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.

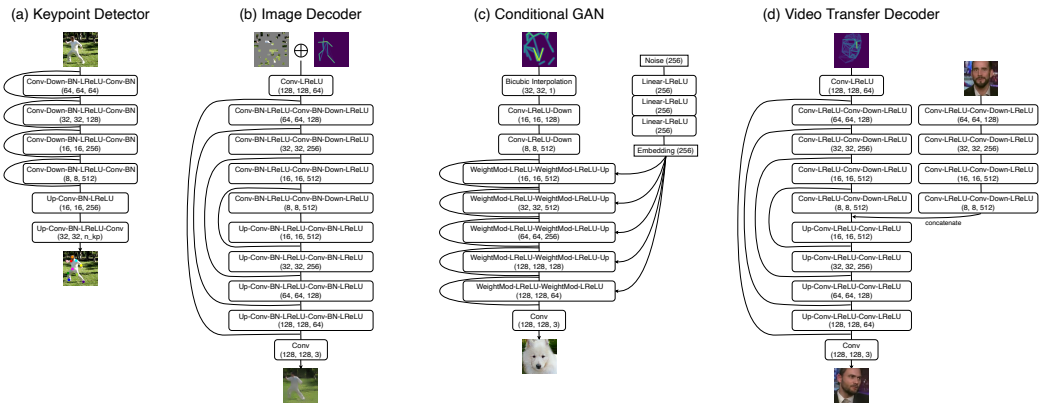

Figure 8: **Network architectures. From left to right: detector (encoder), decoder, conditional GAN, conditional autoencoder.** The shortcuts in (a) and (c) are addition while the shortcuts in (b) and (d) are concatenation.

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.

