AutoLink: Self-supervised Learning of Human Skeletons and Object Outlines by Linking Keypoints

Xingzhe He **Bastian Wandt** **Helge Rhodin**
University of British Columbia
{xingzhe, wandt, rhodin}@cs.ubc.ca

Abstract

Structured representations such as keypoints are widely used in pose transfer, conditional image generation, animation, and 3D reconstruction. However, their supervised learning requires expensive annotation for each target domain. We propose a self-supervised method that learns to disentangle object structure from the appearance with a graph of 2D keypoints linked by straight edges. Both the keypoint location and their pairwise edge weights are learned, given only a collection of images depicting the same object class. The resulting graph is interpretable, for example, AutoLink recovers the human skeleton topology when applied to images showing people. Our key ingredients are i) an encoder that predicts keypoint locations in an input image, ii) a shared graph as a latent variable that links the same pairs of keypoints in every image, iii) an intermediate edge map that combines the latent graph edge weights and keypoint locations in a soft, differentiable manner, and iv) an inpainting objective on randomly masked images. Although simpler, AutoLink outperforms existing self-supervised methods on the established keypoint and pose estimation benchmarks and paves the way for structure-conditioned generative models on more diverse datasets. Project website: https://xingzhehe.github.io/autolink/.

1 Introduction

Object structure representations are widely used in modern computer graphics and computer vision techniques, including keypoints for image generation [67, 68, 94] and skeletons for 3D reconstruction [26, 41, 121, 83, 99]. However, the structure is usually supervised on large annotated datasets [61, 2, 3, 71] or via hand-crafted parametric models [65, 84, 111, 80, 5, 59]. Neither approach generalizes well to new domains and both require additional manual annotation whenever more detail is needed [88].

Our goal is to reconstruct the keypoint locations of an object by learning from an unlabelled image collection, thereby sidestepping the generalization problem. Our key idea is to leverage that the same object shares the same topology by introducing an explicit graph that links the same pairs of keypoints in all instances. By contrast, existing self-supervised keypoint learning methods model objects as a set of independent parts. Their consistency over different instances of the same object is encouraged by either enforcing parts to follow hand-crafted image transformations [102, 124, 42, 66, 37, 62] or by adding implicit bias in the network architecture that encodes such spatial equivariances [35, 34]. Only [43, 93] use an explicit skeleton representation, but both require predefined topology, rely on video input, and [43] is trained in a CycleGAN setting that still requires manually labeled examples.

We propose a simple yet effective method to learn both the keypoints and their links without supervision in terms of a sparse graph serving two purposes. First, the graph acts as a bottleneck that can only store structural information disentangled from appearance. Second, it forms a constraint that associates observations across training images. We enforce the same topology across instances of the same class by learning a shared graph with a single set of edge weights. In the absence of labels,

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

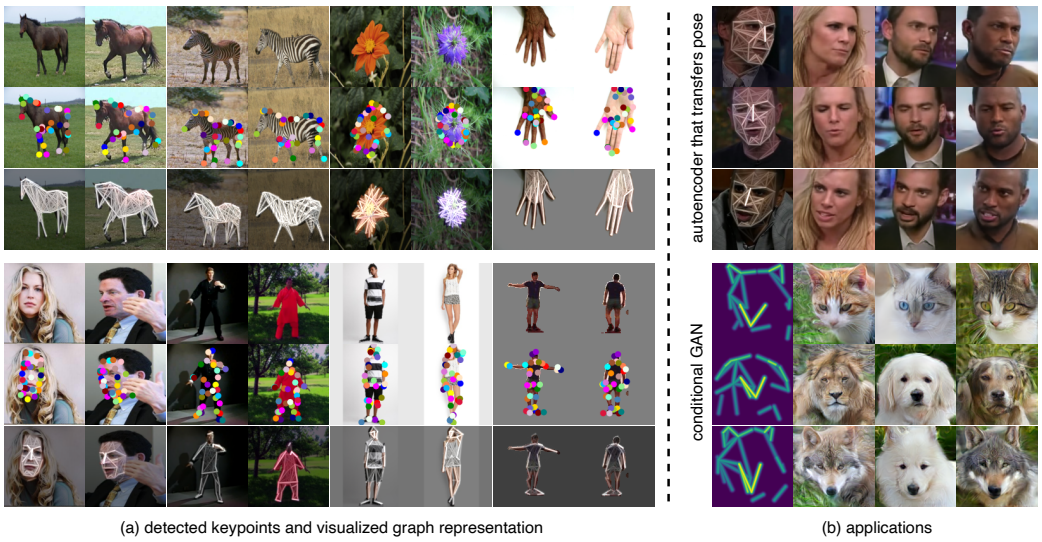

Figure 1: **Teaser.** (a) AutoLink applies to diverse collections of images and automatically yields keypoints linked to a graph without ground truth. It recovers animal and human poses and object shapes in settings where previous methods struggle, including cluttered backgrounds, structured stripe textures, articulated fingers, and detailed faces. (b) Example applications are conditional image generation with autoencoders (top) and GANs (bottom) that are driven by the learned keypoints.

we train AutoLink with only an autoencoder reconstruction objective. Since the graph bottleneck should not model appearance, we additionally feed the decoder with the input image after masking the majority of its pixels. In turn, it is important for the disentanglement that the heavily masked image contains appearance without leaking structural information. This is the case as inpainting methods are unable to infer the original image precisely from only sparse pixel colors [127, 122] unless conditioning on structure representations, such as edge maps [110, 75, 58, 56, 31]. Therefore, by forcing the autoencoder to reconstruct the original image, the detector converges to generate representative structures of images.

We demonstrate on 4 benchmarks that the trained detector has a significantly improved keypoint localization accuracy and on 6 additional datasets that it applies to a broader set of images spanning portraits, persons, animals, hands, and flowers, which we attribute to the explicit modeling of links in the graph. Figure 1 shows the diverse set of image domains it applies to, including challenging textures and uncontrolled background, how both skeleton representations as well as object outlines are learned by varying the number of keypoints, and exemplifies applications to controlled image generation.

Ethics - Risks. The estimated keypoints and edges could be abused for deep fakes as the driving signal for generative models or for unwanted surveillance applications. However, our method works towards improved generality, including objects and animals, and does not improve upon supervised models that already exist in high detail for humans. **Benefits.** Since our method is entirely self-supervised, it can be applied to a diverse set of persons, objects, animals, or situations that have not yet been labeled.

2 Related Work

Most representation learning methods focus on generic feature vectors for entire images to initialize deep networks for improved object classification [108, 11, 79, 32, 33]. By contrast, our method introduces explicit object structure. We review the most related approaches in the following.

Self-supervised Keypoints Detection. The most common idea to discover keypoints in an unsupervised manner is to rely on the notion that keypoints move as the image changes. Various constraints have been used to enforce that the keypoints follow a known transformation, including view changes in a multi-view recording [101, 86, 87] or the natural motion in videos [23, 95, 55, 74, 20, 53, 43].

When only single images are available, artificial image deformation is applied, either from randomized [102, 124, 42, 66] or learned [107, 116] transformations within a pre-defined deformation space. However, learned keypoints may model the background [124, 95] and struggle with large pose variation [37] as image deformations do not separate foreground from background, which are usually tuned for each dataset and bound to be small. Most models leverage multi-branch network architectures to encode the structure and appearance separately and utilize multiple losses that need to be balanced. By contrast, we do not apply artificial transformations, use a single branch, and a single loss which eases and stabilizes training. To overcome the need for artificial image deformation, He et al. [34, 35] exploit GANs to generate images along with corresponding keypoints and later use them to train a detector. However, this leads to even more complex network architectures and comes with instabilities in GAN training, limiting their applicability to complex objects like human bodies.

Skeleton Representations. Bone maps representing the keypoints connectivity as affinity fields [9] or via explicit offsets [82] are used in supervised human, animal, and object pose estimation. We use a similar edge map representation but learn both the location and linking from scratch without annotations. In the weakly-supervised setting, Jakab et al. [43] exploit CycleGAN [128] to translate between image and edge maps. The graph connectivity is predefined to the human skeleton and edges are supervised by a large dataset of unpaired ground truth object edges, which can come from a different dataset but are manually annotated. Schmidtke et al. [93] overcome the manual labeling by deforming a template skeleton. However, they both require the known connectivity of the keypoints and videos for training while ours learns both the keypoints and connectivity from a collection of single images. Noguchi et al. [78] generate a skeleton heuristically by linking the centers of part-wise learned Signed Distance Fields [70]. However, they require videos without the background of the same object, and the learned skeleton does not generalize to other objects of the same class.

