed to refine masks and poses to produce high-quality estimates. It is used, for instance, in BMP ablations (Sec. 4.3) to loop SAM and MaskPose without re-detection. It significantly improves segmentation and pose estimation but is not part of the reported BMP results. J1 could be applied after the BMP loop terminates to further refine masks and bounding boxes, but we avoided this because it introduces additional overhead by requiring extra SAM (and possibly MaskPose) iterations. While such micro-loops and adjustments could further improve the reported results, our focus is on maintaining clarity, showing that two simple loops are sufficient to improve detection, segmentation, and pose estimation.

(a) Two people in matching coats.

(b) Two boys in one pair of pants, wearing matching shirts.

(c) Two players with matching jerseys.

Figure 7. Instances not split even after mask refinement by SAM [24], typically due to similar or identical textures.

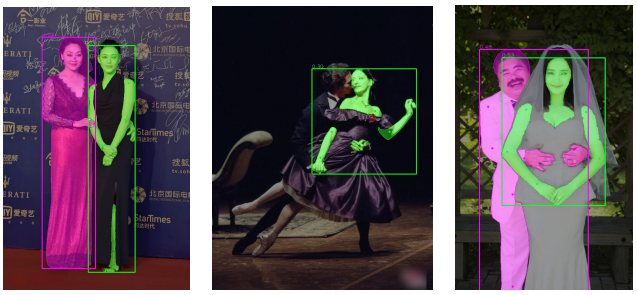
Pose estimation robustness. Pose estimation demonstrates notable robustness to the quality of estimated masks. MaskPose consistently produces accurate poses, even with low-quality masks (e.g., experiment C5 in Tab. 7), and almost always outperforms the ViTPose [36] baseline conditioned by the bounding box. However, achieving the MaskPose-SAM-MaskPose self-improving loop requires employing several hand-crafted tweaks. Among these, the Pose-Mask Consistency, as used in experiment J1 in Tab. 8, is particularly critical. Overall, BMP's pose estimation benefits more from refined detections and re-detection of background instances than from refining masks through SAM. This highlights the importance of robust detection to improve overall performance within the BMP framework.

B. Failure cases analysis

Here, we provide a detailed analysis of BMP failure cases. While the most common issues are discussed in the paper, particularly in Sec. 5 and Fig. 4, this section offers additional examples and introduces a previously unmentioned type of error, instance merging.

Merging instances. Even though BMP is designed to decouple instances merged by the detector, and MaskPose performs well in such cases, SAM can mistakenly merge instances if it is incorrectly prompted or if the instances have similar textures. Prominent examples of these failures are shown in Fig. 7.

BMP struggles to address these issues because bounding box prompting would also fail, given that the detected bounding box already merges the instances. Furthermore, Pose-Mask Consistency (P-Mc) does not help in such cases, as only one instance is detected. Without negative key-



(a) (b) (c)

Figure 8. Oversegmentation. Green instances have incorrect masks – only the skin is segmented, excluding the clothes. This issue commonly occurs with clothing that exposes bare shoulders, such as dresses or jerseys. Keypoints used for SAM prompting are marked (best viewed in zoom).

points, a large mask that merges multiple instances (or even covers the entire image) would still achieve $P - Mc = 1.0$, since all positive keypoints fall within the mask and no negative keypoints are present to penalize the score.

Segmenting clothes instead of the whole person. This issue, illustrated in Fig. 8, is particularly common in OCHuman, where many individuals wear specific clothing. The problem frequently arises when a person has bare shoulders, such as in an evening dress or basketball jersey. In such cases, shoulder, facial, knee, elbow, and wrist keypoints, which are on the skin rather than clothing, prompt SAM to segment only the skin, leaving the clothing unsegmented. Hip and sometimes ankle keypoints could help refine segmentation, but these are typically low-confidence predictions and are often not selected.

Unsegmented clothing causes downstream issues as the masking-out step leaves the clothes visible. In subsequent BMPiterations, the detector identifies these as separate instances, as shown in Fig. 4.

We suggest two potential solutions. The first is to improve SAM prompting to include clothing in the segmentation. The bounding box prompt could address this specific case, but it hinders performance in other scenarios, as detailed in Fig. 3b and Appendix A. The second is to fine-tune the detector to ignore clothing when the skin is masked out. However, this approach risks reducing the detector's generalizability and causing overfitting to scenarios with visible skin and faces, which we believe is not a viable long-term solution.

Missing body parts. When SAM fails to segment a body part, it remains unmasked and may be redetected in the next stage, as shown in Figs. 4 and 9. This issue is even more pronounced when prompting with a bounding box, as detected bounding boxes often exclude disconnected limbs, leaving SAM unable to recover them. For this reason, we avoid prompting with the bounding box in the BMP loop.