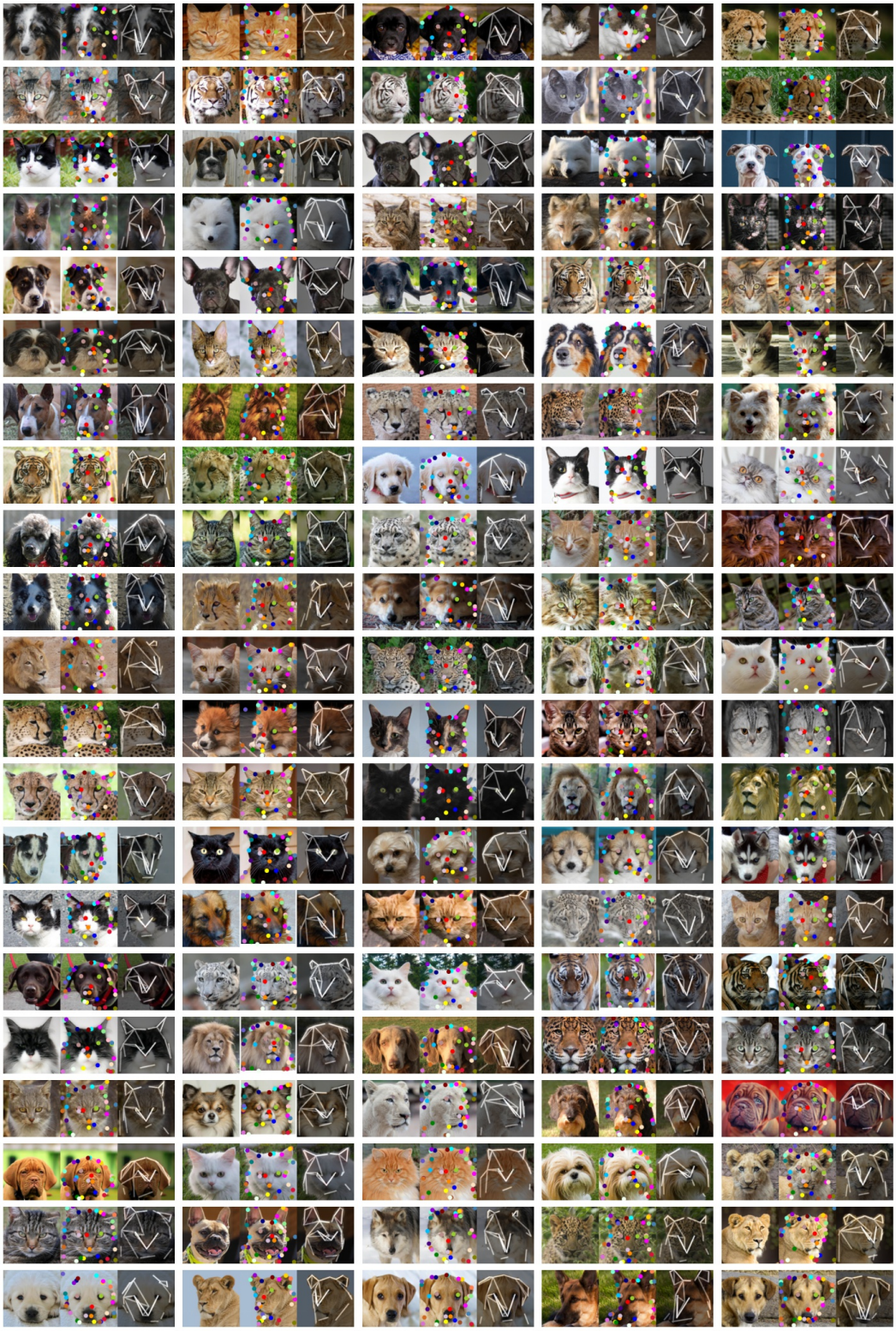

Figure 9: **105 samples from AFHQ (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 10: **105 samples from CelebA-in-The-Wild (32 keypoints),** with the image-points-edge pairs overlaid.