Object Sketch Learning. Sketches are made of strokes drawn by a pen. It is a concise and abstract representation, which can be used in object recognition [119, 115] and image retrieval [104, 120, 114]. There are two common sketch representations used in neural models [113]: black-white raster images [105, 91, 118], often used for image-to-image translation [40, 128, 81, 48] and sequences of points (pen coordinates) [30, 24, 90], which is usually used by recurrent generation models [30, 12, 8, 47, 19, 28]. This graph representation is similar to ours. However, instead of learning to mimic human drawings, ours directly predicts both the keypoints and their connectivity on real natural images.

Structure-enhanced Image Inpainting. When key parts are missing in an image, e.g., eyes on faces or arms of humans, it is hard for inpainting networks to imagine the content accurately from scratch. Therefore, additional structural cues are detected to guide the subsequent image generation. The cues can be supervised segmentation masks [126, 60, 31, 98], foreground contours [110], and landmarks [56, 123, 117], or automatically extracted edges [75, 58, 45, 112, 7] and low-frequency image components [106, 85]. Our reconstruction objective can be seen as such two-stage inpainting, but self-supervised and with the image edges replaced with the learned graph edge representation.

Self-supervised Foreground Segmentation. Traditional methods use color [129], contrast [13], and hand-crafted features [44] to cluster foreground pixels. A recent trend is exploiting inpainting techniques to segment the foreground. Chen et al. [10] and Arandjelović et al. [4] use a GAN to inpaint the background at the predicted segmentation mask, assuming that the object texture can be changed without changing the data distribution due to the independence of foreground and background. Savarese et al. [92] propose to minimize the mutual information between the inpainted foreground and background. When the object is small compared to the background, an additional object detection module [18] and temporal [51] or multi-view [52] information is required. Different from these previous methods, we utilize a form of inpainting to learn sparse keypoints instead of segmentation. Our learned edges form a sparse foreground shape, but further extensions would be necessary to transfer from the edge maps to the boundary-aligned foreground segmentation.

3 Method

We leverage that the objects in the dataset share the same topology and can be represented as a graph that connects keypoints by a shared set of edges. To learn the keypoints and edges, we design an autoencoder that aims to accurately reconstruct the input image, with the graph as the intermediate representation. To encode the input image into a graph, we detect the keypoints and create an edge

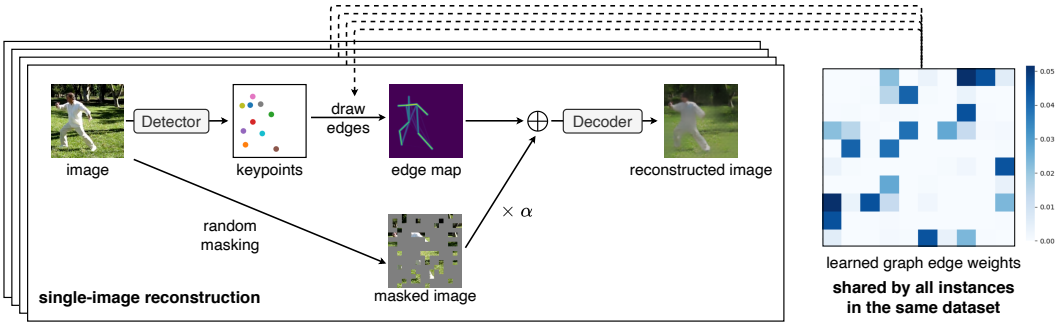

Figure 2: **Overview.** Given an image, we detect keypoints and draw differentiable edges between keypoints according to the learned graph edge weights that is visualized as a color matrix. The method is self-supervised in that the latent edge map and keypoints are learned by reconstructing the masked input images. Note that keypoints are image specific and edge maps are shared.

heatmap based on learnable edge weights. To mostly obtain appearance information, we mask out the majority of the image, which randomizes the structure information and reduces the remainder to a very low level. The edge heatmap is combined with the masked image to reconstruct the original image. Since the missing structure is important to reconstructing the original image, the network is forced to learn the structure of the object in a self-supervised manner. Figure 2 shows an overview of our method.

Formally, given an image $\mathbf{I} \in \mathbb{R}^{H \times W \times 3}$ with height H and width W we aim to learn a set of keypoints $\{\mathbf{k}_i\}_{i=1}^{K}$, where $\mathbf{k}_i \in [-1, 1] \times [-1, 1] \subset \mathbb{R}^2$ is the normalized keypoint coordinate, and K is the number of keypoints. We use a ResNet with upsampling [109] to detect keypoints. Afterward, we draw a differentiable edge [73] between each pair of keypoints (details below). The edge map $\mathbf{S} \in \mathbb{R}^{H \times W}$ is concatenated along the channel dimension with the randomly masked image $\mathbf{I}_m \in \mathbb{R}^{H \times W \times 3}$ and fed into a UNet [89] to obtain the reconstructed image $\mathbf{I}'$. The detailed network architectures can be found in Appendix D.

3.1 Image Structure Representation

In this section, we introduce the generation of keypoints and the edge map from the image. Let $\mathbf{H} \in \mathbb{R}^{H \times W \times K}$ be the K heatmaps generated by a ResNet with upsampling [109] from the image $\mathbf{I}$. The keypoint $\mathbf{k}_i$ is calculated by the differentiable soft-argmax function,

$$\mathbf{k}_i = \sum_{\mathbf{p}} \frac{\exp(\mathbf{H}(\mathbf{p}))}{\sum_{\mathbf{p}}(\exp \mathbf{H}(\mathbf{p}))} \mathbf{p}, \tag{1}$$

where $\mathbf{p} \in [-1, 1] \times [-1, 1]$ is the normalized pixel coordinates.

Given two keypoints $\mathbf{k}_i, \mathbf{k}_j$, we draw a differentiable edge map $\mathbf{S}_{ij}$, where values are 1 on the edge linked by the two keypoints and decrease exponentially based on the distance to the line. Formally, the edge map $\mathbf{S}_{ij}$ is a Gaussian extended along the line [73], defined as

$$\mathbf{S}_{ij}(\mathbf{p}) = \exp\left(d_{ij}^2(\mathbf{p})/\sigma^2\right), \tag{2}$$

where σ is a hyperparameter controlling the thickness of the edge, and $d_{ij}(\mathbf{p})$ is the L_2 distance between the pixel $\mathbf{p}$ and the edge drawn by keypoints $\mathbf{k}_i$ and $\mathbf{k}_j$,

$$\mathbf{d}_{ij}(\mathbf{p}) = \begin{cases} \|\mathbf{p} - \mathbf{k}_i\|_2 & \text{if } t \leq 0, \\ \|\mathbf{p} - (\mathbf{k}_i + t\mathbf{k}_j)\|_2 & \text{if } 0 < t < 1, \\ \|\mathbf{p} - \mathbf{k}_j\|_2 & \text{if } t \geq 1, \end{cases} \quad \text{where} \quad t = \frac{(\mathbf{p} - \mathbf{k}_i) \cdot (\mathbf{k}_j - \mathbf{k}_i)}{\|\mathbf{k}_i - \mathbf{k}_j\|_2^2}. \tag{3}$$

We assign a weight $w_{ij} > 0$ to each edge, which is enforced to be positive by SoftPlus [22]. This weight is learned during training and shared across all object instances in a dataset. Finally, we take the maximum at each pixel of the heatmaps to obtain the final edge map $\mathbf{S} \in \mathbb{R}^{H \times W}$,

$$\mathbf{S}(\mathbf{p}) = \max_{ij} w_{ij} \mathbf{S}_{ij}(\mathbf{p}). \tag{4}$$

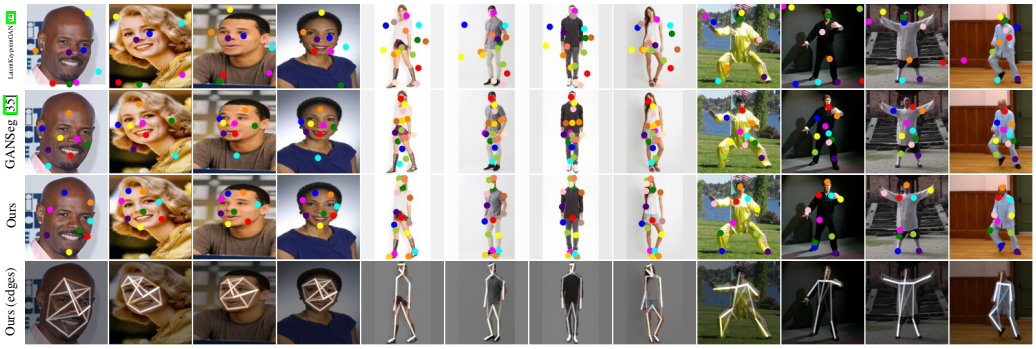

Figure 3: **Qualitative comparison on detected keypoints**. Our model is more robust on wild face poses, and depicts more details on human bodies compared to [34] and [35]. For example, the feet poses in the middle four images are clearly detected.

Taking the maximum at each pixel avoids the entanglement of the edge weights and the convolution kernel weights, which is further explained in Section 4.4.