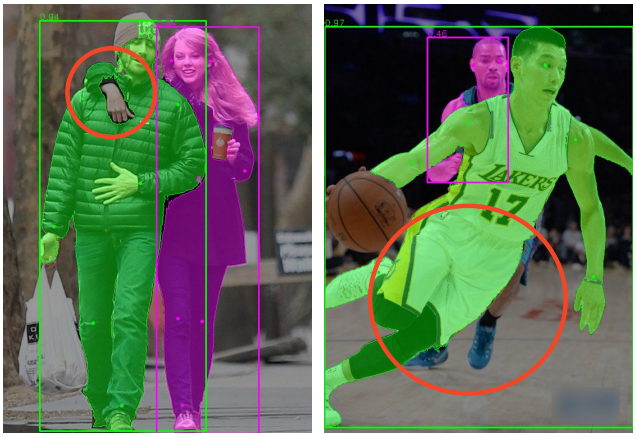


Figure 9. Images where SAM [24] successfully decoupled instances but failed to segment a disconnected body part. These parts remain unmasked and risk being re-detected, as illustrated in Fig. 4c. Keypoints used for SAM prompting are marked (best viewed in zoom).

Missed limbs could potentially be addressed by better alignment between pose and mask. If the refined mask is inconsistent with the prompted pose, SAM could be restarted with different prompts to minimize missed limbs. However, if the limb is also missed by MaskPose, BMP cannot resolve the issue.

Correct examples. BMP performs reliably in most cases, as demonstrated by the quantitative results. Figs. 10 and 11 showcase examples of successful detection and segmentation in challenging multi-body scenarios, including cases where a person is upside down.

In particular, Fig. 10 highlights the ability of BMP to balance segmentation and detection, as discussed in Fig. 5. The improvements are significant, with more precise segmentation and accurate instance counts in the scene. Some small body parts may occasionally be assigned to the wrong instance, but overall performance remains strong.

C. BMP Ablation Study – number of parameters

Tab. 4 in Sec. 4.3 shows the performance change with and without various BMP components. For clarity, we also present Tab. 9, which shows the same result along with the number of trainable parameters of the whole loop. For example, combining the detector (RTMDet-l) with 57M parameters and the pose model (ViTPose-b) with 87M parameters results in 144M trainable parameters.

Omitting SAM from the loop significantly reduces parameters, but also sharply decreases performance. Running the pose estimation again after the SAM refinement increases parameter usage by 40%, from 225M to 312M.

pose	SAM	pose	loops	bbox	pose	params
✓	✓	✗	1×	31.1	45.3	225 M
✓	✓	✗	2×	**32.1**	**48.6**	369 M
✓	✓	✓	1×	31.1	46.4	312 M
✗	✓	✗	2×	<u>31.9</u>	<u>47.3</u>	282 M
✓	✗	✗	2×	30.8	47.0	201 M

Table 9. **Ablation study** of BBox-Mask-Pose components evaluated on OCHuman-val. Bbox and pose evaluated with AP. The sum of trainable parameters approximates computational complexity. First row corresponds to BMP 1×, second to BMP 2×.

Figure 10. Images where BMP improves detection and segmentation using its pose estimates and SAM prompting with selected keypoint. Bounding box prompting did not lead to comparable results. Keypoints used for SAM prompting are marked (best viewed in zoom). Left – RTMDet [21], right – BMP.

