Deformation-aware Unpaired Image Translation for Pose Estimation on Laboratory Animals

Siyuan Li[1], Semih Günel[1,3], Mirela Ostrek[1], Pavan Ramdya[3], Pascal Fua[1], and Helge Rhodin[1,2]

[1]CVLAB, EPFL, Lausanne
[2]Imager Lab, UBC, Vancouver
[3]Neuroengineering Lab, EPFL, Lausanne

Abstract

Our goal is to capture the pose of real animals using synthetic training examples, without using any manual supervision. Our focus is on neuroscience model organisms, to be able to study how neural circuits orchestrate behaviour. Human pose estimation attains remarkable accuracy when trained on real or simulated datasets consisting of millions of frames. However, for many applications simulated models are unrealistic and real training datasets with comprehensive annotations do not exist. We address this problem with a new sim2real domain transfer method. Our key contribution is the explicit and independent modelling of appearance, shape and pose in an unpaired image translation framework. Our model lets us train a pose estimator on the target domain by transferring readily available body keypoint locations from the source domain to generated target images. We compare our approach with existing domain transfer methods and demonstrate improved pose estimation accuracy on Drosophila melanogaster (fruit fly), Caenorhabditis elegans (worm) and Danio rerio (zebrafish), without requiring any manual annotation on the target domain and despite using simplistic off-the-shelf animal characters for simulation, or simple geometric shapes as models. Our new datasets, code and trained models will be published to support future computer vision and neuroscientific studies.

1. Introduction

Deep learning-based pose estimation on images has evolved into a practical tool for a wide range of applications, as long as sufficiently large training databases are available. However, in very specialized domains there are rarely large annotation databases. For example, neuroscientists need to accurately capture the poses of all the appendages of

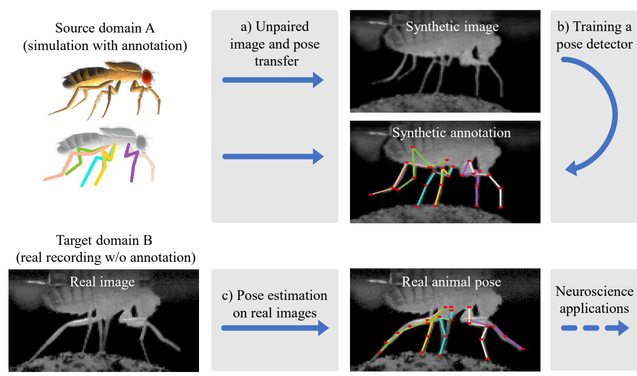

Figure 1. **Approach.** Our most morphologically complex example is the six-legged *Drosophila*: a) We transfer synthetic images and their keypoint annotations to realistically looking images using only unpaired examples of the two domains. b) Our method enables training of a pose detector that c) can be applied to real images for neuroscientific studies.

fruit flies, as pose dynamics are crucial for drawing inferences about how neural populations coordinate animal behavior. Publicly available databases for such studies are rare and current annotation techniques available to create such a database are tedious and time consuming, even when semi-automated. Given the existence of motion simulators, an apparently simple workaround would be to synthesize images of flies in various poses and use these synthetic images for training purposes. Although image generation algorithms can now generate very convincing *deepfakes*, existing image translation algorithms do not preserve pose geometrically when the gap between a synthetic source and a real target is large. This is critical to our application, as creating matching high-fidelity images would be time consuming.

In this paper, we introduce a novel approach to generate realistic images of different kinds of laboratory animals—flies, fish, and worms–from synthetic renderings for which labels such as keypoint annotations are readily available.

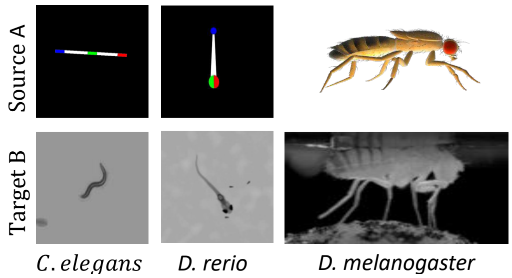

Figure 2. **Domain examples with large discrepancy in appearance, shape and pose.** Translating from rendering to real images requires bridging the domain gap without having pixel nor pose correspondences. It is particularly challenging in our setting, as even the realistic fly character shows significant differences in shape (body and limb width) as well as pose (legs stretched).

Figure 3. **Generalization to humans.** Our method also applies to human pose transfer undergoing large shape changes.

The generated realistic images can then be used to train a deep network that operates on real images, as shown in Fig. 1. The challenge is to condition the generated images in such a way that the keypoints (e.g. skeleton joint positions) in the simulated source transfer to the realistic target; despite large differences in shape and pose as well as for small training sets that are practical, see Fig. 2.

We model the change of 2D pose and shape in terms of a deformation field. This field is then paired with an image-to-image translator that synthesizes appearance while preserving geometry, as shown in Fig. 4. Our approach is inspired by earlier approaches modeling human faces [59] and brain scans [6]. We go beyond these studies in two important ways. First, we introduce silhouettes as an intermediate representation that facilitates independent constraints (loss terms) on shape and appearance. It stabilizes training to succeed without reference images and helps to separate explicit geometric deformation from appearance changes. Furthermore, end-to-end training on unpaired examples is enabled with two discriminators and a straight-through estimator for non-differentiable thresholding operation and patch-wise processing. Second, to cope with large-scale as well as small-scale shape discrepancies, we introduce a hierarchical deformation model to separate global scaling, translation, and rotation from local deformation.

We test our method on flies (*Drosophila melanogaster*), worms (*Caenorhabditis elegans*) and larval zebrafish (*Danio rerio*), see Fig. 2, and compare it against state-of-the-art approaches that rely either on circularity constraint or hand-defined factorizations of style and content. We also show the advantage over classical domain adaptation [47], which can not cope with large geometric differences. Not only does our method generate more realistic images, but more importantly, when we use the images it generates to train pose estimators we get more accurate results. Nothing in our approach is specific to the animals we worked with and that could also be applied just as well to limbed vertebrates, including rodents and primates, as shown in Fig. 3 for person to person transfer with large shape differences.

Our code and fly dataset is available on github.

2. Related Work

We present a method for spatially consistent image domain adaptation and pose estimation. In the following sections, we discuss recent advances towards this goal.