3.2 Image Reconstruction

The masked image $\mathbf{I}_m$ is generated by first uniformly dividing the image $\mathbf{I}$ into a 16×16 grid, and randomly masking out 80% of the grid cells, similar to [33]. We concatenate the masked image with the edge map and feed it into a UNet decoder [89] to reconstruct the original image,

$$\mathbf{I}' = \text{Decoder}(\alpha \mathbf{I}_m \oplus \mathbf{S}) \tag{5}$$

where $\oplus$ means concatenation along the channel dimension and α is a learnable parameter that compensates for the change of the edge weight magnitude during training. α is initialized to 1. We found this parameter to be helpful in training stability. Different to [33], we condition on an edge map. Different to [43], we have no ground truth for the edge map. Our edge map is an unobserved latent variable. Thus we only minimize the difference of the original image and the reconstructed image by the perceptual loss [46],

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \|\Gamma(I_i) - \Gamma(I_i')\|_2^2 \tag{6}$$

where N is the number of examples and Γ is the feature extractor. The perceptual loss is believed to measure the structure similarity [46, 27, 21], and leads to more robust training [42, 43].

3.3 Implementation Details

We use the Adam optimizer [54] with a learning rate of 10^{-4} with $\beta_1 = 0.9$, $\beta_2 = 0.99$. The batch size is 64. We train for 20k iterations. It takes 3 hours to train on a single V100 GPU. All images are resized to 128×128. The learning rate for the edge weights is multiplied by 512 due to the small gradient of SoftPlus [22] when the value is close to 0. To show the robustness of our model, we report all experiments on the sampling strategy of masking 80% of the 16×16 patches. We perform experiments with the same edge thickness of $\sigma^2 = 5e - 5$ for all benchmark datasets. We train 10 times and report the mean and the standard deviation of the evaluation metrics. Although it already outperforms other work in most experiments, we also tune thicknesses to each individual dataset, as others did for their hyperparameters, which further improves the results. The tuned thicknesses can be found in Appendix B. The only other hyperparameter is the number of keypoints, which we set to that of the established benchmarks for quantitative comparisons, ranging from 4 to 32 points.

4 Experiments

In this section, we compare our results to the related methods, showing that our model is simple yet effective. Besides, we perform a number of ablation studies on hyperparameters and algorithm variants, exhibiting the robustness of our model and justifying the necessity of every model component.

Table 1: **Landmark detection on CelebA**. The metric is the landmark regression (without bias) error in terms of L_2 distance normalized by inter-ocular distance (lower is better). While all methods perform well on aligned CelebA, ours is more robust on Wild CelebA. The sign $\star$ means being reported by [37] and † means being reported by [62].

Method	Type	Aligned (K=10)	Wild (K=4)	Wild (K=8)
DFF [17] by [37]	Part Segmentation	-	-	31.30% $\star$
SCOPS [37] (w/o saliency)	Part Segmentation	-	46.62%	22.11%
SCOPS [37] (w/ saliency)	Part Segmentation	-	21.76%	15.01%
Liu et al. [62]	Part Segmentation	-	15.39%	12.26%
Huang et al. [36] (w/ detailed label)	Part Segmentation	-	-	8.40%
GANSeg [35]	Part Segmentation	3.98%	**12.26**%	**6.18**%
Thewlis et al. [102]	Landmark	7.95%	-	31.30% $\star$
Zhang et al. [124]	Landmark	3.46%	-	40.82% $\star$
LatentKeypointGAN [34]	Landmark	5.85%	25.81%	21.90%
Lorenz et al. [66]	Landmark	3.24%	15.49% †	11.41% †
IMM [42]	Landmark	**3.19**%	19.42% †	8.74% †
LatentKeypointGAN-tuned [34]	Landmark	3.31%	12.10%	5.63%
Ours (general)	Landmark	3.92±0.69%	7.72±0.47%	5.66±0.29%
Ours (thickness-tuned)	Landmark	3.54%	**6.11**%	**5.24**%

Table 2: **Landmark detection on Human Body**. Our model outperforms all the other unsupervised baselines. The metric for each dataset follows the corresponding description in the text. The sign † means being reported by [97] and the sign $\star$ means being reported by [6]. The number of keypoints is $K = 16$ for Human3.6m and DeepFashion and $K = 10$ for Taichi.

Method	Supervision	Human3.6m ↓	DeepFashion ↑	Taichi ↓
Jakab et al. [43]	video & unpaired ground truth	2.73	-	-
Newell et al. [43]	paired ground truth	**2.16**	-	-
DFF [17]	testing dataset	-	-	494.48 †
SCOPS [37]	saliency maps	-	-	411.38 †
Siarohin et al. [97]	videos	-	-	389.78
Zhang et al. [125]	videos	-	-	**343.67**
Zhang et al. [124]	videos	4.14	-	-
Schmidtke et al. [93]	video & T-pose template	3.31	-	-
Sun et al. [100]	videos	**2.53**±0.06	-	-
Thewlis et al. [102]	unsupervised	7.51	-	-
Zhang et al. [124]	unsupervised	4.91	-	-
LatentKeypointGAN [34]	unsupervised	-	49%	437.69
Lorenz et al. [66]	unsupervised	2.79	57% $\star$	-
GANSeg [35]	unsupervised	-	59%	417.17
Ours (general)	unsupervised	2.81±0.07	65±1.2%	337.50±25.08
Ours (thickness-tuned)	unsupervised	**2.76**	**66**%	**316.10**

4.1 Datasets and Evaluation Metrics

CelebA-aligned [63] contains 200k celebrity faces aligned in center. We follow [102] splitting it into three subsets: CelebA training set without MAFL (160k images), MAFL training set (19k), MAFL test set (1k). We train our network on the CelebA training set without MAFL. To quantitatively evaluate the consistency of our predicted keypoints, we follow [102] training a linear regression without bias from our detected keypoints to the ground truth keypoints on the MAFL training set and reporting the mean L_2 error normalized by inter-ocular distance on the MAFL test set.

CelebA-in-the-wild [63] contains celebrity faces in unconstrained conditions. We follow [37] and first split it into three subsets as for CelebA-aligned, and then remove the images where a face covers less than 30% of the area, which results in 45,609 images for model training, 5,379 with keypoint labels for regression, and 283 for testing. The evaluation metric is the same as CelebA-aligned.

Human3.6m [39] contains human activity videos in static backgrounds. We follow [124] considering six activities (direction, discussion, posing, waiting, greeting, walking), and using subjects 1, 5, 6, 7, 8, 9 for training and 11 for testing. This results in 796,648 images for training and 87,975 images for testing. The evaluation metric is the regressed (without bias) mean L_2 error normalized by the image size. We remove the background as in [124, 66] to make a fair comparison to others. To underline the robustness against structured backgrounds we also report the numbers including background.

Table 3: **Landmark detection on CUB Birds**. Our model outperforms most other baselines and achieves comparable results with the ones using ground truth segmentation masks. The metric is the landmark regression (without bias) error of L_2 distance normalized by the image size (lower is better). A star $\star$ means being reported by [15], † means being reported by [37], and ‡ means tested by us with their official code; all other numbers are taken from the respective papers. The number of keypoints is $K = 10$ for CUB-aligned and $K = 4$ for CUB-001, CUB-002, CUB-003, and CUB-all.

Method	Supervision	CUB-aligned ↓	CUB-001 ↓	CUB-002 ↓	CUB-003 ↓	CUB-all ↓
SCOPS [37]	GT silhouette	-	18.3 $\star$	17.7 $\star$	17.0 $\star$	12.6 $\star$
Choudhury et al. [15]	GT silhouette	-	**11.3**	**15.0**	**10.6**	**9.2**
DFF [17]	testing dataset	-	22.4†	21.6†	22.0†	-
SCOPS [37]	saliency maps	-	**18.5**	**18.8**	**21.1**	-
Lorenz et al. [66]	unsupervised	3.91	-	-	-	-
ULD [124,102]	unsupervised	-	30.1†	29.4†	28.2†	-
Zhang et al. [124]	unsupervised	5.36	26.9‡	27.6‡	27.1‡	22.4‡
LatentKeypointGAN [34]	unsupervised	5.21‡	22.6‡	29.1‡	21.2‡	14.7‡
GANSeg [35]	unsupervised	**3.23**	22.1‡	22.3‡	21.5‡	12.1‡
Ours (general)	unsupervised	4.15 ± 0.24	20.6 ± 0.54	20.3 ± 0.96	19.7 ± 0.91	11.6 ± 0.33
Ours (thickness-tuned)	unsupervised	3.51	**20.2**	**19.2**	**18.5**	**11.3**

DeepFashion [64] contains 53k in-shop clothes images. We follow [66] only keeping the full-body images. We use 10604 images for training and 1179 images for testing as in [6]. We use the keypoints generated by AlphaPose [25] as the ground truth. The evaluation metric is Percentage of Correct Keypoints of d=6 pixels in resolution 256×256.

