MAS: Multi-view Ancestral Sampling for 3D Motion Generation Using 2D Diffusion - Supplementary Materials

Roy Kapon, Guy Tevet, Daniel Cohen-Or and Amit H. Bermano
Tel Aviv University
roykapon@mail.tau.ac.il

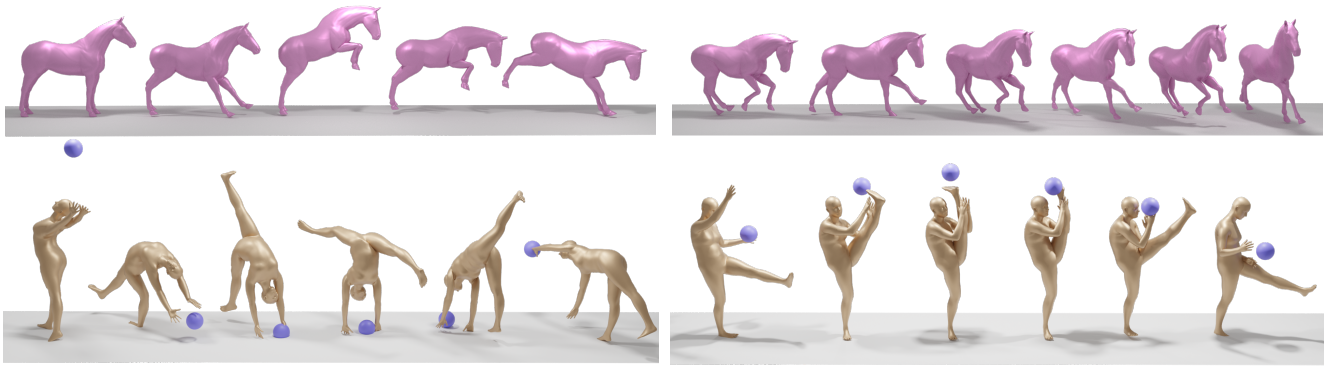

Figure 1. 3D motions generated by Multi-view Ancestral Sampling (MAS) — each one using a different initial noise. Our method generates novel 3D motions using a 2D diffusion model. As such, it enables learning intricate 3D motion synthesis solely from monocular video data.

Abstract

We introduce Multi-view Ancestral Sampling (MAS), a method for 3D motion generation, using 2D diffusion models that were trained on motions obtained from in-the-wild videos. As such, MAS opens opportunities to exciting and diverse fields of motion previously under-explored as 3D data is scarce and hard to collect. MAS works by simultaneously denoising multiple 2D motion sequences representing different views of the same 3D motion. It ensures consistency across all views at each diffusion step by combining the individual generations into a unified 3D sequence, and projecting it back to the original views. We demonstrate MAS on 2D pose data acquired from videos depicting professional basketball maneuvers, rhythmic gymnastic performances featuring a ball apparatus, and horse races. In each of these domains, 3D motion capture is arduous, and yet, MAS generates diverse and realistic 3D sequences. Unlike the Score Distillation approach, which optimizes each sample by repeatedly applying small fixes, our method uses a sampling process that was constructed for the diffusion framework. As we demonstrate, MAS avoids common issues such as out-of-domain sampling and mode-collapse. https://guytevet.github.io/mas-page/

A. The MAS algorithm

Algorithm 1 describes the MAS sampling process.

B. Performance Details

Table 1 displays the time needed for a single sample generation and the GPU memory it consumes.

	MAS	DreamFusion	ElePose	MotionBert
Time[sec]	9	17	$2.3 \cdot 10^{-3}$	1
Memory[MB]	794	794	686	784

Table 1. Time and memory costs per single sample generation.

