An Image-like Diffusion Method for Human-Object Interaction Detection

Supplementary Material

1. More Implementation Details

We follow [5] to use DDETR [6] and obtain human-object detection pairs. For the diffusion forward process, we generate $\{\beta_k\}_{k=1}^K$ by linearly interpolating from 0.001 to 0.2. Then, we compute $\alpha_k = 1 - \beta_k$, $\overline{\alpha}_k = \prod_{s=1}^{k} \alpha_s$. In the diffusion reverse process, we follow [4] to generate the step embedding f_s^k to provide time step information to the diffusion model. We build the diffusion model based on DiT-S [4], with 12 transformer layers.

2. More Architecture Details

Here we provide more details about the model architecture of the diffusion model in our framework as introduced in Sec. 4.3 of the main paper.

For an input *HOI image* of size $H \times W \times 2$, we convert it into $H + W$ tokens via the proposed slice patchification architecture. Specifically, first, we split the $H \times W \times 2$ *HOI image* into H horizontal slice patches (each with size $W \times 2$), and W vertical slice patches (each with size $H \times 2$). Then we convert the obtained patches into tokens using linear projection [4]. In particular, we construct an MLP layer to convert the H horizontal slice patches to H tokens, and an MLP layer to convert the W vertical slice patches to W tokens. All the obtained tokens are of equal dimension (i.e., 384) as DiT token embeddings. The obtained H tokens and W tokens are concatenated together, resulting in a total of $H + W$ *HOI image* tokens.

After constructing the $H + W$ tokens, given a diffusion step k and the appearance feature f_a, as mentioned in Sec 4.2 in the main paper, we aim for the diffusion model to also involve these information during its reverse diffusion process. To achieve this, we first construct the diffusion step embedding f_s^k [4]. Moreover, we convert the appearance feature to embedding of the same dimension as the *HOI image* token (i.e., 384) via an MLP layer. Then, we input the $H + W$ tokens as image token sequence to the DiT blocks and meanwhile input the constructed step embedding and the converted appearance embedding as conditioning embedding to the DiT blocks [4]. This process finally outputs a sequence of $H + W$ tokens.

Finally, in the last part of the diffusion model, we aim to obtain the final *HOI image* of size $H \times W \times 2$ from the above outputted $H + W$ tokens. To achieve this, we first apply linear projection [4] to the $H + W$ tokens. Specifically, we construct an MLP layer to project the H tokens to a tensor with shape of $H \times (W \times 2)$, and construct another MLP layer to project the W tokens to a tensor with shape $W \times (H \times 2)$. Then, we reshape the tensors to obtain two tensors of shape $H \times W \times 2$ [4]. Finally, we fuse the two tensors with an MLP layer to obtain the final *HOI image*.

3. More Ablation Studies and Further Analysis

To evaluate our proposed HOI-IDiff framework more comprehensively, we conduct the following ablation studies on the Default setting of HICO-DET [1].

Impact of the diffusion process. To further evaluate the impact of employing the diffusion process, we compare our HOI-IDiff with the following variants: (1) In *Variant A*, we adopt the same diffusion model architecture as HOI-IDiff but do not perform diffusion process, i.e., *Variant A* is trained to directly predict the HOI image in a single step. (2) *Variant B* is similar to *Variant A*, but we stack the diffusion model architecture multiple times, resulting in a model that has similar computation complexity as our method. As shown in Tab. 1, the performances of the two variants drop significantly, demonstrating the effectiveness of the designed diffusion process.

Method	Full	Rare	Non-rare
Variant A	42.09	42.35	42.01
Variant B	41.77	42.05	41.68
Ours	47.71	48.36	47.52

Table 1. Impact of the diffusion process.

