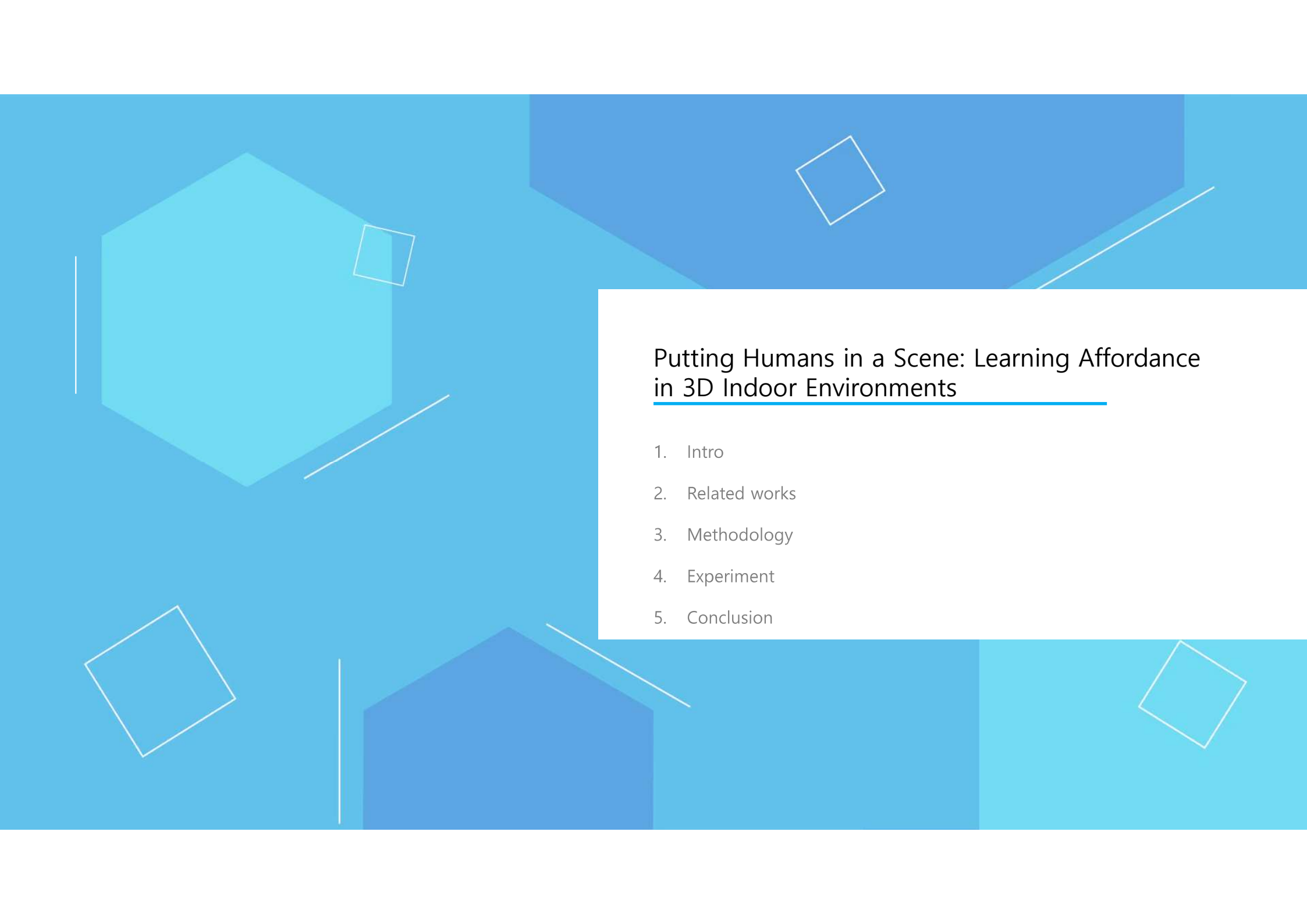


2021

인공지능세미나

김범채
AILab. 연구소장
2021.07.13.



The background of the slide is a solid light blue color. It features several abstract geometric shapes in a darker blue shade, including a large hexagon on the left, a large rectangle on the top right, and a smaller rectangle on the bottom left. There are also several thin white lines and small white squares scattered across the background. A white rectangular box is positioned on the right side of the slide, containing the title and a list of sections.

Putting Humans in a Scene: Learning Affordance in 3D Indoor Environments

1. Intro
2. Related works
3. Methodology
4. Experiment
5. Conclusion

Problem

- We propose an efficient, fully-automatic 3D human pose synthesizer that leverages the pose distributions learned from the 2D world, and the physical feasibility extracted from the 3D world.
- We develop a generative model for 3D affordance prediction which generates plausible human poses with full 3D information, from a single scene image.
- We set a new benchmark for large-scale human-centric affordance prediction on the SUNCG dataset by leveraging the human pose synthesizer and the pose generator.

