This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DiffPose: Toward More Reliable 3D Pose Estimation

Jia Gong[1†] Lin Geng Foo[1†] Zhipeng Fan[2§] Qiuhong Ke[3] Hossein Rahmani[4] Jun Liu[1‡]
[1]Singapore University of Technology and Design
[2]New York University [3]Monash University [4]Lancaster University
{jia_gong,lingeng_foo}@mymail.sutd.edu.sg, zf606@nyu.edu, qiuhong.ke@monash.edu,
h.rahmani@lancaster.ac.uk, jun_liu@sutd.edu.sg

Abstract

Monocular 3D human pose estimation is quite challenging due to the inherent ambiguity and occlusion, which often lead to high uncertainty and indeterminacy. On the other hand, diffusion models have recently emerged as an effective tool for generating high-quality images from noise. Inspired by their capability, we explore a novel pose estimation framework (DiffPose) that formulates 3D pose estimation as a reverse diffusion process. We incorporate novel designs into our DiffPose to facilitate the diffusion process for 3D pose estimation: a pose-specific initialization of pose uncertainty distributions, a Gaussian Mixture Model-based forward diffusion process, and a context-conditioned reverse diffusion process. Our proposed DiffPose significantly outperforms existing methods on the widely used pose estimation benchmarks Human3.6M and MPI-INF-3DHP. Project page: https://gongjia0208.github.io/Diffpose/.

1. Introduction

3D human pose estimation, which aims to predict the 3D coordinates of human joints from images or videos, is an important task with a wide range of applications, including augmented reality [5], sign language translation [21] and human-robot interaction [40], attracting a lot of attention in recent years [23, 46, 50, 52]. Generally, the mainstream approach is to conduct 3D pose estimation in two stages: the 2D pose is first obtained with a 2D pose detector, and then 2D-to-3D lifting is performed (where the lifting process is the primary aspect that most recent works [2, 10, 16, 17, 19, 32, 54] focus on). Yet, despite the considerable progress, monocular 3D pose estimation still remains challenging. In particular, it can be difficult to accurately predict 3D pose from monocular data due to many challenges, including the inherent depth ambiguity and the potential occlusion, which often lead to high indeterminacy and uncertainty.

† Equal contribution; § Currently at Meta; ‡ Corresponding author

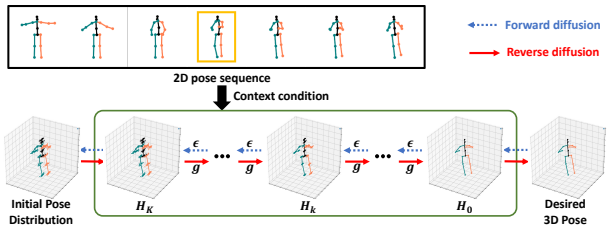


Figure 1. Overview of our DiffPose framework. In the forward process (denoted with blue dotted arrows), we gradually diffuse a "ground truth" 3D pose distribution H_0 with low indeterminacy towards a 3D pose distribution with high uncertainty H_K by adding noise ϵ at every step, which generates intermediate distributions to guide model training. Before the reverse process, we first initialize the indeterminate 3D pose distribution H_K from the input. Then, during the reverse process (denoted with red solid arrows), we use the diffusion model g, conditioned on the context information from 2D pose sequence, to progressively transform H_K into a 3D pose distribution H_0 with low indeterminacy.

On the other hand, diffusion models [12, 38] have recently become popular as an effective way to generate high-quality images [33]. Generally, diffusion models are capable of generating samples that match a specified data distribution (e.g., natural images) from random (indeterminate) noise through multiple steps where the noise is progressively removed [12, 38]. Intuitively, such a paradigm of progressive denoising helps to break down the large gap between distributions (from a highly uncertain one to a determinate one) into smaller intermediate steps [39] and thus successfully helps the model to converge towards smoothly generating samples from the target data distribution.

Inspired by the strong capability of diffusion models to generate realistic samples even from a starting point with high uncertainty (e.g., random noise), here we aim to tackle 3D pose estimation, which also involves handling uncertainty and indeterminacy (of 3D poses), with diffusion models. In this paper, we propose **DiffPose**, a novel framework that represents a new brand of diffusion-based 3D pose estimation approach, which also follows the mainstream two-stage pipeline. In short, DiffPose models the 3D pose esti-

mation procedure as a reverse diffusion process, where we progressively transform a 3D pose distribution with high uncertainty and indeterminacy towards a 3D pose with low uncertainty.

Intuitively, we can consider the determinate ground truth 3D pose as particles in the context of thermodynamics, where particles can be neatly gathered and form a clear pose with low indeterminacy at the start; then eventually these particles stochastically spread over the space, leading to high indeterminacy. This process of particles evolving from low indeterminacy to high indeterminacy is the *forward diffusion process*. The pose estimation task aims to perform precisely the opposite of this process, i.e., *the reverse diffusion process*. We receive an initial 2D pose that is indeterminate and uncertain in 3D space, and we want to shed the indeterminacy to obtain a determinate 3D pose distribution containing high-quality solutions.

Overall, our DiffPose framework consists of two opposite processes: the *forward process* and the *reverse process*, as shown in Fig. 1. In short, the forward process generates supervisory signals of intermediate distributions for training purposes, while the reverse process is a key part of our 3D pose estimation pipeline that is used for both training and testing. Specifically, in the forward process, we gradually diffuse a "ground truth" 3D pose distribution H_0 with low indeterminacy towards a 3D pose distribution with high indeterminacy that resembles the 3D pose's underlying uncertainty distribution H_K. We obtain samples from the intermediate distributions along the way, which are used during training as step-by-step supervisory signals for our diffusion model g. To start the reverse process, we first initialize the indeterminate 3D pose distribution (H_K) according to the underlying uncertainty of the 3D pose. Then, our diffusion model g is used in the reverse process to progressively transform H_K into a 3D pose distribution with low indeterminacy (H_0). The diffusion model g is optimized using the samples from intermediate distributions (generated in the forward process), which guide it to smoothly transform the indeterminate distribution H_K into accurate predictions.

However, there are several challenges in the above forward and reverse process. Firstly, in 3D pose estimation, we start the reverse diffusion process from an estimated 2D pose which has high uncertainty in 3D space, instead of starting from random noise like in existing image generation diffusion models [12, 38]. This is a significant difference, as it means that the underlying uncertainty distribution of each 3D pose can differ. Thus, we cannot design the output of the forward diffusion steps to converge to the same Gaussian noise like in previous image generation diffusion works [12, 38]. Moreover, the uncertainty distribution of 3D poses can be irregular and complicated, making it hard to characterize via a single Gaussian distribution. Lastly, it can be difficult to perform accurate 3D pose estimation with just H_K as input. This is because our aim is not just to generate any realistic 3D pose, but rather to predict accurate 3D poses corresponding to our estimated 2D poses, which often requires more context information to achieve.