Pose Estimation. Deep learning based human pose estimation methods have recently made great progress. This is especially true for capturing human movements for which there is enough annotated data to train deep networks [49, 16, 25, 43, 27, 1]. A large corpus of the literature focuses on prediction of 2D key points from images directly [31, 53, 46, 15, 55, 54]. There is also a wide literature on capturing 3D pose directly from images, or as a function of 2D keypoints instead [35, 29, 39, 32, 61, 44, 33]. Weakly [60] and semi-supervised algorithms [50] can further improve the performance of motion capture systems, for example by using multi-view constraints [38, 56].

Approaches designed primarily for human pose have recently been transferred to study large animals, like cheetahs and lab mice [30]. [63] uses a model based algorithm, trains on synthetic renderings, and refines on real zebra photographs. However, their quadruped body model does not translate to animals with a different number of legs and the suggested direct training on synthetic images for initialization did not succeed in our experiments, likely because realistic models are not available for our cases.

For pose estimation in *Drosophila*, DeepLabCut provides a user-friendly interface to DeeperCut [30], LEAP [34] tracks limb and appendage landmarks, and DeepFly3D leverages multiple views to capture 3D pose [13]. Nevertheless, all these methods require large amounts of manual labels, which are not available for many animals and cannot be reused when recording the same species in different environments and illumination conditions.

Paired Image-to-Image Translation. Supervised image-to-image translation methods aim to translate images across domains (e.g., day-to-night, summer-to-winter, photo-to-painting), often using adversarial methods [17] to learn a mapping from input to output images. More recent studies

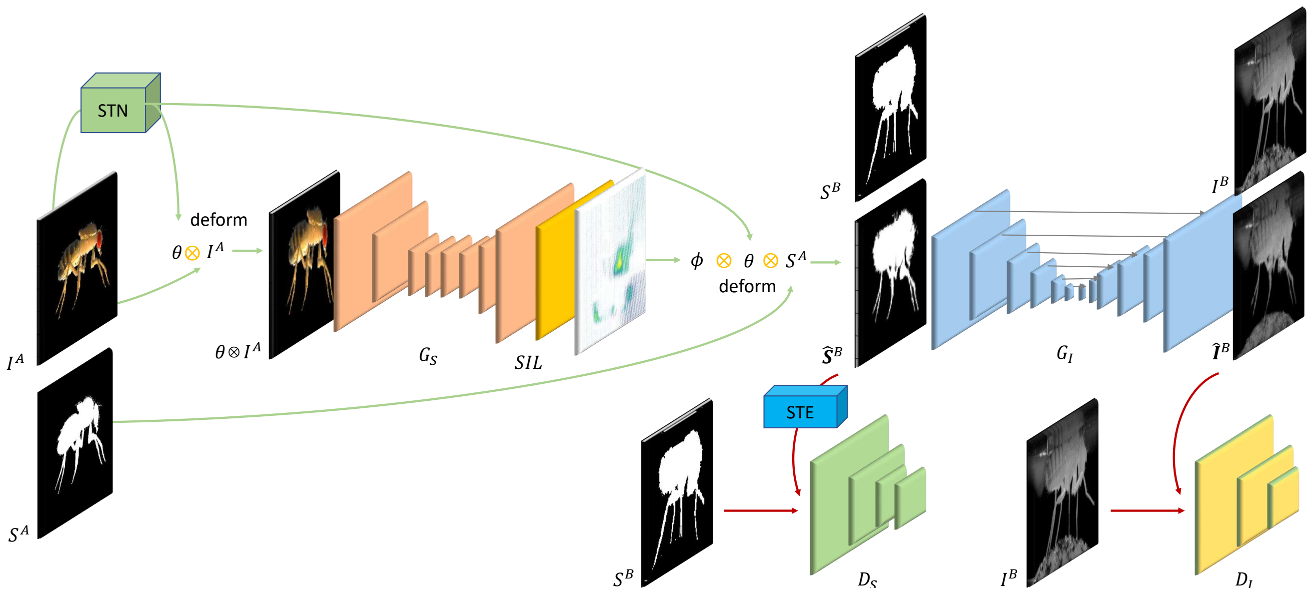

Figure 4. **Overview of our deformation-based image translation method.** Our model has two steps. In the first step, the deformation from source domain A to target domain B is estimated for input image $\mathbf{I}^A$ and it's silhouette $\mathbf{S}^A$ via network G_S and a Spatial Transformer Network (STN). Their output is an explicit deformation field parameterized by the global, affine transformation θ and a local, non-linear warping ϕ, using a spatial integral layer (SIL).Then, the deformed silhouette is transformed into the full output image $\hat{\mathbf{I}}^B$ with image generator G_I. Discriminators D_S and D_I enable unpaired training. D_S uses the Straight Through Estimator (STE) for backpropagation.

have aimed to translate edges to images [41] and cascaded networks are used to condition on semantic label maps [5]. However, in our setting, no paired examples are available.

Style Transfer. Style transfer is an image-to-image translation method that works on unpaired examples, aiming to transfer the input style while preserving the geometry of the target image [11, 20, 9, 48]. Initial deep learning approaches optimized an image with respect to the Gram matrix statistics of deep features of the target image [10, 11]. More recent studies tested other architectures and loss functions [24] and use a contextual loss to transfer the style at the semantic level [28, 19]. Our work differs from style transfer as we support significant changes in pose and shape.

Unsupervised Image Domain Translation. Another line of work trains neural networks on unpaired examples for domain translation, including sim2real mappings. Early approaches used weight-sharing [26, 62] and sharing of specific content features [4, 42]. The cycle consistency in Cycle-GAN, which assumes bijective mapping between two domains, can map from zebra to horse [62, 23, 14], but bridging large deformations across domains, such as for going from cat to dog and even more in our case (see Fig. 2), requires alternative network architectures [12] or intermediate keypoint representations [52]; However, none of the methods discussed above establish a fine-grained, dense spatial correspondence between source and target, which prevents accurate transfer of desired keypoint locations.