Related Works

- Scene understanding
- Object functionality reasoning
- Human affordance prediction
- Instance placement in a scene

Purpose: 3D pose generation

- Map 2D pose annotations to 3D poses
- Train pose generation model to generate 3D poses
- Detailed mapping process
- Instance placement in a scene

Purpose: 3D pose generation (where module)

- 2차원 이미지에서 현재 배경화면의 상황을 인식하여 사람이 있을 수 있는 장소를 heat map 형태로 prediction
- 현 상황 및 장소에 맞춰서 가능한 사람의 자세를 인식 함
 - 이를 위해 사람이 가능한 자세를 크게 30가지로 클러스터링 하여 31개의 클래스로 지정함
 - 31번 클래스는 배경화면(Background)
- 2차원 정보로 예측한 사람에 대한 정보에 Depth를 prediction 하여 3차원 정보로 변경함

3. 3D Pose Synthesis

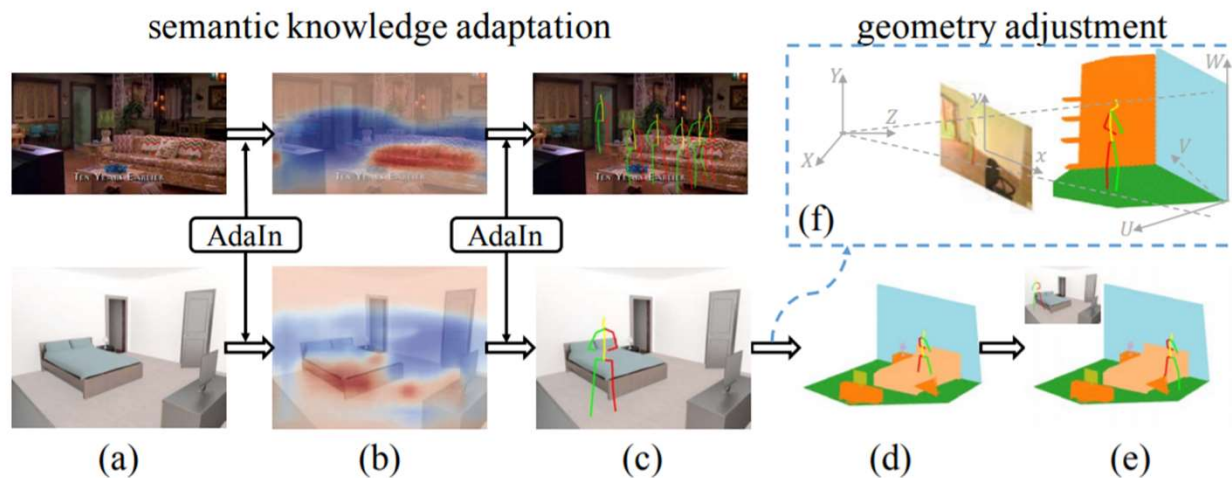
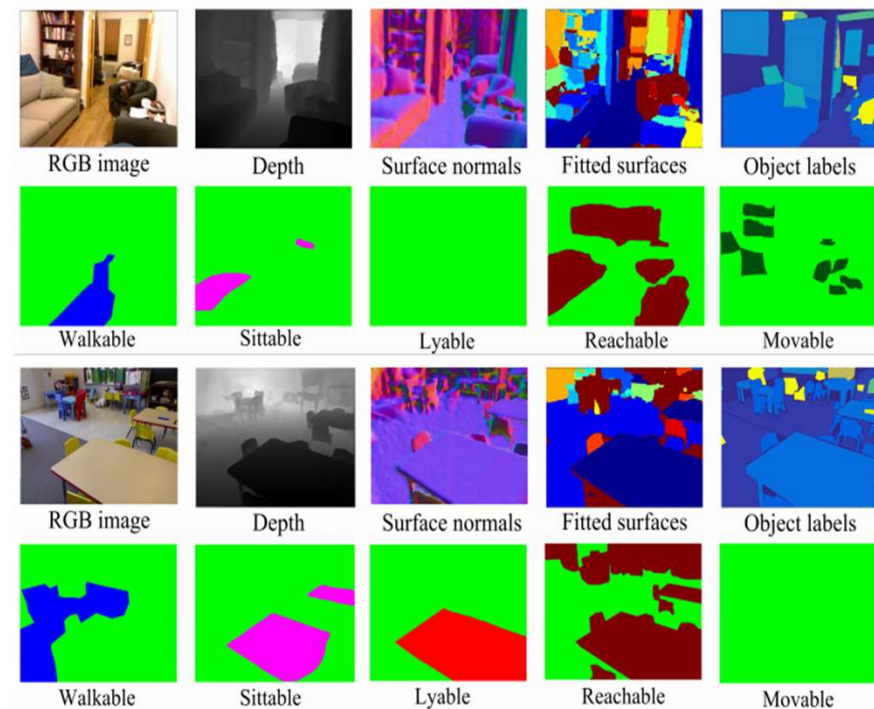


