MEGA: Masked Generative Autoencoder for Human Mesh Recovery

Supplementary Material

	Stochastic						Deterministic
Q	1	5	10	25	50	100	
PVE ↓	84.08	83.27	83.13	83.09	83.02	83.05	83.10
MPJPE ↓	70.88	70.29	70.15	70.09	70.04	70.05	69.95
PA-MPJPE ↓	44.42	43.77	43.66	43.65	43.62	43.61	43.53
Dist. to det.	14.78	9.53	8.56	7.95	7.71	7.61	0.00
SD	-	11.61	12.30	12.65	12.79	12.88	-

Table 4. **Comparison between deterministic and stochastic generation modes.** In stochastic mode, we evaluate the mean mesh obtained with different sample sizes on 10% of the 3DPW [84] dataset, and we provide its distance to the deterministic prediction (Dist. to det.). We also report the standard deviation of the predictions. All metrics are in mm.

A. Link between the deterministic and stochastic modes

To gain deeper insights into the stochastic generation mode, we propose not only evaluating the best sample among the Q generations (the common practice in the literature), but also assessing the mean of the generated meshes. In Appendix A, we compare the performance of the average prediction for different Q using the standard metrics (refer to Sec. 4.1). Additionally, we compute the Euclidean distance between the mean mesh and the one obtained in deterministic mode (Dist. to det., in mm) and the standard deviation of the predictions averaged over all the vertices (SD, in mm). To reduce the computational costs, this study is conducted on a randomly selected 10% subset of images from the 3DPW dataset. Appendix A shows that as Q increases, the average prediction in stochastic mode approaches the deterministic prediction, with a distance around 7 mm for $Q = 100$. The average prediction appears to converge toward a favorable solution, slightly outperforming the deterministic prediction in terms of PVE.

We also examine the distributions of the MPJPE on the 3DPW [84] dataset for MEGA in both deterministic and stochastic modes across various sample sizes Q. In stochastic mode, we analyze the average and best predictions. The results are reported in Fig. 5. Notably, the average prediction error of the stochastic mode appears to converge toward the deterministic prediction error, particularly as Q increases; the distributions are very similar, with overlapping 95% confidence intervals. When Q equals 1, the mean performance is comparatively lower, resulting in higher error values. These observations underscore the importance of having a deterministic mode for rapid and accurate predictions, which can be considered an estimator of the average prediction over Q samples.

When selecting the prediction with the minimum error among N samples, we observe a shift in the distribution shapes, with errors concentrated toward lower values. While the highest error values decrease notably, the lowest remain relatively unchanged. This phenomenon likely occurs because the lowest errors typically correspond to meshes that are easier to predict and exhibit lower standard deviation. Consequently, the stochastic mode proves particularly beneficial for challenging images, where multiple predictions offer valuable insights.

B. Experimental details

Pre-training stage. As detailed in the main body, the pre-training stage is done on a subset of AMASS [62], as introduced in [7]. We pre-train MEGA for 500 epochs, a task accomplished in less than a day on 4 A100 GPUs. MEGA is trained using the AdamW optimizer [60] with a cosine scheduler to adjust the learning rate. The base learning rate is $1e-3$, and we have a warmup of 20 epochs. The optimizer's parameters are $\beta_1 = 0.9$, $\beta_2 = 0.99$, and the weight decay is 0.05.

Supervised training stage. The supervised training stage is done on a mix of standard datasets for HMR as presented in Sec. 4.1. We first train MEGA on MSCOCO [56] for 100 epochs and then train on the whole training set for 10 epochs. Each step takes about 1 day on 4 A100 GPUs. The training settings are exactly the same as the pre-training regarding learning rates and schedulers. For each training step, we start at epoch 0. Note that we have a lower learning rate in practice for the training on the mix of datasets because we stop the training before finishing the warm-up period. For training with HRNet and ResNet-50 backbones, the weights of the backbone are fine-tuned with the same settings as the other parameters of MEGA. When using ViT, the backbone is frozen during the training on MSCOCO, and we only fine-tune the last 10 blocks when training on standard datasets for computing power reasons.