Deformation networks. Explicit deformation has been used in diverse contexts. The spatial transformer network (STN) made affine and non-parametric spatial deformations popular as a differentiable network layer [18]. These approaches have been used to zoom in on salient objects [37], disentangle shape and appearance variations in an image collection [59], and register (brain scan) images to a common, learned template image [6, 2, 22, 40]. [8] introduced global transformation into the Cycle-GAN framework. While similar in spirit, additional advances beyond these approaches are still required to model deformations faithfully on our unpaired translation task.

3. Method

Our goal is to translate pose annotations and images from a synthetic domain A to a target domain B for which only unpaired images $\{\mathbf{I}_i^A\}_{i=1}^N$ and $\{\mathbf{I}_i^B\}_{i=1}^K$ exist. In our application scenario, the target examples are frames of video recordings of a living animal and the source domain are simple drawings or computer graphics renderings of a character animated with random deformations of the limbs. Both domains depict images of the same species, but in different pose, shape, and appearance.

Fig. 4 summarizes our approach. To tackle the problem of translating between domains while preserving pose correspondence, we separately transfer spatially varying shape changes via explicit deformation of the source images via an intermediate silhouette representation $\hat{\mathbf{S}}^B$ (Section 3.1). Subsequently, we locally map from silhouette to real appearance (Section 3.2). The final goal is to train a pose estimator on realistic images from synthetic examples (Section 3.3). Our challenge then becomes to train neural networks for each, without requiring any paired examples or keypoint annotation on the target domain. To this end, we set up adversarial networks that discriminate differences with respect to the target domain statistics. Learning of the image translation is performed jointly on the objective

$$\mathcal{L} = \mathcal{L}_I + \mathcal{L}_S + \mathcal{R}_{\mathcal{D}}, \tag{1}$$

where $\mathcal{L}_I$ and $\mathcal{L}_S$ are the adversarial losses on generated segmentation and image, and $\mathcal{R}_D$ is a regularizer on the deformation grid. Besides images $\mathbf{I}$, our method operates on segmentation masks $\mathbf{S}$ of the same resolution. The domain origin is denoted with superscripts—$\mathbf{I}^A$, and the domain target (real images) is denoted $\mathbf{I}^B$. We use several generator and discriminator networks, which we denote G and D, respectively, with subscripts differentiating the type—G_I. We explain each step in the following section.

3.1. Spatial Deformation

Our experiments showed that using a single, large discriminator, as done by existing techniques, leads to overfitting and forces the generator to hallucinate, due to the limited and unrealistic pose variability of the simulated source. We model shape explicitly through the intermediate silhouette representation and its changes with a per-pixel deformation field, as shown in Fig. 5. The silhouette lets us setup independent discriminators with varying receptive field; large for capturing global shape and small to fill-in texture. Moreover, the deformation field enables the desired pose transfer while bridging large shape discrepancies.

The first stage is a generator that takes a synthetic image $\mathbf{I}^A$ and mask $\mathbf{S}^A$ as input, and outputs a deformed segmentation mask $\hat{\mathbf{S}}^B$ that is similar to the shapes in B. To model global deformation, we use a spatial transformer network (STN) [18] that takes the synthetic image $\mathbf{I}^A \in \mathbb{R}^{C,H,W}$ as input, and outputs an affine matrix $\theta \in \mathbb{R}^{3,4}$, which models global scaling, translation and rotation differences between the source and target domains. It is trained jointly with a fully-convolutional generator network, G_S, which takes the globally transformed image as input and outputs $\phi \in \mathbb{R}^{2,H,W}$, a per-pixel vector field that models fine-grained differences in pose and shape. The vector at pixel location x in ϕ points to the pixel in the source domain that corresponds to x. Overlaying the source pixels of selected rows and columns of ϕ leads to the deformed grid visualized in Fig. 5. This hierarchical representation allows us to cope with varying degrees of discrepancies between the two domains. We refer to the combined application of these two networks as $\phi \otimes \theta \otimes \mathbf{S}^A$, where $\theta = \mathrm{STN}(\mathbf{I}^A)$, $\phi = G_S(\theta \otimes \mathbf{I}^A)$, and $\otimes$ denotes the transformation by global and local deformation.

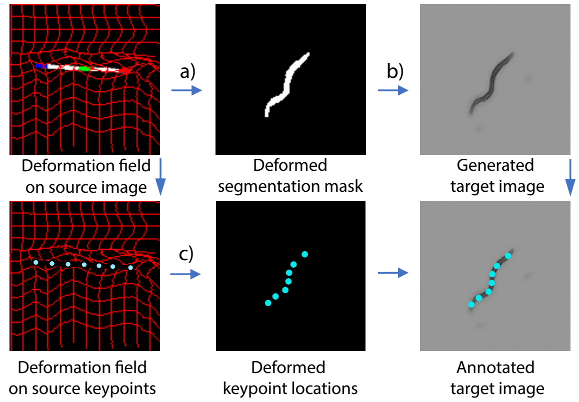

Figure 5. **Explicit deformation ensures transfer of keypoints.** The deformation field is inferred as part of a) source image segmentation to target image segmentation transfer (including global, affine transformation) and b) segmentation to target image translation. c) The same deformation field is applied to transfer known keypoints from source to target.

Training the STN and G_S requires silhouettes in A and B. Silhouettes $\mathbf{S}^A$ in the source domain are trivially obtainable from synthetic characters by rendering them on a black background. It is relatively easy to estimate $\mathbf{S}^B$ on a static background for the target domain as datasets are obtained in controlled lab environments. We will later demonstrate that our model is robust to remaining errors in segmentation.

The difficulty of our task is that all domain examples are unpaired, hence, a constraint can only be set up in the distributional sense. Thus, we train a shape discriminator D_S alongside G_S and STN and train them alternately to minimize and maximize the adversarial loss

$$\begin{aligned}\mathcal{L}_S &= \mathcal{L}_{GAN}(G_{\mathbf{S}}, D_{\mathbf{S}}, \mathbf{S}^A, \mathbf{S}^B) \\ &= \mathbb{E}_{\mathbf{S}^B}[\log D_S(\mathbf{S}^B)] + \mathbb{E}_{\mathbf{S}^A}[\log(1 - D_S(\phi \otimes \theta \otimes \mathbf{S}^A))],\end{aligned} \tag{2}$$

where the expectation is built across the training set of A and B. The adversarial loss is paired with the regularizer

$$\mathcal{R}_D = \alpha(\|\triangledown \phi_x(A)\|^2 + \|\triangledown \phi_y(A))\|^2) + \beta \|\phi(A)\|, \tag{3}$$

to encourage smoothness by penalizing deformation magnitude and the gradients of the deformation field, as in [59].

