Few-shot generative model for skeleton-based human action synthesis using cross-domain adversarial learning

Kenichiro Fukushi Yoshitaka Nozaki Kosuke Nishihara
Kentaro Nakahara
Biometrics Research Laboratories, NEC Corporation, Japan
{k-fukushi, yoshitaka-nozaki, koske, k-nakahara}@nec.com

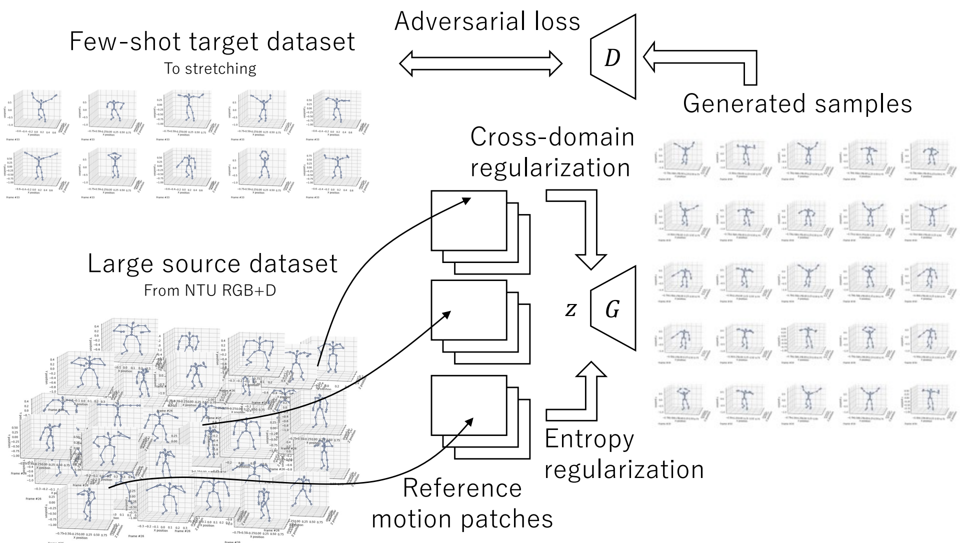


Figure 1. Our generative models learn to synthesize skeleton-based human actions using few samples of the target domain. We exploit a large public dataset to tackle the overfitting problem. Our cross-domain and entropy regularization guide the model to having variations in the generated actions. In this figure, stretching actions are generated using only 10 samples as the target domain data in adversarial training.

Abstract

We propose few-shot generative models of skeleton-based human actions on limited samples of the target domain. We exploit large public datasets as a source of motion variations by introducing novel cross-domain and entropy regularization losses that effectively transfer the diversity of the motions contained in the source to the target domain. First, target samples are divided into patches, which are a set of short motion clips. For each patch, we search for a reference motion from the source dataset that is similar to the patch. Next, in adversarial training, our cross-domain regularization encourages the generated sequences to resemble the reference motion at the patch level. Entropy regularization prevents mode collapse by forcing the generator to follow the distribution of the source dataset. Experiments are performed on public datasets where we utilize three action classes from NTU RGB+D 120 as the target and all data of 60 action classes in NTU RGB+D as the source. Ten samples for each target action class, 30 in total, are selected as target data. The results demonstrate that data augmented with the proposed method improve recognition accuracy by 28 % using a ST-GCN classifier.

1. Introduction

Human action recognition (HAR) has been in the midst of rapid advancement, benefitting from the progress of artificial intelligence such as deep learning. HAR involves many applications such as human-behavior monitoring, human-computer interaction in different areas of education, entertainment, medical, and sports.

Recognizing human actions from a video (i.e., a sequence of RGB images) has become an area of particular interest. Convolutional neural networks (CNNs) became widely used for vision-based HAR. Tran *et al*. [41] proposed a spatio-temporal convolution network called a 3D CNN. 3D CNNs were further extended using other image recognition techniques such as Inception-v1 [3] or ResNet [16], and by training with large datasets, recognition has improved greatly. A video of a person contains a large amount of information on his or her body positioning; thus, high accuracy recognition should be possible with video-based HAR. However, training deep networks with large datasets tends to be computationally intensive due to the data size of videos. In addition, changes in illumination and occlusion reduce the robustness of recognition.

Skeleton-based models are being increasingly utilized for HAR [31, 47]. Skeletons are expressed as a sequence of two- or three-dimensional coordinates of the joints in the human body. Skeletons are a compact but complete representation of human poses; thus, minimal computation is required to process skeleton data without sacrificing accuracy. Spatial temporal graph convolutional network GCN (ST-GCN) [47] is the most well-known skeleton-based model for HAR, which made it possible to apply deep learning to skeletons by extending a GCN so as to model temporal dynamics of human actions. Additionally, the emergence of pose estimation methods from RGB images [2, 10, 21, 38] and related preprocessing methods [15, 25] have enhanced the applicability of skeleton-based models by enabling calculation of skeletons from any videos without special equipment such as depth cameras or motion capture devices.