More experiments on the HOI image formulation. In our framework ($H \times W \times 2$ *HOI image*), we formulate the *HOI image* I^{hoi} as the product of v^{obj} with size H and m^{int} with size $W \times 2$. In Tab. 3 of the main paper, to validate this formulation of *HOI image*, we have compared our framework with variants I, II, IV. We here further extend this ablation study with the following variants. In variant V ($W \times 2$ *HOI image* with H *HOI image* as condition), for each human-object pair, we obtain its object classification results from the off-the-shelf object detector. Then, instead of I^{hoi}, we regard m^{int} (of size $W \times 2$) as the *HOI image* and only generate m^{int}. Note that in this process, v^{obj} of size H is fed to the diffusion model as a condition to guide the generation of the $W \times 2$ *HOI image* m^{int}. In variant VI ($H \times W$ *HOI image*), for each human-object pair, instead of I^{hoi} of size $H \times W \times 2$, we formulate an *HOI image* of size $H \times W$. In this formulated *HOI image*, the pixel value at the h-th row and the w-th column indicates the presence probability of the w-th interaction category, under the assumption that the object in the current pair belongs to h-th object category. During post-processing, we then apply threshold (0.5) to the pixel values in the $H \times W$ *HOI image* predicted by the diffusion model to obtain the interaction prediction re-

sults. Notably, unlike I^{hoi}, the *HOI image* formulated in the above way does not guarantee each of its column (vertical slice) to sum to 1. Thus, in the diffusion process we use in variant VI, unlike our proposed *HOI image* diffusion process, we do not encourage each column (vertical slice) of the *HOI image* to sum to 1. As shown in Tab. 2, our method ($H \times W \times 2$ *HOI image*) outperforms the above two variants. This further shows the superiority of our *HOI image* formulation design.

Method	Full	Rare	Non-rare
I: $W \times 2$ *HOI image*	46.43	47.22	46.19
II: H & $W \times 2$ *HOI images*	46.83	47.47	46.64
III: $H \times W \times 2$ *HOI image*	47.71	48.36	47.52
IV: Box coordinates & $H \times W \times 2$ *HOI image*	47.79	48.36	47.62
V: $W \times 2$ *HOI image* with H *HOI image* as condition	46.60	47.39	46.37
VI: $H \times W$ *HOI image*	42.26	42.57	42.17

Table 2. Further evaluation on the HOI image formulation process.

Impact of the number of diffusion steps K. We investigate the impact of the diffusion step K on the performance of HOI-IDiff. Specifically, we conduct experiments with different diffusion steps ($K = [10, 30, 50, 70]$) as shown in Tab. 3. As shown, the performance of HOI-IDiff increases with K consistently when K is smaller than 50, and becomes stabilized after K reaches 50. Thus, taking the model efficiency also into the consideration, we set K to 50 in our experiments.

K	Full	Rare	Non-rare
10	44.51	44.62	44.48
30	46.46	46.93	46.32
50	47.71	48.36	47.52
70	47.76	48.38	47.58

Table 3. Impact of the number of diffusion steps K.

Impact of the number of trials T in Multinomial distribution. We explore the impact of the number of trials T of the Multinomial distribution on the performance. We conduct experiments with varying number of trials ($T = [500, 1000, 2000, 4000]$) and show the results in Tab. 4. As shown, the model performance improves with T when T is smaller than 2000, and becomes stabilized after T reaches 2000, Thus, we set $T = 2000$ in our experiments.

T	Full	Rare	Non-rare
500	44.81	44.63	44.86
1000	45.50	45.85	45.39
2000	47.71	48.36	47.52
4000	47.73	48.35	47.54

Table 4. Impact of the number of trials T.

Impact of the appearance features f_a. In our framework, we feed the appearance feature f_a into the diffusion model θ to guide its reverse diffusion process (*w/* f_a). To validate the efficacy of f_a, we test a variant (*w/o* f_a) in which we do not feed f_a into θ during the reverse diffusion process. As shown in Tab. 5, our framework with f_a outperforms this variant. This shows that f_a can effectively guide the diffusion model θ in the reverse diffusion process to generate *HOI images* more accurately.