The inputs of the discriminator are binary masks from source domain A and target domain B. However, the deformed masks are no longer binary on the boundary because of the interpolation required for differentiation. Thus, it would be trivial for D_S to discriminate against the real and synthesized masks based on non-binary values. To overcome this issue, we threshold to get a binary mask. Although the threshold operation is not differentiable, we can still estimate the gradients with respect to G_S using a straight through estimator (STE) [58], which treats the threshold as the identity function during backpropagation, and therefore passes the gradients on to the previous layer.

Implementation details. Directly outputting a vector field leads to foldovers that make the training unstable. Instead, we parameterize it as the gradient of the deformation field ϕ, and enforce positivity to prevent foldovers as in [59]. ϕ can be recovered by summing the gradients across the image. The deformation from A to B is implemented with a spatial transformer layer (STL) that infers the value of deformed pixel locations by bilinear interpolation [18] and is differentiable. In contrast to [59], we use a fully convolutional network to learn the local deformation field. The G_S network consists of 3 Resnet blocks between downsampling/upsampling layers. The receptive field of the network is 64 pixels, $1/2$ of the image.

The STN network consists of 5 convolutional layers and a fully connected stub to output θ that is preceded by max-pooling and SELU units (this yielded better results in preliminary experiments, compared to ReLU activations).

3.2. Appearance Transfer

Once the shape discrepancies between the two domains have been estimated by ϕ, θ, we then generate the appearance of the target domain on the deformed silhouettes $\hat{\mathbf{S}}^B = \phi \otimes \theta \otimes \mathbf{S}^A$. We deploy a generator G_I that is configured to preserve the source shape, only filling in texture details. The input is $\hat{\mathbf{S}}^B$ and the output is a realistic image $\hat{\mathbf{I}}^B$ that matches the appearance of the target domain. We use a discriminator D_I for training, as synthetic and real images are unpaired. In addition, our choice of using the silhouette as an intermediate representation allows us to introduce a supervised loss on silhouette $\mathbf{S}_{\mathbf{I}^B}$, computed from real images $\mathbf{I}^B$ using background subtraction. The training objective is

$$\mathcal{L}_I = \lambda \mathcal{L}_{GAN}(G_I, D_I, \mathbf{I}^A, \mathbf{I}^B) + \left\| G_I(\mathbf{S}_{\mathbf{I}^B}) - \mathbf{I}^B \right\|, \quad (4)$$

where the GAN loss is defined as before and the second part is the supervised loss which stabilizes training.

Training the supervised loss in isolation without end-to-end training with the adversarial losses leads to artifacts since neither the synthesized nor silhouettes from real images are perfect, see Fig. 6.

The pose distribution of the simulated character can differ even after local and global deformation as some pose differences cannot be explained by an image deformation. For instance, the occlusion effects of crossing legs on *Drosophila* cannot be undone as a 2D image transformation. A discriminator with a large receptive field could detect these differences and re-position legs at locations without correspondence in the source. To counteract this issue, we make sure D_I has a small receptive field. This is possible without suffering from texture artifacts since the global shape deformation is already compensated by G_S and the texture can be filled in locally.

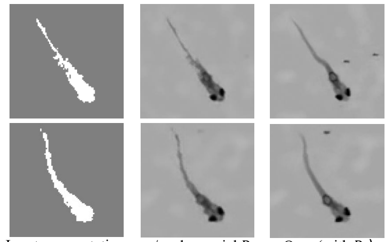
Input segmentation w/o adversarial D_I Ours (with D_I)

Figure 6. **Texture discriminator influence.** Without the adversarial discriminator, the image generator is disturbed by an irregular silhouette boundary. In our model, the adversarial D_I creates a link to the deformed silhouettes $\hat{S}^B$ enabling end-to-end training.

Implementation details. We use a 7-layer U-Net generator as our backbone network for image translation with G_I. The skip connections in the U-Net help the network preserve the spatial information. For D_I, we use a patch-wise discriminator, consisting of three 4x4 convolutional layers; the first one with stride two and the second one with instance normalization. All activation functions are leaky ReLU. The small receptive field of the patch discriminator additionally helps to maintain the spatial structure of the object and was sufficient in our experiment to reproduce the real appearances faithfully.

3.3. Pose Estimation

We use the stacked hourglass network architecture for pose estimation [31]. Stacked hourglass is a fully-convolutional network with several bottlenecks that takes an image $\mathbf{I}$ and outputs a heatmap $\mathbf{H}$ of the same aspect ratio but at four times lower resolution due to pooling at the initial layers. The heatmaps $\mathbf{H}$ are a stack of 2D probability maps with Gaussian distribution, where the maximum value of each channel in the stack indicates one specific joint location. Because our source images are synthesized from 3D character models, we can use the virtual camera matrix to project 3D keypoints, such as the knee joint, onto the image.

To obtain annotations in the target domain, we conveniently use the image deformation operation $\hat{\mathbf{H}}^B = \phi \otimes \theta \otimes \mathbf{H}^A$ to compute the deformed heatmap $\hat{\mathbf{H}}^B$ that matches to the synthesized target domain image $\hat{\mathbf{I}}^B = G_I(\phi \otimes \theta \otimes \mathbf{I}^A)$, with ϕ coming from G_S and θ from the STN. Note, this is only possible due to the explicit handling of deformations.

Having synthesized realistic examples of the target domain and transferred ground truth heatmaps, it remains to train the pose estimation network in a supervised manner. We use the L_2 loss between the predicted and ground truth heatmaps. At test time, we estimate the corresponding joint location as the argmax of the predicted heatmap, as usual in the pose estimation literature. Implementation details are given in the supplemental document.

4. Evaluation

In this section, we qualitatively compare our results to canonical baselines and variants of our algorithm, in order to highlight advantages and remaining shortcomings both visually and quantitatively. This includes the task of 2D keypoint localization on the target domain. We test our approach on different neuroscience model organisms in order to demonstrate varying complexity levels of deformation and generality to different conditions. Additional qualitative results and comparisons are given in the supplemental document.