To address these challenges, we introduce several novel designs in our DiffPose. Firstly, we initialize the indeterminate 3D pose distribution H_K based on extracted heatmaps, which captures the underlying uncertainty of the desired 3D pose. Secondly, during forward diffusion, to generate the indeterminate 3D pose distributions that eventually (after K steps) resemble H_K, we add noise to the ground truth 3D pose distribution H_0, where the noise is modeled by a Gaussian Mixture Model (GMM) that characterizes the uncertainty distribution H_K. Thirdly, the reverse diffusion process is conditioned on context information from the input video or frame in order to better leverage the spatial-temporal relationship between frames and joints. Then, to effectively use the context information and perform the progressive denoising to obtain accurate 3D poses, we design a GCN-based diffusion model g.

The contributions of this paper are threefold: (i) We propose DiffPose, a novel framework which represents a new brand of method with the diffusion architecture for 3D pose estimation, which can naturally handle the indeterminacy and uncertainty of 3D poses. (ii) We propose various designs to facilitate 3D pose estimation, including the initialization of 3D pose distribution, a GMM-based forward diffusion process and a conditional reverse diffusion process. (iii) DiffPose achieves state-of-the-art performance on two widely used human pose estimation benchmarks.

2. Related Work

3D Human Pose Estimation. Existing monocular 3D pose estimation methods can roughly be categorized into two groups: frame-based methods and video-based ones. *Frame-based methods* predict the 3D pose from a single RGB image. Some works [7–9,30,31,42] use Convolutional Neural Networks (CNNs) to output a human pose from the RGB image, while many works [26, 46, 51, 52] first detect the 2D pose and then use it to regress the 3D pose. On the other hand, *video-based methods* tend to exploit temporal dependencies between frames in the video clip. Most video-based methods [2, 3, 6, 10, 14, 32, 34, 35, 44, 45, 54] extract 2D pose sequences from the input video clip via a 2D pose detector, and focus on distilling the crucial spatial-temporal information from these 2D pose sequences for 3D pose estimation. To encode spatial-temporal information, existing works explore CNN-based frameworks with temporal convolutions [3, 32], GCNs [2, 6], or Transformers [34, 54]. Notably, several works [17, 19, 36] aim to alleviate the uncertainty and indeterminacy in 3D pose estimation by designing models that can generate multiple hypothesis solutions from a single input. Different from all the aforemen-

tioned works, DiffPose is formulated as a *distribution-to-distribution* transformation process, where we train a diffusion model to smoothly denoise from the indeterminate pose distribution to a pose distribution with low indeterminacy. By framing the 3D pose estimation procedure as a reverse diffusion process, DiffPose can naturally handle the indeterminacy and uncertainty for 3D pose estimation.

Denoising Diffusion Probabilistic Models (DDPMs). DDPMs (called diffusion models for short) have emerged as an effective approach to learn a data distribution that is straightforward to sample from. Introduced by Sohl-Dickstein et al. [37] for image generation, DDPMs have been further simplified and accelerated [12, 38], and enhanced [1, 28, 29, 53] in recent years. Previous works have explored applying diffusion models to various generation tasks, including image inpainting [25] and text generation [20]. Here, we explore using diffusion models to tackle 3D pose estimation with our DiffPose framework. Unlike these generation tasks [20, 25] that often start the generation process from random noise, our pose estimation process starts from an estimated 2D pose with uncertainty and indeterminacy in 3D space, where the uncertainty distribution differs for each pose and can also be irregular and difficult to characterize. We also design a GCN-based architecture as our diffusion model g, and condition it on spatial-temporal context information to aid the reverse diffusion process and obtain accurate 3D poses.

3. Background on Diffusion Models

Diffusion models [12, 38] are a class of probabilistic generative models that learn to transform noise $h_K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ to a sample h_0 by recurrently denoising h_K, i.e., $(h_K \rightarrow h_{K-1} \rightarrow ... \rightarrow h_0)$. This denoising process is called *reverse diffusion*. Conversely, the process $(h_0 \rightarrow h_1 \rightarrow ... \rightarrow h_K)$ is called *forward diffusion*.

To allow the diffusion model to learn the reverse diffusion process, a set of intermediate noisy samples $\{h_k\}_{k=1}^{K-1}$ are needed to bridge the source sample h_0 and the Gaussian noise h_K. Specifically, *forward diffusion* is conducted to generate these samples, where the posterior distribution $q(h_{1:K}|h_0)$ from h_0 to h_K is formulated as:

$$q(h_{1:K}|h_0) := \prod_{k=1}^{K} q(h_k|h_{k-1}) \tag{1}$$

$$q(h_k|h_{k-1}) := \mathcal{N}_{pdf}\big(h_k|\sqrt{\frac{\alpha_k}{\alpha_{k-1}}}h_{k-1}, (1-\frac{\alpha_k}{\alpha_{k-1}})\mathbf{I}\big), \tag{2}$$

where $\mathcal{N}_{pdf}(h_k|\cdot)$ refers to the likelihood of sampling h_k conditioned on the given parameters, and $\alpha_{1:K} \in (0,1]^K$ is a fixed decreasing sequence that controls the noise scaling at each diffusion step. Using the known statistical results for the combination of Gaussian distributions, the posterior for the diffusion process to step k can be formulated as:

$$\begin{aligned} q(h_k|h_0) :=& \int q(h_{1:k}|h_0)\mathrm{d}h_{1:k-1} \\ =& \mathcal{N}_{pdf}(h_k|\sqrt{\alpha_k}h_0, (1-\alpha_k)\mathbf{I}). \end{aligned} \tag{3}$$

Thus, h_k can be expressed as a linear combination of the source sample h_0 and a noise variable ϵ, where each element of ϵ is sampled from $\mathcal{N}(0,1)$, as follows:

$$h_k = \sqrt{\alpha_k}h_0 + \sqrt{(1-\alpha_k)}\epsilon. \tag{4}$$

Hence, when a long decreasing sequence $\alpha_{1:K}$ is set such that $\alpha_K \approx 0$, the distribution of h_K will converge to a standard Gaussian, i.e., $h_K \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This indicates that the source signal h_0 will eventually be corrupted into Gaussian noise, which conforms to the non-equilibrium thermodynamics phenomenon of the diffusion process [37].

Using the sample h_0 and noisy samples $\{h_k\}_{k=1}^{K}$ generated by forward diffusion, the diffusion model g (which is often a deep network parameterized by θ) is optimized to approximate the reverse diffusion process. Specifically, although the exact formulations may differ [12, 37, 38], each reverse diffusion step can be expressed as a function f that takes in h_k and diffusion model g as input to generate an output h_{k-1} as follows:

$$h_{k-1} = f(h_k, g). \tag{5}$$

Finally, during testing, a Gaussian noise h_K can be easily sampled, and the reverse diffusion step introduced in Eq. 5 can be recurrently performed to generate a high-quality sample h_0 using the trained diffusion model g.

4. Proposed Method: DiffPose

Given an RGB image frame I_t or a video clip $V_t = \{I_\tau\}_{\tau=(t-T)}^{(t+T)}$, the goal of 3D human pose estimation is to predict the 3D coordinates of all the J keypoints of the human body in I_t. In this paper, inspired by diffusion-based generative models that can recurrently shed the indeterminacy in an initial distribution (e.g., Gaussian distribution) to reconstruct a high-quality determinate sample, we frame the 3D pose estimation task as constructing a determinate 3D pose distribution (H_0) from the highly indeterminate pose distribution (H_K) via diffusion models, which can handle the uncertainty and indeterminacy of 3D poses.