Figure 2. Pose synthesis. (a) Input image. (b) Location heat map. The blue and red regions denote the areas suitable for standing and sitting. (c) Generated pose. (d) Corresponding pose in the voxel. (e) Adjusted pose in voxel. (f) Mapping from image to voxel.



- 2차원 이미지에서 현재 배경화면의 상황을 인식하여 사람이 있을 수 있는 장소를 heat map 형태로 prediction
- AdaIn: 전체 장소에서 가구 등의 불필요한 정보(mean)를 제외하고 사람이 있을 수 있는 Style 정보 추출

4.1. Pose Classification

As a first step, given an input image and a location, we first do a categorical prediction of human poses. But what are the right categories? We use a data-driven vocabulary in our case. Specifically, we use randomly sampled 10K poses from the training videos. We then compute the distances between each pair of poses using procrustes analysis over the 2D joint coordinates, and cluster them into 30 clusters using k-mediod clustering. We visualize the centers of the clusters as Fig. 5.

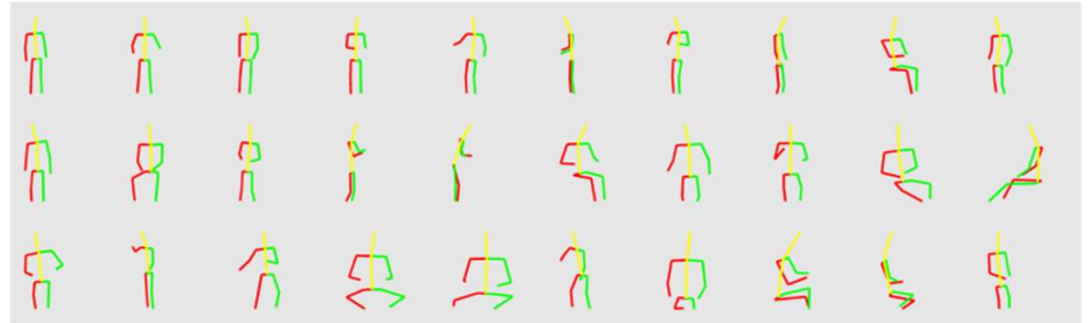


Figure 5. Cluster centers of human poses in sitcom dataset. These clusters are used as pose categories predicted by classification network.

- 이를 위해 사람이 가능한 자세를 크게 30가지로 클러스터링 하여 31개의 클래스로 지정함
 - Procrustes analysis : SVD 를 통해 3D rotation 정보를 지워버림
 - 총 17개의 지점을 기반으로 각 지점간 거리를 기반으로 성능을 측정함

Mean Per Joint Position Error - MPJPE

MPJPE는 모든 관절의 추정 좌표와 정답 좌표의 거리(단위 : mm)를 평균하여 산출되는 지표이다. 이것이 작을수록 정확도가 좋다고 말할 수 있다. Estimated and groundtruth 3D Pose의 root 관절(일반적으로 골반)을 정렬한 후 계산한다. 관절은 또한 root 관절로 정규화된다.

$$\text{MPJPE} = \frac{1}{T} \frac{1}{N} \sum_{t=1}^T \sum_{i=1}^N \| (J_i^{(t)} - J_{root}^{(t)}) - (\hat{J}_i^{(t)} - \hat{J}_{root}^{(t)}) \|_2.$$