Method	Full	Rare	Non-rare
w/o f_a	43.15	43.05	43.18
w/ f_a	47.38	48.18	47.12

Table 5. Evaluation on the appearance features f_a.

4. More Visualizations

Comparisons of typical image diffusion process and our proposed diffusion process. In our framework, we propose an *HOI image* diffusion process tailored for *HOI image* generation. Here, to further evaluate this design, besides the visualization in Fig. 3 in the main paper, we also compare the generated final *HOI images* using typical natural image diffusion process [3] and our proposed *HOI image* diffusion process. As shown in Fig. 1, the *HOI image* generated using typical natural image diffusion process appears to be much more ambiguous than the *HOI image* generated with our diffusion process. The white pixels in Fig. 1 (a) scatter across different rows, which suggests that the predictions for object categories are spread over multiple categories. On the contrary, as shown in Fig. 1 (b), the *HOI image* generated using our proposed *HOI image* diffusion process is much more determinated, where the white pixels are distinctly highlighted and are clearly concentrated on the same row. This shows the efficacy of our proposed *HOI image* diffusion process.

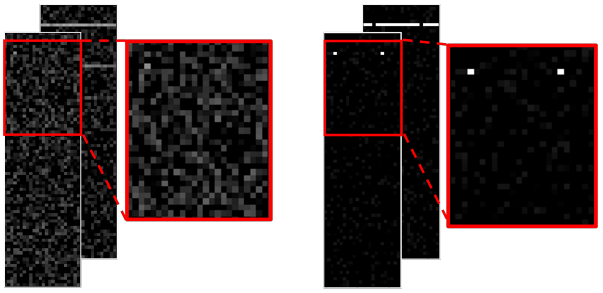

Figure 1. Visualization of the generated final *HOI image* of typical image diffusion process (a) and our *HOI image* diffusion process (b).

5. Analysis and Proofs

5.1. More Analysis about the Unique Property of the HOI Image.

We here provide a more detailed analysis on the unique property of the *HOI image* mentioned in Sec. 4.1 of the main paper. Specifically, in this subsection, we first review the formulation of *HOI image* introduced in Sec. 1 and Sec. 4.1 in the main paper. Then, we elaborate on how each of the vertical slices of the *HOI image* sums to 1. Finally, we discuss how we can uniquely decompose the *HOI image* to v^{obj} and m^{int} if each of its vertical slices sums to 1.

Review of *HOI image* formulation. As introduced in Sec. 1 and Sec. 4.1 in the main paper, the *HOI image* is formed by "multiplying" v^{obj} and m^{int}. Specifically, the object classification output v^{obj} represents a probability distribution of size H, where H is the number of object categories. The interaction prediction output m^{int} of size $W \times 2$ consists of W probability distributions, each with size 2, where W is the number of interaction categories.

Elaboration on how vertical slices in *HOI image* sum to 1. Here we elaborate with the w-th vertical slice in the *HOI image* as an example. Because the *HOI image* is formed as the product of v^{obj} and m^{int}, its w-th vertical slice (i.e., $I^{hoi}[:, w, :]$) of size $H \times 2$ is essentially derived by multiplying v^{obj} of size H and the m-th distribution in m^{int} (i.e., $m^{int}[w]$) of size 2. As v^{obj} and $m^{int}[w]$ both represent probability distributions that each sums to 1, $I^{hoi}[:, w, :]$ represents a joint probability distribution that also sums to 1. Considering all W vertical slices, the *HOI image* essentially contains W joint probability distributions that each sums to 1.