Few-shot scenarios are often the case when training HAR models; this is considered a major obstacle to practical use because collecting human action data and annotating labels correctly are time consuming and labor intensive. Researchers have made significant efforts to disclose a variety of large public datasets: daily behaviors [9, 17, 26, 36], dancing [4], sports [37], gait [20], and work-specific motion such as logistics [29] and nurse care [23]. But what if you want to build a model applicable to a target domain that is not covered the above datasets? In many cases, domains of interest have a very limited collection of data, or it is necessary to obtain training and validation data by yourself where you may want to begin with just a few samples. However, small datasets potentially do not cover the entire distribution of the target domain, which will result in a reduction of recognition accuracy, known as overfitting.

Thus, augmenting few-shot training samples is fundamental in HAR. Some studies have utilized model-based approaches where they implement physical simulators to synthesize physically plausible human motion [1, 18, 34]. Cabrera *et al*. proposed a one-shot augmentation of hand gestures where arm movements are simulated as a set of inverse kinematic solutions with the constraints of minimum jerk and energy expenditure. Jiang *et al*. [18] considered musculotendon characteristics of the body to produce realistic behaviors. The difficulty with the model-based approaches is scalability and versatility because they rely on the hand-crafted formulation of the human body using domain knowledge.

In our work, we will explore generative approaches, which automatically find the patterns of human motion variation from training data. The limitation of state-of-the-art generative models for human action synthesis [6, 8, 11, 33, 40, 42, 46] is that they are validated with a large number of training samples. In few-shot scenarios, they suffer from overfitting problem resulting in generated samples of low quality. Hence, we aim at realizing a method to train generative models with limited target samples.

The proposed method leverages a large public dataset as a source domain and transfer the information on diversity of motion from source to target. The contributions of our study are as follows:

- The first few-shot generative models of human actions, to the best of our knowledge
- Novel cross-domain and entropy regularization losses from exploiting the variation within large public datasets
- Improved recognition accuracy using data augmented by the proposed method

2. Related work

Approaches for human action synthesis include video- and skeleton-based approaches, analogous to HAR. As previously mentioned, our approach is skeleton-based, though video generation is also an important research field because it has inspired skeleton-based methods. We begin by reviewing the video-based generative models and discussing studies related to few-shot generation, followed by skeleton-based generative models.

2.1. Video-based generative models

There has been a growing interest in using deep networks for generative modeling of visual data, particularly images and videos. Prior studies have been focused on video prediction that predicts future frames given some of the previous frames. Kalchbrenner *et al*. [19] proposed an encoder-decoder architecture where CNN encoders compute the temporal dependencies of the video tensor, and the PixelCNN decoder computes dependencies along the space and color dimensions. To generate new sequences rather than predict them, subsequent studies have used adversarial approaches. Saito *et al*. [35] proposed a two-phase model consisting of temporal and image generators. Tulyakov *et al*. [43] modeled the latent spaces for content and motion separately from which video frames are synthesized. Clark *et al*. [5] introduced two discriminators for spatial and temporal domains. Sun *et al*. [39] utilized two parallel generators that process content and motion individually. Gupta *et al*. [14] designed a recurrent-based generator and CNN-based discriminator.

2.2. Few-shot generation approaches

The purpose of few-shot generation is to successfully train generative models while avoiding overfitting. Most studies use an adaptation strategy, where a pre-trained model is guided to the target domain with a small number of real samples while inheriting the diversity of the source domain. Wang *et al*. [45] employed a transfer learning approach to fine-tune pre-trained GANs with as few as 1000 target images. Noguchi and Harada [30] proposed another transfer method where only scale and shift parameters in the generator are updated using –100 target images. Liu *et al*. [27] connected a detector and a GAN to explicitly improve the produced images for the downstream object detection task. Wang *et al*. [44] introduced the process of mining of GANs where subregions of the pre-trained generators are identified to generate samples close to the target domain. Mo *et al*. [28] showed that the freezing lower layers of the discriminator improved the effectiveness of fine-tuning. Li *et al*. [24] demonstrated image generation with less than 10 samples by regularizing the changes of the weights at each layer of the network. Subsequent studies have utilized cross-domain adversarial learning. Ojha *et al*. [32] used only ten training samples for image generation by introducing cross-domain consistency. Kwon and Ye [22] improved [32] by exploiting the CLIP space to achieve one-shot adaptation.

2.3. Skeleton-based generative models

Previous studies mostly involved autoregressive models, such as RNN [8, 11] and LSTM [42], which generate frames one by one. Subsequent studies have shown that generating entire sequences from latent vectors improves the quality of generated motion by capturing the long-term temporal structure using generative adversarial networks (GANs). Yan *et al*. [46] proposed a CNN-based model where the skeleton sequence is generated using latent vectors from a Gaussian process. Degardin *et al*. [6] proposed Kinetic-GAN, which uses ST-GCN to produce the generated samples from the latent space representation of a noise vector. Petrovich *et al*. [33] designed a Transformer-based architecture. Tevet *et al*. [40] devised a diffusion-based model.

Historically, most of these studies emerged from the field of computer graphics, so data augmentation is not necessarily their main research interest. However, some studies validated the effectiveness of generative models for data augmentation in HAR. Tu *et al*. [42] reported that the HAR accuracy increased by 4.2% when their augmented data was used to train a recognition model. Petrovich *et al*. [33] found that the augmented training is especially effective on low-data regimes. However, these methods are designed for scenarios that contain thousands of training motion samples.