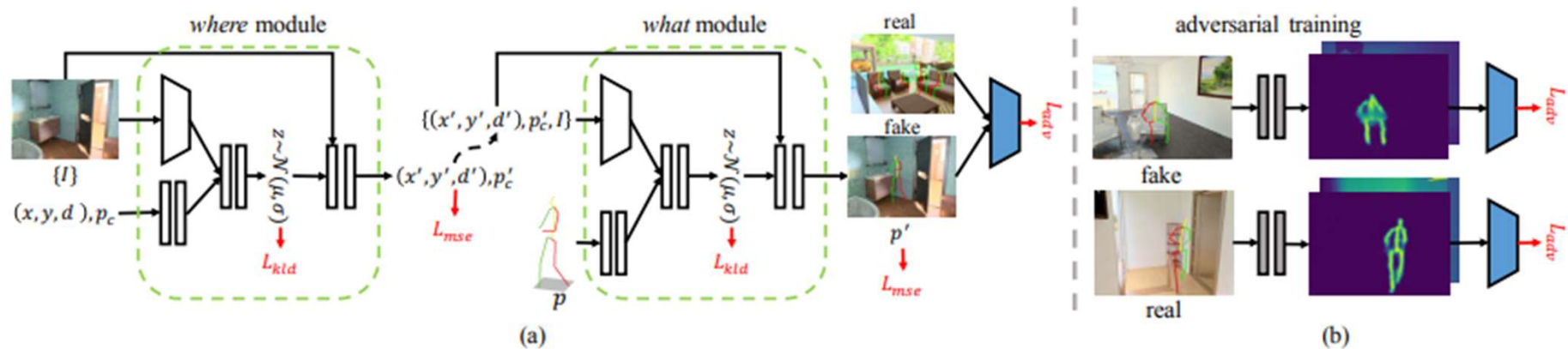


Figure 4. **Overview of the 3D affordance learning model.** (a) Our end-to-end framework consists of a *where* (Section 4.1) and a *what* (Section 4.2) component for pose location and gesture prediction respectively. (b) Detailed illustration of our adversarial training (blue block in (a), detailed in Section 4.3). Grey blocks convert joint coordinates and depth to a “depth heat map”, which are pretrained and fixed when jointly training the *where* and the *what* module. Blocks with same color share parameters.

4.1. The *Where* Module: Pose Locations Prediction

Given a scene image I , we build a *where* VAE to encode pose location in the 3D scene, by simultaneously reconstructing pose pelvis joint coordinates (x, y) and depth d , as well as the most likely pose class p_c at the predicted location. The standard variational equality is represented as:

$$\begin{aligned} & \log P(Y|I) - KL(Q(z|Y, I)||P(z|Y, I)) \\ &= E_{z \sim Q}(\log P(Y|z, I)) - KL(Q(z|Y, I)||P(z|I)) \end{aligned} \quad (3)$$

$P(z|I)$ and $Q(z|Y, I)$ are two normal distributions $\mathcal{N}(0, 1)$ and $\mathcal{N}(\mu(Y, I), \sigma(Y, I))$ and KL represents the Kullback-Leibler divergence.

representations, i.e., each $p_c \in \mathcal{R}^{3 \times 17}$ (each pose contains 17 joints).

The objectives of the *where* module. We use three losses in training the *where* module. First, we minimize the Euclidean distance on the estimated pose class, depth and pelvis coordinates by $L_{mse} = \|Y^* - Y\|$. Second, we minimize the KL-divergence between the estimated distribution Q and the normal distribution $\mathcal{N}(0, 1)$ by $L_{kld} = KL[Q(z|\mu(Y, I), \sigma(Y, I))||\mathcal{N}(0, 1)]$. In addition, to better associate predicted pelvis joint depth and pixel coordinates, we minimize the Euclidean distance between ground truth and predicted pelvis coordinates under the world coordinate system using camera parameters for each scene. We refer this loss as *geometry loss* and represent it as $L_{geo} = \|M_e M_i[x^*, y^*, d^*] - M_e M_i[x, y, d]\|$, where M_e and M_i are camera extrinsic and intrinsic matrices. Our final objective is:

$$L = \lambda_{mse} L_{mse} + \lambda_{kld} L_{kld} + \lambda_{geo} L_{geo}, \quad (4)$$

where λ_{mse} , λ_{kld} , λ_{geo} are the weights that balance the

4.2. The *What* Module: Pose Gestures Prediction

The *what* module takes pelvis joint coordinates (x, y) , depth d and pose class p_c predicted by the *where* module as well as a scene image I as inputs, and learns to predict coordinates and depth of each joint in $p \in \mathbb{R}^{3 \times 17}$, so that the generated pose p can align well with its surrounding context. In other words, the *what* module needs to understand the scene context, and be able to sample poses conditioned on it. Similarly, we model the pose appearance distribution with a conditional VAE, which is represented as:

$$\begin{aligned} & \log(P(S|R, I)) - KL(Q(z|S, R, I)||P(z|S, R, I)) \quad (5) \\ & = E_{z \sim Q}(\log P(P|z, R, I)) - KL(Q(z|S, R, I)||P(z|R, I)), \end{aligned}$$

where S represents the coordinates and depth $\{x, y, d\}$ for each joint, R denotes $\{x, y, d, p_c\}$ predicted by the *where* module. Other symbols follow those in Section 4.1.