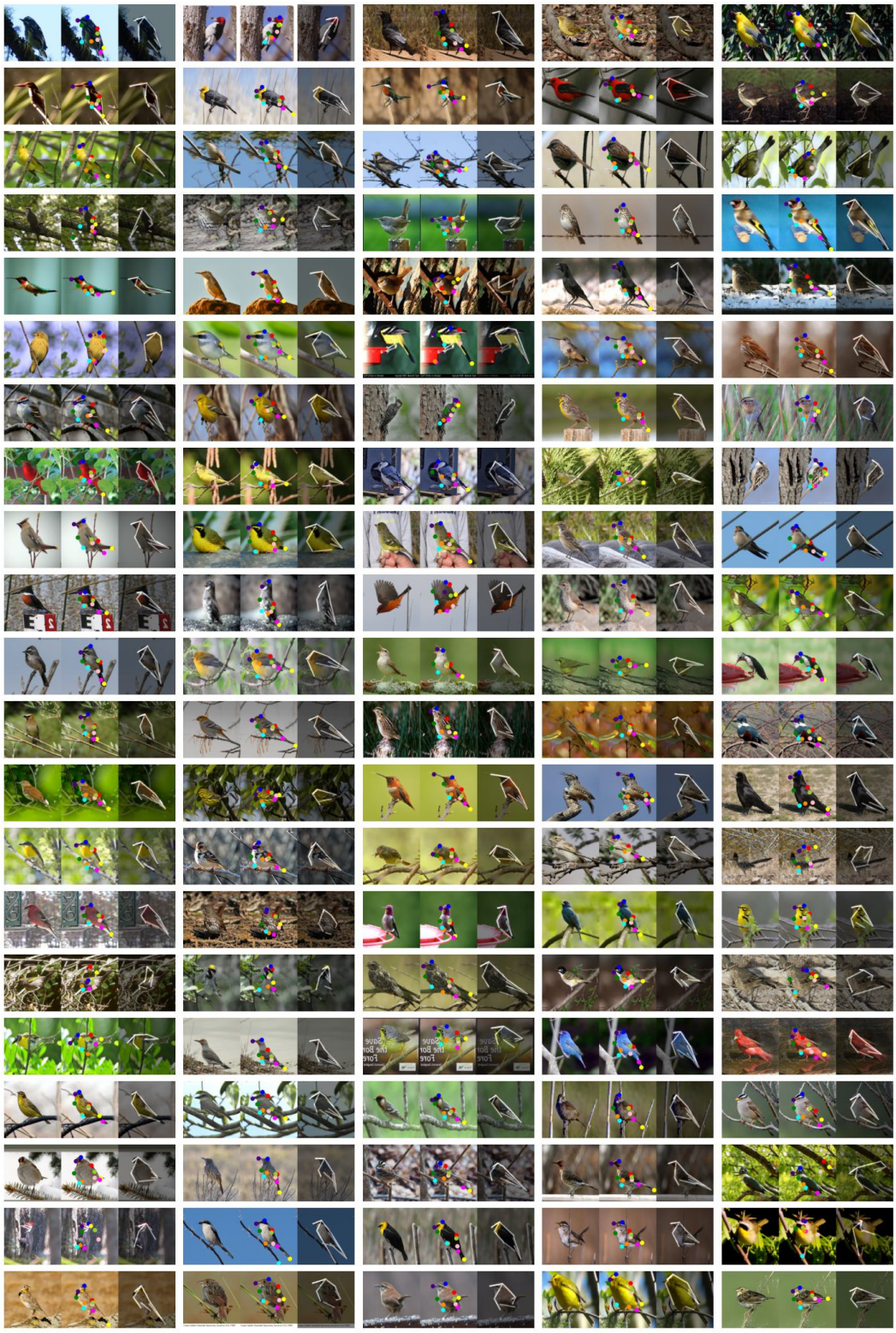

Figure 11: **105 samples from CUB-aligned (10 keypoints),** with the image-points-edge pairs overlaid.