3. Proposed Method

Our approach is to extend an existing generative model by introducing a novel regularization that is effective in few-shot scenarios. One of the state-of-the-art models, Kinetic-GAN [6], is used as a generative model in this paper. Thus, we start with the problem formulation on the basis of Kinetic-GAN and then describe the objective function using our method. WGAN-GP [12] is used in [6], which is expressed as:

$$\mathcal{L}_{adv}(G, D) = D(G(z)) - D(x) \tag{1}$$

$$\mathcal{L}_{gp}(D) = (\|\nabla_{\hat{x}} D(\hat{x})\| - 1)^2 \tag{2}$$

$$\mathbb{E}_{z \sim p_z(z), x \sim \mathcal{D}_t, \hat{x} \sim \mathbb{P}_{\hat{x}}} = \arg\min_G \max_D \mathcal{L}_{adv}(G, D) + \lambda_{gp} \mathcal{L}_{gp}(D) \tag{3}$$

where $\mathcal{D}_t$ is the target dataset, and $\mathbb{P}_{\hat{x}}$ is sampled uniformly along straight lines between pairs of points sampled from the target dataset $\mathcal{D}_t$ and generator distribution. The loss weight λ_{gp} for gradient penalty is set to 10 in all experiments.

Our goal is to successfully train a generator G on a small target dataset $\mathcal{D}_t$, given a large source dataset $\mathcal{D}_s$ which is different from the target domain. With $\mathcal{D}_t$ only, the training samples can be memorized by a discriminator. This causes overfitting where the discriminator forces the generator to make the samples from $\mathcal{D}_t$.

The key idea is to make the generator use the source dataset as a hint for valid motion variations. We hypothesize that, if the target motion is similar to the motion contained in the source dataset (we call this motion the reference motion), then the variations in the target motion should also be similar to that of the reference motion. We will handle target and source motion at the patch level, i.e., short motion clips, and find the reference motion patches for each target motion patch.

Before explaining the modified objective function, we will define the reference motion patches as illustrated in Fig. 2. For the k-th patch motion of the target samples, we first calculate $\mathbf{p}^*(k)$ as a nearest neighbor of the source patch motion:

$$\mathbf{p}^*(k) = \mathop{\arg\max}_{\mathbf{p} \in \{W(\overline{\mathcal{D}_s}(i), j)\}_{ij}} \mathrm{sim}\{W(\overline{\mathcal{D}_t}, k), \mathbf{p}\} \tag{4}$$

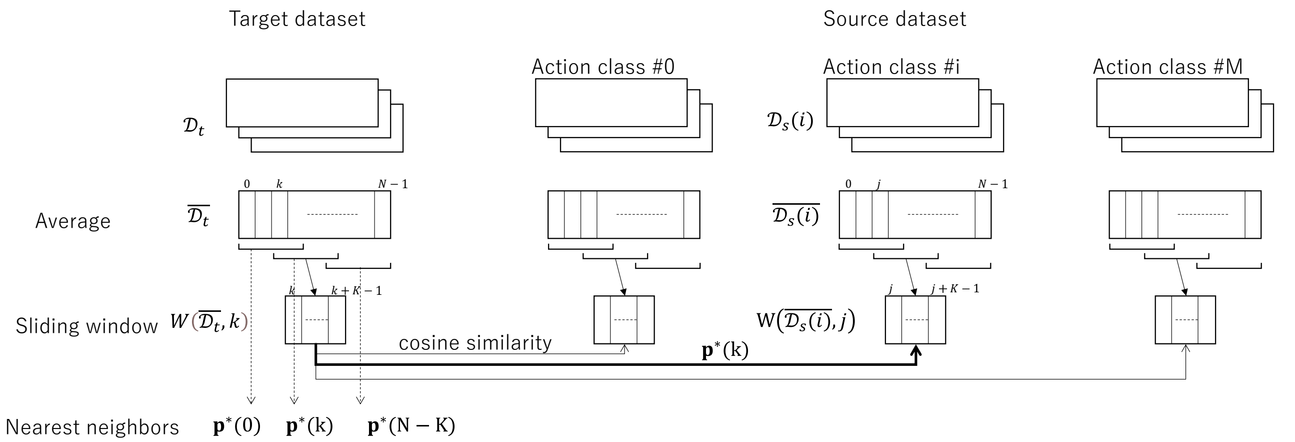


Figure 2. Reference motion patches are defined as a set of source motion patches from action class i that are similar to the target motion. Searching reference motion patches P_k for k-th target motion patch $W(\overline{\mathcal{D}_t}, k)$ is to find action class i and patch position j within the source dataset where the averaged motion $W(\overline{\mathcal{D}_s}(i), j)$ is the nearest neighbor $\mathbf{p}^*(k)$.