As shown in Fig. 2, we conduct pose estimation in two stages: (i) Initializing the indeterminate 3D pose distribution H_K based on extracted heatmaps, which capture the underlying uncertainty of the input 2D pose in 3D space; (ii) Performing the *reverse diffusion process*, where we use a diffusion model g to progressively denoise the initial distribution H_K to a desired high-quality determinate distribution H_0, and then we can sample $h_0 \in \mathbb{R}^{3\times J}$ from the pose distribution H_0 to synthesize the final 3D pose h_s.

In Sec. 4.1, we describe how to initialize the 3D distribution H_K from an input 2D pose that effectively captures the uncertainty in the 3D space. Then, we explain our forward

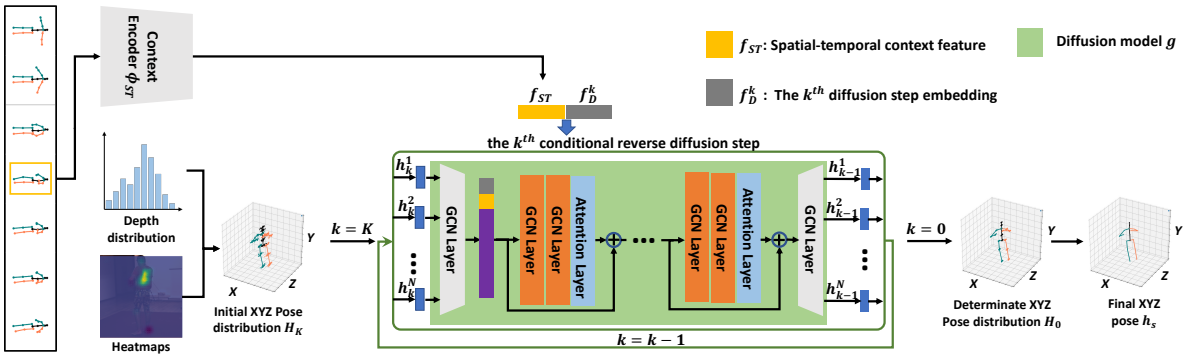


Figure 2. Illustration of our DiffPose framework during inference. First, we use the Context Encoder ϕ_{ST} to extract the spatial-temporal context feature f_{ST} from the given 2D pose sequence. We also generate diffusion step embedding f_D^k for each k^{th} diffusion step. Then, we initialize the indeterminate pose distribution H_K using heatmaps derived from an off-the-shelf 2D pose detector and depth distributions that can either be computed from the training set or predicted by the Context Encoder ϕ_{ST}. Next, we sample N noisy poses $\{h_K^i\}_{i=1}^N$ from H_K, which are required for performing distribution-to-distribution mapping. We feed these N poses into the diffusion model K times, where diffusion model g is also conditioned on f_{ST} and f_D^k at each step, to obtain $\{h_0^i\}_{i=1}^N$ which represents the high-quality determinate distribution H_0. Lastly, we use the mean of $\{h_0^i\}_{i=1}^N$ as our final 3D pose h_s.

diffusion process in Sec. 4.2 and the reverse diffusion process in Sec. 4.3. After that, we present the detailed training and testing process in Sec. 4.4. Finally, the architecture of our diffusion network is detailed in Sec. 4.5.

4.1. Initializing 3D Pose Distribution H_K

In previous diffusion models [11, 12, 38], the reverse diffusion process often starts from random noise, which is progressively denoised to generate a high-quality output. However, in 3D pose estimation, our input here is instead an estimated 2D pose that has its own uncertainty characteristics in 3D space. To aid our diffusion model in handling the uncertainty and indeterminacy of each input 2D pose in 3D space, we would like to initialize a corresponding 3D pose distribution H_K that captures the uncertainty of the 3D pose. Thus, the reverse diffusion process can start from the distribution H_K with sample-specific knowledge (in contrast to Gaussian noise with no prior information), which leads to better performance. Below, we describe how we construct the x, y, and z uncertainty distribution for each joint of an input pose.

Initializing (x, y, z) distribution. Intuitively, the x and y uncertainty distribution contains information regarding the likely regions in the image where the joints are located, and can roughly be seen as the outcome of "outwards" diffusion from the ground-truth positions. It can be difficult to capture such 2D pose uncertainty distributions, which are often complicated and also vary for different joints of the given pose. To address this, we take advantage of the available prior information to model the uncertainty of the 2D pose. Notably, the 2D pose is often estimated from the image with an off-the-shelf 2D pose detector (e.g., CPN [4]), which first extracts heatmaps depicting the likely area on the image where each joint is located, before making predictions of 2D joint locations based on the extracted heatmaps. Therefore, these heatmaps naturally reveal the uncertainty of the 2D pose predictions. Hence, for the input 2D pose, we use the corresponding heatmaps from the off-the-shelf 2D pose detector as the x and y distribution.

However, we are unable to obtain the z distribution in the same way, as it is not known by the 2D pose detector. Instead, one way we can compute the z distribution is by calculating the occurrence frequencies of z values in the training data, where we obtain a histogram for every joint. We also explore another approach, where the uncertain z distribution is initialized using the Context Encoder (which is introduced in Sec. 4.3), which we empirically observe to lead to faster convergence.

4.2. Forward Pose Diffusion

After initializing the indeterminate distribution H_K, the next step in our 3D pose estimation pipeline is to progressively reduce the uncertainty $(H_K \rightarrow H_{K-1} \rightarrow ... \rightarrow H_0)$ using the diffusion model g through the reverse diffusion process. However, to attain the progressive denoising capability of the diffusion model g, we require "ground truth" intermediate distributions as supervisory signals to train g. Here, we obtain samples from these intermediate distributions via the *forward diffusion process*, where we take a ground truth 3D pose distribution H_0 and gradually add noise to it, as shown in Fig. 1. Specifically, given a desired determinate pose distribution H_0, we define the forward diffusion process as $(H_0 \rightarrow H_1 \rightarrow ... \rightarrow H_K)$, where K is the maximum number of diffusion steps. In this process, we aim to progressively increase the indeterminacy of H_0 towards the underlying pose uncertainty distribution H_K as obtained in Sec. 4.1, such that we can obtain samples from intermediate distributions that correspond to $H_1, ..., H_K$, which will allow us to optimize the diffusion model g to smoothly perform the step-by-step denoising.

DiffPose Forward Diffusion. For DiffPose, we do not want to diffuse our 3D pose towards a standard Gaussian noise. This is because our indeterminate distribution H_K is not random noise, but is instead a (x, y, z) distribution according to the 3D pose uncertainty, and has more complex characteristics. This has several implications. For example, the region of uncertainty for each joint and each coordinate of the initial pose distribution H_K can be different. Secondly, the mean locations of all joints should not be treated as equal to the origin (i.e., 0 along all dimensions), due to the constraints of the body structure. Due to these reasons, the basic generative diffusion process (in Sec. 3) cannot appropriately model the uncertainty of the initialized pose distribution H_K (as described in Sec. 4.1) for our 3D pose estimation task, which motivates us to design a new forward diffusion process.