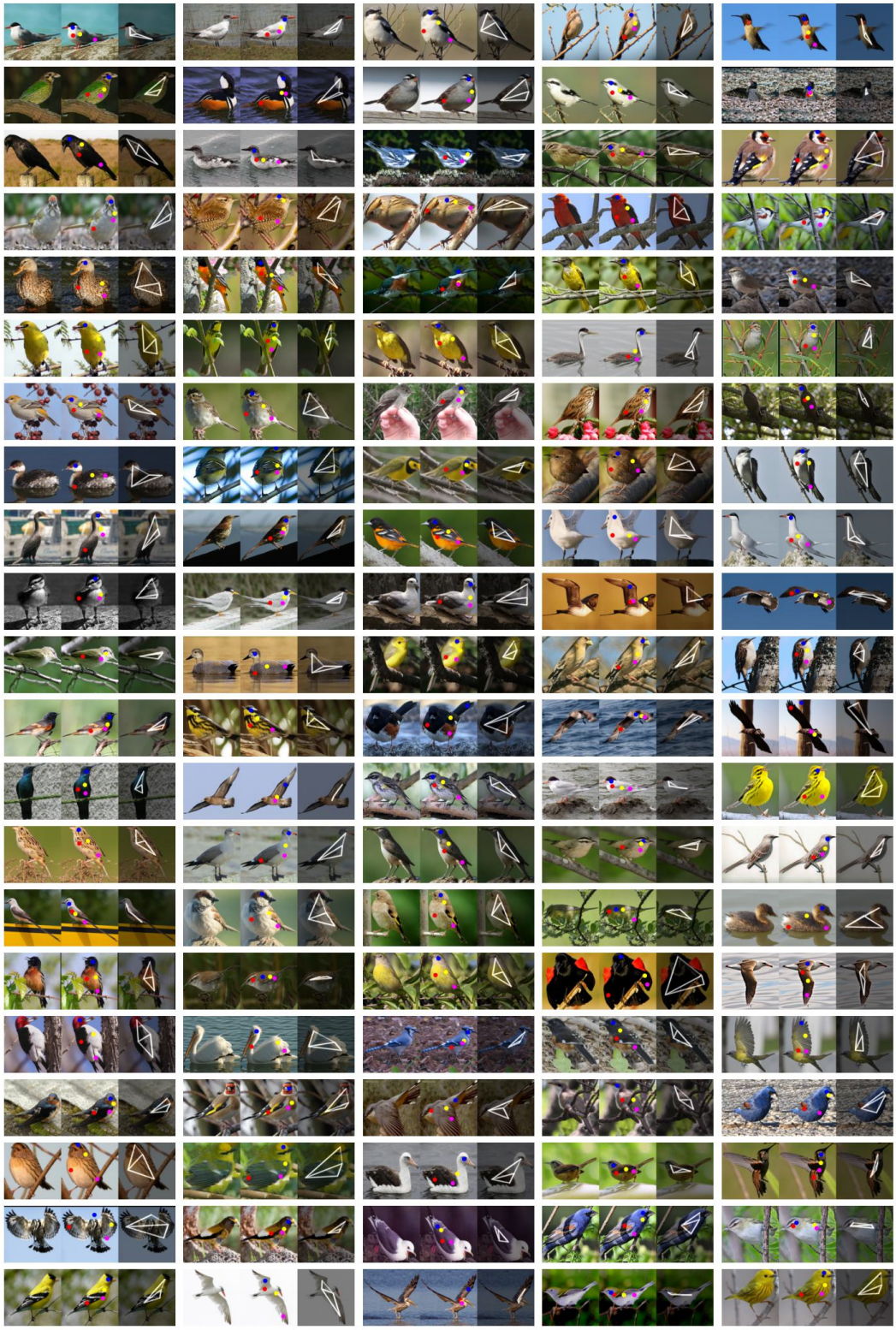

Figure 12: **105 samples from CUB (4 keypoints),** with the image-points-edge pairs overlaid.

Figure 13: **105 samples from DeepFashion (16 keypoints),** with the image-points-edge pairs overlaid.