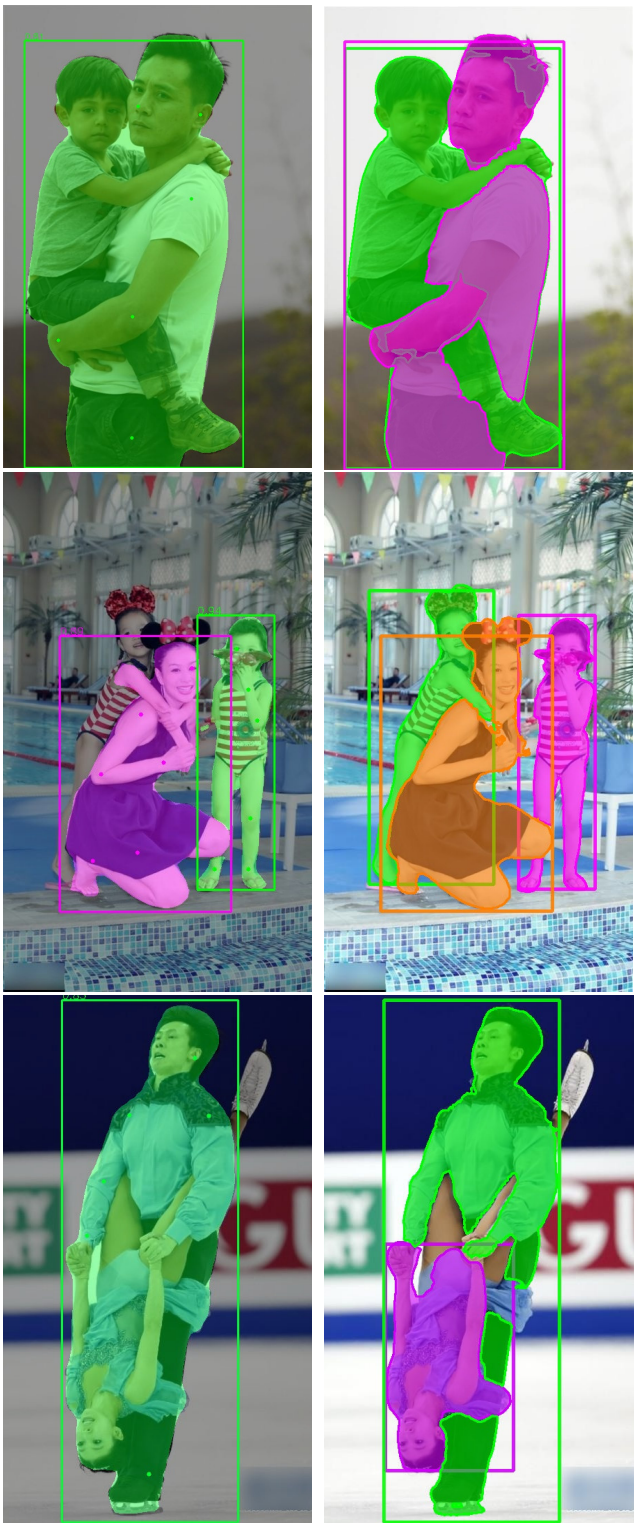


Figure 11. Two iterations of BMP successfully decouple merged instances, even in challenging images with upside-down people. Left – RTMDet [21], right – BMP.

Figure 12. Qualitative results on the OCHuman dataset. Left – RTMDet [21], right – BMP 2×.

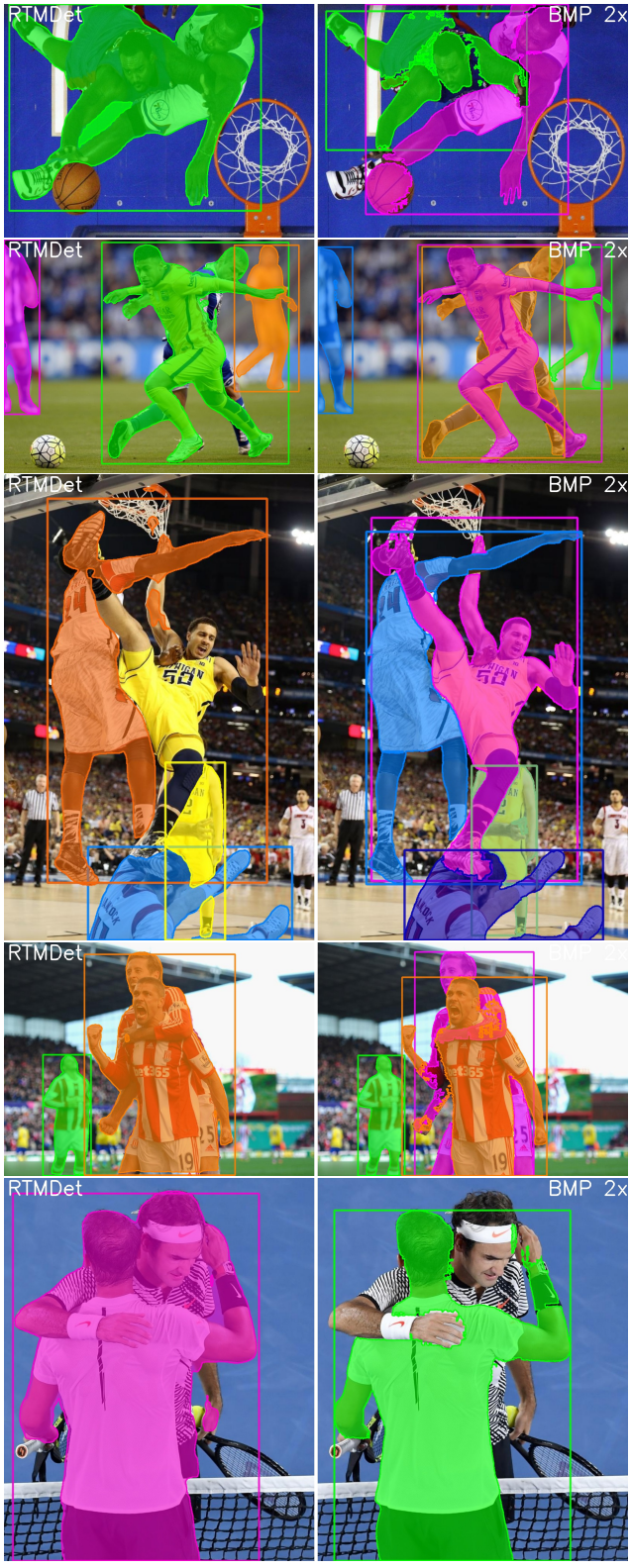


Figure 13. More qualitative results on the OCHuman dataset. Left – RTMDet [21], right – BMP 2×.