All input and output images are of dimension $(128, 128)$. We operate on gray-scale images, i.e. channel dimension $C = 1$, obtained from infrared cameras, which are commonly used in neuroscience experiments in order to avoid inadvertent visual stimulation. Nevertheless, our method extends naturally to color images.

Datasets. We test on available zebrafish and worm image datasets, by [21] and [57, 45], using 500 and 1000 real images for unpaired training. To quantify pose estimation accuracy, we manually annotate a test set of 200 frames with three keypoints (tail and eyes) for the zebrafish and two points (head and tail) for the worm. In these datasets, the background is monochrome and is removed by color keying to obtain the foreground masks. Because of the simplicity of these models, we use a simple, static stick figure as a source image that is augmented by uniformly random rotation and translation. Fig. 7 gives example images.

For human examples in Fig. 3, we use two walking sequences from endlessreference.com of a slim and ample man with length 250 and 137 frames, respectively, to train our unpaired image translation model.

Our most challenging test case is the *Drosophila* fly. We use the subset of the dataset published alongside [13], which contains transitions between different speeds of walking, grooming and standing captured from a side view and includes annotations for five keypoints for each of the fully-visible legs (four joints and tarsus tip). In this dataset, the fly is tethered to a metal stage of a microscope and the body remains stationary, yet the fly can walk on a freely rotating ball (spherical treadmill), see Fig. 7. To get the target domain segmentation masks, we first crop out the ball and background clutter with a single coarse segmentation mask. This mask is applied to all images due to the static camera setup. The body, including the legs, is then segmented by color keying on the remaining black background. Please note, that at test time, no manual segmentation is used. We use 815 real images for unpaired training and 200 manually annotated images for testing. On the source side, we render

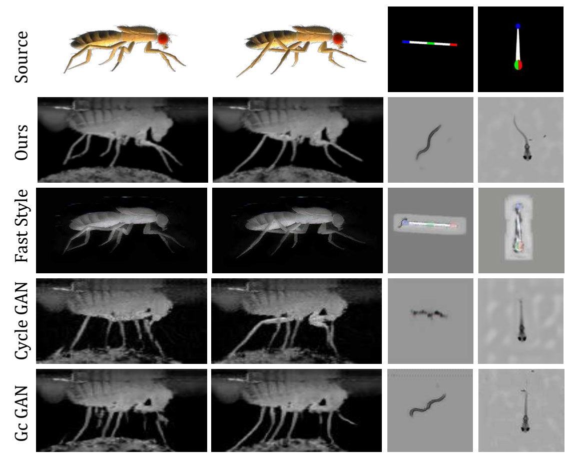

Figure 7. **Qualitative comparison.** Existing unpaired image translation methods can generate realistic images on worm and fish, but exhibit artifacts for the thin legs of the *Drosophila* and zebrafish examples. Ours succeeds on all three classes.

1500 synthetic images using an off-the-shelf Maya model from turbosquid.com. The source motion is a single robotic walk cycle from [36] which we augment by adding random Gaussian noise to the character control handles. This increases diversity but may lead to unrealistic poses that our deformation network helps to correct.

Metrics. The pose estimation accuracy is estimated as the root mean squared error (RMSE) of predicted and ground truth 2D location and percentage of correct keypoints (PCK), the ratio of predicted keypoints below a set threshold. We report results for thresholds ranging from 2 to 45 pixels. We also provide accumulated error histograms and the average PCK difference as the area under the curve (AUC) of the error histogram, to analyze the consistency of the improvements.

In many cases, it is impossible, even for a human, to uniquely identify the leg identity for *Drosophila*. As in [13], we therefore only evaluate the three entirely visible legs. Moreover, we find at test time the optimal leg assignment across the three legs and refer to these permutation invariant (PI) metrics PI-RMSE, PI-PCK, and PI-AUC. Because the worm is tail-head symmetric, we compute errors for front-to-back and back-to-front ordering of joints and return the minimum. The pose estimation task lets us quantify the made improvements, both due to more realistically generated images (image quality), as well as the preservation of correspondences (geometric accuracy) since the lack of one would already lead to poor pose estimation.

To independently quantify the image quality, we use the structural similarity (SSIM) index [51]. We measure the similarity between all generated images $\hat{\mathbf{I}}^B$ (for every $\mathbf{I}^A$ in A) with a pseudo-randomly sampled reference image $\mathbf{I}^B$.

Task	*D.M.*	*C.E.*	*D.E.*
Fast-Style-Transfer	0.3932	0.0539	0.6385
Cycle-GAN	0.6543	0.9034	0.8504
Gc-GAN	0.6392	0.8915	0.8586
Ours	**0.6746**	**0.9076**	**0.8771**

Table 1. **Structured similarity (SSIM) comparison**. The explicit modeling of deformation outperforms baselines, particularly on the complex *Drosophila* images showing complex poses.

Baselines. We compare to Fast-Style-Transfer [7], which combines [10, 20, 48], Cycle-GAN [62] and Gc-GAN [8]. With the latter being a state-of-the-art method for image to image translation and the former used to validate that simpler solutions do not succeed.

We compare pose estimation with the same architecture, trained directly on the synthetic images, images generated by the above mentioned methods, and on manual annotations of real training images (185 for *Drosophila*, 100 for worm, and 100 for fish). To also compare to domain adaptation methods, we adopt the pipeline of ADDA [47] for pose estimation. The original ADDA [47] transfers domains in a vector feature space. Instead, we use the hourglass network for feature extraction, replacing the vector space into spatial feature maps which preserves the spatial pose information. The supplemental document provides additional details.

4.1. Quality of Unpaired Image Translation

The quality of Cycle and Gc-GAN is comparable to ours on the simple worm and fish domains, as reflected visually in Fig. 7 and quantitatively in terms of SSIM in Table 1. For *Drosophila*, our method improves image quality (0.67 vs. 0.39, 0.63 and 0.65). Albeit the core of explicit deformation was to transfer pose annotations across domains, this analysis shows that an explicit mapping and incorporation of silhouettes regularizes and leads to improved results. For instance, it ensures that thin legs of the fly are completely reconstructed and that exactly six legs are synthesized, while Cycle-GAN and Gc-GAN hallucinate additional partial limbs.