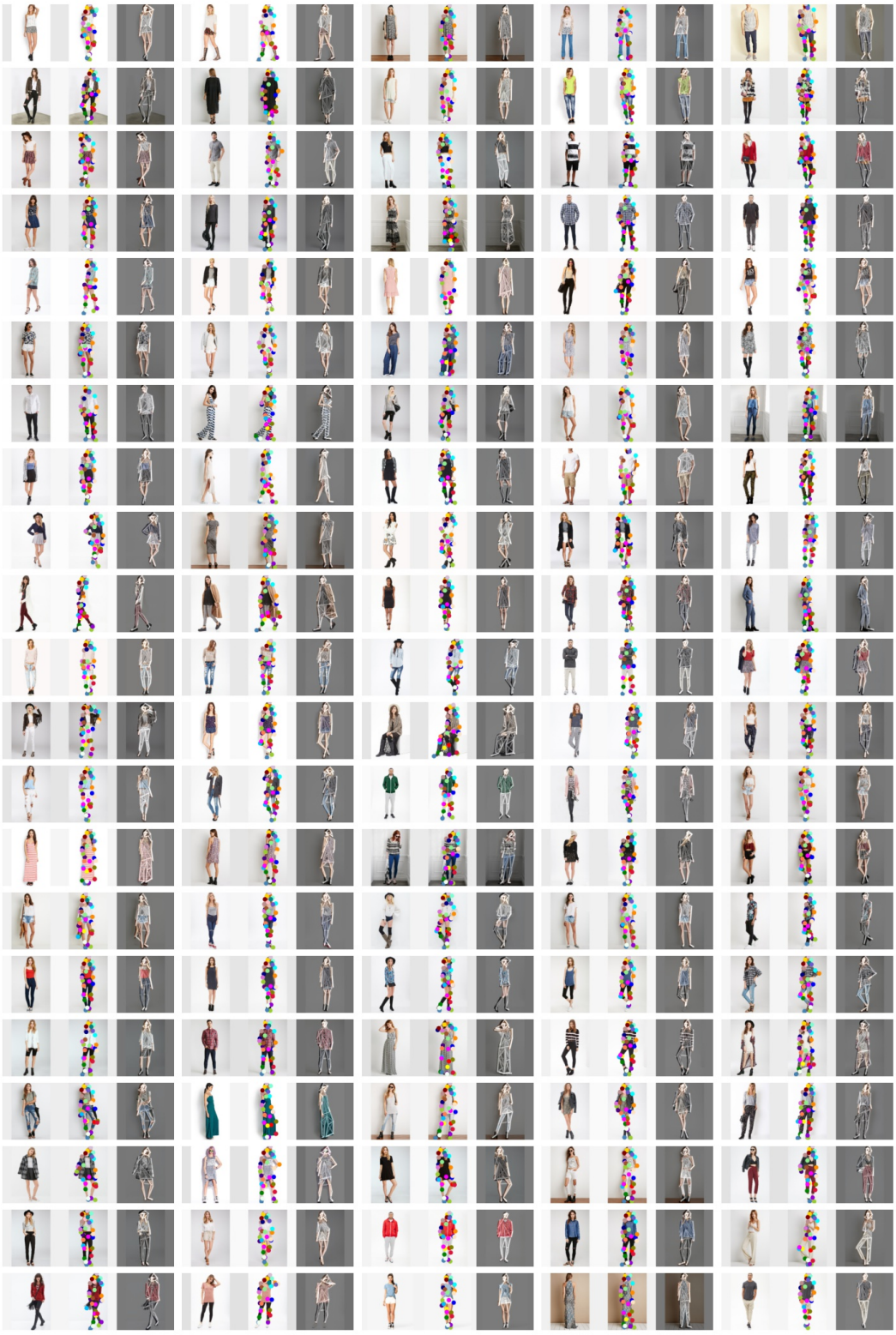

Figure 14: **105 samples from DeepFashion (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 15: **105 samples from Flower (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 16: **105 samples from Human3.6m (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 17: **105 samples from Human3.6m without background (16 keypoints),** with the image-points-edge pairs overlaid.

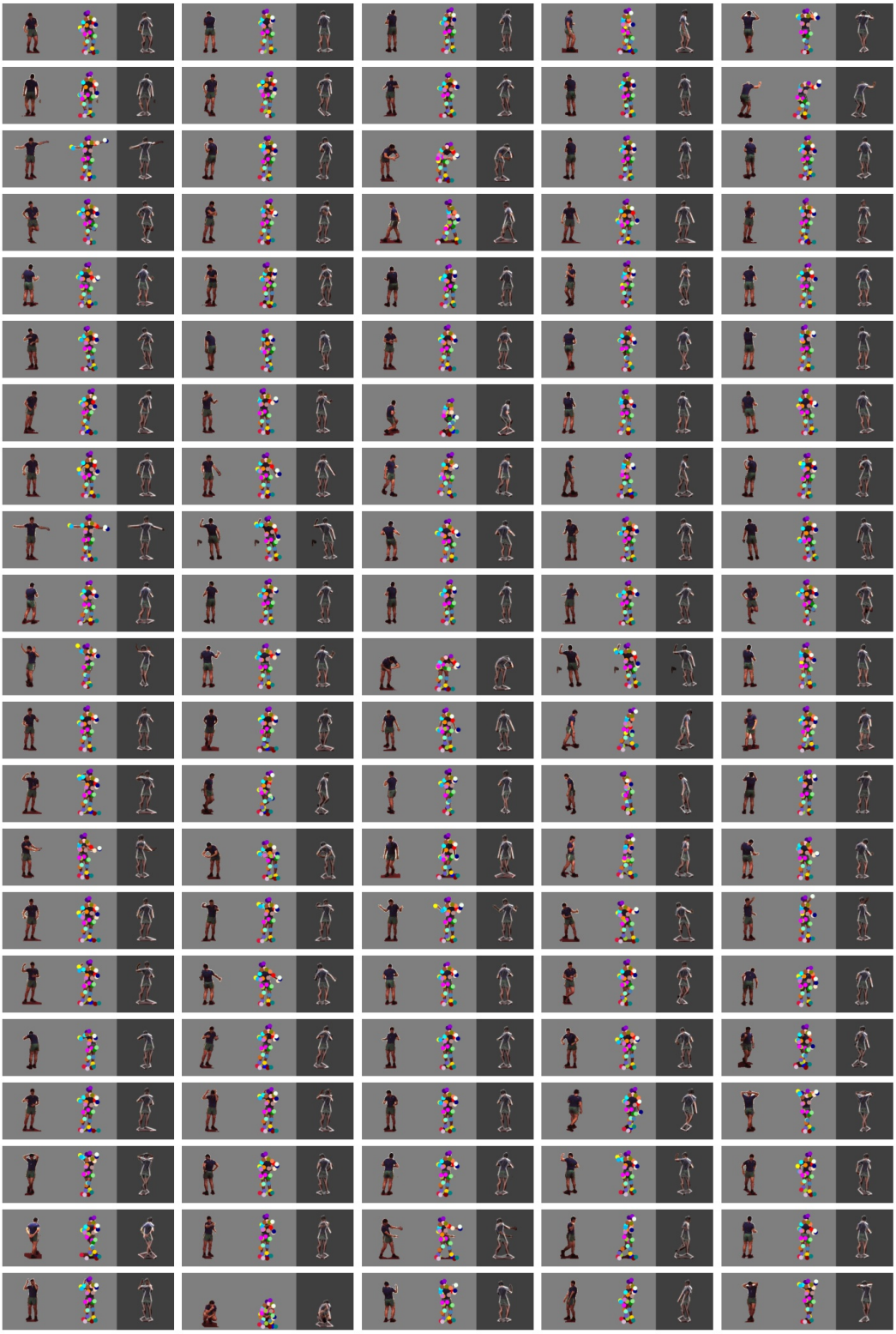

Figure 18: **105 samples from Human3.6m without background (32 keypoints),** with the image-points-edge pairs overlaid.