Elaboration on how to decompose to v^{obj} and m^{int}. Here, we also show that if each of the vertical slices in an obtained matrix M (of size $H \times W \times 2$) represents a probability distribution, it forms an *HOI image*, and we can uniquely decompose the *HOI image* into the corresponding v^{obj} and m^{int}. First, consider the w-th vertical slice of the obtained matrix $M[:, w, :]$ of size $H \times 2$. We can obtain a vector v_1 of size H by summing along the second dimension, and a vector v_2 of size 2 by summing along the first dimension. Consequently, both the sum of v_1 and the sum of v_2 are essentially equal to the sum of $M[:, w, :]$, i.e., v_1 and v_2 both sum to 1. Then, for all the W vertical slices in the matrix M, we can obtain W vectors of size H, and W vectors of size 2. Each of the obtained vector sums to 1. We can then uniquely derive v^{obj} (of size H) by taking the average of the W vectors of size H. Then, we can obtain m^{int} (of size $W \times 2$) by collecting the W vectors of size 2 to form a matrix of size $W \times 2$. Via the above, we can then decompose M back to its corresponding v^{obj} and m^{int}.

5.2. Derivation of Eq. 5 in the Main Paper

In this subsection, we show how we derive Eq. 5 in the main paper. To better elaborate on this, we first provide more details and notations of the Multinomial distribution.

Details of Multinomial Distribution. We here first provide more details about the Multinomial distribution for better understanding.

We denote Multinomial distribution as $P^{Mu}(T, p)$, where T is the number of trials and p (of size N) represents the probabilities of N categories, p sums to 1. $P^{Mu}(T, p)$ represents the distribution of counts of the picked categories in T trials conducted with replacement, where in each trail, one category is picked with the probabilities provided in p. The likelihood function of the number of counts for all categories is defined as:

$$P^{Mu}(x; T, d) = \frac{T!}{\prod_{n=1}^{N} x_n!} \prod_{n=1}^{N} p_n^{x_n}, \tag{1}$$

where x represents the non-negative integer counts across all categories. Notably, elements in x sums to the total number of trials T. Here, we can also easily compute the Multinomial likelihood ϵ^{Mu} of categories (ϵ^{Mu} sums to 1), by:

$$P^{Mu}_{T\epsilon^{Mu}}(\epsilon^{Mu}; T, d) = \frac{T!}{\prod_{n=1}^{N} (T\epsilon_n^{Mu})!} \prod_{n=1}^{N} p_n^{T\epsilon_n^{Mu}} \tag{2}$$

We use $P_T^{Mu}(T, p)$ to represent sampling ϵ^{Mu} from Eq. (2), which is sampling from $P^{Mu}(T, p)$ and dividing the resultant counts by T such that elements in ϵ^{Mu} sum to 1. Note that, for simplicity, in the following we use the formulation of x in Eq. (1) that sums to T, but the statements and derived results also hold for ϵ that sums to 1.

Derivation of Eq. 5 in the main paper. Then, we show how to derive Eq. 5 in the main paper. For simplicity, same as Sec. 4.2 in the main paper, we illustrate this by focusing a single vertical slice of the *HOI image*, i.e., the w-th vertical slice. For convenience, we first repeat Eq. 5 in the main paper here:

$$d_k = \overline{\alpha}_k d_0 + (1 - \overline{\alpha}_k)\bar{\epsilon}^{Mu}, \tag{3}$$

where $\alpha_k = 1 - \beta_k$, $\overline{\alpha}_k = \prod_{s=1}^{k} \alpha_s$, $\bar{\epsilon}^{Mu} \sim P^{Mu}_{S_k T}(S_k T, d_{init})$, and $S_k = \frac{(1-\overline{\alpha}_j)^2}{(\prod_{j=2}^{k} \alpha_j)^2 \beta_1^2 + (\prod_{j=3}^{k} \alpha_j)^2 \beta_2^2 + \ldots + \beta_k^2}$.

Specifically, as mentioned in Eq. 4 of the main paper, in each diffusion step k in the forward process, we inject a small noise to d_{k-1} as:

$$d_k = (1 - \beta_k) d_{k-1} + \beta_k \epsilon^{Mu}, \tag{4}$$

where $\epsilon^{Mu} \sim P_T^{Mu}(T, d_{init})$, and d_{init} is the w-th vertical slice in the initialized noisy *HOI image*. Expanding Eq. (4)

we can get:

$$d_k = \overline{\alpha}_k d_0 + (\prod_{j=2}^{k} \alpha_j)\beta_1 \epsilon^{Mu,1} + (\prod_{j=3}^{k} \alpha_j)\beta_2 \epsilon^{Mu,2} + ... + \beta_k \epsilon^{Mu,k}, \tag{5}$$

where $\epsilon^{Mu,j} \sim P_T^{Mu}(T, d_{init})$ denotes the sampled noise from the Multinomial in the j-th step.