Taichi [96] contains 3049 training videos and 285 test videos of people performing Tai-Chi, with the various appearance of foreground and background. We follow [97] using 5000 and 300 images (not contained in training data) for training a linear regression and for testing, respectively. The evaluation metric, mean average error (MAE), is calculated as the sum of the L_2 error in resolution 256×256.

CUB-200-2011 [103] consists of 11,788 images of birds. We follow two established protocols [66, 15] to evaluate our method: 1) Images are cropped based on the bird landmarks, aligned to face to the left [66], and seabirds are removed; 2) Birds are cropped based on the given bounding box and the train/val/test split of [15] is used. In both cases, the evaluation metric is the regressed (without bias) mean L_2 error normalized by the cropped image size.

Flower [77], **11k Hands** [1], **Horses** [128], and **Zebras** [128] are used for qualitative experiments. **VoxCeleb2** [16] and **AFHQ** [14] are used for pose transfer and conditional image generation, respectively. Horses and Zebras are extracted from the CycleGAN dataset [128] by removing the images with multiple horses and aligning them to face left. Note that the horses and zebra are trained separately, yet the model learns similar structures. The train/test split of Flower follows [10]. All the other datasets follow the train/test split specified by the dataset.

4.2 Qualitative Analysis

We qualitatively compare our detected keypoints with other methods and show the examples of the learned edges in Figure 3. For visualization purposes, we scale the edge weights to obtain visible edges. We use the same number of keypoints as the previous method [35] for a fair comparison, which are 8 for CelebA-in-the-Wild, 16 for DeepFashion, and 10 for Taichi. We will discuss more on the choice of the number of keypoints in Ablation Study 4.4. As shown in Figure 3, our model not only detects consistent keypoints but also learns reasonable edges, such as human skeletons in DeepFashion and Taichi. For example, the feet are clearly connected with the corresponding knees, and there is no edge between the left and right hands. We show 105 images with detected keypoints and visualized graph structure for each dataset in Figure 9-23 in the Appendix, demonstrating that our model works on various classes of objects of diverse appearance and complex backgrounds.

4.3 Quantitative Analysis

We compare the keypoint detection results with other methods in Table 1 (CelebA), Table 2 (Human3.6, DeepFashion, Taichi), and Table 3 (CUB). Our simple model outperforms all other unsupervised methods in all benchmarks, except [37] on three CUB subsets, which however requires saliency maps, and for the most constrained setting CelebA-aligned and CUB-aligned where all methods perform well and the results are comparable. The results on CelebA in Table 1 confirms that our

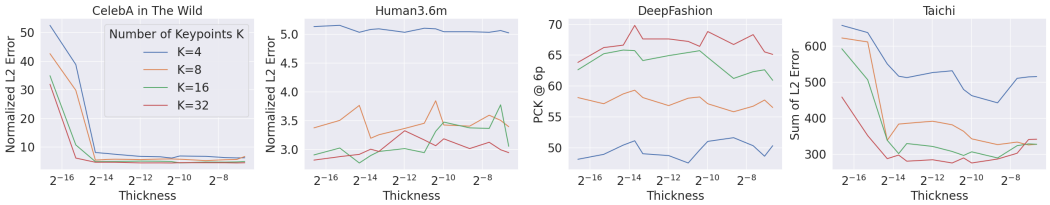

Figure 4: **Ablation tests on the number of keypoints and edge thickness.** While the model shows better performance with more keypoints, it is robust to the edge thickness.

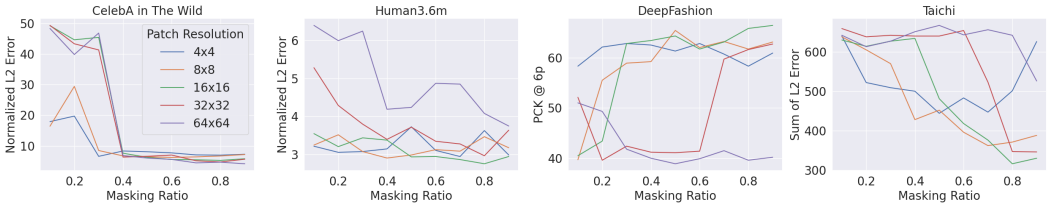

Figure 5: **Ablation tests on masking strategy.** Overall, the performance increases as the mask ratio increases. Too small or too large patch sizes can decrease the performance. Empirically, masking 80% of the 16×16 patches is a golden rule.

model is more robust to poses in the wild. Since self-supervised part segmentation methods are usually more robust on wild faces [62, 37, 35], we also include them for comparison, demonstrating the robustness of our model over existing baselines. The results on Human3.6m and DeepFashion in Table 2 show the capability of our model to detect keypoints on human bodies of either similar or diverse appearances. The result on Taichi in Table 2 demonstrates the general applicability of our model to human bodies of diverse poses in complex backgrounds.

4.4 Ablation Tests

In this section, we analyze the hyperparameters and demonstrate the robustness of our model. We also discuss the possible variants of our model and show the superiority of our design.

Number of Keypoints & Thickness. We show ablation test results on the different numbers of keypoints and edge thicknesses in Figure 4. The exact numbers can be found in Table 5 in Appendix B. Our model shows very strong robustness to edge thickness. The accuracy remains state-of-the-art while the thickness of the edges varies by multiple orders of magnitude. On the other hand, with the increasing number of keypoints the accuracy increases. This is expected since more keypoints are able to capture structure in more detail, as shown in Figure 6. Yet, some other methods fail for a large number of keypoints [35].

Masking Strategy. In our standard setting, the image is divided into 16×16 patches and 80% of the patches are randomly masked. We investigate how the patch size and masking ratio affect the model performance. Figure 5 shows that a too low masking ratio enables the network to directly infer the structure from the masked image which is undesired in our case. In these cases, the network would not choose to infer a set of compact keypoints from the original image. Figure 5 illustrates that the patch size cannot be too small (4×4) or too large (64×64). Although in some cases, such as CelebA-in-the-Wild, a different masking strategy gives better results (4.14 vs 5.24), we choose to report a unified strategy that we mask 80% of 16×16 patches for simplicity and conciseness.

Variants of Edge Heatmap Generation. Besides the heatmap generation described in Method 3, we test four more ways to generate the edge maps: 1) we define the thickness as a globally learnable parameter; 2) we learn each edge thickness independently as a parameter; 3) we treat each edge heatmap as an independent channel of the feature map, instead of making them a single channel heatmap with Equation 4; 4) instead of using the edge heatmap, we generate Gaussian heatmaps for the keypoints and use Equation 4 to combine them in a single channel heatmap. The results are listed in Table 4. Overall, these variants have worse performance. Although in some cases, the model

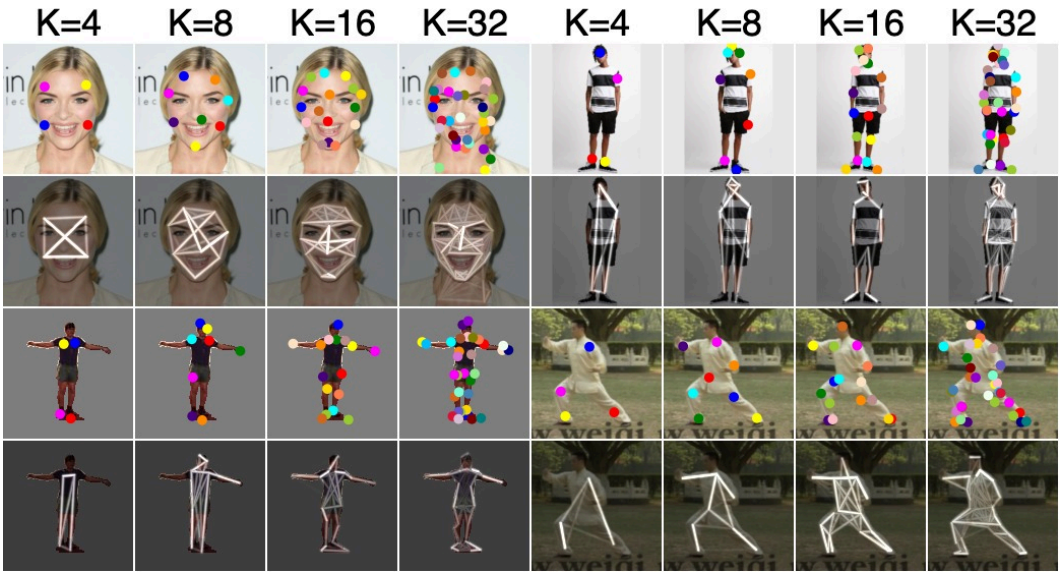

Figure 6: **Examples of different numbers of detected keypoints.** With very few keypoints, the model only models a very basic shape, such as the box on the face, and sometimes cannot fully capture the structure. For example, with K=4, only legs are modeled on humans. With abundant keypoints, it is able to model details.

