A Comparative Analysis of 2D Human Pose Estimation Models in the MMPose Zoo: Paradigms, Architectures, and Performance Benchmarks

Executive Summary

The field of 2D human pose estimation has witnessed a significant evolution, moving from foundational academic concepts to highly optimized, deployment-ready solutions. The MMPose toolbox, a core component of the OpenMMLab ecosystem, stands as a comprehensive repository of state-of-the-art models, encapsulating this progress.[1] This report provides an exhaustive comparative analysis of the 2D multi-person pose estimation models within the MMPose model zoo, structured around the two primary methodological paradigms: top-down and bottom-up.

Historically, the choice between these paradigms involved a distinct trade-off: top-down methods offered superior accuracy at the cost of computational complexity that scaled with the number of individuals, while bottom-up methods provided constant-time inference but often with a precision penalty, particularly in complex scenes. However, this analysis reveals a fundamental shift in this dynamic. The advent of hyper-efficient real-time object detectors has largely nullified the traditional speed bottleneck of the top-down approach, redirecting the optimization focus toward the single-person pose estimator (SPPE) itself.

This shift is epitomized by models like RTMPose, a top-down architecture that achieves an unprecedented balance of speed and accuracy by integrating a lightweight backbone with an efficient coordinate classification head.[3] Concurrently, the rise of one-stage models such as RTMO and YOLOX-Pose, which unify detection and pose estimation into a single network, further blurs the lines between paradigms, leveraging architectural innovations from the object detection domain to push performance boundaries.[5]

This report deconstructs these paradigms, analyzes the specific model-backbone combinations available in MMPose, and presents a rigorous quantitative benchmark based on accuracy (Average Precision on COCO), model complexity (GFLOPs, parameters), and inference speed (FPS). The findings culminate in a strategic framework for model selection, offering tailored recommendations for diverse application scenarios, from high-accuracy offline analysis to resource-constrained, real-time deployment on edge devices.

Section 1: Foundational Paradigms in Multi-Person Pose Estimation

Understanding the core methodologies of multi-person pose estimation is essential for navigating the complex landscape of available models. The field is primarily defined by two competing philosophies—top-down and bottom-up—with modern approaches beginning to synthesize their respective strengths.

1.1 The Top-Down Approach: A "Detect-then-Estimate" Framework

The top-down paradigm operates on a straightforward, two-stage principle that decomposes the complex multi-person problem into a series of simpler single-person tasks.[7]
Methodology:
The process begins with an independent, pre-trained object detector that scans the input image to identify and localize every person, enclosing each in a bounding box. These bounding box regions are then cropped from the original image, resized to a standardized input dimension, and individually fed into a Single-Person Pose Estimator (SPPE). This second-stage network is responsible for predicting the keypoint locations for the single person contained within its input crop.9
Inherent Advantages:
This approach has historically set the benchmark for accuracy. By isolating each individual, the SPPE is shielded from the ambiguity of inter-person keypoint association, a significant challenge in crowded scenes. Furthermore, normalizing the input size for the SPPE allows the model to learn scale-invariant features more effectively, leading to more robust and precise keypoint localization.7
Classical Limitations and the Modern Re-evaluation:
The primary drawbacks of the top-down approach have traditionally been its performance dependency and computational scaling. The final accuracy is fundamentally capped by the person detector's ability to find every individual; a missed detection results in a completely missed pose. More critically, the inference time historically scaled linearly with the number of people in the frame, as the SPPE had to be executed once for each detected person. This made the approach potentially prohibitive for real-time applications in crowded environments.9
However, this long-held view of top-down methods as inherently slow is being actively challenged. The remarkable progress in real-time object detection has produced detectors like RTMDet that are so efficient their contribution to the total pipeline latency is minimal.[13] The RTMPose research explicitly states that "benefiting from the excellent efficiency of real-time detectors... the detection part is no longer a bottleneck of the inference speed of top-down methods".[4] This crucial development reframes the debate entirely. The new

bottleneck is the SPPE itself, and the focus of high-performance top-down design has shifted from mitigating the multi-pass problem to engineering exceptionally fast and efficient SPPE architectures.