Then, we can simplify Eq. (5) by taking advantage of the property of Multinomial distribution. Specifically, the sum of samples from Multinomials with the same probability p results in a sample from another Multinomial. In other words, let $x_3 = x_1 + x_2$, $x_1 \sim P^{Mu}(T_1, p)$, and $x_2 \sim P^{Mu}(T_2, p)$. Then $x_3 \sim P^{Mu}(T_1 + T_2, p)$. We show this using the characteristic functions of Multinomial distribution. Specifically, note that the characteristic function of the Multinomial distribution can be represented as $\varphi_x(t) = \mathbb{E}[e^{itx}] = (\sum_{n=1}^{N} p_n e^{it_n})^T$, where $x \sim P^{Mu}(T, p)$, and $i^2 = -1$. As the sum of two random variables is equal to the product of the corresponding characteristic functions [2], we can get:

$$\begin{aligned}
\varphi_{x_1+x_2}(t) &= \varphi_{x_1}(t)\varphi_{x_2}(t) \\
&= (\sum_{n=1}^{N} p_n e^{it_n})^{T_1} (\sum_{n=1}^{N} p_n e^{it_n})^{T_2} \\
&= (\sum_{n=1}^{N} p_n e^{it_n})^{T_1+T_2}, \\
&= \varphi_{x_3}(t)
\end{aligned} \tag{6}$$

where $x_3 \sim P^{Mu}(T_1 + T_2, p)$. Thus, we can see that the sum of samples from Multinomials results in a sample from another Multinomial. However, we also need to apply scaling factors $\{\beta_j\}_{j=1}^k$ that re-weights the samples, which complicates the scenario. We then aim to derive a good approximation of the linear combination of the samples. Specifically, c_k is constructed as:

$$c_k = (\prod_{j=2}^{k} \alpha_j)\beta_1 x_1 + (\prod_{j=3}^{k} \alpha_j)\beta_2 x_2 + ... + \beta_k x_k \tag{7}$$

where $x_j \sim P^{Mu}(T, p)$. Then, Lemma 1 is introduced below:

Lemma 1 *If for c_k defined in Eq. (7), $x_j \sim P^{Mu}(T, p)$ is independently sampled for $j \in [1, ..., k]$, then the following approximately holds: $\frac{S_k}{1-\overline{\alpha}_k} c_k \sim P^{Mu}(S_k T, p)$, where $\overline{\alpha}_k = \prod_{s=1}^{k} \alpha_s$, and $S_k = \frac{(1-\overline{\alpha}_k)^2}{(\prod_{j=2}^{k} \alpha_j)^2 \beta_1^2 + (\prod_{j=3}^{k} \alpha_j)^2 \beta_2^2 + ... + \beta_k^2}$*

We then provide proof of Lemma 1. First, we take advantage of the property of Multinomial distributions that when the number of trials T becomes large, the likelihood over each dimension can be approximated with a Gaussian distribution [2]. Specifically, for $P^{Mu}(T, p)$, the mean and variance of the approximating Gaussian of dimension m are Tp_m and $Tp_m(1 - p_m)$ respectively, where p_m represents the corresponding m-th element of the probabilities provided in p. Thus, by setting T to be large, each dimension m of x_j can be approximated as $\mathcal{N}(Tp_m, Tp_m(1 - p_m))$.