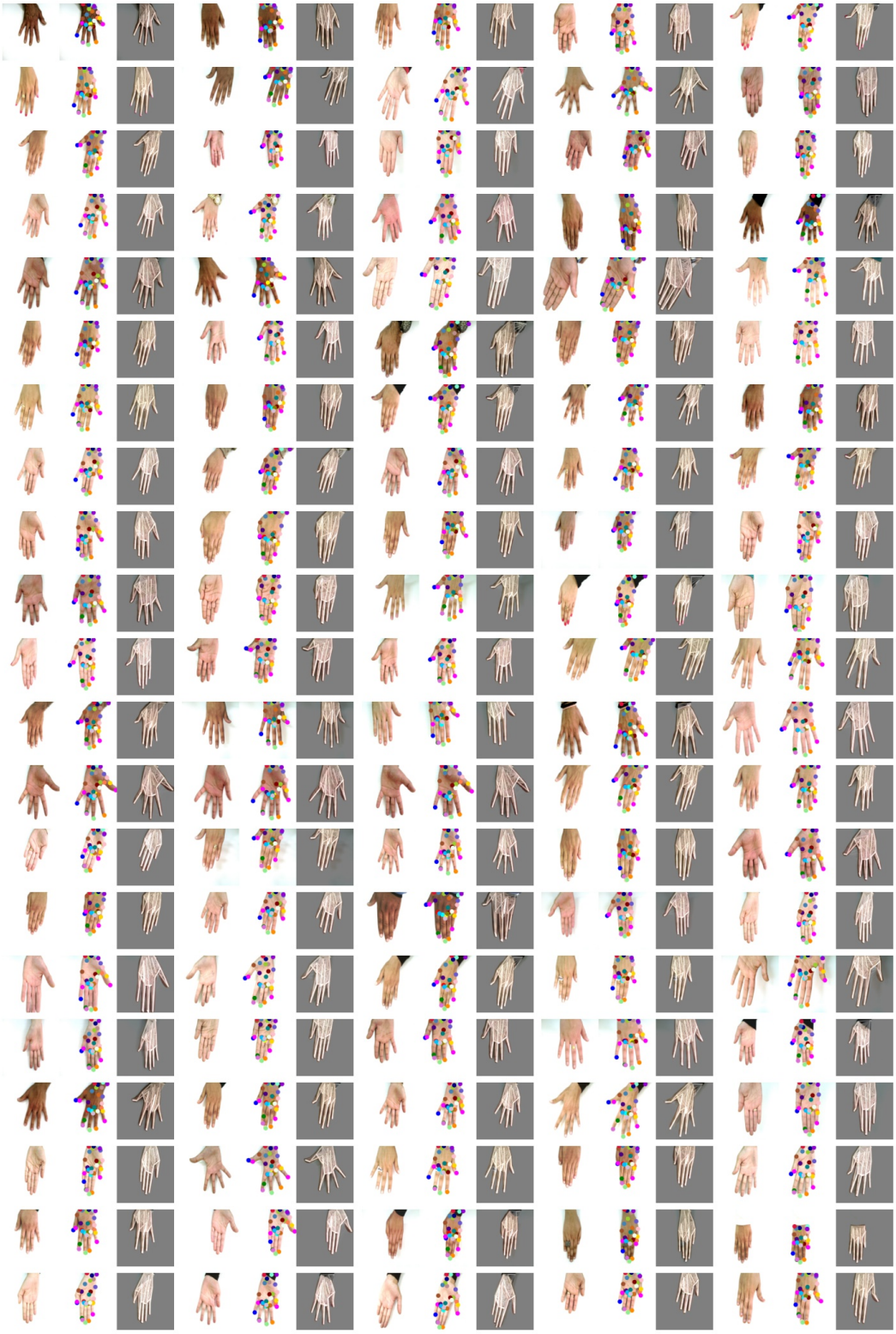

Figure 19: **105 samples from 11k Hands without background (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 20: **105 samples from Horses (32 keypoints),** with the image-points-edge pairs overlaid.