C. Dynamic View-point Sampling

Keeping the optimized views constant could theoretically lead to overfitting a motion to the optimized views, while novel views might have a lower quality. Note that this problem arises only at a lower number of views (< 5). For this reason, we suggest a way to re-sample the viewing-points: After every step, we can save $X^{(i)}$ and the 3D noise sample used $\epsilon_{3D}^{(i)}$. When trying to sample x_t for a newly sampled view v we can then take all $X^{(0)}, ..., X^{(T-t)}$, and all $\epsilon_{3D}^{(0)}, ..., \epsilon_{3D}^{(T-t)}$ and project them to view v. We can then apply a sampling loop using the projections, just like we did

Algorithm 1 Multi-view Ancestral Sampling (MAS)

Sample camera views: $v_{1:V} \sim \mathcal{V}$
Initialize 3D noise: $\varepsilon_{3D} \sim \mathcal{N}_{L\times J\times 3}(0, I)$
Initialize views by projection: $x_T^{1:V} = P(\varepsilon_{3D}, v_{1:V})$
for $t = T, T-1, ..., 0$ **do**
  $\hat{x}_0^{1:V} = G_{2D}\left(x_t^{1:V}\right)$
  Triangulate: $X = \underset{X'\in\mathbb{R}^{L\times J\times 3}}{\operatorname{argmin}} ||P(X', v_{1:V}) - \hat{x}_0^{1:V}||_2^2$ ▷ $X, \varepsilon_{3D} \in \mathbb{R}^{L\times J\times 3}$
  Back-project: $\tilde{x}_0^{1:V} = P(X, v_{1:V})$
  ▷ $x_t^{1:V}, \hat{x}_0^{1:V}, \tilde{x}_0^{1:V}, \varepsilon^{1:V} \in \mathbb{R}^{V\times L\times J\times 2}$
  Sample noise: $\varepsilon_{3D} \sim \mathcal{N}_{L\times J\times 3}(0, I)$
  Project noise: $\varepsilon^{1:V} = P(\varepsilon_{3D}, v_{1:V})$
  Denoising step: $x_{t-1}^{1:V} = \frac{\beta_t\sqrt{\bar{\alpha}_{t-1}}}{1-\bar{\alpha}_t}x_t^{1:V} + \frac{(1-\bar{\alpha}_{t-1})\sqrt{\alpha_t}}{1-\bar{\alpha}_t}\tilde{x}_0^{1:V} + \frac{\beta_t(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\varepsilon^{1:V}$
end for
Output triangulation: $\underset{X'}{\operatorname{argmin}} ||P(X', v_{1:V}) - x_0^{1:V}||_2^2$

in the original algorithm. We observe that in our setting, this method does not lead to significant improvement so we present it as an optional addition.

D. Data Collection

To demonstrate the merits of MAS we collected three 2D motion datasets, extracted from in-the-wild videos.

NBA videos. We collected about $10K$ videos from the NBA online API[1]. We then applied multi-person tracking using ByteTrack [9], and AlphaPose [1] for 2D human pose estimation (based on the tracking results). We finally processed and filtered the data by centering the people, filtering short motions, crowd motions, and motions of low quality, splitting discontinuous motions (caused typically by tracking errors), mirroring, and applying smoothing interpolations.

Horse jumping contests. We collected 3 horse jumping contest videos (around 2-3 hours each) from YouTube.com. We then apply YoloV7 [5] for horse detection and tracking and VitPose [6] trained on APT-36K [7] for horse pose estimation. The post-processing pipeline was similar to the one described above.

Rhytmic-ball gymnastics. We used the Rhythmic Gymnastics Dataset [8] to get 250 videos, about 1.5 minutes long each, of high-standard international competitions of rhythmic gymnastics performance with a ball. We followed the pipeline described for NBA videos to obtain athletes' motions and also use YoloV7 [5] for detecting bounding boxes of sports balls. We take the closest ball to the athlete at each frame and add the center of the bounding box as an additional "joint" in the motion representation.