Moreover, the convolution of two independent Gaussians is a Gaussian, with the expectation and variance being the sum of the two Gaussians, i.e., $\mathcal{N}(\mu_1, \sigma_1^2) * \mathcal{N}(\mu_2, \sigma_2^2) = \mathcal{N}(\mu_1 + \mu_2, \sigma_1^2 + \sigma_2^2)$. Thus, the likelihood of each multinomial can be approximated with a Gaussian. Then, parameters to approximate c_k can be found by computing the mean and variance of each element x_j and summing them up. Specifically, at diffusion step k, the m-th dimension of c_k can be approximated as $\mathcal{N}\big(\mathbb{E}(c_{k,m}), \mathbb{V}(c_{k,m})\big)$, with mean and variance derived as:

$$\begin{aligned}
\mathbb{E}(c_{k,m}) &= (\prod_{j=2}^{k} \alpha_j)\beta_1 \mu_{1,m} + (\prod_{j=3}^{k} \alpha_j)\beta_2 \mu_{2,m} + ... + \beta_k \mu_{k,m} \\
&= \big((\prod_{j=2}^{k} \alpha_j)\beta_1 + (\prod_{j=3}^{k} \alpha_j)\beta_2 + ... + \beta_k\big) T p_m \\
&= (1 - \overline{\alpha}_k) T p_m
\end{aligned} \tag{8}$$

$$\begin{aligned}
\mathbb{V}(c_{k,m}) &= (\prod_{j=2}^{k} \alpha_j)^2 \beta_1^2 \sigma_{1,m}^2 + (\prod_{j=3}^{k} \alpha_j)^2 \beta_2^2 \sigma_{2,m}^2 + ... + \beta_k^2 \sigma_{k,m}^2 \\
&= ((\prod_{j=2}^{k} \alpha_j)^2 \beta_1^2 + (\prod_{j=3}^{k} \alpha_j)^2 \beta_2^2 + ... + \beta_k^2) T p_m (1 - p_m) \\
&= \eta_k T p_m (1 - p_m)
\end{aligned} \tag{9}$$

where $\eta_k = \big((\prod_{j=2}^{k} \alpha_j)^2 \beta_1^2 + (\prod_{j=3}^{k} \alpha_j)^2 \beta_2^2 + ... + \beta_k^2\big)$.

With the above derived, we then aim to find the Multinomial such that its likelihood can be approximated by a Gaussian with mean and variance obtained in Eq. (8) and Eq. (9) respectively. This means that, a sample u_k from this Multinomial should also have expectation and variance as Eq. (8) and Eq. (9). Specifically, for this to hold, the sampling probability of the Multinomial should be p as well. Then, to match the expectation of the Multinomial to Eq. (8), and design the variance of the Multinomial to Eq. (9), we can form the Multinomial as $P^{Mu}(S_k T, p)$, where we scale the number of trials T by S_k and accordingly scale the sample u_k from the Multinomial by $\frac{(1-\overline{\alpha}_k)}{S_k}$. Then, the expectation of the m-th dimension of $\frac{(1-\overline{\alpha}_k)}{S_k} u_k$, denoted as $\tau_{k,m}$, is computed as:

$$\begin{aligned}
\mathbb{E}[\tau_{k,m}] &= \frac{(1 - \overline{\alpha}_k)}{S_k} S_k T p_m \\
&= (1 - \overline{\alpha}_k) T p_m
\end{aligned} \tag{10}$$

which is equal to Eq. (8). The variance of $\tau_{k,m}$ can also be computed as:

$$\begin{aligned}
\mathbb{V}[\tau_{k,m}] &= (\frac{1 - \overline{\alpha}_k}{S_k})^2 S_k T p_m (1 - p_m) \\
&= \frac{(1 - \overline{\alpha}_k)^2}{S_k} T p_m (1 - p_m)
\end{aligned} \tag{11}$$

Then, for Eq. (11) to equal to Eq. (9), we can derive:

$$S_k = \frac{(1-\overline{\alpha}_k)^2}{\eta_k} \tag{12}$$

where $\eta_k = ((\prod_{j=2}^{k}\alpha_j)^2\beta_1^2 + (\prod_{j=3}^{k}\alpha_j)^2\beta_2^2 + ... + \beta_k^2)$.