Table 4: **Ablation tests on variants of edge heatmap generation**. The original design is proved to be the most robust one. Although in some cases it is not optimal, the difference is almost trivial.

Model	CelebA in The Wild ↓	Human3.6m ↓	DeepFashion ↑	Taichi ↓
original model	**5.24**%	**2.76**%	65.8%	316
fixed α	6.39%	2.87%	**66.0**%	374
shared learnable thickness	6.12%	3.25%	49.1%	425
independent learnable thickness	5.94%	3.73%	50.2%	**311**
edge-specific heatmap	5.65%	3.83%	65.1%	407
only using keypoints without edges	6.55%	3.58%	52.9%	722

has slightly better results on specific datasets, the performance boost does not hold in general. We observe that with only keypoints without edges, the model may degenerate, as shown in Figure 7a. Interestingly, assigning each edge a different channel performs worse than simply combining all edges into a single channel. We believe it is caused by entangling the edge weights with the convolution kernel weights. As visualized in Figure 7b, there exist dummy edges that do not model the object structure. In addition, we tried to remove the learnable α in Equation 5, fixing $\alpha = 1$, but the overall performance decreases as shown in Table 4.

Does Texture Matter? We trained two networks on horses and zebras separately. As shown in Figure 1a, the horse and the zebra share similar shape structures but only one is textured. The striped texture not having a significant impact on the learned structure shows that our model primarily learns the structure instead of texture features.

What if the model is trained on images with a structured background? We tested on Human3.6m with a background, where all images are taken in a single room. The error is 5.02. As shown in Figure 7c, our model captures the entrance in the background. It is expected since we assume the foreground object is structured. If we apply spectral clustering [76] on the learned graph, the keypoints are clearly divided into two clusters, one for the room and one for the person.

5 Limitations and Future Work

If the background is highly structured, the keypoints will appear on the background. Yet, we showed an avenue for future work, as already a simple graph clustering could separate the object from

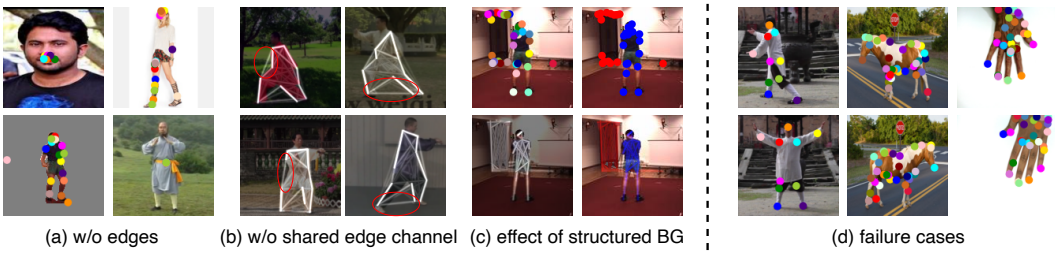

Figure 7: (a) If we do not model the edges, the model may degenerate. (b) If we give different edges a different channel in the feature map, the model would give dummy edges. (c) If the model is trained on the dataset with structured background, the background would be modeled. However, the keypoints can be separated into two sets by spectral clustering. (d) failure case: left) the model cannot model the occlusion well; right) the model has left and right ambiguity.

the background on the Human3.6M dataset. Similar to the previous 2D self-supervised methods [66, 97, 35], our model cannot model occlusion well. We show in Figure 7d left that the occluded right arm becomes the back when the person turns to the left. In addition, as for all other methods, the model cannot distinguish the left and right sides of the objects as shown in Figure 7d middle and right. We believe it is necessary to model the structure in 3D to solve these problems.

6 Conclusion

We presented a simple approach for learning a spatial graph representation from unlabelled image collections by reconstructing masked images. The crucial part is our learnable graph design that models the relationship between different keypoints. It is simpler than existing alternatives and opens up a path for image understanding, image editing, and learning 3D models from 2D images.

Acknowledgement

This work was supported by the Compute Canada GPU servers, and a Huawei-UBC Joint Lab project.

References

[1] M. Afifi. 11k hands: gender recognition and biometric identification using a large dataset of hand images. *Multimedia Tools and Applications*, 2019. doi: 10.1007/s11042-019-7424-8. URL https://doi.org/10.1007/s11042-019-7424-8.

[2] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[3] M. Andriluka, U. Iqbal, E. Insafutdinov, L. Pishchulin, A. Milan, J. Gall, and S. B. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018.

[4] R. Arandjelović and A. Zisserman. Object discovery with a copy-pasting gan. *arXiv preprint arXiv:1905.11369*, 2019.

[5] V. Blanz and T. Vetter. A morphable model for the synthesis of 3d faces. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 187–194, 1999.

[6] S. Braun. unsupervised-disentangling. https://github.com/theRealSuperMario/unsupervised-disentangling/tree/reproducing_baselines, 2020.

[7] C. Cao and Y. Fu. Learning a sketch tensor space for image inpainting of man-made scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14509–14518, 2021.

[8] N. Cao, X. Yan, Y. Shi, and C. Chen. Ai-sketcher : A deep generative model for producing high-quality sketches. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01): 2564–2571, Jul. 2019. doi: 10.1609/aaai.v33i01.33012564. URL https://ojs.aaai.org/index.php/AAAI/article/view/4103.

[9] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017.

[10] M. Chen, T. Artières, and L. Denoyer. Unsupervised object segmentation by redrawing. In *Advances in Neural Information Processing Systems 32 (NIPS 2019)*, pages 12705–12716, 2019.

[11] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[12] Y. Chen, S. Tu, Y. Yi, and L. Xu. Sketch-pix2seq: a model to generate sketches of multiple categories. *arXiv preprint arXiv:1709.04121*, 2017.

[13] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu. Global contrast based salient region detection. In *CVPR 2011*, pages 409–416, 2011. doi: 10.1109/CVPR.2011.5995344.

[14] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020.

[15] S. Choudhury, I. Laina, C. Rupprecht, and A. Vedaldi. Unsupervised part discovery from contrastive reconstruction. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=iHXQPrISusS.

[16] J. S. Chung, A. Nagrani, and A. Zisserman. Voxceleb2: Deep speaker recognition. In *INTERSPEECH*, 2018.

[17] E. Collins, R. Achanta, and S. Susstrunk. Deep feature factorization for concept discovery. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 336–352, 2018.

[18] E. Crawford and J. Pineau. Spatially invariant unsupervised object detection with convolutional neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3412–3420, 2019.

[19] A. Das, Y. Yang, T. Hospedales, T. Xiang, and Y.-Z. Song. Bézierskетch: A generative model for scalable vector sketches. In *European Conference on Computer Vision*, pages 632–647. Springer, 2020.

[20] X. Dong, S.-I. Yu, X. Weng, S.-E. Wei, Y. Yang, and Y. Sheikh. Supervision-by-registration: An unsupervised approach to improve the precision of facial landmark detectors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 360–368, 2018.

[21] A. Dosovitskiy and T. Brox. Generating images with perceptual similarity metrics based on deep networks. *Advances in neural information processing systems*, 29, 2016.

[22] C. Dugas, Y. Bengio, F. Bélisle, C. Nadeau, and R. Garcia. Incorporating second-order functional knowledge for better option pricing. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000. URL https://proceedings.neurips.cc/paper/2000/file/44968aece94f667e4095002d140b5896-Paper.pdf.

[23] A. Dundar, K. J. Shih, A. Garg, R. Pottorf, A. Tao, and B. Catanzaro. Unsupervised disentanglement of pose, appearance and background from images and videos. *arXiv preprint arXiv:2001.09518*, 2020.

[24] M. Eitz, J. Hays, and M. Alexa. How do humans sketch objects? *ACM Trans. Graph. (Proc. SIGGRAPH)*, 31(4):44:1–44:10, 2012.

[25] H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu. RMPE: Regional multi-person pose estimation. In *ICCV*, 2017.

[26] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou. Joint 3d face reconstruction and dense alignment with position map regression network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 534–551, 2018.

[27] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2414–2423, 2016.

[28] S. Ge, V. Goswami, L. Zitnick, and D. Parikh. Creative sketch generation. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=gwnoVHIES05.

[29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[30] D. Ha and D. Eck. A neural representation of sketch drawings. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hy6GHpkCW.

[31] X. Han, Z. Wu, W. Huang, M. R. Scott, and L. S. Davis. Finet: Compatible and diverse fashion image inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[32] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[33] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

[34] X. He, B. Wandt, and H. Rhodin. Latentkeypointgan: Controlling gans via latent keypoints. *arXiv preprint arXiv:2103.15812*, 2021.