All motions are represented as $x \in \mathbb{R}^{L\times J\times 2}$, where NBA is using the AlphaPose body model with 16 joint, horses represented according to APT-36K with 17 joints and the gymnastics dataset is represented with the COCO body model [2] with 17 joints plus additional joint for the ball. All 2D pose predictions are accompanied by confidence predictions per joint per frame which are used in the diffusion training process.

[1] https://github.com/swar/nba_api

E. Additional Experiments

Table 2 presents a comparison of our method with off-the-shelf SOTA methods for supervised pose lifting - MotionBERT [11], unsupervised pose lifting - ElePose [4], and DreamFussion [3] adaptation. MAS is on par with the lifting method for the more challenging side views.

Table 3 depicts an ablation study for the number of views, camera distance, and diffusion steps.

Figure 2 presents a screenshot from the user study presented in the paper, including the wording of the questions for each of the three aspects - *Precision, Diversity, and Quality*.

F. Gradient Update Formula

In order to clarify the difference between SDS and our method, we calculate the gradient update formula w.r.t our optimized loss. Denote by $X^{(i)}$ the optimizing motion at iteration i. When differentiating our loss w.r.t $X^{(i)}$ we get:

$$\nabla_{X^{(i)}} \|P\left(X^{(i)}\right) - \hat{x_0}\|_2^2 \quad (1)$$

$$= \left(P\left(X^{(i)}\right) - \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\phi(x_t)}{\sqrt{\bar{\alpha}_t}}\right)\frac{\partial p}{\partial X^{(i)}} \quad (2)$$

which is clearly differers from $\nabla\mathcal{L}_{\text{SDS}}$. Let us observe substituting our x_t sampling with a simple forward diffusion: $x_t = \sqrt{\bar{\alpha}_t}P\left(X^{(i-1)}\right) + \left(\sqrt{1-\bar{\alpha}}\right)\varepsilon$ - as used in DreamFusion. (This formulation is also analyzed in HIFA [10]):

	FID↓		Diversity→		Precision↑		Recall↑	
View Angles	All	Side	All	Side	All	Side	All	Side
Human3.6M (GT)	$7.34^{\pm 0.18}$		$10.74^{\pm 0.15}$		$0.52^{\pm 0.01}$		$0.91^{\pm 0.005}$	
ElePose	$11.20^{\pm 0.36}$	$24.13^{\pm 0.16}$	$10.67^{\pm 0.05}$	$10.24^{\pm 0.08}$	$0.47^{\pm 0.02}$	$\mathbf{0.41^{\pm 0.01}}$	$0.80^{\pm 0.01}$	$0.25^{\pm 0.01}$
MotionBert	$14.05^{\pm 0.14}$	$24.12^{\pm 0.29}$	$11.46^{\pm 0.07}$	$\mathbf{11.18^{\pm 0.06}}$	$0.32^{\pm 0.01}$	$0.21^{\pm 0.01}$	$0.88^{\pm 1.21e-03}$	$0.56^{\pm 0.02}$
MAS (ours)	$\mathbf{15.15^{\pm 0.16}}$		$11.94^{\pm 0.07}$		$0.21^{\pm 0.01}$		$\mathbf{0.92^{\pm 0.01}}$	

Table 2. **Comparison with pose lifting on Human3.6M dataset.** MAS has a competitive performance to lifting methods that were designed for this dataset. However, MAS outperforms the lifting methods when evaluated from the side view. Here, **bold** marks the best results when comparing to the side view.