The MMPose Ecosystem Advantage:

The OpenMMLab ecosystem provides a significant practical advantage in this context. MMPose is tightly integrated with MMDetection, a comprehensive toolbox for object detection.14 This synergy allows practitioners to seamlessly pair a state-of-the-art pose estimator from MMPose with a state-of-the-art detector from MMDetection, creating a powerful and highly optimized top-down pipeline with minimal effort.13

1.2 The Bottom-Up Approach: An "Estimate-then-Group" Framework

In direct contrast to the top-down methodology, the bottom-up approach tackles the problem holistically in a single pass, first identifying all keypoints in an image and then assembling them into individual skeletons.

Methodology:

The network processes the entire image at once to predict instance-agnostic keypoints—for example, generating heatmaps for all left elbows, all right knees, etc., present in the scene. Following this, a sophisticated grouping mechanism is employed to associate the detected keypoints into distinct person instances. This is often achieved using techniques like Part Affinity Fields (PAFs), which are 2D vector fields that encode the location and orientation of limbs, or associative embeddings, where the network predicts an additional "tag" for each keypoint, and keypoints with similar tags are clustered together.8

Inherent Advantages:

The primary strength of the bottom-up approach is its computational efficiency and robustness to the number of people. The network performs a single forward pass regardless of how crowded the scene is, resulting in a runtime that remains constant.11 This makes it theoretically ideal for applications involving dense crowds. The global context provided by processing the entire image at once can also aid in resolving poses in cases of heavy occlusion.

Classical Limitations:

The main challenge and source of error for bottom-up methods lies in the grouping stage. Associating keypoints correctly is a complex combinatorial problem, especially in scenes where individuals are close together or overlapping. This can lead to incorrect limb associations and a higher rate of false positives, resulting in lower overall precision compared to top-down alternatives.8

1.3 The Modern Synthesis: One-Stage and Hybrid Approaches

Emerging architectures seek to combine the benefits of both paradigms into a unified, end-to-end framework. These are often referred to as one-stage models.

Methodology:
One-stage models, such as YOLOX-Pose and RTMO, perform person detection and keypoint estimation simultaneously within a single network architecture, eliminating the need for separate pre-processing (person detection) or post-processing (keypoint grouping) steps.5
These models are typically built upon successful one-stage object detectors like YOLO, modifying the prediction head to output keypoint coordinates and visibility scores in addition to bounding boxes.6
Bridging the Gap:
These models represent a significant architectural evolution. They retain the single-pass efficiency characteristic of bottom-up methods while adopting a more person-centric representation akin to top-down approaches, often regressing keypoints relative to a detected instance center. This hybrid nature allows them to achieve a compelling balance of speed and accuracy, positioning them as strong contenders in the modern pose estimation landscape.5

Section 2: In-Depth Analysis of Top-Down Architectures in MMPose

The MMPose model zoo features a rich and diverse collection of top-down models, reflecting the paradigm's dominance in achieving state-of-the-art accuracy. These models can be broadly categorized by their output representation strategy—heatmap-based or coordinate-based—which is a primary determinant of their performance profile.

2.1 Canonical Heatmap-Based Models: The Bedrock of High Accuracy

Heatmap-based methods have long been the foundation of high-precision pose estimation. They reframe keypoint localization as a dense prediction problem, offering robustness and sub-pixel accuracy.
Core Concept:
Instead of directly regressing coordinates, these models predict a set of 2D heatmaps, one for each keypoint type. Each heatmap represents a probability distribution over the image space, where the intensity at a given pixel corresponds to the likelihood of that keypoint being present. The final coordinate is typically determined by finding the pixel with the maximum activation (the peak of a Gaussian-like distribution) in each heatmap.17 This probabilistic approach is less sensitive to small errors and allows for more nuanced confidence scoring.
Architectures:

- **Simple Baselines (SimpleBaseline):** Introduced by Xiao et al. (2018), this architecture provides a powerful and straightforward foundation for pose estimation. It employs a standard deep backbone, such as ResNet, followed by a series of transposed convolution (deconvolution) layers to upsample the feature maps and generate the final high-resolution heatmaps. MMPose offers numerous variants of this approach, pairing the SimpleBaseline head with a wide range of backbones, including **ResNet**, **ResNeXt**, and the transformer-based **Swin Transformer**.[6]

- **High-Resolution Networks (HRNet):** A key innovation in dense prediction tasks, HRNet is designed to overcome the information loss inherent in standard downsampling-upsampling architectures. It maintains high-resolution feature representations throughout the network via parallel multi-resolution branches. Crucially, it performs repeated fusions across these branches, allowing for a rich exchange of information between high-resolution (precise spatial detail) and low-resolution (strong semantic context) features. This makes HRNet exceptionally well-suited for spatially sensitive tasks like pose estimation and consistently places it at the upper end of accuracy benchmarks.[6] MMPose also includes
  Lite-HRNet, a lightweight variant designed for applications where computational efficiency is a greater priority.[6]
- **Stacked Hourglass Networks:** This architecture consists of one or more sequential "hourglass" modules. Each module first downsamples the feature maps to capture global context and then upsamples them to reintegrate spatial information. By stacking these modules, the network can iteratively refine its keypoint predictions, leveraging the output of one stage to inform the next, which is effective for capturing complex spatial relationships between joints.[6]
- **Transformer-Based Models (ViTPose & HRFormer):** Leveraging the success of Vision Transformers (ViT) in image classification, these models adapt the transformer architecture for pose estimation. **ViTPose** employs a plain ViT backbone to process image patches as tokens, followed by a simple decoder to generate heatmaps. Its strength lies in the powerful global attention mechanism of the transformer, which allows it to model long-range dependencies between joints effectively.[6]
  HRFormer, in contrast, is a hybrid model that combines the efficiency of convolutional operations with the global context modeling of transformers, creating a hierarchical architecture that balances performance and computational cost.[6]

2.2 Coordinate Representation Models: The Key to Real-Time Performance

While heatmap-based methods are highly accurate, they come with a significant computational cost. The generation and processing of high-resolution heatmaps, especially in the upsampling decoder, are resource-intensive. Furthermore, the final decoding step (finding the argmax) is non-differentiable, which can complicate end-to-end training. Coordinate representation models aim to bypass these bottlenecks for faster inference.
A crucial observation is that the choice of output representation and its corresponding decoder head can be a more significant factor in a model's efficiency than the backbone itself. An ablation study on a SimCC-based model revealed that removing the transposed convolutional layers typical of heatmap decoders saved 35% of the model's parameters and

26.7% of its computation, with only a minor 0.8 AP drop in accuracy.[3] This demonstrates that the decoder architecture is a critical optimization lever. An efficient backbone paired with a heavy heatmap decoder can be slower than a more powerful backbone paired with a lightweight coordinate-based head.

Methods:

- **Direct Regression (DeepPose, Integral Regression):** These are early approaches that attempt to directly regress the (x,y) coordinates of each keypoint using fully connected layers at the end of a CNN backbone.[6] While conceptually simple and fast, they often suffer from lower accuracy as they discard spatial information and struggle with the highly non-linear mapping from image pixels to coordinates. MMPose includes advanced versions of this concept, such as models incorporating **Unbiased Data Processing (UDP)** and **Residual Log-Likelihood Estimation (RLE)**, which are techniques designed to improve the precision and robustness of the regression process.[6]
- **Coordinate Classification (SimCC):** This innovative method, central to many modern real-time models, re-frames keypoint localization not as a regression task but as two separate classification problems. For each keypoint, the model predicts two 1D probability distributions: one over a set of discrete bins along the horizontal axis and one over bins along the vertical axis. The final coordinate is then calculated from the expected value of these distributions. This **Sim**ple **C**oordinate **C**lassification approach brilliantly sidesteps the need for 2D heatmaps and computationally expensive upsampling layers, replacing them with lightweight linear classifiers. This results in a dramatic reduction in model complexity and a significant boost in inference speed, making it a cornerstone of high-performance real-time pose estimation.[5]

2.3 RTMPose: A Case Study in High-Performance Top-Down Design

RTMPose stands as a prime example of modern, efficient top-down model design, synthesizing the latest advancements in network architecture and output representation to achieve an exceptional balance of speed and accuracy.