HMR. For recovering human meshes from images in deterministic mode, we predict all images in a single step without randomness. In stochastic mode, we have to set the number of steps for generating the sequence of human mesh tokens and the amount of noise injected for the Gumble-max sampling. Note that we did not test MEGA with a ViT in stochastic mode. With HRNet, we generate meshes in 5 steps, and the initial noise temperature is 1. The generation process with ResNet-50 is made in 2 steps, and the initial noise temperature is fixed to 10. The amount of noise at step t is $A \times (1 - \frac{t}{T})$ where A is the initial noise tempera-

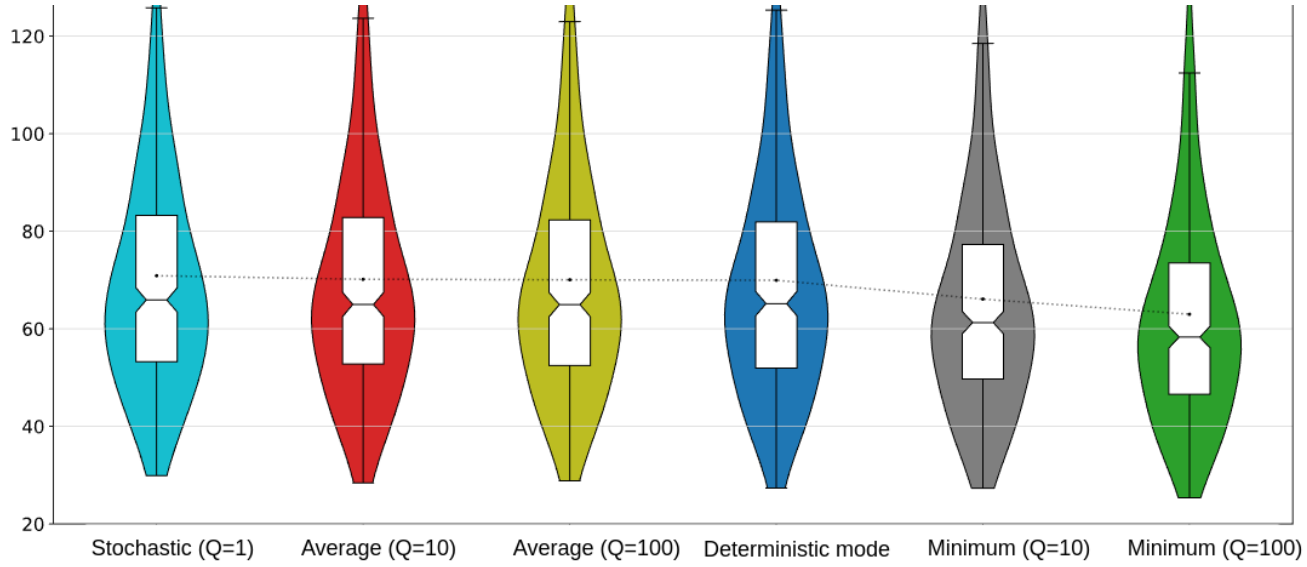

Figure 5. **Error distribution.** We visualize the distribution of the MPJPE in mm on the 3DPW dataset.

ture [9, 10].

Random meshes generation. We generate random meshes in 20 steps (see next section for more details). We want the generation to be completely random for the first steps so that the predictions are diverse. However, the last steps should be almost deterministic to obtain realistic meshes. The initial noise temperature is $A = 1.2$, and the amount of noise at step t is given by $(A \times (1 - \frac{t}{T}))^6$.

Inference time In deterministic mode, with a batch size of 1, a forward pass takes 0.07 seconds with the HRNet version of MEGA on a GeForce GTX 1070 GPU. With similar settings, the ResNet version is real-time (0.03 seconds). In the stochastic mode, generating a single prediction with the ResNet model takes 0.04 seconds, which is still real-time, and generating 16 predictions takes 0.23 seconds.

C. Random meshes generation

We propose to use MEGA pre-trained in a self-supervised manner (see Sec. 3.3) for generating random human meshes. For comparison, we assess its capabilities against VPoser [69] and NRDF [32]. VPoser is a conventional pose prior in VAE form, while NRDF is a SOTA pose prior based on neural fields [88]. Although these models do not explicitly model body shapes, making them not directly comparable to MEGA, which directly generates meshes with diverse poses and shapes, they are the most suitable for comparison purposes. As far as we know, MEGA is the first model generating unconditioned random human body meshes with pose and shape diversity. We assess all 3 models in terms of diversity using the average pairwise distance (APD) in cm, representing the average distance between the joints of all pairs of samples. For plausibility evaluation, we compute the Fréchet inception distance (FID) with the fully convolutional mesh autoencoder introduced in [97] trained on AMASS [62], with a latent space dimension of 7×9. The FID compares the latent representation of generated meshes with that of a representative subset of AMASS introduced in [7].