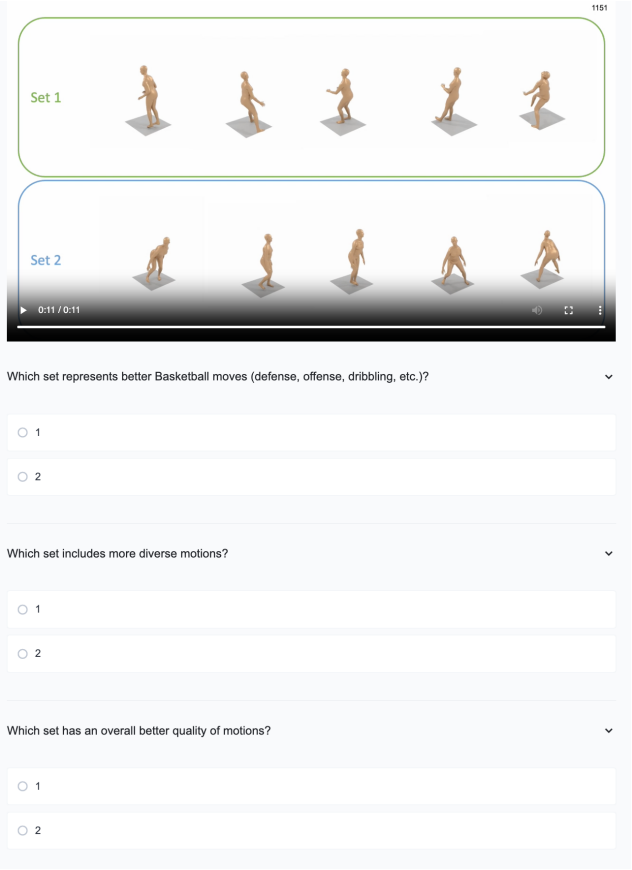

Figure 2. **NBA User study screenshot.** A screenshot from the user study conducted with https://www.pollfish.com/.

$$\nabla_{X^{(i)}} \| P\left(X^{(i)}\right) - \hat{x_0} \| = \quad (3)$$

$$\left(P\left(X^{(i)}\right) - \frac{x_t - \sqrt{1-\bar{\alpha}_t}\epsilon_\phi\left(x_t\right)}{\sqrt{\bar{\alpha}_t}} \right) \frac{\partial p}{\partial X_i} = \quad (4)$$

$$\left(P\left(X^{(i)}\right) - \frac{\sqrt{\bar{\alpha}_t} P\left(X^{(i-1)}\right) + \sqrt{1-\bar{\alpha}_t}\varepsilon - \sqrt{1-\bar{\alpha}_t}\epsilon_\phi\left(x_t\right)}{\sqrt{\bar{\alpha}_t}} \right) \frac{\partial p}{\partial X_i} = \quad (5)$$

$$\left(P\left(X^{(i)}\right) - P\left(X^{(i-1)}\right) + \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} \left(\varepsilon - \epsilon_\phi\left(x_t\right)\right) \right) \frac{\partial p}{\partial X_i} \quad (6)$$