Architectural Synthesis:

At its core, RTMPose is a top-down model that strategically combines a lightweight yet powerful backbone, CSPNeXt, with the highly efficient SimCC head.4 The CSPNeXt backbone is derived from the architecture used in the RTMDet real-time object detector, inheriting its focus on speed and efficiency. By pairing this optimized feature extractor with the SimCC coordinate classification head, RTMPose avoids the computational overhead of traditional heatmap-based decoders, creating a streamlined and remarkably fast pipeline.3

Performance Profile:

The design choices in RTMPose translate directly into state-of-the-art real-time performance. Benchmarks show that the RTMPose-m model achieves 75.8% AP on the COCO dataset while running at over 90 FPS on an Intel i7 CPU and over 430 FPS on an NVIDIA GTX 1660 Ti GPU.4

Its efficiency extends to mobile platforms, with the
RTMPose-s model achieving 72.2% AP at over 70 FPS on a Snapdragon 865 chip.[3] This level of performance provides compelling evidence for the re-evaluation of the top-down paradigm, demonstrating that with the right architecture, "detect-then-estimate" can be both highly accurate and extremely fast.

Furthermore, the performance of these models can be significantly enhanced through strategic training. For instance, the rtmpose-l-aic-coco model, pre-trained on a combination of the COCO and AIC datasets, achieves 76.5% AP, a notable 0.7 AP improvement over the 75.8% AP of the base rtmpose-l model trained on COCO alone.[6] This gain, achieved solely through more diverse training data, is comparable to the performance jump from some smaller to larger model variants, highlighting that data strategy can be as impactful as architectural scaling.

Section 3: In-Depth Analysis of Bottom-Up & One-Stage Architectures in MMPose

While top-down models have excelled in accuracy, bottom-up and one-stage architectures offer compelling alternatives, particularly for scenarios where inference time must be independent of the number of people in the scene. These models operate without a prior person detection stage, instead relying on sophisticated internal mechanisms for keypoint grouping or integrated detection and estimation.

3.1 Keypoint Grouping Methodologies

Traditional bottom-up models focus on the core challenge of correctly associating detected keypoints into coherent individual skeletons. MMPose implements several powerful approaches to solve this problem.

- **Associative Embedding (AE):** This is a widely used technique for instance grouping. In addition to predicting heatmaps for keypoint locations, the network simultaneously predicts an embedding vector, or "tag," for each detected keypoint. The training objective encourages keypoints belonging to the same person to have similar embedding tags, while keypoints from different people have dissimilar tags. During inference, a simple clustering algorithm groups keypoints with nearby tags to form complete person instances. MMPose provides models that combine this technique with strong backbones, such as the Associative Embedding + Hrnet configuration.[6]
- **Disentangled Keypoint Regression (DEKR):** DEKR represents a more advanced bottom-up approach that aims for a more streamlined pipeline. It learns disentangled representations for each keypoint type and uses adaptively activated convolutions to directly regress keypoint locations from these features. This method includes a

multi-branch structure where each branch focuses on a specific keypoint, and it incorporates a mechanism to link these regressed points to instance centers, effectively performing detection and grouping implicitly. This avoids the need for explicit heatmaps and post-hoc clustering, leading to an efficient and effective bottom-up framework. The DEKR + Hrnet models in MMPose exemplify this approach.[6]

3.2 Integrated One-Stage Models

One-stage models represent the latest evolution in this domain, unifying person detection and keypoint estimation into a single, end-to-end trainable network. This architectural convergence with the object detection field is a dominant trend, as advances in one area are now rapidly transferable to the other. Backbones and head designs pioneered for real-time object detection are being repurposed to create highly efficient pose estimators. The modular structure of the OpenMMLab project, where libraries like MMDetection and MMPose share common components, greatly facilitates this cross-pollination of ideas.[1]