Designing such a forward diffusion process can be challenging, because the uncertainty distribution H_K, which is based on heatmaps, often has irregular and complex shapes, and it is not straightforward to express H_K mathematically. To overcome this, we propose to use a Gaussian Mixture Model (GMM) to model the uncertainty distribution H_K for 3D pose estimation, as it can characterize intractable and complex distributions [18, 28], and is very effective to represent heatmap-based distributions [43]. Then, based on the fitted GMM model, we perform a corresponding GMM-based forward diffusion process. Specifically, we set the number of Gaussian components in the GMM at M, and use the Expectation-Maximization (EM) algorithm to optimize the GMM parameters ϕ_{GMM} to fit the target distribution H_K as follows:

$$\max_{\phi_{GMM}} \prod_{i=1}^{N_{GMM}} \sum_{m=1}^{M} \pi_m \mathcal{N}_{pdf}(h_K^i | \mu_m, \Sigma_m), \quad (6)$$

where $h_K^1, ..., h_K^{N_{GMM}}$ are N_{GMM} poses sampled from the pose distribution H_K, and $\phi_{GMM} = \{\mu_1, \Sigma_1, \pi_1, ..., \mu_M, \Sigma_M, \pi_M\}$ refers to the GMM parameters. Here, $\mu_m \in \mathbb{R}^{3J}$ and $\Sigma_m \in \mathbb{R}^{3J \times 3J}$ are the mean values and covariance matrix of the m^{th} Gaussian component. $\pi_m \in [0, 1]$ is the probability that any sample h_K^i is drawn from the m^{th} mixture component ($\sum_{m=1}^{M} \pi_m = 1$).

Next, we want to run the forward diffusion process on the ground truth pose distribution H_0 such that after K steps, the generated noisy distribution becomes equivalent to the fitted GMM distribution ϕ_{GMM}, which we henceforth denote as $\hat{H}_K$ because it is a GMM-based representation of H_K. To achieve this, we can modify Eq. 4 as follows:

$$\hat{h}_k = \mu^G + \sqrt{\alpha_k}(h_0 - \mu^G) + \sqrt{(1 - \alpha_k)} \cdot \epsilon^G. \quad (7)$$

where $\hat{h}_k$ is a generated sample from the generated distribution $\hat{H}_k$ (which does not have a superscript since it describes how to generate a single sample), $\mu^G = \sum_{m=1}^{M} \mathbf{1}_m \mu_m$, $\epsilon^G \sim \mathcal{N}(0, \sum_{m=1}^{M}(\mathbf{1}_m \Sigma_m))$, and $\mathbf{1}_m \in \{0, 1\}$ is a binary indicator for the m^{th} component such that $\sum_{m=1}^{M} \mathbf{1}_m = 1$ and $Prob(\mathbf{1}_m = 1) = \pi_m$. In other words, we first select a component $\hat{m}$ via sampling according to the respective probabilities π_m, and set only $\mathbf{1}_{\hat{m}}$ to 1. Then, we sample the Gaussian noise from that component $\hat{m}$ using $\mu_{\hat{m}}$ and $\Sigma_{\hat{m}}$. Notably, as $\alpha_K \approx 0$, $\hat{h}_K$ is drawn from the fitted GMM model, i.e., $\hat{h}_K = \mu^G + \epsilon^G \sim \mathcal{N}(\sum_{m=1}^{M}(\mathbf{1}_m \mu_m), \sum_{m=1}^{M}(\mathbf{1}_m \Sigma_m))$. Thus, this allows us to generate samples from $\{\hat{H}_1, ..., \hat{H}_K\}$ as supervisory signals. More details can be found in Supplementary.

4.3. Reverse Diffusion for 3D Pose Estimation

As shown in Fig. 1, the reverse diffusion process aims to recover a determinate 3D pose distribution H_0 from the indeterminate pose distribution H_K, where H_K has been discussed in Sec. 4.1. In the previous subsection, we represent H_K via a GMM model to generate intermediate distributions $\{\hat{H}_1, ..., \hat{H}_K\}$. Here, we use these distributions to optimize our diffusion model g (parameterized by θ) to learn the reverse diffusion process ($\hat{H}_K \rightarrow ... \rightarrow \hat{H}_1 \rightarrow H_0$), and progressively shed the indeterminacy from $\hat{H}_K$ to reconstruct the determinate source distribution H_0. The architecture of the diffusion model g is described in Sec. 4.5.

Context Encoder ϕ_{ST}. However, it is difficult to directly perform the reverse diffusion process using only $\hat{H}_K$ as the input of the diffusion model g. This is because g will not observe much context information from the input videos/images, leading to difficulties for g to generate accurate poses from the indeterminate distribution H_K. Therefore, we propose to utilize the available context information from the input to guide g to achieve more accurate predictions. The context information can constrain the model's denoising based on the observed inputs, and guide the model to produce more accurate predictions.

Specifically, to guide the diffusion model g, we leverage the *spatial-temporal context*. The context information can be extracted from the 2D pose sequence derived from V_t (or just a single 2D pose derived from I_t if V_t is not available). This context information aids the reverse diffusion process, providing additional information to the diffusion model g that helps to reduce uncertainty and generate more accurate 3D poses. To achieve that, we introduce the Context Encoder ϕ_{ST} to extract spatial-temporal information f_{ST} from the 2D pose sequence, and condition the reverse diffusion process on f_{ST} (as shown in Fig. 2).

Reverse Diffusion Process. Overall, our reverse diffusion process aims to recover a determinate pose distribution H_0 from the indeterminate pose distribution $\hat{H}_K$ (during training) or H_K (during testing). Here, we describe the reverse diffusion process during training and use $\hat{H}_K$ notation. We first use Context Encoder ϕ_{ST} to extract f_{ST} from

the 2D pose sequence. Moreover, to allow the diffusion model to learn to denoise samples appropriately at each diffusion step, we also generate the unique step embedding f_D^k to represent the k^{th} diffusion step via the sinusoidal function. Then, for a noisy pose $\hat{h}_k$ sampled from $\hat{H}_k$, we use diffusion model g, conditioned on the diffusion step k and the spatial-temporal context feature f_{ST}, to progressively reconstruct $\hat{h}_{k-1}$ from $\hat{h}_k$ as follows:

$$\hat{h}_{k-1} = g_\theta(\hat{h}_k, f_{ST}, f_D^k), \;\; k \in \{1, ..., K\}. \tag{8}$$

4.4. Overall Training and Testing Process

Overall, for each sample during training, we (i) initialize H_K; (ii) use H_0 and H_K to generate supervisory signals $\{\hat{H}_1, ..., \hat{H}_K\}$ via the forward process; (iii) run K steps of the reverse process starting from $\hat{H}_K$ and optimize with our generated signals. During testing, we (i) initialize H_K; (ii) run K steps of the reverse process starting from H_K to obtain final prediction h_s. More details are described below.

