MimicMotion: High-Quality Human Motion Video Generation with Confidence-aware Pose Guidance

Joowhan Song

목차

- **MimicMotion Architecture**
- **Confidence-aware Pose Guidance**
- **Regional Loss Amplification**
- **Progressive Latent Fusion**

MimicMotion Architecture

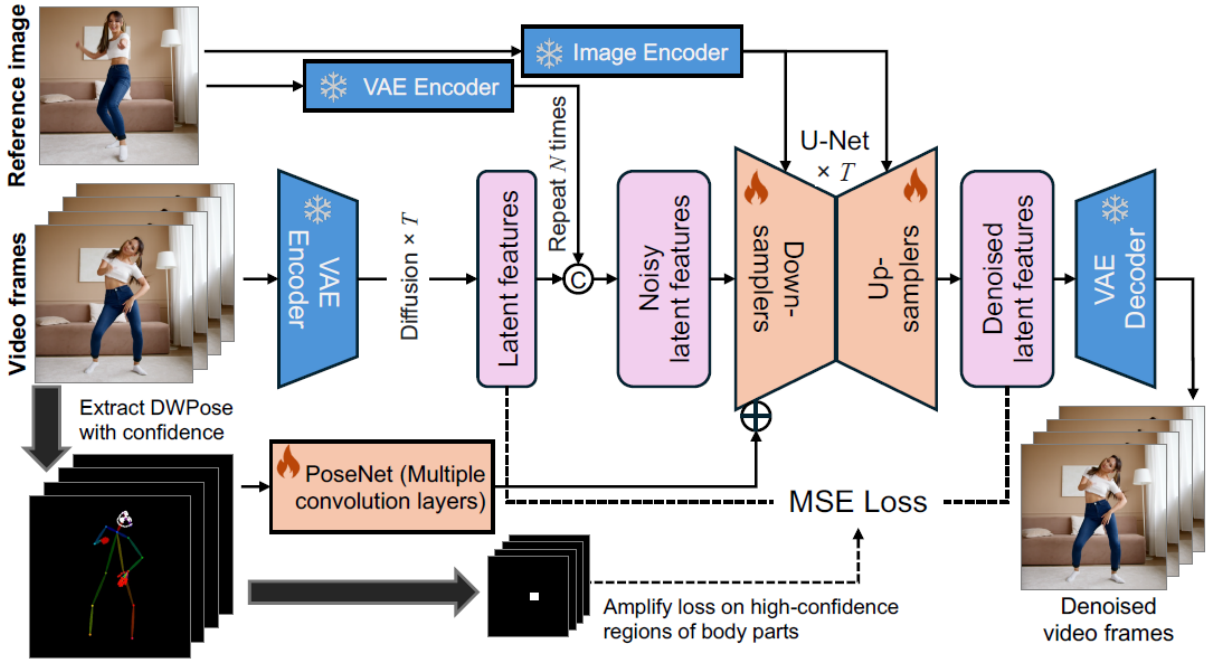
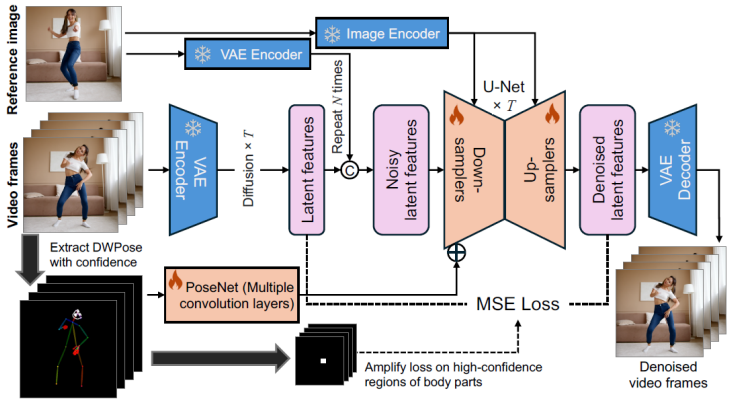


Figure 2: MimicMotion integrates an image-to-video diffusion model with novel confidence-aware pose guidance. The model's trainable components consist of a spatiotemporal U-Net and a PoseNet for introducing pose sequence as the condition. Key features of confidence-aware pose guidance include: 1) The pose sequence condition is accompanied by keypoint confidence scores, enabling the model to adaptively adjust the influence of pose guidance based on the score. 2) The regions with high confidence are given greater weight in the loss function, amplifying their impact in training.