where $\overline{\mathcal{D}_t}$ is the mean target motion and $\overline{\mathcal{D}_s}(i)$ is the mean source motion for the action class i, which are also formulated as $\overline{\mathcal{D}_t} = \mathbb{E}_{\mathbf{x}_t \sim \mathcal{D}_t}(\mathbf{x}_t)$ and $\overline{\mathcal{D}_s}(i) = \mathbb{E}_{\mathbf{x}_s \sim \mathcal{D}_s(i)}(\mathbf{x}_s)$, respectively. Given the motion sample $\mathbf{x} = \{x_1, x_2, \cdots\}$, the sampling function $W(\mathbf{x}, k) = \{x_m | x_m \in \mathbf{x}, k \leq m < k + K\}$ enumerates a subset of $\mathbf{x}$ starting at the k-th frame with sliding window length K. sim represents the cosine similarity. Then the reference motion patches for the k-th patch of the target sample are defined as follows:

$$P_k = \{W(\mathbf{x}, j) | \mathbf{x} \in \mathcal{D}_s(i)\} \quad \text{for } i, j \text{ s.t.} \mathbf{p}^*(k) = W(\overline{\mathcal{D}_s}(i), j) \tag{5}$$

The proposed method transfers diversity from P_k, a subspace of the source domain conditioned on the target samples, while the existing method [32] transfers from an entire space. For example, upper body-dominant samples are selected as P_k when the target samples are also upper body-dominant.

By transferring the variation in the reference motion patches to the target domain, overfitting should be prevented, and learning should be feasible with few target samples. To carry this out in GAN training, we propose two regularization terms in the loss function. Cross-domain regularization guides the generator to remain consistent with the reference motions at the patch level. Entropy regularization encourages the generator to capture the distribution of the reference motions to enhance diversity. The concept of the two regularizations is illustrated in Fig. 3. Note that Ojha *et al*. [32] also employ patch-based loss; however, they apply it to the discriminator while we use it for the generator.

3.1. Cross-domain regularization

We encourage the generated samples to resemble the reference motion at the patch level. We formulate this as

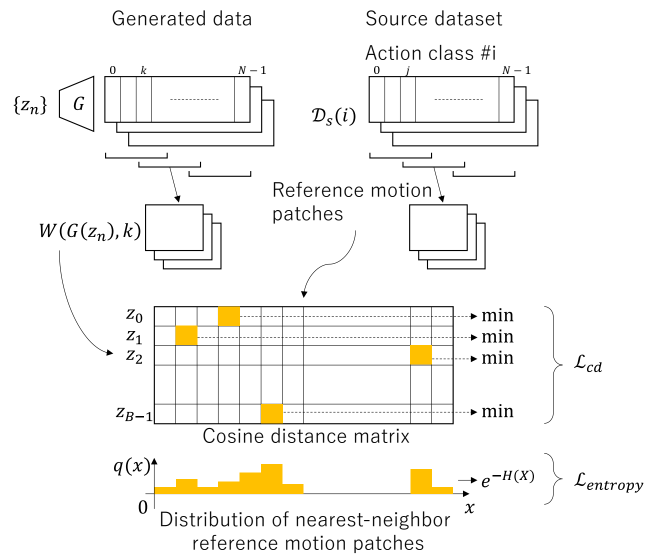


Figure 3. Cross-domain regularization $\mathcal{L}_{cd}$ is defined as the mean of minimum values along each row of the cosine distance matrix, which correspond to the nearest-neighbors within the reference motion patches for each noise vector z_n. Entropy regularization $\mathcal{L}_{entropy}$ is defined as the entropy for the normalized frequency of the nearest-neighbor reference motion patches along each column of the cosine distance matrix. These two losses are calculated with reference motion patches sampled from action class i, as shown in Eq. (5).

the distance of nearest-neighbor reference motion patches. Given the generated samples from a batch of B noise vectors $\{z_n\}_0^{B-1}$, the loss function is expressed as:

$$\mathcal{L}_{cd}(G) = \frac{\sum_k \min_{\mathbf{p} \in P_k} \{1 - \mathrm{sim}(W(G(z_n), k), \mathbf{p})\}}{N - K + 1} \tag{6}$$

As the generated motion becomes closer to the reference motion at the patch level, the loss become smaller. This causes the generator to be guided by the source dataset.

3.2. Entropy regularization

We encourage the generated samples to follow the distribution of reference motion patches. With the cross-domain regularization only, the generator may fall into using a small subset of reference motion patches. This results in limited variations of the target samples, i.e., mode collapse. We formulate the variability of the target samples as entropy:

$$\mathcal{L}_{entropy}(G) = \frac{\sum_k e^{-H(P_k;G)}}{N - K + 1} \tag{7}$$

$$H(P_k;G) = -\sum_{\mathbf{p}\in P_k} q(\mathbf{p};G) \cdot \ln q(\mathbf{p};G) \tag{8}$$

where $q(x_i)$ is the distribution of nearest-neighbor reference motion patches, which can be expressed as:

$$q(\mathbf{p};G) = |\Omega(\mathbf{p};G)| / B \tag{9}$$

$$\Omega(\mathbf{p};G) = \left\{ n | \mathbf{p} = \underset{\hat{\mathbf{p}}\in P_k}{\arg\max} \, \mathrm{sim}\left(W\left(G(z_n), k\right), \hat{\mathbf{p}}\right) \right\} \tag{10}$$

When entropy H is high, it means target samples have different nearest-neighbor reference motion patches. Consequently, a high H value results in small loss, which corresponds to capturing the distribution of the reference motion in the generated samples. This approach is similar to Prescribed GAN [7], where the distribution of the target domain is considered, while ours considers that of the source domain.