- **YOLOX-Pose:** This model is a direct extension of the high-performance, anchor-free YOLOX object detector. The core YOLOX architecture is adapted by modifying its prediction head to not only predict bounding boxes, objectness scores, and class categories but also the locations and confidences of human keypoints. This creates a fully unified pipeline that outputs all necessary information in a single forward pass, making it highly efficient. The Yoloxpose + Yoloxpose models in the zoo demonstrate this powerful integration.[6]
- **RTMO (Real-Time Multi-person pose estimation with One-stage approach):** RTMO stands as a state-of-the-art one-stage model that achieves an exceptional balance of speed and accuracy. It ingeniously integrates the highly efficient SimCC coordinate classification scheme directly into a YOLO-based architecture.[5] By representing keypoints as dual 1D heatmaps (one for the x-axis, one for the y-axis), RTMO avoids the computational burden of 2D heatmaps while retaining high localization precision. This design allows it to achieve accuracy comparable to leading top-down methods while preserving the high-speed, single-pass inference characteristic of one-stage models, making it particularly advantageous when processing images with multiple people.[5]

Section 4: Quantitative Benchmarking and Performance Analysis

A rigorous, data-driven comparison is essential for understanding the practical trade-offs between different models and paradigms. This section synthesizes performance metrics from the MMPose model zoo to provide a clear, evidence-based analysis of accuracy, computational cost, and inference speed.

4.1 Accuracy vs. Computational Cost

The relationship between a model's predictive accuracy and its computational demand (measured in GFLOPs—billions of floating-point operations) is the most critical trade-off in model selection. The following tables provide a comprehensive overview of the top-down, bottom-up, and one-stage models available in MMPose, evaluated on the COCO val2017 dataset.

The data reveals several key patterns. High-accuracy models like **ViTPose-H** (78.8 AP) and **HRNet-W48+UDP** (77.3 AP) occupy the top-left of the performance spectrum, offering the best precision but at a substantial computational cost. In contrast, the **RTMPose** series defines the modern "Pareto frontier"—the set of models offering the best possible accuracy for a given computational budget. For example, **RTMPose-l** achieves 75.8 AP with just 4.16 GFLOPs, outperforming many heavier models like **ResNet-152** (73.6 AP at 12.77 GFLOPs). This highlights the exceptional efficiency of its architecture.

Among the bottom-up and one-stage models, **DEKR+HRNet-W48** and **RTMO-l** demonstrate strong performance, with 71.4 AP and 72.4 AP respectively. Notably, RTMO achieves this accuracy at a performance point that is highly competitive with top-down models, confirming its status as a leading one-stage solution.

Table 1: Comprehensive Performance of Top-Down Models on COCO val2017

Model	Backbone	Input Size	COCO AP	Params (M)	GFLOPs	GPU FPS (GTX 1660 Ti)	CPU FPS (Intel i7)
ViTPose-H*	ViT-H	256x192	78.8	632.2	125.9	-	-
ViTPose-L*	ViT-L	256x192	78.2	304.4	60.1	-	-
HRNet-W48+UDP	HRNet-W48	384x288	77.3	63.6	35.5	~15	~0.7
RTMPose-l (AIC+COCO)	CSPNeXt-l	384x288	77.3	27.8	9.4	130+	-
Swin-L	Swin-L	384x288	76.3	205.1	53.6	-	-
RTMPose-l	CSPNeXt-l	256x192	75.8	27.7	4.2	430+	90+
HRNet-W32	HRNet-W32	256x192	74.9	28.5	7.7	~23	~2.7
RTMPose-m	CSPNeXt-m	256x192	74.6	13.6	1.9	-	-
ResNet-15	ResNet-152	256x192	73.6	68.6	12.8	~32	~1.9

2							
RTMPose-s	CSPNeXt-s	256x192	71.6	5.5	0.7	-	-
LiteHRNet-30	LiteHRNet-30	256x192	67.6	1.8	0.4	~12	~5.8
MobileNetV2	MobileNetV2	256x192	64.8	9.6	1.6	~92	~10.4

Note: Performance metrics are consolidated from [6], and.[4] FPS figures are approximate and depend heavily on the specific hardware, batch size, and deployment framework. RTMPose FPS figures are reported using optimized frameworks (TensorRT on GPU, ONNX Runtime on CPU). ViTPose models are converted from official repos and may have different performance characteristics.