[35] X. He, B. Wandt, and H. Rhodin. Ganseg: Learning to segment by unsupervised hierarchical image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1225–1235, 2022.

[36] Z. Huang and Y. Li. Interpretable and accurate fine-grained recognition via region grouping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8662–8672, 2020.

[37] W.-C. Hung, V. Jampani, S. Liu, P. Molchanov, M.-H. Yang, and J. Kautz. Scops: Self-supervised co-part segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 869–878, 2019.

[38] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[39] C. Ionescu, D. Papava, V. Olaru, and C. 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, 2014. doi: 10.1109/TPAMI.2013.248.

[40] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

[41] A. S. Jackson, A. Bulat, V. Argyriou, and G. Tzimiropoulos. Large pose 3d face reconstruction from a single image via direct volumetric cnn regression. In *Proceedings of the IEEE international conference on computer vision*, pages 1031–1039, 2017.

[42] T. Jakab, A. Gupta, H. Bilen, and A. Vedaldi. Unsupervised learning of object landmarks through conditional image generation. In *Advances in neural information processing systems*, pages 4016–4027, 2018.

[43] T. Jakab, A. Gupta, H. Bilen, and A. Vedaldi. Self-supervised learning of interpretable keypoints from unlabelled videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8787–8797, 2020.

[44] H. Jiang, J. Wang, Z. Yuan, Y. Wu, N. Zheng, and S. Li. Salient object detection: A discriminative regional feature integration approach. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2083–2090, 2013. doi: 10.1109/CVPR.2013.271.

[45] Y. S. Jie Yang, Zhiquan Qi. Learning to incorporate structure knowledge for image inpainting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12605–12612, 2020.

[46] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016.

[47] K. Kaiyrbekov and M. Sezgin. Deep stroke-based sketched symbol reconstruction and segmentation. *IEEE computer graphics and applications*, 40(1):112–126, 2019.

[48] M. Kampelmuhler and A. Pinz. Synthesizing human-like sketches from natural images using a conditional convolutional decoder. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3203–3211, 2020.

[49] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[50] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[51] I. Katircioglu, H. Rhodin, V. Constantin, J. Spörri, M. Salzmann, and P. Fua. Self-supervised human detection and segmentation via background inpainting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[52] I. Katircioglu, H. Rhodin, J. Sporri, M. Salzmann, and P. Fua. Human detection and segmentation via multi-view consensus. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2855–2864, 2021.

[53] Y. Kim, S. Nam, I. Cho, and S. J. Kim. Unsupervised keypoint learning for guiding class-conditional video prediction. In *Advances in Neural Information Processing Systems*, pages 3814–3824, 2019.

[54] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[55] T. D. Kulkarni, A. Gupta, C. Ionescu, S. Borgeaud, M. Reynolds, A. Zisserman, and V. Mnih. Unsupervised learning of object keypoints for perception and control. In *Advances in neural information processing systems*, pages 10724–10734, 2019.

[56] A. Lahiri, A. K. Jain, S. Agrawal, P. Mitra, and P. K. Biswas. Prior guided gan based semantic inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[57] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4): 541–551, 1989. doi: 10.1162/neco.1989.1.4.541.

[58] J. Li, F. He, L. Zhang, B. Du, and D. Tao. Progressive reconstruction of visual structure for image inpainting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[59] T. Li, T. Bolkart, M. J. Black, H. Li, and J. Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6): 194:1–194:17, 2017.

[60] L. Liao, J. Xiao, Z. Wang, C.-W. Lin, and S. Satoh. Guidance and evaluation: Semantic-aware image inpainting for mixed scenes. In *European Conference on Computer Vision*, pages 683–700. Springer, 2020.

[61] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[62] S. Liu, L. Zhang, X. Yang, H. Su, and J. Zhu. Unsupervised part segmentation through disentangling appearance and shape. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8355–8364, 2021.

[63] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[64] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[65] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015.

[66] D. Lorenz, L. Bereska, T. Milbich, and B. Ommer. Unsupervised part-based disentangling of object shape and appearance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10955–10964, 2019.

[67] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. *Advances in neural information processing systems*, 30, 2017.

[68] L. Ma, Q. Sun, S. Georgoulis, L. Van Gool, B. Schiele, and M. Fritz. Disentangled person image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 99–108, 2018.

[69] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *in ICML Workshop on Deep Learning for Audio, Speech and Language Processing*. Citeseer, 2013.

[70] R. Malladi, J. Sethian, and B. Vemuri. Shape modeling with front propagation: a level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2):158–175, 1995. doi: 10.1109/34.368173.

[71] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. doi: 10.1109/3dv.2017.00064. URL `http://gvv.mpi-inf.mpg.de/3dhp_dataset`.

[72] L. Mescheder, A. Geiger, and S. Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018.

[73] D. Mihai and J. Hare. Differentiable drawing and sketching. *arXiv preprint arXiv:2103.16194*, 2021.

[74] M. Minderer, C. Sun, R. Villegas, F. Cole, K. P. Murphy, and H. Lee. Unsupervised learning of object structure and dynamics from videos. In *Advances in Neural Information Processing Systems*, pages 92–102, 2019.

[75] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi. Edgeconnect: Structure guided image inpainting using edge prediction. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[76] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, page 849–856, Cambridge, MA, USA, 2001. MIT Press.

[77] M.-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, Dec 2008.

[78] A. Noguchi, U. Iqbal, J. Tremblay, T. Harada, and O. Gallo. Watch it move: Unsupervised discovery of 3D joints for re-posing of articulated objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[79] A. v. d. Oord, Y. Li, and O. Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[80] A. A. Osman, T. Bolkart, and M. J. Black. Star: Sparse trained articulated human body regressor. In *European Conference on Computer Vision*, pages 598–613. Springer, 2020.

[81] K. Pang, D. Li, J. Song, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Deep factorised inverse-sketching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 36–52, 2018.

[82] G. Papandreou, T. Zhu, L.-C. Chen, S. Gidaris, J. Tompson, and K. Murphy. Personlab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In *Proceedings of the European conference on computer vision (ECCV)*, pages 269–286, 2018.

[83] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10975–10985, 2019.

[84] G. Pavlakos, V. Choutas, N. Ghorbani, T. Bolkart, A. A. A. Osman, D. Tzionas, and M. J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 10975–10985, 2019.

[85] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li. Structureflow: Image inpainting via structure-aware appearance flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 181–190, 2019.

[86] H. Rhodin, M. Salzmann, and P. Fua. Unsupervised geometry-aware representation for 3d human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 750–767, 2018.

[87] H. Rhodin, V. Constantin, I. Katircioglu, M. Salzmann, and P. Fua. Neural scene decomposition for multi-person motion capture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[88] I. K. Riza Alp Güler, Natalia Neverova. Densepose: Dense human pose estimation in the wild. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[89] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[90] P. Sangkloy, N. Burnell, C. Ham, and J. Hays. The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (proceedings of SIGGRAPH)*, 2016.

[91] K. Sasaki, S. Iizuka, E. Simo-Serra, and H. Ishikawa. Learning to Restore Deteriorated Line Drawing. *The Visual Computer (Proc. of Computer Graphics International 2018)*, 34(6-8): 1077–1085, 2018.

[92] P. Savarese, S. S. Kim, M. Maire, G. Shakhnarovich, and D. McAllester. Information-theoretic segmentation by inpainting error maximization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4029–4039, 2021.

[93] L. Schmidtke, A. Vlontzos, S. Ellershaw, A. Lukens, T. Arichi, and B. Kainz. Unsupervised human pose estimation through transforming shape templates. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2484–2494, 2021.

[94] A. Siarohin, E. Sangineto, S. Lathuiliere, and N. Sebe. Deformable gans for pose-based human image generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3408–3416, 2018.

[95] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2377–2386, 2019.

[96] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. First order motion model for image animation. In *Conference on Neural Information Processing Systems (NeurIPS)*, December 2019.

[97] A. Siarohin, S. Roy, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. Motion-supervised co-part segmentation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9650–9657. IEEE, 2021.

[98] Y. Song, C. Yang, Y. Shen, P. Wang, Q. Huang, and C.-C. J. Kuo. Spg-net: Segmentation prediction and guidance network for image inpainting. *arXiv preprint arXiv:1805.03356*, 2018.

[99] S.-Y. Su, F. Yu, M. Zollhoefer, and H. Rhodin. A-neRF: Articulated neural radiance fields for learning human shape, appearance, and pose. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=lwwEh0OM61b.

[100] J. J. Sun, S. Ryou, R. H. Goldshmid, B. Weissbourd, J. O. Dabiri, D. J. Anderson, A. Kennedy, Y. Yue, and P. Perona. Self-supervised keypoint discovery in behavioral videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2171–2180, 2022.