We also remark that the covariance between different dimensions of c_k and $\frac{(1-\overline{\alpha}_k)}{S_k}u_k$ are approximately equal as well when setting S_k as Eq. (12) [2]. Overall, considering all elements (dimensions) in c_k, we can approximately have:

$$\frac{S_k}{1-\overline{\alpha}_k}c_k \sim P^{Mu}(S_kT, p) \tag{13}$$

where $c_k = (\prod_{j=2}^{k}\alpha_j)\beta_1 x_1 + (\prod_{j=3}^{k}\alpha_j)\beta_2 x_2 + ... + \beta_k x_k$. This finishes the proof of Lemma 1.

To this point, with the proved Lemma 1, Eq. (5) can be approximated as:

$$d_k = \overline{\alpha}_k d_0 + (1-\overline{\alpha}_k)\bar{\epsilon}^{Mu} \tag{14}$$

where $\bar{\epsilon}^{Mu} \sim P^{Mu}_{S_kT}(S_kT, d_{init})$, d_{init} provides the sampling probability in each trial, and $S_k = \frac{(1-\overline{\alpha}_j)^2}{(\prod_{j=2}^{k}\alpha_j)^2\beta_1^2 + (\prod_{j=3}^{k}\alpha_j)^2\beta_2^2 + ... + \beta_k^2}$.

5.3. Derivation of Eq. 6 in the Main Paper

Here we show how to derive $q(d_{k-1}|d_k, d_0)$ as Eq. 6 in the main paper. Specifically, from Bayes' rule we have:

$$q(d_{k-1}|d_k, d_0) = \frac{q(d_k|d_{k-1}, d_0)\cdot q(d_{k-1}|d_0)}{q(d_k|d_0)} \tag{15}$$

Then, based on that the forward process of our diffusion can be modeled as a Markov chain (shown below in Sec. 5.4), we have $q(d_k|d_{k-1}, d_0) = q(d_k|d_{k-1})$ [2]. Thus, the above equation can be further re-written as:

$$\begin{aligned} q(d_{k-1}|d_k, d_0) &= \frac{q(d_k|d_{k-1})\cdot q(d_{k-1}|d_0)}{q(d_k|d_0)} \\ &= \frac{q(d_k|d_{k-1})\cdot q(d_{k-1}|d_0)}{\sum_{d_{k-1}} q(d_k|d_{k-1})\cdot q(d_{k-1}|d_0)} \end{aligned} \tag{16}$$

We then reformulate Eq. 5 in the main paper (i.e., Eq. (3) in Supplementary) as:

$$q(d_k|d_0) = P^{Mu}_{\frac{S_kT(d_k-\overline{\alpha}_k d_0)}{(1-\overline{\alpha}_k)}}(d_k; S_kT, d_{init}, d_0) \tag{17}$$

Thus, we have $q(d_{k-1}|d_0)$ as:

$$q(d_{k-1}|d_0) = P^{Mu}_{\frac{S_{k-1}T(d_{k-1}-\overline{\alpha}_{k-1}d_0)}{(1-\overline{\alpha}_{k-1})}}(d_{k-1}; S_{k-1}T, d_{init}, d_0) \tag{18}$$

$q(d_k|d_{k-1})$ can also be obtain as:

$$q(d_k|d_{k-1}) = q(d_{k-1}|d_k) = P^{Mu}_{\frac{T(d_k-(1-\beta_k)d_{k-1})}{\beta_k}}(d_{k-1}; T, d_{init}, d_k) \tag{19}$$