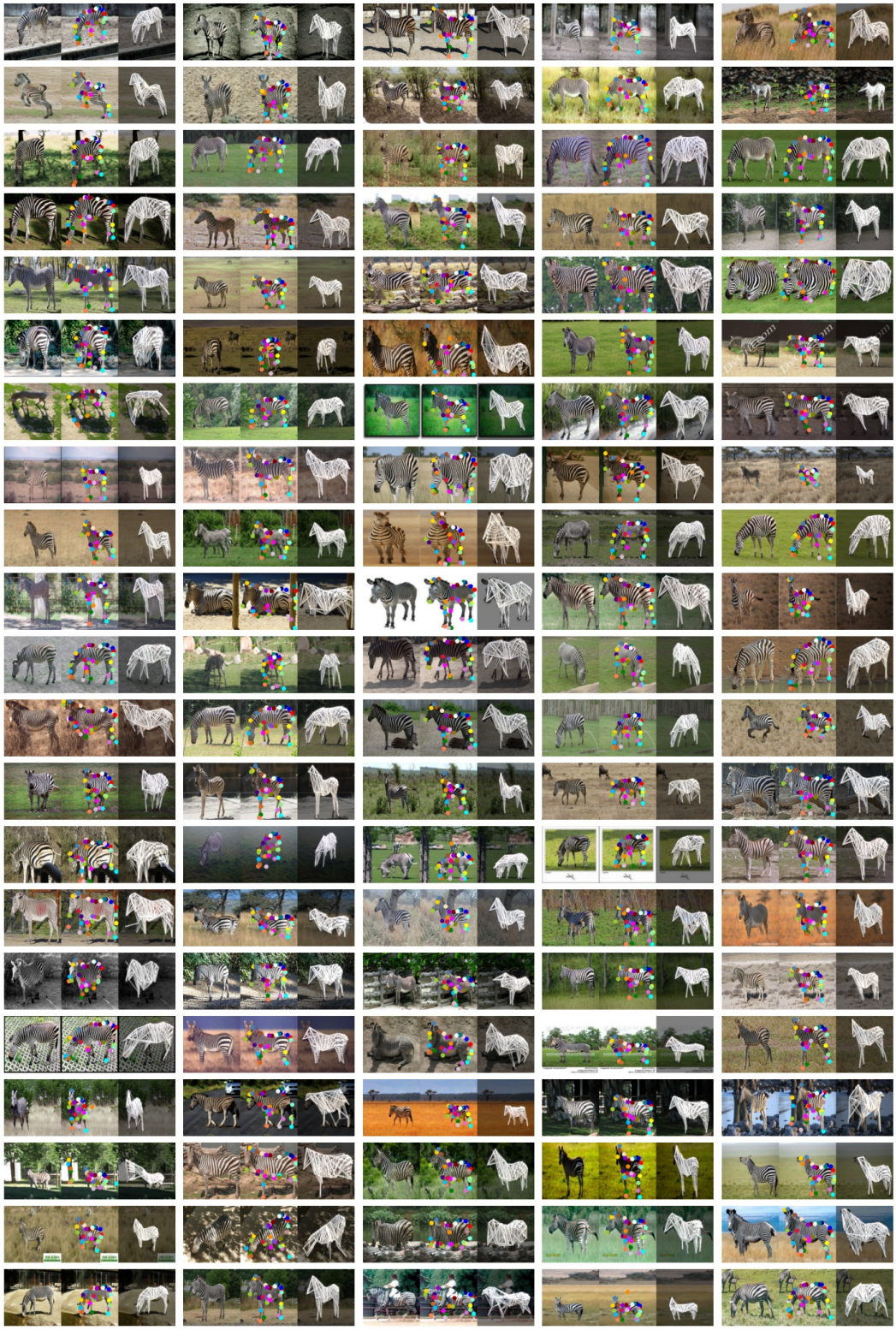

Figure 21: **105 samples from Zebra (32 keypoints),** with the image-points-edge pairs overlaid.

Figure 22: **105 samples from Taichi (10 keypoints),** with the image-points-edge pairs overlaid.

Figure 23: **105 samples from Taichi (32 keypoints),** with the image-points-edge pairs overlaid.