[101] S. Suwajanakorn, N. Snavely, J. J. Tompson, and M. Norouzi. Discovery of latent 3d keypoints via end-to-end geometric reasoning. In *Advances in neural information processing systems*, pages 2059–2070, 2018.

[102] J. Thewlis, H. Bilen, and A. Vedaldi. Unsupervised learning of object landmarks by factorized spatial embeddings. In *Proceedings of the IEEE international conference on computer vision*, pages 5916–5925, 2017.

[103] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[104] F. Wang, L. Kang, and Y. Li. Sketch-based 3d shape retrieval using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1875–1883, 2015.

[105] H. Wang, S. Ge, Z. Lipton, and E. P. Xing. Learning robust global representations by penalizing local predictive power. In *Advances in Neural Information Processing Systems*, pages 10506–10518, 2019.

[106] J. Wang, C. Wang, Q. Huang, Y. Shi, J.-F. Cai, Q. Zhu, and B. Yin. *Image Inpainting Based on Multi-Frequency Probabilistic Inference Model*, page 1–9. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450379885. URL https://doi.org/10.1145/3394171.3413891.

[107] W. Wu, K. Cao, C. Li, C. Qian, and C. C. Loy. Transgaga: Geometry-aware unsupervised image-to-image translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8012–8021, 2019.

[108] Z. Wu, Y. Xiong, X. Y. Stella, and D. Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[109] B. Xiao, H. Wu, and Y. Wei. Simple baselines for human pose estimation and tracking. In *European Conference on Computer Vision (ECCV)*, 2018.

[110] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo. Foreground-aware image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5840–5848, 2019.

[111] H. Xu, E. G. Bazavan, A. Zanfir, B. Freeman, R. Sukthankar, and C. Sminchisescu. Ghum & ghuml: Generative 3d human shape and articulated pose models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (Oral)*, pages 6184–6193, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Xu_GHUM__GHUML_Generative_3D_Human_Shape_and_Articulated_Pose_CVPR_2020_paper.html.

[112] H. Xu, X. Su, M. Wang, X. Hao, and G. Gao. An edge information and mask shrinking based image inpainting approach. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, Los Alamitos, CA, USA, jul 2020. IEEE Computer Society. doi: 10.1109/ICME46284.2020.9102892. URL https://doi.ieeecomputersociety.org/10.1109/ICME46284.2020.9102892.

[113] P. Xu. Deep learning for free-hand sketch: A survey. *IEEE transactions on pattern analysis and machine intelligence*, PP, 2022.

[114] P. Xu, Y. Huang, T. Yuan, K. Pang, Y.-Z. Song, T. Xiang, T. M. Hospedales, Z. Ma, and J. Guo. Sketchmate: Deep hashing for million-scale human sketch retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8090–8098, 2018.

[115] P. Xu, C. K. Joshi, and X. Bresson. Multigraph transformer for free-hand sketch recognition. *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[116] Y. Xu, C. Yang, Z. Liu, B. Dai, and B. Zhou. Unsupervised landmark learning from unpaired data. *arXiv preprint arXiv:2007.01053*, 2020.

[117] Y. Yang and X. Guo. Generative landmark guided face inpainting. In *Chinese Conference on Pattern Recognition and Computer Vision (PRCV)*, pages 14–26. Springer, 2020.

[118] R. Yi, Y.-J. Liu, Y.-K. Lai, and P. L. Rosin. APDrawingGAN: Generating artistic portrait drawings from face photos with hierarchical gans. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR '19)*, pages 10743–10752, 2019.

[119] Q. Yu, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales. Sketch-a-net that beats humans. In *BMVC*, 2015.

[120] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy. Sketch me that shoe. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[121] T. Yu, K. Guo, F. Xu, Y. Dong, Z. Su, J. Zhao, J. Li, Q. Dai, and Y. Liu. Bodyfusion: Real-time capture of human motion and surface geometry using a single depth camera. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 910–919, 2017. doi: 10.1109/ICCV.2017.104.

[122] Y. Yu, F. Zhan, R. WU, J. Pan, K. Cui, S. Lu, F. Ma, X. Xie, and C. Miao. *Diverse Image Inpainting with Bidirectional and Autoregressive Transformers*, page 69–78. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450386517. URL https://doi.org/10.1145/3474085.3475436.

[123] X. Zhang, C. Shi, X. Wang, X. Wu, X. Li, J. Lv, and I. Mumtaz. Face inpainting based on gan by facial prediction and fusion as guidance information. *Applied Soft Computing*, 111:107626, 2021. ISSN 1568-4946. doi: https://doi.org/10.1016/j.asoc.2021.107626. URL https://www.sciencedirect.com/science/article/pii/S1568494621005470.

[124] Y. Zhang, Y. Guo, Y. Jin, Y. Luo, Z. He, and H. Lee. Unsupervised discovery of object landmarks as structural representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2694–2703, 2018.

[125] Y. Zhang, Q. Liang, K. Zou, Z. Li, W. Sun, and Y. Wang. Self-supervised part segmentation via motion imitation. *Image and Vision Computing*, 120:104393, 2022. ISSN 0262-8856. doi: https://doi.org/10.1016/j.imavis.2022.104393. URL https://www.sciencedirect.com/science/article/pii/S0262885622000221.

[126] Z. Zhao, W. Liu, Y. Xu, X. Chen, W. Luo, L. Jin, B. Zhu, T. Liu, B. Zhao, and S. Gao. Prior based human completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7947–7957, 2021.

[127] C. Zheng, T.-J. Cham, and J. Cai. Pluralistic image completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1438–1447, 2019.

[128] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017.

[129] W. Zhu, S. Liang, Y. Wei, and J. Sun. Saliency optimization from robust background detection. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2814–2821, 2014. doi: 10.1109/CVPR.2014.360.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 1
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [No] This paper does not work towards theoretical analysis.
   (b) Did you include complete proofs of all theoretical results? [No] This paper does not work towards theoretical analysis.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] The code, instructions, and the pre-trained models are released in our GitHub. The data we used is already publicly available.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 3.3 and Appendix B,D.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section 3.3 and Table 1& 2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section 3.3
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No] All the datasets are commonly used in research papers and their licenses allow the usage for research purposes.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] The code, instructions, and the pre-trained models are released in our GitHub.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] All the datasets are commonly used in research papers and the licenses allow the usage.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] All the datasets are commonly used in research papers and all datasets that contain humans are established datasets.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A More Results

We show supplemental videos in https://xingzhehe.github.io/autolink/.

To verify the robustness and generality, we show 105 images for each dataset we used in the main paper with keypoints and visualized graph representation in Figure 9-23 in the supplemental materials.

B Edge-Map Ablation Tests

Table 5 shows the results on the different numbers of keypoints and edge thickness. While a larger number of keypoints gives better details and thus higher accuracy, the performance is robust to the thickness. Table 6 shows the results on different masking ratios and mask patch sizes. A too small masking ratio significantly decreases the performance since the structure can be directly extracted from the masked image with a low masking ratio. The thicknesses used in the main paper are those marked bold in Table 5.

Table 5: **Ablation Tests on the numbers of keypoints and edge thickness**. We remove the % sign in metrics for simplicity. The best one for each number of keypoints is marked in bold.

	CelebA-wild ↓					Human3.6m ↓					DeepFashion ↑					Taichi ↓				
σ^2	K=2	K=4	K=8	K=16	K=32	K=2	K=4	K=8	K=16	K=32	K=2	K=4	K=8	K=16	K=32	K=2	K=4	K=8	K=16	K=32
1.0e-5	60.9	52.5	42.6	34.9	31.9	5.67	5.13	3.37	2.90	**2.81**	17.8	48.1	58.1	62.6	63.8	667	657	622	592	458
2.5e-5	50.5	38.8	29.8	10.7	6.11	5.64	5.15	3.50	3.02	2.87	29.5	48.9	57.1	65.2	66.2	665	637	611	506	351
5.0e-5	49.4	8.06	5.41	4.88	4.65	6.10	5.03	3.76	**2.76**	2.91	22.5	50.4	58.7	**65.8**	66.6	654	550	338	338	287
7.5e-5	58.3	7.71	5.62	4.92	4.65	5.49	5.08	**3.19**	2.89	3.00	43.5	51.1	**59.3**	65.7	**69.8**	650	516	383	301	297
1.0e-4	54.5	7.44	5.71	4.93	4.64	5.56	5.09	3.25	2.96	2.96	44.5	49.0	58.1	64.1	67.6	647	512	385	329	280
2.5e-4	11.3	6.68	5.57	5.05	4.42	5.49	5.03	3.36	3.01	3.32	42.9	48.7	56.8	64.9	67.6	624	526	391	321	284
5.0e-4	11.4	6.56	5.77	5.01	4.43	5.49	5.10	3.45	2.94	3.16	42.0	47.5	58.0	65.4	67.2	612	531	381	307	**275**
7.5e-4	**10.6**	**6.11**	5.81	4.86	**4.39**	5.57	5.09	3.84	3.31	3.06	35.6	49.6	58.2	65.7	66.4	604	479	363	296	289
1.0e-3	13.3	6.84	5.70	**4.43**	4.50	5.49	5.04	3.42	3.47	3.18	40.4	51.0	57.1	64.6	68.8	608	462	342	306	275
2.5e-3	15.8	6.70	**5.24**	4.72	4.49	5.48	5.04	3.40	3.37	3.01	43.6	**51.6**	55.8	61.2	66.7	609	**442**	326	**289**	286
5.0e-3	12.6	6.29	5.53	4.69	4.48	5.52	5.03	3.59	3.36	3.12	**44.7**	50.3	56.7	62.3	68.3	598	510	333	323	302
7.5e-3	11.6	6.16	5.61	4.76	4.41	5.54	5.06	3.50	3.77	2.99	39.4	48.6	57.7	62.6	65.5	604	514	**325**	328	340
1.0e-2	11.2	6.22	6.66	4.89	4.47	**5.45**	**5.02**	3.39	3.05	2.94	44.6	50.3	56.5	60.9	65.1	**593**	515	327	327	341

Table 6: **Ablation Tests on masking ratio and patch size**. We remove the % sign in metrics for simplicity. The best one for each number of keypoints is marked in bold.