Table 2: Comprehensive Performance of Bottom-Up & One-Stage Models on COCO val2017

Model	Backbone	Input Size	COCO AP	Params (M)	GFLOPs
RTMO-l	-	640x640	72.4	-	-
DEKR	HRNet-W48	640x640	71.4	-	-
YOLOX-Pose-l	YOLOX-l	640x640	71.2	-	-
CID	HRNet-W48	512x512	70.4	-	-
Associative Embedding	HRNet-W32	512x512	65.6	28.5	41.1
YOLOX-Pose-s	YOLOX-s	640x640	64.1	-	-

Note: Performance metrics are consolidated from [6] and.[5] GFLOPs and Params data for some one-stage models are not consistently reported in the same format.

4.2 Inference Speed and The Deployment Multiplier

While GFLOPs provide a theoretical measure of computational cost, real-world inference speed, measured in Frames Per Second (FPS), is the ultimate metric for practical applications. The provided data shows a wide range of performance, from single-digit FPS for heavy heatmap models on CPU to several hundred FPS for optimized models on GPU.[20]

A critical factor often overlooked is the **deployment multiplier**: the significant speed-up achieved by using optimized inference frameworks like TensorRT or ONNX Runtime instead of the native PyTorch environment. The performance of RTMPose provides a clear example. The RTMPose-m model, when deployed with TensorRT on a GTX 1660 Ti GPU, achieves over 430 FPS. On an Intel i7 CPU with ONNX Runtime, it reaches over 90 FPS.[4] These speeds are

drastically higher than what would be achieved with a naive PyTorch implementation and demonstrate that the choice of deployment backend is as important as the model architecture itself for achieving real-time performance.
Furthermore, for top-down models, batching can significantly improve throughput. While a single-person inference might take a certain amount of time, processing a batch of 10 detected people simultaneously can lead to a much higher overall FPS, as the GPU can parallelize the computations effectively. This is reflected in benchmarks where the batched FPS is often substantially higher than the single-instance FPS multiplied by the batch size.[20]

4.3 The Role of the Backbone

For any given pose estimation algorithm or head, the choice of backbone network is a primary lever for tuning the balance between accuracy and cost. Using the Topdown Heatmap + SimpleBaseline approach as a constant, we can isolate the impact of the backbone.

- **ResNet-50:** Achieves 71.8% AP with 5.46 GFLOPs.
- **ResNet-101:** Increases accuracy to 72.8% AP (+1.0 AP) at 9.11 GFLOPs (+67%).
- **ResNet-152:** Further improves accuracy to 73.6% AP (+0.8 AP) at 12.77 GFLOPs (+40%).
- **Swin-B:** A transformer-based backbone, achieves 73.7% AP at a significantly higher 28.0 GFLOPs.[6]

This comparison clearly illustrates the law of diminishing returns. Scaling from ResNet-50 to ResNet-152 more than doubles the computational cost for a relatively modest gain of 1.8 AP. This analysis empowers practitioners to make an informed decision: if a 1-2 AP point improvement is critical and computational resources are not a constraint, scaling up the backbone is a valid strategy. Otherwise, a smaller backbone or a more efficient overall architecture like RTMPose may be a more prudent choice.

Section 5: Synthesis and Strategic Model Selection

The extensive analysis of paradigms, architectures, and performance benchmarks culminates in a set of strategic recommendations. Selecting the optimal model-backbone combination is not a one-size-fits-all decision but rather a nuanced process guided by the specific constraints and goals of the application.

5.1 A Framework for Model Selection

The following decision-making framework is designed to guide practitioners toward the most suitable model in the MMPose zoo for their use case.
For Maximum Accuracy (Research, Offline Analysis, Data Annotation):
When the primary objective is to achieve the highest possible precision and computational

resources are not a limiting factor, the choice should gravitate towards large, heatmap-based transformer models.