Then, Eq. 6 in the main paper can be derived, i.e.:

$$\begin{aligned} q(d_{k-1}|d_k, d_0) = \Big(&\gamma_k\big(P^{Mu}_{\frac{T(d_k-(1-\beta_k)d_{k-1})}{\beta_k}}(d_{k-1}; T, d_{init}, d_k)\big) \\ &\times \big(P^{Mu}_{\frac{S_{k-1}T(d_{k-1}-\overline{\alpha}_{k-1}d_0)}{1-\overline{\alpha}_{k-1}}}(d_{k-1}; S_{k-1}T, d_{init}, d_0)\big)\Big) \end{aligned} \tag{20}$$

where $\gamma_k = \Big(\sum_{d_{k-1}}\big(\big(P^{Mu}_{\frac{T(d_k-(1-\beta_k)d_{k-1})}{\beta_k}}(d_{k-1}; T, d_{init}, d_k)\big) \times \big(P^{Mu}_{\frac{S_{k-1}T(d_{k-1}-\overline{\alpha}_{k-1}d_0)}{1-\overline{\alpha}_{k-1}}}(d_{k-1}; S_{k-1}T, d_{init}, d_0)\big)\big)\Big)^{-1}$.

5.4. Markov Chain Modeling of the Diffusion Process

Here we show that the forward process of our diffusion process can be modeled as a (discrete-time) Markov chain [2]. Specifically, given Eq. 4 of the main paper (i.e., Eq. (4) in Supplementary), at the k-th diffusion step, the step-wise transition from d_{k-1} to d_k can be formulated with a Markov transition matrix M_k. For simplicity, we flatten the vertical slides d_k and d_{k-1} to vectors with length $(H \times 2)$, then M_k is of size $(H \times 2) \times (H \times 2)$, where the element corresponding to the i-th row and j-th column represents the probability of the i-th state transitioning to the j-th state. We also flatten ϵ^{Mu} in Eq. (4) to a vector of length $(H \times 2)$ accordingly.

Then, according to Eq. (4), we have the transition matrix M_k at the k-th diffusion step to be:

$$[M_k]_{ij} = \begin{cases} 1-\beta_k+\beta_k\epsilon_j^{Mu}, & \text{if } i = j \\ \beta_k\epsilon_j^{Mu}, & \text{if } i \neq j \end{cases} \tag{21}$$

where $[M_k]_{ij}$ represents the element in the i-th row and j-th column of the transition matrix, and ϵ_i^{Mu} denotes the i-th entry of the (flattened) vector ϵ^{Mu}.

The above transition matrix is verified by computing $d_{k-1}M_k$ [2], which should produce d_k as formulated in Eq. (4). According to Eq. (21), we can have the j-th element in d_k (denoted as $d_{k,j}$) as:

$$\begin{aligned} d_{k,j} &= (1-\beta_k+\beta_k\epsilon_j^{Mu})d_{k-1,j} + \sum_{i\neq j}(\beta_k\epsilon_j^{Mu}d_{k-1,i}) \\ &= (1-\beta_k)d_{k-1,j} + \beta_k\sum_i(\epsilon_j^{Mu}d_{k-1,i}) \\ &= (1-\beta_k)d_{k-1,j} + \beta_k\epsilon_j^{Mu}, \end{aligned} \tag{22}$$

where $i \in \{1, ..., (H \times 2)\}$. The above transition holds for all $(H \times 2)$ elements in d_k. Thus, we verify that the above defined Markov chain is equivalent to Eq. 4 in the main paper (i.e., Eq. (4) in Supplementary). This shows that, the forward process of our diffusion process can be modeled as a Markov chain.

References

[1] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision (wacv)*, pages 381–389. IEEE, 2018. 1

[2] Lin Geng Foo, Tianjiao Li, Hossein Rahmani, and Jun Liu. Action detection via an image diffusion process. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18351–18361, 2024. 4, 5

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[4] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 1

[5] Hangjie Yuan, Shiwei Zhang, Xiang Wang, Samuel Albanie, Yining Pan, Tao Feng, Jianwen Jiang, Dong Ni, Yingya Zhang, and Deli Zhao. Rlipv2: Fast scaling of relational language-image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21649–21661, 2023. 1

[6] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. 1