	CelebA-wild ↓					Human3.6m ↓					DeepFashion ↑					Taichi ↓				
ratio	4x4	8x8	16x16	32x32	64x64	4x4	8x8	16x16	32x32	64x64	4x4	8x8	16x16	32x32	64x64	4x4	8x8	16x16	32x32	64x64
10%	17.9	16.4	49.2	49.3	48.3	3.22	3.25	3.55	5.28	6.39	58.3	39.7	40.5	52.1	51.0	640	638	630	659	642
20%	19.7	29.4	44.6	43.3	39.8	3.06	3.52	3.21	4.29	5.99	62.1	55.5	43.4	39.6	49.3	522	605	614	638	613
30%	6.56	8.46	45.4	41.3	46.8	3.08	3.08	3.44	3.80	6.24	62.8	58.9	62.8	42.4	41.8	509	570	627	642	627
40%	8.31	6.71	7.58	6.31	6.81	3.15	2.91	3.38	3.40	4.19	62.5	59.2	63.4	41.2	40.0	500	428	634	640	651
50%	8.06	6.39	6.23	6.58	5.99	3.73	2.99	2.94	3.72	4.24	61.3	65.4	64.3	41.1	38.9	444	452	481	640	667
60%	7.69	6.22	5.52	6.98	5.56	3.11	3.13	2.95	3.35	4.87	62.8	62.0	61.7	41.4	39.9	483	396	418	654	643
70%	7.03	6.38	5.44	5.11	4.43	2.95	3.09	2.87	3.28	4.85	60.7	63.2	63.1	59.7	41.5	447	362	376	523	656
80%	6.95	6.68	5.24	4.73	4.65	3.63	3.47	2.76	2.97	4.08	58.3	61.7	65.8	61.6	39.6	501	371	316	347	642
90%	7.24	7.15	5.77	5.62	4.14	2.99	3.18	2.95	3.64	3.75	60.9	63.1	66.4	62.7	40.2	626	388	330	346	526

C Applications

In this section, we briefly describe how we created the two applications, conditional Generative Adversarial Networks and pose transfer networks, based on the learned graph representation, as shown in the teaser. Note that, although we apply the graph representation to videos for pose transfer, it is only learned from the collections of single images.

C.1 Conditional Generative Adversarial Network

The conditional GAN is a simplified StyleGAN2 [50], where the spatial noise injection is removed and the starting tensor is replaced by the feature map generated from the edge map. For simplicity, we do not use EqualLinear [49] or Path Regularization [49].

Formally speaking, given an edge map $\mathbf{S} \in \mathbb{R}^{H \times W}$, where H, W are the spatial size, we use bicubic interpolation to downsample it to 32×32 and feed it into a two-layer convolution network to generate a feature map $\mathbf{F} \in \mathbb{R}^{8 \times 8 \times 512}$. In parallel, we generate the embedding vector $\mathbf{w} \in \mathbb{R}^{256}$ from a noise vector $\mathbf{z} \in \mathbb{R}^{256}$ by a three linear-layer MLP. The feature map $\mathbf{F}$ is fed into a residual convolution-based generator where the kernel weights are modulated by the embedding vector $\mathbf{w}$ [50]. The final image $\mathbf{x} \in \mathbb{R}^{H \times W \times 3}$ is generated by a single convolution layer.

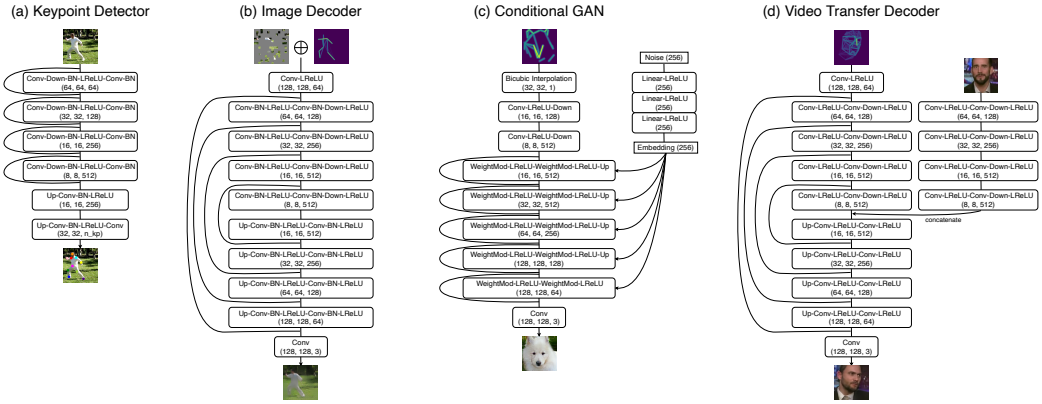

Figure 8: **Network architectures. From left to right: detector (encoder), decoder, conditional GAN, conditional autoencoder.** The shortcuts in (a) and (c) are addition while the shortcuts in (b) and (d) are concatenation.

We denote $\mathcal{G}$ as the generator and $\mathcal{D}$ as the discriminator. We use the non-saturating loss [29],

$$\mathcal{L}_{\text{GAN}}(\mathcal{G}) = \mathbb{E}_{\mathbf{z}\sim\mathcal{N}} \log(\exp(-\mathcal{D}(\mathcal{G}(\mathbf{z}))) + 1) \tag{7}$$

for the generator, and logistic loss,

$$\mathcal{L}_{\text{GAN}}(\mathcal{D}) = \mathbb{E}_{\mathbf{z}\sim\mathcal{N}} \log(\exp(\mathcal{D}(\mathcal{G}(\mathbf{z}))) + 1) + \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log(\exp(-\mathcal{D}(\mathbf{x})) + 1) \tag{8}$$

for the discriminator, with gradient penalty [72] applied only on real data,

$$\mathcal{L}_{\text{gp}}(\mathcal{D}) = \mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \nabla\mathcal{D}(\mathbf{x}). \tag{9}$$

C.2 Pose Transfer Network

We train the pose transfer network on videos. We randomly sample two frames $\mathbf{I}_1, \mathbf{I}_2 \in \mathbb{R}^{H\times W\times 3}$, where H, W are the spatial size. We use keypoint detector on the frame $\mathbf{I}_1$ and generate the edge map $\mathbf{S}_1 \in \mathbb{R}^{H\times W}$. The edge map is fed into a UNet [89] to reconstruct $\mathbf{I}_1$. The smallest feature map in the UNet is concatenated by a feature map of the appearance information, which is generated by frame $\mathbf{I}_2$ to provide the appearance information. The loss is a combination of Mean Squared Error, VGG perceptual loss [46] and the GAN loss in Eq 7, 8, and 9.

D Network Architecture

Figure 8 shows the architectures we used in the main paper. The keypoint detector is a ResNet with upsampling [109], which is a simple baseline used in human pose estimation. The decoder and the pose transfer network are UNets [89]. The conditional GAN is a simplified StyleGAN2 [50], where the spatial noise injection is removed and the starting tensor is replaced by the feature map generated from the edge map. In Figure 8, we denote Conv for 3x3 convolution [57], BN for Batch Normalization [38], LReLU for Leaky ReLU [69], Up for 2x bilinear upsampling, and Down for 2x bilinear downsampling.

E Edge Map Visualization

For visualization purposes, we scale the edge weights by dividing the maximum value to obtain visible edges. For DeepFashion, before dividing the maximum value, we further add 0.01 to the edge weights larger than 0.0001. Although the models are trained on different thicknesses, we draw them in the same thickness $\sigma^2 = 5 \times 10^{-4}$ for pleasing visualization.