3.3. Final objective

Our final objective consists of these three terms: $\mathcal{L}_{adv}$ for the appearance of the target, $\mathcal{L}_{cd}$, which directly leverages the diversity contained in the source dataset, and $\mathcal{L}_{entropy}$, which prevents mode collapse:

$$\begin{aligned} G = \mathbb{E}_{z\sim p_z(z), x\sim \mathcal{D}_t, \hat{x}\sim \mathbb{P}_{\hat{x}}} \Big\{ &\arg\min_G \max_D \mathcal{L}_{adv}(G,D) \\ &+ \lambda_{gp}\mathcal{L}_{gp}(D) \Big\} + \lambda_{cd}\mathcal{L}_{cd}(G) + \lambda_{entropy}\mathcal{L}_{entropy}(G) \end{aligned} \tag{11}$$

We determined the optimal loss weight by grid search, and used loss weight $\lambda_{cd} = 1.0, \lambda_{entropy} = 1.5$ in all experiments except for the ablation study.

4. Experiments

In this section, we evaluate our method by training ST-GCN as a standard action recognition model using our generated sequences.

4.1. Dataset

NTU RGB+D [36] This dataset contains RGB videos, depth videos, and skeleton data calculated by Kinect for 60 action classes. The total number of samples is 56,880. The dataset is split into training and validation data on the basis of the authors' definition of "cross-subject." We utilize all data of 60 action classes (A001 – A060) of the training data as the source dataset.

NTU120 RGB+D [26] This dataset is a superset of NTU RGB+D where additional 60 action classes (A061 – A120) are collected, altogether amounting to 120 action classes. The total number of samples is 114,480. The dataset is split into training and validation data on the basis of the authors' definition of "cross-subject." We utilize three characteristsic action classes from the training data that include upper body-dominant, lower body-dominant, and fast movements — *Run on the spot* (A099), *Side kick* (A102), and *Stretch oneself* (A104) — as the target dataset. The validation data for these three action classes is used in accuracy evaluation of the ST-GCN classifier (Sec. 4.3).

Preprocessing Due to the inaccuracy of 3D joint annotations in the original NTU RGB+D dataset, we re-estimate the 3D joint rotations extracted from only RGB videos using the VIBE method [21].

4.2. Architecture and training

We configure the discriminator and generator networks on the basis of Kinetic-GAN with two exceptions. One of them is that, for simplification of implementation, we disable the action conditioning of Kinetic-GAN by removing the embedded class representation y from its mapping network. Instead, we train individual networks for each action class. The second exception is that we use the rotation representation for datasets, while the original Kinetic-GAN is trained with the position of each joint calculated by Kinect from depth images. We use VIBE [21] to obtain the rotation representation of each joint using the RGB images.

Networks are trained using the Adam optimizer with learning rate 0.0002 and weight decay parameters b1 = 0.5 and b2 = 0.999. For the target samples, we manually checked and excluded samples with estimation errors of 3D joint rotations by VIBE, then picked ten samples for each action class A099, A102, and A104. These 30 samples are then sampled randomly to compose 38400 samples to be used as training data. Training was performed with a batch size of 32 for 10 epochs using a single NVIDIA Quadro RTX 5000 GPU.

4.3. Data augmentation for ST-GCN classifier

We validate our method by applying it to data augmentation for action recognition models.

4.3.1 Recognition model

We use a standard model, ST-GCN [47], for action recognition. We train the model with the concatenated data of the real and generated samples. The real samples are identical to that used in Sec. 4.2. In data augmentation, A total of 1149 samples are generated (i.e., 383 samples for each action class), which is as big as the original real samples. For the training parameters, we follow their implementation on GitHub[1] with learning rate 0.1, weight decay 0.0001, momentum 0.9, batch size 64, and number of epochs 80. The tests are performed on the validation data three times with different random seeds, and the median of resultant accuracy is reported.

4.3.2 Comparison methods

The state-of-the-art methods Kinetic-GAN [6] and ACTOR [33] are compared with our method. Since the problem of few-shot human action synthesis is new and we could not find any existing few-shot learning methods, these methods are not designed for few-shot scenarios and do not have the capability for cross-domain learning. Therefore, these models are trained using target samples only.

We additionally perform an ablation study to investigate the effect of the proposed loss functions: Cross-domain only ($\lambda_{entropy}$ is set to zero) and Entropy only (λ_{cd} is set to 0). Note that Kinetic-GAN can be considered as a part of the ablation study since it is equivalent to setting the weight of both regularization terms to zero (i.e., $\lambda_{cd} = \lambda_{entropy} = 0$) in our method.

4.3.3 Results

Tab. 1 summarizes the results. When trained with only real training data, the top-1 accuracy is 58.4% (*w/o* augmentation). All the results with data augmentation show better accuracy. This indicates that when there are few real samples, ST-GCN suffers from overfitting, which causes low accuracy; however, the accuracy improves when generated samples are added.

The best accuracy of 86.4 % (Ours) is obtained with the proposed method. This confirms that the proposed method can successfully augment motion data to improve recognition accuracy even in few-shot scenarios. Ablation results suggest that both the cross-domain and entropy losses are essential for effective augmentation. We will analyse the role of these losses later in Sec. 4.4.1.