- **Primary Recommendation: ViTPose-H** or **ViTPose-L**. These models leverage the power of large Vision Transformer backbones to achieve state-of-the-art accuracy on the COCO benchmark, with AP scores reaching 78.8%.[6]
- **Alternative: HRNet-W48** with Unbiased Data Processing (UDP). This remains one of the most powerful CNN-based architectures, offering accuracy that is highly competitive with transformer models.[6]
- **Strategy for Enhancement:** For any of these models, seek out checkpoints that have been pre-trained on combined datasets (e.g., COCO+AIC). As demonstrated with RTMPose and HRNet, this can provide a significant accuracy boost with no additional inference cost.[6]

For Real-Time Desktop/Server Applications (Interactive Systems, Sports Analytics):
In scenarios requiring a balance of high accuracy and real-time feedback on server-grade hardware (CPU or GPU), the RTMPose family is the standout choice.

- **Primary Recommendation: RTMPose-l** or **RTMPose-m**. These models provide an exceptional balance of accuracy (74-76% AP) and speed. Their performance is particularly impressive when deployed with an optimized framework.[3]
- **Deployment Strategy:** It is crucial to leverage an optimized inference backend. Use **TensorRT** for NVIDIA GPUs or **ONNX Runtime** for CPUs to unlock the full speed potential of these models, achieving hundreds of FPS on GPU and near 100 FPS on modern CPUs.[4]

For Resource-Constrained Edge/Mobile Devices (On-Device AI, Mobile Apps):
When deploying on hardware with limited computational power and memory, such as mobile phones or embedded systems, model efficiency is paramount.

- **Primary Recommendation: RTMPose-s** or **RTMPose-t**. These models are specifically designed for this domain. Benchmarks showing RTMPose-s achieving over 70 FPS on a Snapdragon mobile chip while maintaining a respectable 72.2% AP underscore their suitability for on-device applications.[4]
- **Alternatives: Lite-HRNet** or **MobileNetV2-based** models are also viable options, especially if a heatmap-based representation is required for the application logic. They offer a good trade-off between size, speed, and accuracy.[6]

For Highly Crowded Scenes (Public Spaces, Event Monitoring):
In environments with a large and variable number of people, the constant-time inference of bottom-up or one-stage models is theoretically advantageous.

- **Primary Recommendation: RTMO** or **DEKR**. These models process the entire image in a single pass, meaning their runtime does not increase as more people enter the scene. RTMO, in particular, offers accuracy that is highly competitive with top-down methods, making it a strong candidate.[5]
- **Practical Consideration:** The speed of modern top-down models like RTMPose is so high that they can often still provide real-time performance even in moderately crowded scenes (e.g., up to 6-10 people), where they may still hold an accuracy advantage.[3]

Therefore, the choice should be based on empirical testing for the specific density of people expected in the target application.

5.2 Concluding Remarks and Future Outlook

The landscape of 2D human pose estimation, as captured by the MMPose model zoo, is characterized by rapid innovation and a blurring of traditional boundaries. The classical dichotomy of "accurate but slow" top-down versus "fast but inaccurate" bottom-up is an oversimplification in the modern era. The development of highly efficient SPPEs like RTMPose has enabled top-down pipelines to achieve real-time performance without sacrificing their accuracy advantage. Simultaneously, one-stage models like RTMO are closing the accuracy gap, offering a compelling synthesis of speed and precision.
Looking forward, several trends are likely to shape the future of the field. The architectural convergence between object detection and pose estimation will continue, with innovations in one domain quickly propagating to the other. The role of transformers, particularly in modeling complex, long-range dependencies between joints, will likely expand, with ongoing research focused on making these powerful models more efficient. Finally, the emphasis on deployment-friendliness, exemplified by the OpenMMLab ecosystem's focus on tools like MMDeploy and benchmarks on diverse hardware from servers to mobile devices, will remain a key driver. The ultimate goal is the continued development of models that are not only accurate but also robust, efficient, and readily deployable in real-world applications.