4.2. Pose Domain Transformation

Fig. 8 shows that our method faithfully transfers 2D keypoints, obtained for free on synthetic characters, to the target domain. The transferred head and tail keypoints on the worm and fish correspond precisely to the respective locations in the synthesized images, despite having a different position and constellation in the source. This transfer works equally well for the more complex *Drosophila* case. Only occasional failures happen, such as when a leg is behind or in front of the torso, rendering it invisible in the silhouette. Moreover, the eyes of the fish are not well represented in the silhouette and therefore sometimes missed by our silhouette deformation approach.

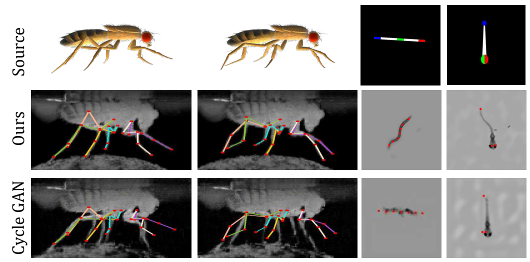

Figure 8. **Automatic Pose Annotation.** Our method faithfully transfers poses across domains, while Cycle-GAN, the best performing baseline, loses correspondence on all three datasets.

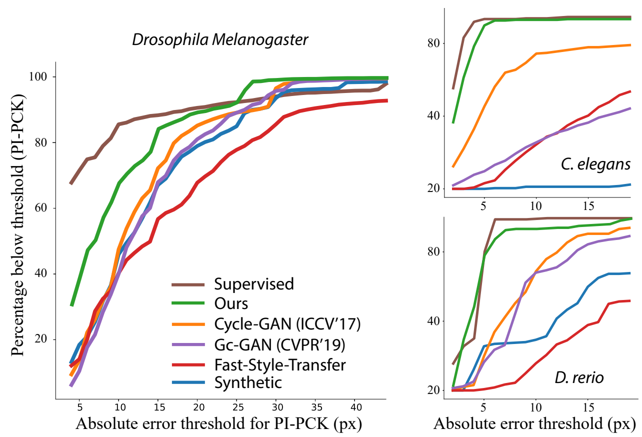

Figure 9. **Pose estimation accuracy.** The accumulated error curves show the accuracy (vertical axis) for different PCK thresholds (horizontal axis). Our method clearly outperforms the baselines and approaches the manually supervised reference.

Metric	*Drosophila Melanogaster* PI-PCK ↑ (5 pix)	PI-PCK ↑ (15 pix)	PI-AUC ↑ (4-45 pix)	PI-RMSE ↓ (pix)
Synthetic	19.8	67.9	75.75	13.456
Fast-Style-Transfer	15.4	57.6	68.9	17.309
Gc-GAN	11.9	68.7	76.3	13.175
Cycle-GAN	15.0	72.9	78.4	12.302
Ours	**40.0**	**84.7**	**86.0**	**8.823**
Supervised	72.2	88.8	90.35	6.507

Table 2. **Pose estimation accuracy comparison on *Drosophila Melanogaster*.** A similar improvement as for Drosophila is attained on the other tested laboratory animals, with a particularly big improvements on the zebrafish.

By contrast, existing solutions capture the shape shift between the two domains, but only implicitly, thereby loosing the correspondence. Poses that are transferred one-to-one from the source do no longer match with the keypoint location in the image. Keypoints are shifted outside of the body, see last column of Fig. 8. The style transfer maintains the pose of the source, however, an appearance domain mismatch remains. We show in the next section that all of the above artifacts lead to reduced accuracy on the downstream task of pose estimation.

Metric	*Caenorhabditis elegans* PI-PCK ↑ (5 pix)	PI-AUC ↑ (2-20 pix)	PI-RMSE ↓ (pix)	*Danio rerio* PCK ↑ (10 pix)	AUC ↑ (2-20 pix)	RMSE ↓ (pix)
Synthetic	0.0	0.9	67.29	29.3	37.4	20.15
Fast-Style-Transfer	3.1	25.0	20.50	15.6	20.8	19.25
Gc-GAN	9.7	25.0	27.38	68.2	54.5	27.38
Cycle-GAN	45.3	63.2	14.71	68.7	59.1	9.70
Ours	**90.3**	**87.6**	**5.36**	**93.9**	**83.1**	**4.50**
Supervised	94.6	92.3	3.77	99.6	86.5	3.91

Table 3. **Pose estimation accuracy on *C. elegans* and *D. rerio*.** Our method significantly outperforms all baselines and approaches the supervised baseline. Units are given in round brackets.

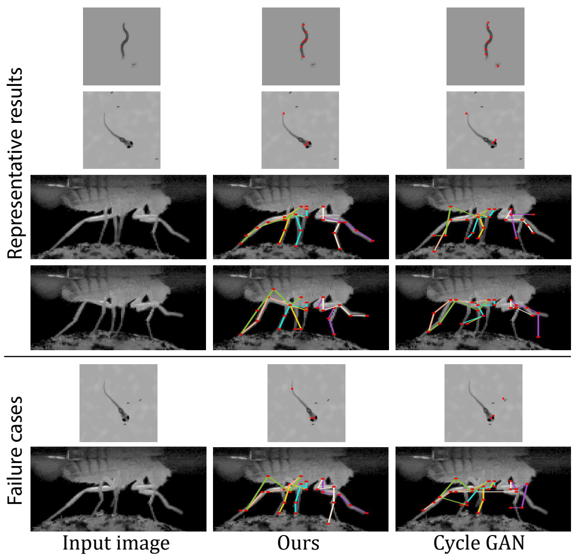

Figure 10. **Qualitative pose estimation results.** The estimator provides decent results across all three animals. Occasional failures (last two rows) happen when legs cross, at occlusions, and for the fine fish tail. Training on Cycle-GAN images does not succeed.

4.3. 2D Pose Estimation

The primary objective of this study is to demonstrate accurate keypoint detection on a target domain for which only annotations on synthetic images with different shape and pose exist. Fig. 10 shows qualitative results. We compare the performance of the same keypoint detector trained on images and keypoints generated by ours and the baseline methods. The absolute errors (tables 2 and 3) and accumulated error histograms (Fig. 9) show significant (PCK 15: 84.7 vs. 72.9 Cycle-GAN) and persistent (AUC 86.0 vs 78.4) improvements for *Drosophila* and the other domains. Even bigger gains are visible for the simpler worm and zebrafish datasets. Although there remains a gap compared to training on real images with manual labels for small error thresholds, our method comes already close to the supervised reference method in PCK 15 and above and has a large margin on existing unpaired image translation methods.