Training. First, from the input sequence V_t (or frame I_t), we extract the 2D heatmaps together with the estimated 2D pose via an off-the-shelf 2D pose detector [4]. Then, we compute the z distribution, either from the training set or predicted by the Context Encoder ϕ_{ST}. After that, we initialize H_K based on the 3D distribution for each joint and use the EM algorithm to get the best-fit GMM parameters $\phi_{GMM} = \{\mu_1, \Sigma_1, \pi_1, ..., \mu_M, \Sigma_M, \pi_M\}$ for H_K. Based on ϕ_{GMM}, we use the ground truth 3D pose h_0 to *directly* generate N sets of $\hat{h}_1, ..., \hat{h}_K$ via Eq. 7, i.e., $\big\{\{\hat{h}_1^i, ..., \hat{h}_K^i\}\big\}_{i=1}^N$. Specifically, we first sample a component $\hat{m}^i$ for each i^{th} set according to probabilities $\{\pi_m\}_{m=1}^M$, and use the $\hat{m}^i$-th Gaussian component to directly add noise for the i^{th} set $\{\hat{h}_1^i, ..., \hat{h}_K^i\}$. Next, we extract the spatial-temporal context f_{ST} using the Context Encoder ϕ_{ST}. Then, we want to optimize the model parameters θ to reconstruct $\hat{h}_{k-1}^i$ from $\hat{h}_k^i$ in a step-wise manner. Following previous works on diffusion models [12, 38], we formulate our loss $\mathcal{L}$ as follows (where $\hat{h}_0^i = h_0$ for all i):

$$\mathcal{L} = \sum_{i=1}^{N} \sum_{k=1}^{K} \Big\| g_\theta(\hat{h}_k^i, f_{ST}, f_D^k) - \hat{h}_{k-1}^i \Big\|_2^2. \tag{9}$$

Testing. Similar to the start of the training procedure, during testing we first initialize H_K and also extract f_{ST}. Then, we perform the reverse diffusion process, where we sample N poses from H_K $(h_K^1, h_K^2, ..., h_K^N)$ and recurrently feed them into diffusion model g for K times, to obtain N high-quality 3D poses $(h_0^1, h_0^2, ..., h_0^N)$. We need N noisy poses here, because we are mapping from a distribution to another distribution. Then, to obtain the final high-quality and reliable pose h_s, we calculate the mean of the N denoised samples $\{h_0^1, \ldots, h_0^N\}$.

4.5. DiffPose Architecture

Our framework consists of two sub-networks: a diffusion network (g) that performs the steps in the reverse process and a Context Encoder (ϕ_{ST}) that extracts the context feature from the 2D pose sequence (or frame).

Main Diffusion Model g. We adopt a lightweight GCN-based architecture for g to perform 3D pose estimation via diffusion, which is modified from [52]. The graph convolution layer treats the human skeleton as a graph (with joints as the nodes), and effectively encodes topological information between joints for 3D human pose estimation. Moreover, we interlace GCN layers with Self-Attention layers, which can encode global relationships between non-adjacent joints and allow for better structural understanding of the 3D human pose as a whole. As shown in Fig. 2, our diffusion model g mainly consists of 3 stacked GCN-Attention Blocks with residual connections, where each GCN-Attention Block comprises of two standard GCN layers and a Self-Attention layer. A GCN layer is added at the front and back of these stacked GCN-Attention Blocks to control the embedding size of GCN-Attention Blocks.

Specifically, the starting GCN layer maps the input $h_k \in \mathbb{R}^{J\times 3}$ to a latent embedding $E \in \mathbb{R}^{J\times 128}$. On the other hand, we extract spatial-temporal context information $f_{ST} \in \mathbb{R}^{J\times 128}$. In order to provide information to the model regarding the current step number k, we also generate a diffusion step embedding $f_D^k \in \mathbb{R}^{J\times 256}$ using the sinusoidal function. Then, we combine these embeddings to form features $v_1 \in \mathbb{R}^{J\times 256}$, where E and f_{ST} are first concatenated along the second dimension, before adding f_D^k to the result. Features v_1 are then fed into the stack of 3 GCN-Attention Blocks, which all have the exact same structure. The output features from the last GCN-Attention Block are fed into the final GCN layer to be mapped into an output pose $h_{k-1} \in \mathbb{R}^{J\times 3}$. Then, we feed h_{k-1} back to g as input again to perform another reverse step. At the final K-th step, we obtain an output pose $h_0 \in \mathbb{R}^{J\times 3}$.

Context Encoder ϕ_{ST}. In this paper, we leverage a transformer-based network [49] to capture the spatial-temporal context information in the 2D pose sequence V_t. Note that, if we do not have the video, we only input a single frame I_t, and utilize [52] instead.

5. Experiments

We evaluate our method on two widely used datasets for 3D human pose estimation: Human3.6M [15] and MPI-INF-3DHP [27]. Specifically, we conduct experiments to evaluate the performance of our method in two scenarios: video-based and frame-based 3D pose estimation.

Human3.6M [15] is the largest benchmark for 3D human pose estimation, consisting of 3.6 million images captured from four cameras, where 15 daily activities are per-

formed by 11 subjects. For video-based 3D pose estimation, we follow previous works [3, 24, 32] to train on five subjects (S1, S5, S6, S7, S8) and test on two subjects (S9 and S11). For frame-based 3D pose estimation, we follow [46, 51, 52] to train on (S1, S5, S6, S7, S8) subjects and test on (S9, S11) subjects. We report the mean per joint position error (MPJPE) and Procrustes MPJPE (P-MPJPE). The former computes the Euclidean distance between the predicted joint positions and the ground truth positions. The latter is the MPJPE after the predicted results are aligned to the ground truth via a rigid transformation. Due to page limitations, we move P-MPJPE results to Supplementary.

MPI-INF-3DHP [27] is a large 3D pose dataset captured in both indoor and outdoor environments, with 1.3 million frames. Following [3, 22, 27, 54], we train DiffPose using all activities from 8 camera views in the training set and evaluate on valid frames in the test set. Here, we report metrics of MPJPE, Percentage of Correct Keypoints (PCK) with the threshold of 150 mm, and Area Under Curve (AUC) for a range of PCK thresholds to compare our performance with other methods on the video-based setting.

Implementation Details. We set the number of pose samples N to 5 and number of reverse diffusion steps K to 50. We fit $\hat{H}_K$ via a GMM model with 5 kernels ($M = 5$) for forward diffusion, and accelerate our diffusion inference procedure for all experiments via an acceleration technique DDIM [38], where only five steps are required to complete the reverse diffusion process. For video pose estimation, we set the Context Encoder ϕ_{ST} to follow [49], and for frame-based pose estimation, we set ϕ_{ST} to follow [52]. The Context Encoder ϕ_{ST} is pre-trained on the training set to predict (x, y, z), then frozen during diffusion model training; we use it to produce features f_{ST} and also to initialize the z distribution. For video-based pose estimation, we follow [2, 32] to use detected 2D pose (using CPN [4]) and ground truth 2D pose on Human3.6M, and use ground truth 2D pose on MPI-INF-3DHP. For frame-based pose estimation, we follow [51, 52] to use the 2D pose detected by [4] and ground truth 2D pose to conduct experiments on Human3.6M. More details are in Supplementary.

5.1. Comparison with State-of-the-art Methods

Video-based Results on Human3.6M. We follow [32, 48, 49] to use 243 frames for 3D pose estimation and compare our method against existing works on Human3.6M in Tab. 1. As shown in the top of Tab. 1, our method achieves the best MPJPE results using the detected 2D pose, and significantly outperforms the SOTA method [49] by around 4 mm. This shows that DiffPose can effectively improve monocular 3D pose estimation. Moreover, we also conduct experiments using the ground truth 2D pose as input, and report our results at the bottom of Tab. 1. Our DiffPose again outperforms all previous methods by a large margin.