We further investigate how the effectiveness of data augmentation changes with the number of real samples. In Fig. 4, the dotted red line indicates the top-1 accuracy *without* augmentation, i.e., real data only. The accuracy is 58.4 % when the amount of real data is 30, as already shown in Tab. 1. As the number of real data increases, the accuracy also increases. When 150 samples (i.e., 50 samples per action class) are used, the accuracy improves to 85 %. With more than 300 samples (i.e., 100 samples per action class), the accuracy, 95%, is almost at the maximum. This demonstrates that ST-GCN recognition models are poorly trained in the few-shot scenarios, which can be attributed to overfitting. However, augmented learning with the proposed method lessen the accuracy reduction. The solid red line indicates the accuracy *with* augmentation where real data and 1149 generated samples are used for training. Even when only 30 real samples are used, augmentation is effective and the accuracy increases by 28 %. The variability of the generated samples may have prevented overfitting. The difference in accuracy between *with* and *without* augmentation decreases as the number of real data increases. Hence,

[1] https://github.com/open-mmlab/mmskeleton

Table 1. Data augmentation: Action recognition with the ST-GCN model improves most when using the samples generated by the proposed method (Ours), compared to state-of-the-art methods (ACTOR [33] and Kinetic-GAN [6]). Application of only one of cross-domain or entropy regularization degrades the the quality of augmentation. The number of real samples was 30 (i.e., 10 samples per action class), and the number of generated samples was 1149 (i.e., 383 samples per action class).

	Top-1 accuracy (%) ↑
w/o augmentation	58.4
ACTOR	73.6
Kinetic-GAN	81.7
Ours (Cross-domain only)	70.9
Ours (Entropy only)	72.1
Ours	**86.4**

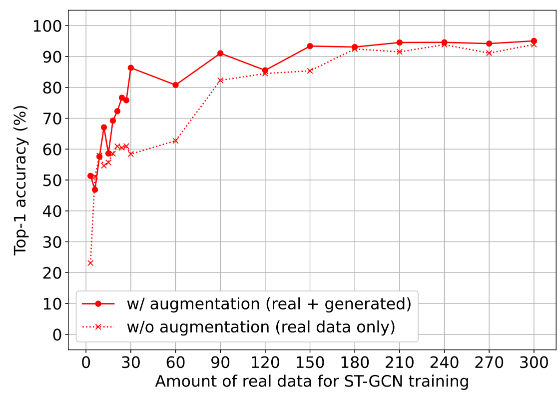


Figure 4. Effect of data augmentation is prominent with few real samples, while the accuracy of both *w/* and *w/o* augmentation approaches asymptotically to the same value with larger number of real data.

Table 2. Diversity evaluation: Diversity measures the overall variance across all action classes. Multimodality measures the variance within each action class. Refer to [13] for the difinition.

	Diversity	Multimodality
w/o augmentation	29.7	24.9
Kinetic-GAN	27.6	23.9
Ours (Cross-domain only)	29.9	24.7
Ours (Entropy only)	**30.1**	25.4
Ours	29.7	**25.8**

our method can be considered more effective in scenarios with few data.

4.4. Diversity evaluation

4.4.1 Quantitative analysis

We examine the diversity and multimodality, which are employed in [13], for the real and augmented training data used in Sec. 4.3. For both metrics, the motion sample $\mathbf{x}$ itself is used as a feature vector. In Tab. 2, we observe the highest multimodality for our method. The highest accuracy of the proposed method (in Tab. 1) should be attributed to this. The ablation results suggest that the proposed cross-domain and entropy losses successfully transfer the variation of source domain to the generated samples. With only cross-domain loss, less diversity is observed. With only entropy loss, greater diversity is obtained. However, the reference motion is not taken into account without the cross-domain loss, which prevents accuracy improvement. Our method combines both losses, resulting in generations which can contribute to effective augmentation.

4.4.2 Qualitative analysis

We qualitatively analyze the generated samples to demonstrate that our method is capable of injecting both spatial and temporal diversity. Fig. 5 shows the resulting 25 generated samples for each action class *Run on the spot*, *Side kick*, and *Stretch oneself*. Our method successfully synthesizes different ways to produce a given action. They inherit the motion from real samples of the target domain while also maintaining spatial diversity. We found that the results for *Stretch oneself* were more varied; this may have been because most of the actions in the source dataset involve the upper body, which is also the case for stretching. Fig. 6 shows the skeletal sequences of real and generated samples for each action, representing both the temporal variation and plausibility of the output.

It should be noted that the generated samples are not completely different each other. This, however, does not necessarily mean the limitation of our method. Because the goal of our model is to serve as a data augmentation method for HAR, the generated samples should follow the variation of the target domain. In this context, the motion that is likely to appear in the target domain should be generated more frequently.

Table 3. Top-3 source action classes corresponding to the target action class. The rankings are ordered by the percentage of action classes to which the reference motion patches used for cross-domain learning belong.