MimicMotion Architecture

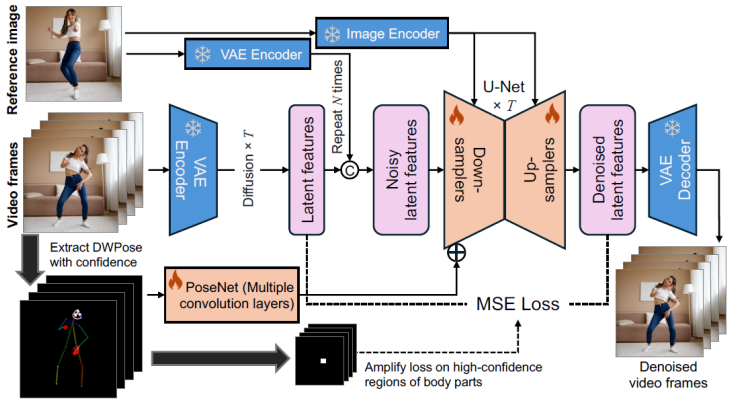


The model structure of MimicMotion is designed to integrate a pre-trained Stable Video Diffusion (SVD) model to leverage its image-to-video generation capabilities.

Figure 2 shows the structure of our model. The core structure of our model is a latent video diffusion model with a U-Net for progressive denoising in latent space. The VAE encoder on input video frames and the corresponding decoder for getting denoised video frames are both adopted from SVD and these parameters are frozen. The VAE encoder is applied independently to each frame of the input video as well as to the conditional reference image, operating on a per-frame basis without considering temporal or cross-frame interactions. Differently, the VAE decoder processes the latent features, which undergo spatiotemporal interaction from U-Net. To ensure the generation of a smooth video, the VAE decoder incorporates temporal layers alongside the spatial layers, mirroring the architecture of the VAE encoder.

In addition to the input video frames, the reference image and the sequence of poses are two other inputs of the model. The reference image is fed into the diffusion model along two separate pathways. One pathway involves feeding the image into each block of the U-Net. Specifically, through a visual encoder like CLIP [37], the image feature is extracted and fed into the cross-attention of every U-Net block for finally controlling the output results. The other pathway targets the input latent features. Similar to the raw video frames, the input reference image is encoded with the same frozen VAE encoder to get its representation in the latent space. The latent feature of the single reference image is then duplicated along the temporal dimension to align with the features of input video frames. The duplicated latent reference images are concatenated with latent video frames along the channel dimension and then fed into U-Net for diffusion altogether.

MimicMotion Architecture



For introducing the guidance of poses, PoseNet, which is implemented with multiple convolution layers, is designed as a trainable module for extracting features of the input sequence of poses. The reason for not using the VAE encoder is that the pixel value distribution of the pose sequence is different from that of common images on which the VAE autoencoder is trained. With PoseNet, the features of poses are extracted and then element-wisely added to the output of the first convolution layer of U-Net. In this way, the influence of the posture guidance can take effect from the very beginning of denoising. We do not add pose guidance to every U-Net block for the following considerations: a) the sequence pose is extracted frame by frame without any temporal interaction so it may confuse the spatio-temporal layers within U-Net when it takes effect on these layers directly; b) excessive involvement of the pose sequence may degrade the performance of the pre-trained image-to-video model.

Confidence-aware Pose Guidance

(a) The occluded areas have lower confidence. (b) The motion blurred areas have lower confidence.