Works cited

1. MMPose - PyTorch - Model Zoo, accessed on September 1, 2025, https://modelzoo.co/model/mmpose
2. guxiaowei1/mmpose - GitHub, accessed on September 1, 2025, https://github.com/guxiaowei1/mmpose
3. RTMPOSE: REAL-TIME MODELS FOR MULTI-PERSON POSE ESTIMATION - OpenReview, accessed on September 1, 2025, https://openreview.net/pdf?id=STxmh1ZLOI
4. RTMPose: Real-Time Multi-Person Pose Estimation based on MMPose - ResearchGate, accessed on September 1, 2025, https://www.researchgate.net/publication/369233598_RTMPose_Real-Time_Multi-Person_Pose_Estimation_based_on_MMPose
5. RTMO: Towards High-Performance One-Stage Real-Time Multi-Person Pose Estimation, accessed on September 1, 2025, https://arxiv.org/html/2312.07526v1
6. Overview — MMPose 1.3.2 documentation, accessed on September 1, 2025, https://mmpose.readthedocs.io/en/latest/model_zoo.html
7. Full-BAPose: Bottom Up Framework for Full Body Pose Estimation - PMC, accessed on September 1, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC10098694/
8. (PDF) The comparison of top-down and bottom-up methods in multi-person

pose estimation, accessed on September 1, 2025, https://www.researchgate.net/publication/382370655_The_comparison_of_top-down_and_bottom-up_methods_in_multi-person_pose_estimation
9. What does top down and bottom up approaches mean? - Stack Overflow, accessed on September 1, 2025, https://stackoverflow.com/questions/56121558/what-does-top-down-and-bottom-up-approaches-mean
10. RZ/V2MA AI Implementation Guide MMPose HRNet revision 7.20 - Renesas Electronics Corporation, accessed on September 1, 2025, https://www.renesas.com/en/document/apn/rzv2ma-ai-implementation-guide-mmpose-hrnet-revision-720
11. Training and Optimizing a 2D Pose Estimation Model with NVIDIA TAO Toolkit, Part 1, accessed on September 1, 2025, https://developer.nvidia.com/blog/training-optimizing-2d-pose-estimation-model-with-tao-toolkit-part-1/
12. RTMW: Real-Time Multi-Person 2D and 3D Whole-body Pose Estimation - arXiv, accessed on September 1, 2025, https://arxiv.org/html/2407.08634v1
13. [2303.07399] RTMPose: Real-Time Multi-Person Pose Estimation based on MMPose - ar5iv, accessed on September 1, 2025, https://ar5iv.labs.arxiv.org/html/2303.07399
14. Model Zoo, accessed on September 1, 2025, https://platform.openmmlab.com/modelzoo/
15. open-mmlab/mmdetection: OpenMMLab Detection Toolbox and Benchmark - GitHub, accessed on September 1, 2025, https://github.com/open-mmlab/mmdetection
16. mmpose-webcam-demo/README_en.md at main - GitHub, accessed on September 1, 2025, https://github.com/open-mmlab/mmpose-webcam-demo/blob/main/README_en.md
17. An In-Depth Analysis of 2D and 3D Pose Estimation Techniques in Deep Learning: Methodologies and Advances - MDPI, accessed on September 1, 2025, https://www.mdpi.com/2079-9292/14/7/1307
18. RTMPose: The All-In-One Real-time Pose Estimation Solution for Application and Research!, accessed on September 1, 2025, https://openmmlab.medium.com/rtmpose-the-all-in-one-real-time-pose-estimation-solution-for-application-and-research-6404f17cd52f
19. mmpose-0.29.0/README.md · camenduru/show at 33365020c067ef6fa051c355bc31fd56fe608e3b - Hugging Face, accessed on September 1, 2025, https://huggingface.co/camenduru/show/blob/33365020c067ef6fa051c355bc31fd56fe608e3b/mmpose-0.29.0/README.md
20. Inference Speed — MMPose 0.29.0 documentation, accessed on September 1, 2025, https://mmpose.readthedocs.io/en/0.x/inference_speed_summary.html