Ablation Study on Fly. We compared our full model at PI-PCK-15 (84.7), to not using one of our core contributions: no deformation (64.9), only global affine (57.4), only local non-linear (79.2), and directly encoding a vector field (69.1). The numbers and Fig. 11 shows that all contributions are important. Also end-to-end training with D_I is important, as shown in Fig. 6, and by additional examples in the supplemental document. Moreover, using ADDA (55.5 PI-PCK-15), did not suffice to bridge the large domain gap.

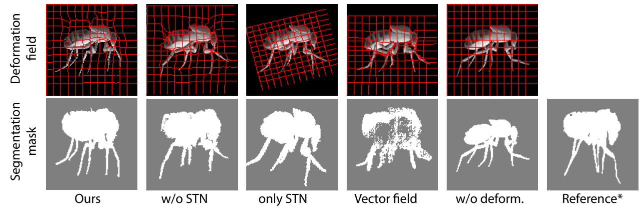

Figure 11. **Ablation study.** All our contributions are important: removing the global STN reduces local details (bends legs), only global transformation misses pose differences (thinner, straight legs), and predicting the vector field directly produces foldovers. *The reference silhouette is from an unpaired target image.

5. Limitations and Future Work

For some domains the assumption of a target segmentation mask is constraining. For instance, for transferring synthetic humans to real images on cluttered backgrounds. We plan on integrating unsupervised segmentation, as demonstrated by [3] for single-domain image generation. Although we could synthesize a variety of poses for the worm and fish using a single stylized source image, our method was not able to synthesize entirely unseen *Drosophila* poses, because crossing legs could not be modeled using a 2D image deformation. Moreover, symmetries and self-similarities can lead to flipped limb identities (see bottom of Fig. 10). We plan to use temporal cues and multiple views to find a consistent assignment in the future, following ideas used in [56] for humans and monkeys.

6. Conclusion

In this paper, we have presented an approach for translating synthetic images to a real domain via explicit shape and pose deformation that consistently outperforms existing image translation methods. Our method allows us to train a pose estimator on synthetic images that generalize to real ones; without requiring manual keypoint labels.

One of our test cases is on *Drosophila* tethered to a microscope used to measure neural activity. By combining pose estimation with state-of-the-art microscopy, we anticipate more rapid advances in understanding the relationship between animal behaviour and neural activity.

Acknowlegments

SG, PR, and PF acknowledge support from an EPFL SV iPhD grant

References

[1] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *CVPR*, pages 3686–3693, 2014. 2

[2] Guha Balakrishnan, Amy Zhao, Mert Sabuncu, John Guttag, and Adrian V. Dalca. Voxelmorph: A learning framework for deformable medical image registration. *IEEE TMI: Transactions on Medical Imaging*, 2019. 3

[3] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. *arXiv preprint arXiv:1905.12663*, 2019. 8

[4] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *CVPR*, volume 1, page 7, 2017. 3

[5] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *ICCV*, volume 1, page 3, 2017. 3

[6] Adrian V Dalca, Marianne Rakic, John Guttag, and Mert R Sabuncu. Learning conditional deformable templates with convolutional networks. *NeurIPS*, 2019. 2, 3

[7] Logan Engstrom. Fast style transfer. `https://github.com/lengstrom/fast-style-transfer/`, 2016. 7

[8] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, Kun Zhang, and Dacheng Tao. Geometry-Consistent Generative Adversarial Networks for One-Sided Unsupervised Domain Mapping. In *CVPR*, 2019. 3, 7

[9] Leon A Gatys, Matthias Bethge, Aaron Hertzmann, and Eli Shechtman. Preserving color in neural artistic style transfer. *arXiv preprint arXiv:1606.05897*, 2016. 3

[10] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015. 3, 7

[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *CVPR*, pages 2414–2423, 2016. 3

[12] Aaron Gokaslan, Vivek Ramanujan, Daniel Ritchie, Kwang In Kim, and James Tompkin. Improving shape deformation in unsupervised image to image translation. In *ECCV*, 2018. 3

[13] Semih Gunel, Helge Rhodin, Daniel Morales, João Compagnolo, Pavan Ramdya, and Pascal Fua. Deepfly3d, a deep learning-based approach for 3d limb and appendage tracking in tethered, adult drosophila. In *eLife*, 2019. 2, 6

[14] Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *NeurIPS*, pages 820–828, 2016. 3

[15] Eldar Insafutdinov, Leonid Pishchulina, Bjoern Andres, Mykhaylo Andriluka, and Bernt Schiele. Deepercut: A deeper, stronger, and faster multiperson pose estimation model. In *ECCV*, 2016. 2

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *TPAMI*, 2014. 2

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017. 2

[18] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial Transformer Networks. In *NeurIPS*, pages 2017–2025, 2015. 3, 4, 5

[19] Yongcheng Jing, Yezhou Yang, Zunlei Feng, Jingwen Ye, Yizhou Yu, and Mingli Song. Neural style transfer: A review. *arXiv*, 05 2017. 3

[20] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711. Springer, 2016. 3, 7

[21] Robert Evan Johnson, Scott Linderman, Thomas Panier, Caroline Lei Wee, Erin Song, Kristian Joseph Herrera, Andrew Miller, and Florian Engert. Probabilistic models of larval zebrafish behavior reveal structure on many scales. *Current Biology*, 30(1):70–82, 2020. 6

[22] Boah Kim, Jieun Kim, June-Goo Lee, Dong Hwan Kim, Seong Ho Park, and Jong Chul Ye. Unsupervised deformable image registration using cycle-consistent cnn. In *MICCAI*, pages 166–174. Springer, 2019. 3

[23] Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. *ICML*, 2017. 3

[24] Chuan Li and Michael Wand. Combining markov random fields and convolutional neural networks for image synthesis. In *CVPR*, pages 2479–2486, 2016. 3

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755. Springer, 2014. 2

[26] Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *NeurIPS*, pages 469–477, 2016. 3