Figure 3: Pose estimation shows uncertainty in some areas with occlusion or dynamic blurring caused by body motions. The confidence score is lower in these areas. We convey this information to Unet.

For this problem, we propose confidence-aware pose guidance, which leverages the confidence scores associated with each keypoint from the pose estimation model. These scores reflect the likelihood of accurate detection, with higher values indicating higher visibility, less occlusion, and motion blur. Instead of applying a fixed confidence threshold to filter the keypoints, as commonly adopted in prior works [38, 4], we utilize brightness on the pose guidance frame to represent the confidence level of pose estimation. Specifically, we integrate the confidence scores of the pose and keypoints into their respective drawing colors. This means that we multiply the color assigned to each keypoint and limb by its confidence score. Consequently, keypoints and corresponding limbs with higher confidence scores will appear more significant on the pose guidance map. This method enables the model to prioritize more reliable pose information in its guidance, thereby enhancing the overall accuracy of pose-guided generation.

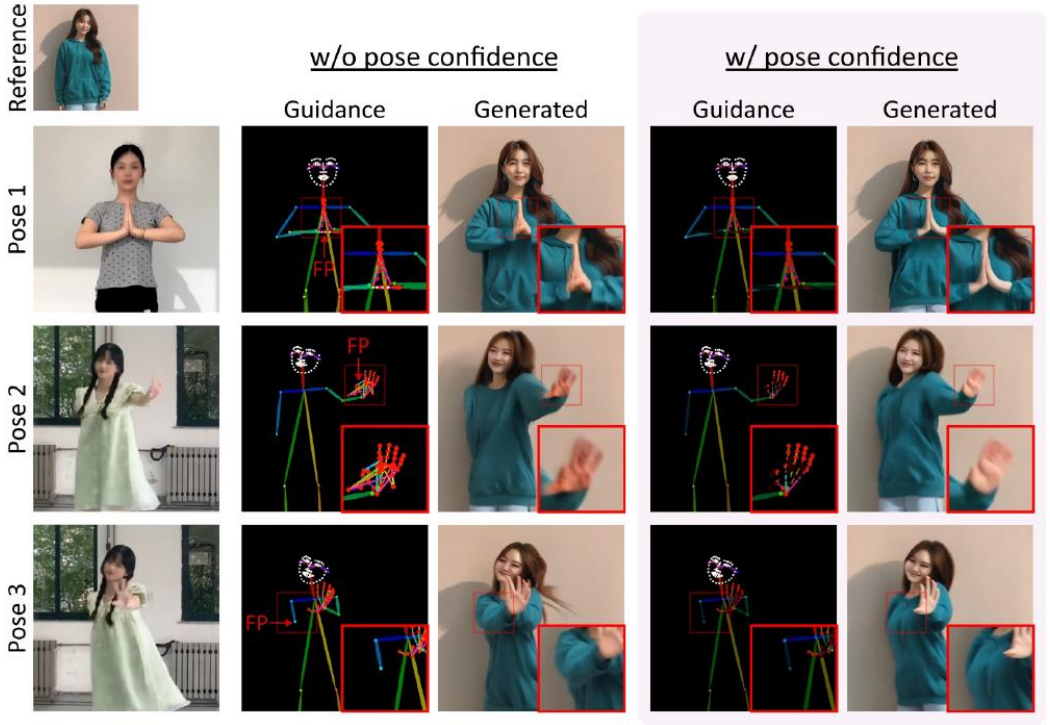


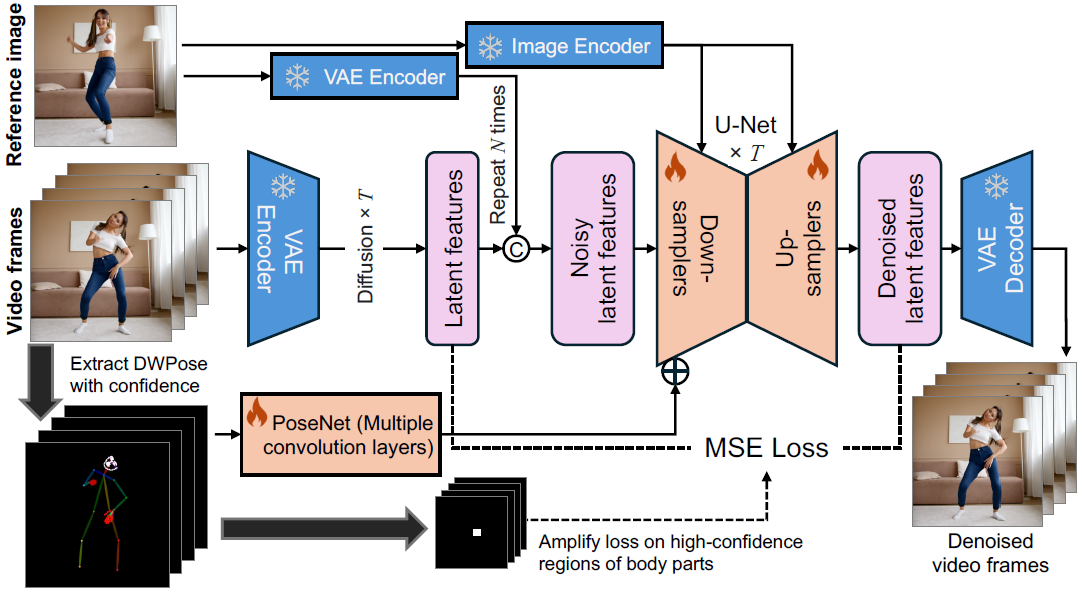
Figure 8: Confidence-aware pose guiding. This design enhances generation robustness to false guiding signals (Pose 1&2) and provides visibility hints to tackle pose ambiguity (Pose 3).

Regional Loss Amplification