If we observe the first iteration of optimization, we have

	FID↓	Diversity→	Precision↑	Recall↑
Ground Truth	$1.05^{\pm .02}$	$8.97^{\pm .05}$	$0.73^{\pm .01}$	$0.73^{\pm .01}$
#views=2 (120°)	$5.17^{\pm .12}$	$9.86^{\pm .04}$	$0.42^{\pm .03}$	$0.77^{\pm .01}$
#views=3	$4.01^{\pm .15}$	$9.55^{\pm .04}$	$0.53^{\pm .02}$	$0.70^{\pm .01}$
#views=5 (ours)	$\mathbf{3.92^{\pm .15}}$	$\mathbf{9.47^{\pm .07}}$	$\mathbf{0.56^{\pm .03}}$	$\mathbf{0.67^{\pm .01}}$
#views=9	$3.94^{\pm .12}$	$9.48^{\pm .05}$	$\mathbf{0.56^{\pm .02}}$	$\mathbf{0.67^{\pm .01}}$
#views=21	$3.94^{\pm .12}$	$9.48^{\pm .05}$	$\mathbf{0.56^{\pm .02}}$	$\mathbf{0.67^{\pm .01}}$
camera dist=2[m]	$7.59^{\pm .13}$	$\mathbf{9.36^{\pm .05}}$	$0.46^{\pm .01}$	$0.46^{\pm .01}$
camera dist=3[m]	$4.78^{\pm .11}$	$9.45^{\pm .05}$	$0.53^{\pm .01}$	$0.63^{\pm .02}$
camera dist=5[m]	$3.99^{\pm .12}$	$9.47^{\pm .05}$	$\mathbf{0.57^{\pm .02}}$	$\mathbf{0.67^{\pm .01}}$
camera dist=7[m] (ours)	$\mathbf{3.92^{\pm .15}}$	$9.47^{\pm .07}$	$0.56^{\pm .03}$	$\mathbf{0.67^{\pm .01}}$
camera dist=11[m]	$4.04^{\pm .13}$	$9.48^{\pm .05}$	$0.55^{\pm .02}$	$0.66^{\pm .01}$
camera dist=30[m]	$4.29^{\pm .14}$	$9.49^{\pm .05}$	$0.55^{\pm .02}$	$0.65^{\pm .01}$
diff steps=20	$5.14^{\pm .13}$	$9.04^{\pm .01}$	$\mathbf{0.68^{\pm .01}}$	$0.42^{\pm .01}$
diff steps=50	$5.49^{\pm .13}$	$\mathbf{8.99^{\pm .04}}$	$\mathbf{0.68^{\pm .02}}$	$0.36^{\pm .01}$
diff steps=100 (ours)	$\mathbf{3.92^{\pm .15}}$	$9.47^{\pm .07}$	$0.56^{\pm .03}$	$\mathbf{0.67^{\pm .01}}$

Table 3. **NBA Dataset Ablations.** Performance saturates for number of views ≥ 5; Optimal performance achieved at camera distance (dist) around 7 meters; Fewer diffusion steps harm recall and FID.

$X^{(i)} = X^{(i-1)}$ so we get:

$$\nabla_X \| P\left(X\right) - \hat{x_0} \|_2^2 = \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}} \left(\varepsilon - \epsilon_\phi\left(x_t\right)\right) \frac{\partial p}{\partial X} \quad (7)$$

This shows that SDS loss is a special case of our loss when sampling $x_t = \sqrt{\bar{\alpha}_t} P\left(X^{(i-1)}\right) + \left(\sqrt{1-\bar{\alpha}}\right)\varepsilon$ (where $\varepsilon \sim \mathcal{N}\left(0, I\right)$), and applying only a single optimization step (after the first step, $X^{(i)} \neq X^{(i-1)}$).

G. Theorems

Theorem 1. *Let* $\varepsilon = \begin{pmatrix} x_\varepsilon \\ y_\varepsilon \\ z_\varepsilon \end{pmatrix} \sim \mathcal{N}\left(0, I_{3\times 3}\right)$ *and let* $P \in \mathbb{R}^{2\times 3}$ *be an orthogonal projection matrix, then* $P \cdot \varepsilon \sim \mathcal{N}\left(0, I_{2\times 2}\right)$.

Proof. First, $P \cdot \varepsilon$ has a normal distribution as a linear combination of normal variables.
In addition, $\mathbb{E}\left[P \cdot \varepsilon\right] = P \cdot \mathbb{E}\left[\varepsilon\right] = 0$.
Now we will prove that $\mathrm{Var}\left[P \cdot \varepsilon\right] = I_{2\times 2}$:
Denote $O = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ then we know that $P = O \cdot P'$ where P' is a rotation matrix, i.e. $P' \cdot \left(P'\right)^T = I_{2\times 2}$. Then

$$P \cdot P^T = (O \cdot P') (O \cdot P')^T =$$

$$O \cdot \overbrace{(P' P'^T)}^{I} O^T = OO^T = I$$

Furthermore, $\mathbb{E}\left[\varepsilon \cdot \varepsilon^T\right] = \mathbb{E}\left[\varepsilon \cdot \varepsilon^T\right] - \overbrace{\mathbb{E}\left[\varepsilon\right]\mathbb{E}\left[\varepsilon\right]^T}^{0} = \mathrm{Var}\left[\varepsilon\right] = I$.
Therefore:

$$\mathrm{Var}\left[P \cdot \varepsilon\right] = \mathbb{E}\left[(P \cdot \varepsilon)(P \cdot \varepsilon)^T\right] - \overbrace{\mathbb{E}\left[P \cdot \varepsilon\right]}^{0} \cdot \overbrace{\mathbb{E}\left[P \cdot \varepsilon\right]^T}^{0} =$$

$$\mathbb{E}\left[P \cdot \varepsilon \cdot \varepsilon^T \cdot P^T\right] =$$

$$P \cdot \overbrace{\mathbb{E}\left[\varepsilon \cdot \varepsilon^T\right]}^{I} \cdot P^T = P \cdot P^T = I$$

□

Theorem 2. *Let $X \in \mathbb{R}^3$, and denote by $p_{orth}(X), p_{pers}(X)$ the orthographic and perspective projections of X to the same view, respectively. We assume that the subject is centered in the origin and is bounded in a sphere with radius 1 ($\|X\|_\infty \leq 1$). We also assume the perspective projection is dome from distance d from the origin. Then $\|p_{orth}(X) - p_{pers}(X)\|_\infty = O\left(\frac{1}{d-1}\right)$.*

Proof. First, denote the rotation matrix that corresponds to the view by $R \in \mathbb{R}^{3\times3}$ and $R_{xy} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot R$, $R_z = \begin{pmatrix} 0 & 0 & 1 \end{pmatrix} \cdot R$. Then

$$p_{\text{orth}}(X) = R_{xy} \cdot X, p_{\text{pers}}(X) = \frac{R_{xy} \cdot X}{d + R_z \cdot X} \cdot d$$

So

$$p_{\text{orth}}(X) - p_{\text{pers}}(X) =$$

$$R_{xy} \cdot X - \frac{R_{xy} \cdot X}{d + R_z \cdot X} \cdot d = \frac{R_{xy} \cdot X \cdot (d + R_z \cdot X - d)}{d + R_z \cdot X} =$$

$$\frac{R_{xy} \cdot X \cdot R_z \cdot X}{d + R_z \cdot X}$$

Assume $\|X\|_\infty \leq 1$, then $\left\| \frac{R_{xy} \cdot X \cdot R_z \cdot X}{d + R_z \cdot X} \right\|_\infty \leq \frac{1}{d-1}$. □

References

[1] Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[2] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2015. 2

[3] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2

[4] Bastian Wandt, James J. Little, and Helge Rhodin. Elepose: Unsupervised 3d human pose estimation by predicting camera elevation and learning normalizing flows on 2d poses, 2021. 2

[5] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7464–7475, 2023. 2

[6] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 2

[7] Yuxiang Yang, Junjie Yang, Yufei Xu, Jing Zhang, Long Lan, and Dacheng Tao. Apt-36k: A large-scale benchmark for animal pose estimation and tracking. *Advances in Neural Information Processing Systems*, 35:17301–17313, 2022. 2

[8] Ling-An Zeng, Fa-Ting Hong, Wei-Shi Zheng, Qi-Zhi Yu, Wei Zeng, Yao-Wei Wang, and Jian-Huang Lai. Hybrid dynamic-static context-aware attention network for action assessment in long videos. In *Proceedings of the 28th ACM international conference on multimedia*, pages 2526–2534, 2020. 2

[9] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European Conference on Computer Vision*, pages 1–21. Springer, 2022. 2

[10] Junzhe Zhu and Peiye Zhuang. Hifa: High-fidelity text-to-3d with advanced diffusion guidance, 2023. 2

[11] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations, 2023. 2