Table 1. Video-based results on Human3.6M in millimeters under MPJPE. Top table shows the results on detected 2D poses. Bottom table shows the results on ground truth 2D poses.

MPJPE(CPN)	Dir	Disc	Eat	Greet	Phone	Photo	Pose	Pur	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Pavllo [32]	45.2	46.7	43.3	45.6	48.1	55.1	44.6	44.3	57.3	65.8	47.1	44.0	49.0	32.8	33.9	46.8
Liu [24]	41.8	44.8	41.1	44.9	47.4	54.1	43.4	42.2	56.2	63.6	45.3	43.5	45.3	31.3	32.2	45.1
Zeng [48]	46.6	47.1	43.9	41.6	45.8	49.6	46.5	40.0	53.4	61.1	46.1	42.6	43.1	31.5	32.6	44.8
Zheng [54]	41.5	44.8	39.8	42.5	46.5	51.6	42.1	42.0	53.3	60.7	45.5	43.3	46.1	31.8	32.2	44.3
Li [19]	39.2	43.1	40.1	40.9	44.9	51.2	40.6	41.3	53.5	60.3	43.7	41.1	43.8	29.8	30.6	43.0
Shan [34]	38.4	42.1	39.8	40.2	45.2	48.9	40.4	38.3	53.8	57.3	43.9	41.6	42.2	29.3	29.3	42.1
Zhang [49]	37.6	40.9	37.3	39.7	42.3	49.9	40.1	39.8	51.7	55.0	42.1	39.8	41.0	27.9	27.9	40.9
Ours	**33.2**	**36.6**	**33.0**	**35.6**	**37.6**	**45.1**	**35.7**	**35.5**	**46.4**	**49.9**	**37.3**	**35.6**	**36.5**	**24.4**	**24.1**	**36.9**
MPJPE(GT)	Dir	Disc	Eat	Greet	Phone	Photo	Pose	Pur	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Pavllo [32]	35.2	40.2	32.7	35.7	38.2	45.5	40.6	36.1	48.8	47.3	37.8	39.7	38.7	27.8	29.5	37.8
Liu [24]	34.5	37.1	33.6	34.2	32.9	37.1	39.6	35.8	40.7	41.4	33.0	33.8	33.0	26.6	26.9	34.7
Zeng [48]	34.8	32.1	28.5	30.7	31.4	36.9	35.6	30.5	38.9	40.5	32.5	31.0	29.9	22.5	24.5	32.0
Zheng [54]	30.0	33.6	29.9	31.0	30.2	33.3	34.8	31.4	37.8	38.6	31.7	31.5	29.0	23.3	23.1	31.3
Li [19]	27.7	32.1	29.1	28.9	30.0	33.9	33.0	31.2	37.0	39.3	30.0	31.0	29.4	22.2	23.0	30.5
Shan [34]	28.5	30.1	28.6	27.9	29.8	33.2	31.3	27.8	36.0	37.4	29.7	29.5	28.1	21.0	21.0	29.3
Zhang [49]	21.6	22.0	20.4	21.0	20.8	24.3	24.7	21.9	26.9	24.9	21.2	21.5	20.8	14.7	15.7	21.6
Ours	**18.6**	**19.3**	**18.0**	**18.4**	**18.3**	**21.5**	**21.5**	**19.1**	**23.6**	**22.3**	**18.6**	**18.8**	**18.3**	**12.8**	**13.9**	**18.9**

Table 3. Frame-based results on Human3.6M in millimeters under MPJPE. Top table shows the results on detected 2D poses. Bottom table shows the results on ground truth 2D poses.

MPJPE(CPN)	Dir	Disc	Eat	Greet	Phone	Photo	Pose	Pur	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Pavlakos [31]	67.4	71.9	66.7	69.1	72.0	77.0	65.0	68.3	83.7	96.5	71.7	65.8	74.9	59.1	63.2	71.9
Martinez [26]	51.8	56.2	58.1	59.0	69.5	78.4	55.2	58.1	74.0	94.6	62.3	59.1	65.1	49.5	52.4	62.9
Sun [41]	52.8	54.8	54.2	54.3	61.8	**53.1**	53.6	71.7	86.7	61.5	67.2	53.4	47.1	61.6	53.4	59.1
Yang [47]	51.5	58.9	50.4	57.0	62.1	65.4	49.8	52.7	69.2	85.2	57.4	58.4	**43.6**	60.1	47.7	58.6
Hossain [13]	48.4	50.7	57.2	55.2	63.1	72.6	53.0	51.7	66.1	80.9	59.0	57.3	62.4	46.6	49.6	58.3
Zhao [51]	48.2	60.8	51.8	64.0	64.6	53.6	51.1	67.4	88.7	**57.7**	73.2	65.6	48.9	64.8	51.9	60.8
Liu [23]	46.3	52.2	47.3	50.7	55.5	67.1	49.2	46.0	60.4	71.1	51.5	50.1	54.5	40.3	43.7	52.4
Xu [46]	45.2	49.9	47.5	50.9	54.9	66.1	48.5	46.3	59.7	71.5	51.4	48.6	53.9	39.9	44.1	51.9
Zhao [52]	45.2	50.8	48.0	50.0	54.9	65.0	48.2	47.1	60.2	70.0	51.6	48.7	54.1	39.7	43.1	51.8
Ours	**42.8**	**49.1**	**45.2**	**48.7**	**52.1**	63.5	**46.3**	**45.2**	**58.6**	66.3	**50.4**	**47.6**	52.0	**37.6**	**40.2**	**49.7**
MPJPE(GT)	Dir	Disc	Eat	Greet	Phone	Photo	Pose	Pur	Sit	SitD	Smoke	Wait	WalkD	Walk	WalkT	Avg
Martinez [26]	37.7	44.4	40.3	42.1	48.2	54.9	44.4	42.1	54.6	58.0	45.1	46.4	47.6	36.4	40.4	45.5
Hossain [13]	35.2	40.8	37.2	37.4	43.2	44.0	38.9	35.6	42.3	44.6	39.7	39.7	40.2	32.8	35.5	39.2
Zhao [51]	37.8	49.4	37.6	40.9	45.1	41.4	40.1	48.3	50.1	42.2	53.5	44.3	40.5	47.3	39.0	43.8
Liu [23]	36.8	40.3	33.0	36.3	37.5	45.0	39.7	34.9	40.3	47.7	37.4	38.5	38.6	29.6	32.0	37.8
Xu [46]	35.8	38.1	31.0	35.3	35.8	43.2	37.3	31.7	38.4	45.5	35.4	36.7	36.8	27.9	30.7	35.8
Zhao [52]	32.0	38.0	30.4	34.4	34.7	43.3	35.2	31.4	38.0	46.2	34.2	35.7	36.1	27.4	30.6	35.2
Ours	**28.8**	**32.7**	**27.8**	**30.9**	**32.8**	**38.9**	**32.2**	**28.3**	**33.3**	**41.0**	**31.0**	**32.1**	**31.5**	**25.9**	**27.5**	**31.6**

Video-based Results on MPI-INF-3DHP. We also evaluate our method on MPI-INF-3DHP. Here, we use 81 frames as our input due to the shorter video length of this dataset. The results in Tab. 2 demonstrate that our method achieves the best performance, showing the efficacy of our DiffPose in improving performance in outdoor scenes.