Hand region enhancement Moreover, we employ pose estimation and the associated confidence scores to alleviate region-specific artifacts, such as hand distortion, which are prevalent in the diffusion-based image and video generation models. Specifically, we identify reliable regions via thresholding keypoint confidence scores. By setting a threshold, we can distinguish between keypoints that are confidently detected and those that may be ambiguous or incorrect due to factors like occlusion or motion blur. Keypoints with confidence scores above the threshold are considered reliable. We implement a masking strategy that generates masks based on a confidence threshold. We unmask areas where confidence scores surpass a predefined threshold, thereby identifying reliable regions. When computing the loss of the video diffusion model, the loss values corresponding to the unmasked regions are amplified by a certain scale so they can have more effect on the model training than other masked regions.

Specifically, to mitigate hand distortion, we compute masks using a confidence threshold for keypoints in the hand region. Only hands with all keypoint confidence scores exceeding this threshold are considered reliable, as a higher score correlates to higher visual quality. We then construct a bounding box around the hand by padding the boundary of these keypoints, and the enclosed rectangle is designated as unmasked. This region is subsequently assigned a larger weight in the loss calculation during the training of the video diffusion model. This selective unmasking and weighting process biases the model's learning towards hands, especially hands with higher visual quality, effectively reducing distortion and improving the overall realism of the generated content.

Figure 9: Hand region enhancement. Training with hand enhancement consistently reduces hand distortion and improves visual appeal, using the same reference image and pose guidance.