	Target action class		
	Run on the spot	Side kick	Strech oneself
1	Touch head (headache)	Drop	Cheer up
2	Take off jacket	Kicking something	Throw
3	Jump up	Rub two hands together	Taking a selfie

4.4.3 Target $\leftrightarrow$ source correspondence

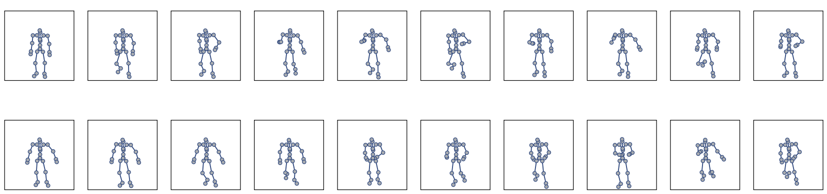
The diversity transferred to the target domain is based on the distribution of reference motion patches drawn from a certain action class of the source dataset. Tab. 3 lists the correspondence between target and source action classes in our experiments. Note that there exist multiple source action classes in the list, because a source motion is searched individually for each patch of the mean target motion divided by a sliding window, as shown in Fig. 2. As in the "Side kick" $\leftrightarrow$ "Kicking something" or "Strech oneself" $\leftrightarrow$ "Cheer up", we can observe intuitive mapping examples. On the other hand, in somes cases irrelevant actions appear to be selected, such as "Run on the spot" $\leftrightarrow$ "Touch head" or "Side kick" $\leftrightarrow$ "Drop". This may be due to the paucity of actions involving lower body movements in NTU RGB+D, suggesting the need for sufficient diversity in the source data set. Also, the current action classes, which follow the definition of NTU RGB+D, may not be optimal for effective cross-domain learning. Redefining optimal action classes could be an interesting prospect for future research.

5. Conclusion

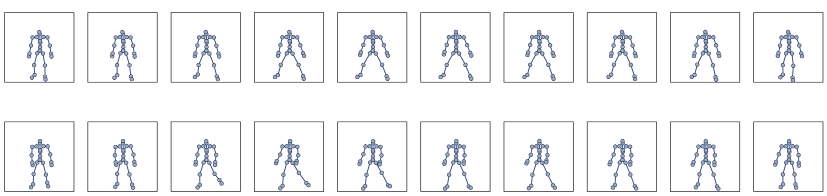
We presented a few-shot learning method for skeleton-based motion synthesis. To our knowledge, our method is the first for learning generative models of motion with few samples. It is based on cross-domain regularization and entropy regularization, which are effective for transferring the diversity of the large dataset of the source domain to the target domain. Experimental results demonstrated that the generated samples had high diversity even with very limited training samples, and they could be used as augmented data to train action recognition models.

(a) Run on the spot (b) Side kick (c) Stretch oneself

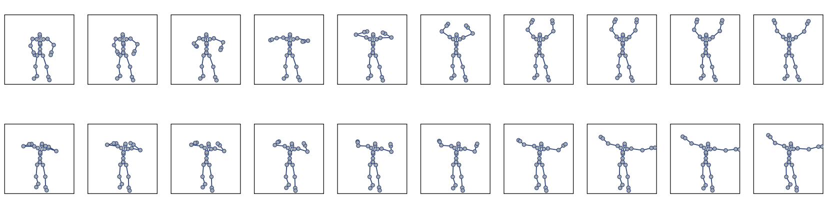
Figure 5. (Top) Real samples and (bottom) 10-shot human action synthesis results. We generated 25 different samples for each action and visualized one frame from the skeletal sequences of these samples. We selected the same frame across the same action. We observed diversity in the generated samples, which also differs from the real samples used as training data.



(a) Run on the spot: (Top) Real, (Bottom) Generated



(b) Side kick: (Top) Real, (Bottom) Generated



(c) Stretch oneself: (Top) Real, (Bottom) Generated

Figure 6. Sequences of real and generated samples. Temporal order is from left to right in the horizontal direction.

References

[1] Maria Eugenia Cabrera and Juan Pablo Wachs. Biomechanical-based approach to data augmentation for one-shot gesture recognition. In *IEEE Int. Conf. Automatic Face & Gesture Recognition*, pages 38–44, May 2018. 2

[2] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Trans. Pattern Anal. Machine Intell.*, 43(1):172–186, Jan. 2019. 2

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 4724–4733, July 2017. 2

[4] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei Efros. Everybody dance now. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 5932–5941, Oct. 2019. 2

[5] Aidan Clark, Jeff Donahue, and Karen Simonyan. Adversarial video generation on complex datasets. *arXiv preprint arXiv:1907.06571*, 2019. 3

[6] Bruno Degardin, Joao Neves, Vasco Lopes, Joao Brito, Ehsan Yaghoubi, and Hugo Proenca. Generative adversarial graph convolutional networks for human action synthesis. In *IEEE/CVF Winter Conf. Appl. of Comput. Vis. (WACV)*, pages 2753–2762, Jan. 2022. 2, 3, 6

[7] Adji B. Dieng, Francisco J. R. Ruiz, David M. Blei, and Michalis K. Titsias. Prescribed generative adversarial networks. *arXiv preprint arXiv:1910.04302*, 2019. 5

[8] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 4346–4354, Dec. 2015. 2, 3