Table 2. Video-based results on MPI-INF-3DHP.

Method	PCK ↑	AUC ↑	MPJPE ↓
Pavllo [32]	86.0	51.9	84.0
Wang [44]	86.9	62.1	68.1
Zheng [54]	88.6	56.4	77.1
Li [24]	93.8	63.3	58.0
Zhang [49]	94.4	66.5	54.9
Ours	**98.0**	**75.9**	**29.1**

Frame-based Results on Human3.6M. To further investigate the efficacy of DiffPose, we evaluate it in a more challenging setting: frame-based 3D pose estimation. Here, we only extract context information from the single input frame via our Context Encoder ϕ_{ST}. Our results on Human3.6M are reported in Tab. 3. As shown at the top of Tab. 3, our DiffPose surpasses all existing methods in average MPJPE using detected 2D poses. At the bottom of Tab. 3, we observe that DiffPose also outperforms all methods with a large margin when ground truth 2D poses are used.

Qualitative results. In the first four columns of Fig. 3,

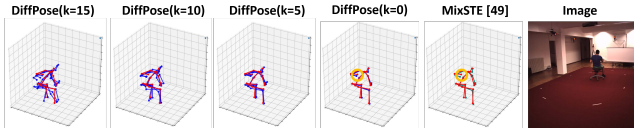


Figure 3. Qualitative results. Red colored 3D pose corresponds to the ground truth. Under occlusion, our DiffPose predicts a pose that is more accurate than previous methods (circled in orange).

we provide visualizations of the reverse diffusion process, where the step k decreases from 15 to 0. We observe that DiffPose can progressively narrow down the gap between the sampled poses and the ground-truth pose. Moreover, we compare our method with the current SOTA method [49], which shows that our method can generate more reliable 3D pose solutions, especially for ambiguous body parts.

5.2. Ablation Study

To verify the impact of each proposed design, we conduct extensive ablation experiments on Human3.6M dataset using the detected 2D poses in the video-based setting.

Impact of Diffusion Process. We first evaluate the diffusion process's effectiveness. Here we build two baseline models: (1) **Baseline A**: It has the same structure as our diffusion model but the 3D pose estimation is conducted in a single step. (2) **Baseline B**: It has the nearly same architecture as our diffusion model but the diffusion model is stacked multiple times to approximate the computational complexity of DiffPose. Note that both baselines are optimized to predict 3D human pose instead of learning the reverse diffusion process. We report the results of the baselines and DiffPose in Tab. 4. The performance of both baselines are much worse than our DiffPose, which indicates that the performance improvement of our method comes from the designed diffusion pipeline.

Table 4. Ablation study for diffusion pipeline.

Method	MPJPE	P-MPJPE
Baseline A	44.3	33.7
Baseline B	41.1	32.8
DiffPose	36.9	28.6

Impact of GMM. To validate the effect of the GMM design, we consider two alternative ways to train our diffusion model: (1) **Stand-Diff**: we directly adopt the basic forward diffusion process introduced in Eq. 4 for model training. (2) **GMM-Diff**: we utilize GMM to fit the initial 3D pose distribution H_K to generate intermediate distributions for model training. Moreover, we test the number of kernels in GMM M (from 1 to 9) to investigate the characteristics of GMM in pose diffusion. We report the results with different M in Tab. 5. Experiments show that our GMM-based design significantly outperforms the baseline Stand-Diff, which shows the effectiveness of using a GMM to approximate H_K. Moreover, we can observe that using 5 kernels ($M = 5$) is sufficient to effectively capture the uncertainty distribution.

Table 5. Ablation study for GMM design

Method	MPJPE	P-MPJPE
Stand-Diff	40.1	31.1
GMM-Diff(M=1)	38.0	29.7
GMM-Diff(M=5)	36.9	28.6
GMM-Diff(M=9)	36.5	28.5

Impact of context f_{ST}. Another crucial component to explore is the role of spatial-temporal context f_{ST} in our method. Here, we evaluate the performance when using various context encoders [34, 49] to obtain f_{ST}. As shown in Tab. 6, our DiffPose achieves good performance using both models. We also find that DiffPose significantly outperforms both context encoders, which verifies the efficacy of our approach.

Table 6. Ablation study for f_{ST}.

Method	MPJPE	P-MPJPE
[34]	42.1	34.4
Ours + [34]	39.3	31.8
[49]	40.9	32.6
Ours + [49]	36.9	28.7

Impact of reverse diffusion steps K and sample number N. To further investigate the characteristics of our pose diffusion process, we conduct several experiments with different diffusion step numbers (K) and sample numbers (N) and plot the results in Fig. 4. We observe that MPJPE first drops significantly till $K = 50$, and shows minor improvements when $K > 50$. Thus, we use 50 diffusion steps ($K = 50$) in our method, which can effectively and efficiently shed indeterminacy. On the other hand, we find that model performance improves with the number of samples N until $N = 5$, where our performance stays roughly consistent.

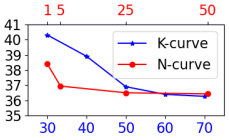


Figure 4. Evaluation of parameters K and N.

Inference Speed. In Tab. 7, we compare the speed of DiffPose with existing methods in terms of Frames Per Second (FPS). Our DiffPose with DDIM acceleration can achieve a competitive speed compared with the current SOTA [49] while obtaining better performance. Moreover, even without DDIM acceleration, the FPS of our model is still higher than 170 FPS, which satisfies most real-time requirements.

Table 7. Analysis of speed. Our method can run efficiently, yet outperforms SOTA significantly.

Method	MPJPE	FPS
Li [19]	43.0	328
Zhang [49]	40.9	974
DiffPose w/o DDIM	36.7	173
DiffPose w/ DDIM	36.9	671

6. Conclusion

This paper presents DiffPose, a novel diffusion-based framework that handles the uncertainty and indeterminacy in monocular 3D pose estimation. DiffPose first initializes the indeterminate 3D pose distribution and then recurrently sheds the indeterminacy in this distribution to obtain the final high-quality 3D human pose distribution for reliable pose estimation. Extensive experiments show that the proposed DiffPose achieves state-of-the-art performance on two widely used benchmark datasets.

Acknowledgments. This work is supported by MOE AcRF Tier 2 (Proposal ID: T2EP20222-0035), National Research Foundation Singapore under its AI Singapore Programme (AISG-100E-2020-065), and SUTD SKI Project (SKI 2021_02_06). This work is also supported by TAILOR, a project funded by EU Horizon 2020 research and innovation programme under GA No 952215.