Progressive Latent Fusion

Progressive latent fusion is training-free and is integrated into the denoising process of the latent diffusion model during inference. Figure 4 shows an overview of this process. We omit the VAE for brevity. The denoising process is done in latent space in our method. In general, there are T denoising steps in total and our latent fusion is applied within each step. For a long given pose sequence, we use a pre-defined strategy for splitting the whole sequence into segments, consisting of a fixed number of frames per segment (denoted as N), with a certain number (C) of overlapped frames between every two adjacent segments. For the sake of generation efficiency, it is common to assume that $C \ll N$. During each denoising step, video segments are firstly denoised separately with the trained model, conditioning on the same reference image and the corresponding sub-sequence of poses. Algorithm 1 shows the specific details of progressive latent fusion. As inputs, the reference image is denoted as I_{ref} (c.f. Sec. 3.2) and pose frames corresponding to j-th frame in i-th video segment is denoted as P_i^j. We use $\mathbf{z}_i^j$ to denote the latent feature of j-th frame in i-th video segment. The denoising process starts from the maximum time step T and the latent features are initialized with a normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Within each denoising step at time step t, the reversed diffusion process defined by the trained model (DM) is applied to the latent features of each video segment numbered i separately, with $\mathbf{z}_i$, I_{ref}, P_i and t as inputs. During the latent fusion stage, for every two adjacent video segments, the involved video frames are then fused. To avoid the corruption of temporal smoothness near video segment boundaries after latent fusion, we propose progressive latent fusion. For a video frame involved in latent fusion, its fusion weight is determined by its relative position in the video segment it belongs to. Specifically, if a frame is close to the segment it belongs to, it will be assigned a heavier weight. For implementation, a fusion scale is pre-defined as $\lambda_{\text{fusion}} = 1/(C+1)$ for controlling the level of latent fusion.

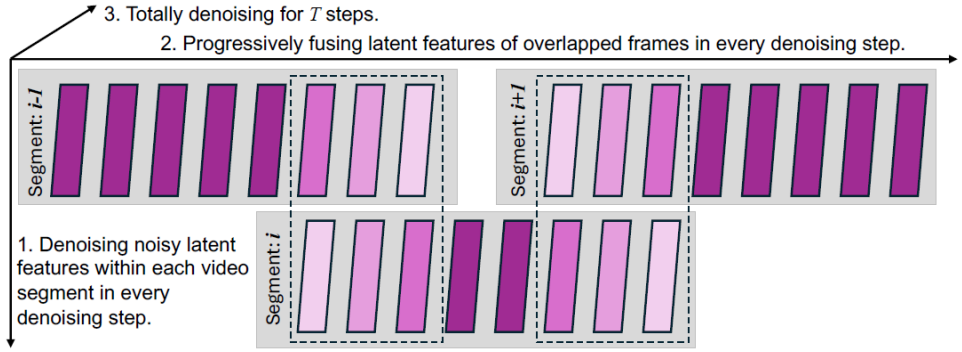


Figure 4: This is an overview of our approach for long video generation. The latent features of video frames are represented with colored boxes. The darkness of color means the weight in latent fusion and darker color means a heavier weight. The dashed boxes represent video frame features involved in latent fusion.

Algorithm 1: Progressive frame-level latent fusion for long video generation.

Input: I_{ref}: Reference image. P_i^j: Pose frame corresponding to j-th frame in i-th video segment; $\mathbf{z}_i^j$: The latent feature of j-th frame in i-th video segment; N: the number of frames in a video segment; C: the number of overlapped frames.

Output: $\mathbf{z}'$: A long sequence of latent features of video frames.

$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$; // Random initialization of video latent features.

$\lambda_{\text{fusion}} \leftarrow 1/(C+1)$; // Set a scale of latent fusion.

for $t = T$ **to** 1 **do** // Denoising from noisy latent features step by step.

  for $i = 1, 2, \ldots$ **do**

    $\mathbf{z}_i \leftarrow \text{DM}(\mathbf{z}_i, I_{\text{ref}}, P_i, t)$ // Separately denoise each segment.

  for $i = 1, 2, \ldots$ **do** // Within each video segment.

    for $j = 1$ **to** N **do** // Start latent fusion for each frame.

      if $i > 1$ ***and*** $j \leq C$ **then** // Latent fusion with the previous segment.