We randomly sample 500 meshes with each method. Regarding plausibility, MEGA outperforms other methods, achieving an FID of 0.001 compared to 0.007 for VPoser and 0.033 for NRDF. This result is not surprising, as other methods use the average shape for all meshes, whereas MEGA produces diverse results in poses and shapes. NRDF generates more diverse meshes, with an APD of 28.61 cm, while VPoser and MEGA achieve APDs of 18.32 and 20.77 cm, respectively. In summary, MEGA clearly outperforms VPoser, as our generated samples are more diverse and plausible. NRDF produces more diverse poses, but the distribution of the generation samples of MEGA is more representative of the AMASS dataset. In Fig. 6, we present some qualitative samples of MEGA's generation, which exhibit diverse and realistic poses and shapes.

In Fig. 7, similar to the main paper, we visualize the generation process after 1, 5, 9, 13, 16, and 20 steps. The initial mesh appears almost identical across all generations, as only one token is predicted in the first step (with all others set to 0 for visualization). However, diversity quickly emerges in subsequent steps, with the final steps refining the meshes to be more realistic. This pattern was anticipated, as the initial steps involve considerable randomness, whereas the later steps tend to become more deterministic.

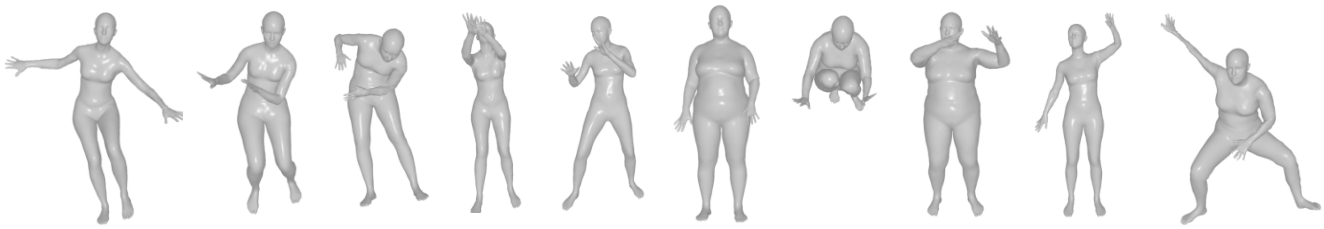

Figure 6. **Random mesh generations.** We use MEGA pre-trained in a self-supervised fashion to generate random human meshes.

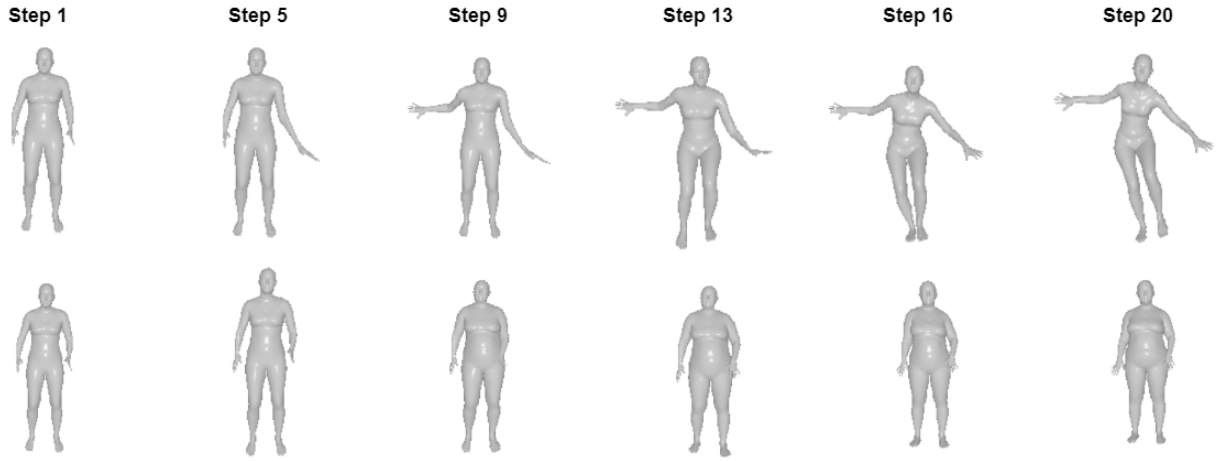

Figure 7. **Prediction process iterations.** We visualize the predictions for intermediate random generations. All masked tokens are replaced by the first token of the codebook, corresponding to index 0.