References

[1] Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. *Advances in Neural Information Processing Systems*, 34:17981–17993, 2021. 3

[2] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2272–2281, 2019. 1, 2, 7

[3] Tianlang Chen, Chen Fang, Xiaohui Shen, Yiheng Zhu, Zhili Chen, and Jiebo Luo. Anatomy-aware 3d human pose estimation with bone-based pose decomposition. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):198–209, 2021. 2, 7

[4] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018. 4, 6, 7

[5] Manuela Chessa, Guido Maiello, Lina K Klein, Vivian C Paulun, and Fabio Solari. Grasping objects in immersive virtual reality. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 1749–1754. IEEE, 2019. 1

[6] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2262–2271, 2019. 2

[7] Zhipeng Fan, Jun Liu, and Yao Wang. Adaptive computationally efficient network for monocular 3d hand pose estimation. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 127–144. Springer, 2020. 2

[8] Zhipeng Fan, Jun Liu, and Yao Wang. Motion adaptive pose estimation from compressed videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11719–11728, 2021. 2

[9] Lin Geng Foo, Jia Gong, Zhipeng Fan, and Jun Liu. System-status-aware adaptive network for online streaming video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 2

[10] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, Qiuhong Ke, and Jun Liu. Unified pose sequence modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023. 1, 2

[11] Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. Stochastic trajectory prediction via motion indeterminacy diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17113–17122, 2022. 4

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 2, 3, 4, 6

[13] Mir Rayat Imtiaz Hossain and James J Little. Exploiting temporal information for 3d human pose estimation. In *ECCV*, pages 68–84, 2018. 7

[14] Wenbo Hu, Changgong Zhang, Fangneng Zhan, Lei Zhang, and Tien-Tsin Wong. Conditional directed graph convolution for 3d human pose estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 602–611, 2021. 2

[15] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE transactions on pattern analysis and machine intelligence*, 36(7):1325–1339, 2013. 6

[16] Rawal Khirodkar, Visesh Chari, Amit Agrawal, and Ambrish Tyagi. Multi-instance pose networks: Rethinking top-down pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3122–3131, 2021. 1

[17] Chen Li and Gim Hee Lee. Generating multiple hypotheses for 3d human pose estimation with mixture density network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9887–9895, 2019. 1, 2

[18] Jonathan Li and Andrew Barron. Mixture density estimation. *Advances in neural information processing systems*, 12, 1999. 5

[19] Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13147–13156, 2022. 1, 2, 7, 8

[20] Xiang Lisa Li, John Thickstun, Ishaan Gulrajani, Percy Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *arXiv preprint arXiv:2205.14217*, 2022. 3

[21] Xing Liang, Anastassia Angelopoulou, Epaminondas Kapetanios, Bencie Woll, Reda Al Batat, and Tyron Woolfe. A multi-modal machine learning approach and toolkit to automate recognition of early stages of dementia among british sign language users. In *European Conference on Computer Vision*, pages 278–293. Springer, 2020. 1

[22] Jiahao Lin and Gim Hee Lee. Trajectory space factorization for deep video-based 3d human pose estimation. In *BMVC*, 2019. 7

[23] Kenkun Liu, Rongqi Ding, Zhiming Zou, Le Wang, and Wei Tang. A comprehensive study of weight sharing in graph networks for 3d human pose estimation. In *European Conference on Computer Vision*, pages 318–334. Springer, 2020. 1, 7

[24] Ruixu Liu, Ju Shen, He Wang, Chen Chen, Sen-ching Cheung, and Vijayan Asari. Attention mechanism exploits temporal contexts: Real-time 3d human pose reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5064–5073, 2020. 7

[25] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 3

[26] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *IEEE ICCV*, pages 2640–2649, 2017. 2, 7

[27] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 6, 7

[28] Eliya Nachmani, Robin San Roman, and Lior Wolf. Non gaussian denoising diffusion models. *arXiv preprint arXiv:2106.07582*, 2021. 3, 5

[29] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 3

[30] Sungheon Park, Jihye Hwang, and Nojun Kwak. 3d human pose estimation using convolutional neural networks with 2d pose information. In *European Conference on Computer Vision*, pages 156–169. Springer, 2016. 2

[31] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G Derpanis, and Kostas Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. In *IEEE CVPR*, pages 7025–7034, 2017. 2, 7

[32] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019. 1, 2, 7

[33] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 1

[34] Wenkang Shan, Zhenhua Liu, Xinfeng Zhang, Shanshe Wang, Siwei Ma, and Wen Gao. P-stmo: Pre-trained spatial temporal many-to-one model for 3d human pose estimation. In *ECCV*, page 461–478, 2022. 2, 7, 8

[35] Wenkang Shan, Haopeng Lu, Shanshe Wang, Xinfeng Zhang, and Wen Gao. Improving robustness and accuracy via relative information encoding in 3d human pose estimation. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 3446–3454, 2021. 2

[36] Saurabh Sharma, Pavan Teja Varigonda, Prashast Bindal, Abhishek Sharma, and Arjun Jain. Monocular 3d human pose estimation by generation and ordinal ranking. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2325–2334, 2019. 2

[37] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 4, 6, 7

[39] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019. 1

[40] Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. Investigating the dexterity of multi-finger input for mid-air text entry. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3643–3652, 2015. 1

[41] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional human pose regression. In *IEEE ICCV*, pages 2602–2611, 2017. 7

[42] Xiao Sun, Bin Xiao, Fangyin Wei, Shuang Liang, and Yichen Wei. Integral human pose regression. In *Proceedings of the European conference on computer vision (ECCV)*, pages 529–545, 2018. 2

[43] Chen Wang, Feng Zhang, Xiatian Zhu, and Shuzhi Sam Ge. Low-resolution human pose estimation. *Pattern Recognition*, 126:108579, 2022. 5

[44] Jingbo Wang, Sijie Yan, Yuanjun Xiong, and Dahua Lin. Motion guided 3d pose estimation from videos. In *European Conference on Computer Vision*, pages 764–780. Springer, 2020. 2, 7

[45] Jingwei Xu, Zhenbo Yu, Bingbing Ni, Jiancheng Yang, Xiaokang Yang, and Wenjun Zhang. Deep kinematics analysis for monocular 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 899–908, 2020. 2

[46] Tianhan Xu and Wataru Takano. Graph stacked hourglass networks for 3d human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16105–16114, 2021. 1, 2, 7

[47] Wei Yang, Wanli Ouyang, Xiaolong Wang, Jimmy Ren, Hongsheng Li, and Xiaogang Wang. 3d human pose estimation in the wild by adversarial learning. In *IEEE CVPR*, pages 5255–5264, 2018. 7

[48] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 7

[49] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13232–13242, 2022. 6, 7, 8

[50] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N. Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3425–3435, 2019. 1

[51] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *IEEE CVPR*, pages 3425–3435, 2019. 2, 7

[52] Weixi Zhao, Weiqiang Wang, and Yunjie Tian. Graformer: Graph-oriented transformer for 3d pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision*

and Pattern Recognition, pages 20438–20447, 2022. 1, 2, 6, 7

[53] Yunqing Zhao, Tianyu Pang, Chao Du, Xiao Yang, Ngai-Man Cheung, and Min Lin. A recipe for watermarking diffusion models. *arXiv preprint arXiv: 2303.10137*, 2023. 3

[54] Ce Zheng, Sijie Zhu, Matias Mendieta, Taojiannan Yang, Chen Chen, and Zhengming Ding. 3d human pose estimation with spatial and temporal transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11656–11665, 2021. 1, 2, 7