        $\mathbf{z}_i^j \leftarrow j\lambda_{\text{fusion}}\mathbf{z}_i^j + (1 - j\lambda_{\text{fusion}})\mathbf{z}_{i-1}^{N-C+j}$

      else if $j > N - C$ **then** // Latent fusion with the next segment.

        $\mathbf{z}_i^j \leftarrow (N+1-j)\lambda_{\text{fusion}}\mathbf{z}_i^j + (C-N+j)\lambda_{\text{fusion}}\mathbf{z}_{i+1}^{C-N+j}$

return $\mathbf{z}' = \mathit{Merge}(\mathbf{z})$; // Merge multi-segment features following Listing 1.

Progressive Latent Fusion

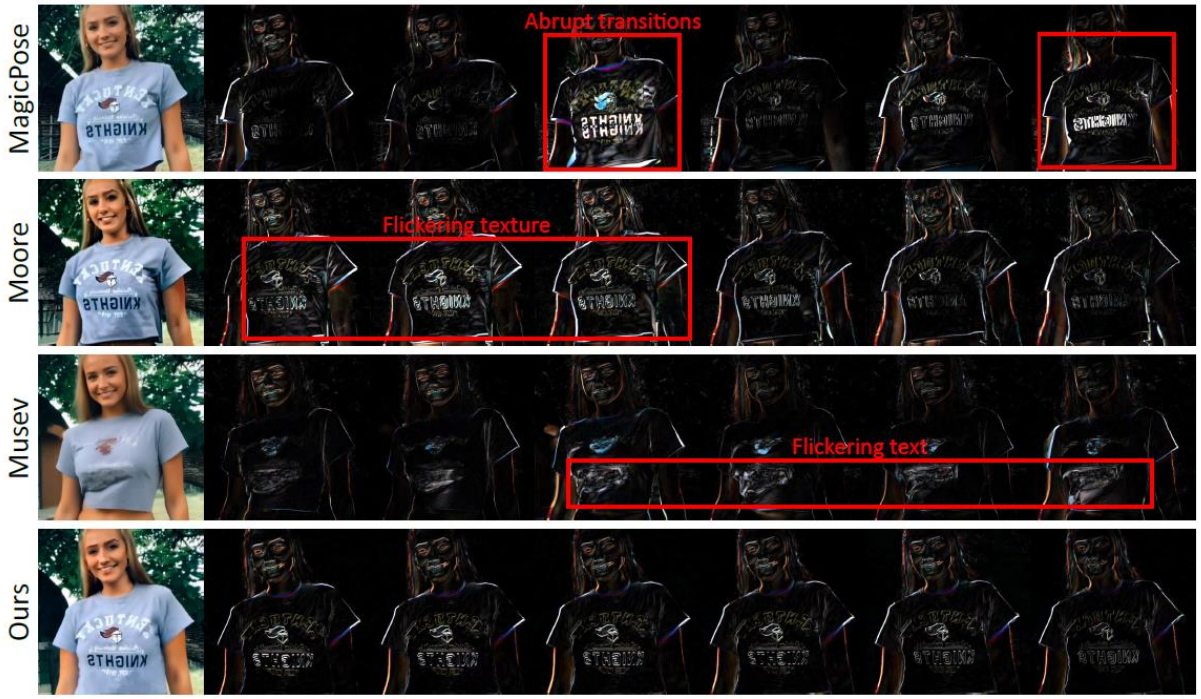


Figure 6: Comparison of temporal smoothness with state-of-the-art methods. We visualize the 106th frame from seq 338 of the TikTok dataset and the pixel-wise difference between consecutive frames. MagicPose exhibits abrupt transitions, while Moore and MuseV show instability in texture and text. In contrast, our method demonstrates stable inter-frame differences and better temporal smoothness.

Figure 10: Progressive latent fusion enables smooth segment (clip) transitions for long video generation. (a) Simple averaging in overlapped regions causes temporal discontinuity, appearing as vertical strips in the Y-T slice. (b) Progressive latent fusion enables smooth transitions.