D. Interpreting diversity in predictions

In the stochastic mode, MEGA makes diverse predictions given a single image. After making multiple predictions given an image, we can compute the standard deviation of the position of each vertex and interpret this value as a measure of the uncertainty in predictions. Indeed, if all samples are similar, we can conclude that the model is "certain" about this prediction. When the results are very diverse given a single image, we can interpret that as high uncertainty.

Some qualitative samples are shown in Fig. 8. The two images on the top show non-occluded bodies, and the relative depth between body parts is easy to perceive. Thus, the standard deviation is low; the model consistently makes accurate predictions. The body is partially occluded in the two bottom images, and the depth of some body parts (such as the left arm in the left image) is hard to estimate. The model makes diverse predictions, which can be interpreted as high uncertainty.

E. Qualitative results and failure cases

We present several failure cases in Fig. 9. Extreme poses can result in prediction errors, occasionally leading to non-anthropomorphic predictions as we do not rely on the SMPL parameters [59]. Notably, the standard deviation of the vertices' position is exceptionally high in such instances.

Fig. 10 presents qualitative samples from in-the-wild datasets. We can observe that in some cases (for instance, images in the third and fourth rows on the right), our predictions appear even more accurate than the ground truth. While this result is encouraging, it underscores the limited value of striving for fractions of millimeters of accuracy on datasets like 3DPW when the ground truth itself is imperfect.

F. Further discussions

Tokenizing vertices vs. SMPL parameters. Predicting the SMPL parameters presents weaknesses [24], such as error accumulation in the kinematic tree, that are addressed when working on the 3D vertices. The SMPL model parameters are attractive because they are a dense representation of human meshes, but once tokenized, this advantage is no longer leveraged. Even if the tokenization of Mesh-VQ-VAE starts from high-dimensional data, the obtained discrete representation is much denser than the representation proposed in TokenHMR in terms of sequence length (54 vs. 160) and

Figure 8. **Visualization of the predictions diversity.** We visualize the standard deviation of the 3D location of each vertex. Bluish regions in the mesh indicate low standard deviation, while reddish areas signify higher standard deviation.

Figure 9. **Failure cases.** In failure cases, it is worth noting that our model predicts very diverse results, which can be interpreted as high uncertainty.

codebook size (512×8 vs. 2048×256): it is easier to predict 54 tokens among 512 values than 160 tokens with 2048 possible values.

Limitations. While MEGA generally produces accurate predictions, it struggles with extreme poses significantly divergent from the training data. Fig. 9 in Appendix E provides visualizations of these failure cases.

Future work. MEGA's adaptable framework suggests potential applications beyond its current scope. Future research could explore generating human meshes conditioned on text inputs [18]. We could also complement image embedding with more observations, such as 2D pose tokens [26] or tokenized meshes of other individuals to model social interactions [29, 65]. Extending this work to videos by incorporating temporal masking during training [71, 72] or including more extreme poses in training data [80] may improve performance. Future works may also focus on other generative models such as discrete diffusion [3].

Potential applications. A direct application of MEGA is to generate solutions until we have a satisfactory answer, similar to LLMs. ScoreHypo [87] proposed a scoring network to select the best prediction among a range of outputs, increasing the accuracy of predictions. We can also choose the most suitable prediction depending on the use case: the solution that minimizes the re-projection error in sports applications for higher precision, the most visually appealing result in animation... The diversity of predictions can also be interpreted as a measure of per-vertex uncertainty [48] (see Sec. D).

Broader impact. MEGA contributes to the understanding of human perception from images. While there is concern about potential misuse for intrusive surveillance, MEGA does not reconstruct facial features, preserving anonymity. MEGA could have applications in healthcare, such as motor assessment of patients. This application would be positive, but potential prediction errors could negatively affect the care pathway.

Figure 10. **Qualitative examples on 3DPW and EMDB.** The first column is the groundtruth, the second and third are the image and the reprojection of the deterministic prediction, and the fourth is the prediction using the deterministic mode of MEGA with an HRNet backbone.