[27] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. Amass: Archive of motion capture as surface shapes. In *ICCV*, Oct 2019. 2

[28] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. *ECCV*, 2018. 3

[29] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theoballt. Vnect: Real-Time 3D Human Pose Estimation with a Single RGB Camera. In *SIGGRAPH*, 2017. 2

[30] Tanmay Nath, Alexander Mathis, An Chi Chen, Amir Patel, Matthias Bethge, and Mackenzie W Mathis. Using deeplabcut for 3d markerless pose estimation across species and behaviors. *Nature Protocols*, 2019. 2

[31] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked Hourglass Networks for Human Pose Estimation. *ECCV*, pages 483–499, 2016. 2, 5

[32] Georgios Pavlakos, Xiaowei Zhou, Konstantinos G. Derpanis, and Kostas Daniilidis. Harvesting Multiple Views for Marker-Less 3D Human Pose Annotations. In *CVPR*, 2017. 2

[33] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *CVPR*, 2019. 2

[34] Talmo D Pereira, Diego E Aldarondo, Lindsay Willmore, Mikhail Kislin, Samuel S-H Wang, Mala Murthy, and Joshua W Shaevitz. Fast animal pose estimation using deep neural networks. *Nature methods*, 16(1):117, 2019. 2

[35] Alin-Ionut Popa, Mihai Zanfir, and Cristian Sminchisescu. Deep Multitask Architecture for Integrated 2D and 3D Human Sensing. In *CVPR*, 2017. 2

[36] Pavan Ramdya, Robin Thandiackal, Raphael Cherney, Thibault Asselborn, Richard Benton, Auke Jan Ijspeert, and Dario Floreano. Climbing favours the tripod gait over alternative faster insect gaits. *Nature communications*, 8:14494, 2017. 6

[37] Adria Recasens, Petr Kellnhofer, Simon Stent, Wojciech Matusik, and Antonio Torralba. Learning to zoom: a saliency-based sampling layer for neural networks. In *ECCV*, pages 51–66, 2018. 3

[38] Helge Rhodin, Victor Constantin, Isinsu Katircioglu, Mateus Salzmann, and Pascal Fua. Neural Scene Decomposition for Human Motion Capture. In *CVPR*, 2019. 2

[39] Grégory Rogez, Philippe Weinzaepfel, and Cordelia Schmid. Lcr-Net: Localization-Classification-Regression for Human Pose. In *CVPR*, 2017. 2

[40] Daniel Rueckert, Paul Aljabar, Rolf A Heckemann, Joseph V Hajnal, and Alexander Hammers. Diffeomorphic registration using b-splines. In *MICCAI*, pages 702–709. Springer, 2006. 3

[41] Patsorn Sangkloy, Jingwan Lu, Chen Fang, Fisher Yu, and James Hays. Scribbler: Controlling deep image synthesis with sketch and color. In *CVPR*, volume 2, 2017. 3

[42] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, volume 2, page 5, 2017. 3

[43] L. Sigal, A. Balan, and M. J. Black. Humaneva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. *IJCV*, 2010. 2

[44] Xiao Sun, Jiaxiang Shang, Shuang Liang, and Yichen Wei. Compositional Human Pose Regression. In *ICCV*, 2017. 2

[45] Balazs Szigeti, Padraig Gleeson, Michael Vella, Sergey Khayrulin, Andrey Palyanov, Jim Hokanson, Michael Currie, Matteo Cantarelli, Giovanni Idili, and Stephen D. Larson. Openworm: an open-science approach to modeling caenorhabditis elegans. *Front. Comput. Neurosci.*, 2014. 6

[46] Wei Tang and Wu Ying. Deeply learned compositional models for human pose estimation. In *ECCV*, 2018. 2

[47] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial Discriminative Domain Adaptation. In *CVPR*, pages 7167–7176, 2017. 2, 7

[48] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 3, 7

[49] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, and Ivan Laptev an Cordelia Schmid. Learning from Synthetic Humans. In *CVPR*, 2017. 2

[50] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *CVPR*, June 2019. 2

[51] Zhou Wang, Alan Conrad Bovik, Hamid Rahim Sheikh, Student Member, Eero P. Simoncelli, and Senior Member. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13:600–612, 2004. 6

[52] Wayne Wu, Kaidi Cao, Cheng Li, Chen Qian, and Chen Change Loy. Transgaga: Geometry-aware unsupervised image-to-image translation. In *CVPR*, 2019. 3

[53] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking. In *ECCV*, pages 466–481, 2018. 2

[54] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018. 2

[55] W. Yang, S. Li, W. Ouyang, and X. Wang. Learning feature pyramids for human pose estimation. In *ICCV*, 2017. 2

[56] Yuan Yao, Yasamin Jafarian, and Hyun Soo Park. Monet: Multiview semi-supervised keypoint detection via epipolar divergence. In *ICCV*, pages 753–762, 2019. 2, 8

[57] Eviatar Yemini, Tadas Jucikas, Laura J Grundy, André E X Brown, and William R Schafer. A database of caenorhabditis elegans behavioral phenotypes. In *Nature Methods*, 2013. 6

[58] Penghang Yin, Jiancheng Lyu, Shuai Zhang, Stanley J. Osher, Yingyong Qi, and Jack Xin. Understanding straight-through estimator in training activation quantized neural nets. In *ICLR*, 2019. 5

[59] Riza Alp Guler Dimitris Samaras Nikos Paragios Zhixin Shu, Mihir Sahasrabudhe and Iasonas Kokkinos. Deforming autoencoders: Unsupervised disentangling of shape and appearance. In *ECCV*, 2018. 2, 3, 4, 5

[60] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Towards 3d human pose estimation in the wild: a weakly-supervised approach. In *ICCV*, pages 398–407, 2017. 2

[61] Xingyi Zhou, Qixing Huang, Xiao Sun, Xiangyang Xue, and Yichen Wei. Weakly-supervised transfer for 3d human pose estimation in the wild. In *ICCV*, volume 3, page 7, 2017. 2

[62] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *ICCV*, 2017. 3, 7

[63] Silvia Zuffi, Angjoo Kanazawa, Tanya Berger-Wolf, and Michael J. Black. Three-d safari: Learning to estimate zebra pose, shape, and texture from images "in the wild". In *ICCV*, 2019. 2