[9] Saeed Ghorbani, Kimia Mahdaviani, Anne Thaler, Konrad Kording, Douglas James Cook, Gunnar Blohm, and Nikolaus F. Troje. MoVi: A large multi-purpose human motion and video dataset. *PLOS ONE*, 16(6):e0253157, June 2021. 2

[10] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa, and Jitendra Malik. Humans in 4D: Reconstructing and tracking humans with transformers. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 14783–14794, Oct. 2023. 2

[11] Zhichen Gong and Huanhuan Chen. Model-based oversampling for imbalanced sequence classification. In *ACM Int. Conf. Inf. Knowledge Management*, pages 1009–1018, New York, NY, USA, Oct. 2016. 2, 3

[12] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, pages 5769—5779, Dec. 2017. 3

[13] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned generation of 3D human motions. In *Int. Conf. Multimedia*, pages 2021—-2029, Oct. 2020. 7

[14] Sonam Gupta, Arti Keshari, and Sukhendu Das. RV-GAN: Recurrent GAN for unconditional video generation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. Workshops (CVPRW)*, pages 2023–2032, June 2022. 3

[15] Praful Hambarde, Subrahmanyam Murala, and Abhinav Dhall. UW-GAN: Single-image depth estimation and image enhancement for underwater images. *IEEE Trans. Instrum. Meas.*, 70:1–12, Oct. 2021. 2

[16] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Learning spatio-temporal features with 3D residual networks for action recognition. In *IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, pages 3154–3160, Oct. 2017. 2

[17] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large scale datasets and predictive methods for 3D human sensing in natural environments. *IEEE Trans. Pattern Anal. Machine Intell.*, 36(7):1325–1339, 2014. 2

[18] Yifeng Jiang, Tom Van Wouwe, Friedl De Groote, and C. Karen Liu. Synthesis of biologically realistic human motion using joint torque actuation. *ACM Trans. Graph.*, 38(4):72:1–72:12, July 2019. 2

[19] Nal Kalchbrenner, Aaron van den Oord, Karen Simonyan, Ivo Danihelka, Oriol Vinyals, Alex Graves, and Koray Kavukcuoglu. Video pixel networks. In *Int. Conf. Machine Learning*, volume 70, pages 1771–1779, Aug. 2017. 2

[20] Yoshiyuki Kobayashi, Naoto Hida, Kanako Nakajima, Masahiro Fujimoto, and Masaaki Mochimaru. AIST gait database 2019, 2019. `https://unit.aist.go.jp/harc/ExPART/GDB2019_e.html`. 2

[21] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. VIBE: Video inference for human body pose and shape estimation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5252–5262, June 2020. 2, 5

[22] Gihyun Kwon and Jong Chul Ye. One-shot adaptation of GAN in just one CLIP. *arXiv preprint arXiv:2203.09301*, 2022. 3

[23] Paula Lago, Sayeda Shamma Alia, Shingo Takeda, Tittaya Mairittha, Nattaya Mairittha, Farina Faiz, Yusuke Nishimura, Kohei Adachi, Tsuyoshi Okita, Francois Charpillet, and Sozo Inoue. Nurse care activity recognition challenge: summary and results. In *ACM Int. Joint Conf. Pervasive and Ubiquitous Comput. and ACM Int. Symp. Wearable Comput.*, pages 746–751, New York, NY, USA, Sept. 2019. 2

[24] Yijun Li, Richard Zhang, Jingwan Lu, and Eli Shechtman. Few-shot image generation with elastic weight consolidation. In *Int. Conf. Neural Inf. Process. Syst. (NeurIPS)*, pages 15885–15896, Dec. 2020. 3

[25] Kang Liao, Chunyu Lin, Yao Zhao, and Moncef Gabbouj. DR-GAN: Automatic radial distortion rectification using Conditional GAN in real-time. *IEEE Trans. Circuits Syst. Video Technol.*, 30(3):725–733, Mar. 2020. 2

[26] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C. Kot. NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding. *IEEE Trans. Pattern Anal. Machine Intell.*, 42(10):2684–2701, Oct. 2020. 2, 5

[27] Lanlan Liu, Michael Muelly, Jia Deng, Tomas Pfister, and Li-Jia Li. Generative modeling for small-data object detection.

In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 6072–6080, Oct. 2019. 3

[28] Sangwoo Mo, Minsu Cho, and Jinwoo Shin. Freeze the Discriminator: A simple baseline for fine-tuning GANs. *arXiv preprint arXiv:2002.10964*, 2020. 3

[29] Friedrich Niemann, Christopher Reining, Fernando Moya Rueda, Nilah Ravi Nair, Janine Anika Steffens, Gernot A. Fink, and Michael ten Hompel. LARa: Creating a dataset for human activity recognition in logistics using semantic attributes. *Sensors*, 20(15):4083, Jan. 2020. 2

[30] Atsuhiro Noguchi and Tatsuya Harada. Image generation from small datasets via batch statistics adaptation. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 2750–2758, Oct. 2019. 3

[31] Yuya Obinata and Takuma Yamamoto. Temporal extension module for skeleton-based action recognition. In *Int. Conf. Pattern Recog. (ICPR)*, pages 534–540, Jan. 2021. 2

[32] Utkarsh Ojha, Yijun Li, Jingwan Lu, Alexei A. Efros, Yong Jae Lee, Eli Shechtman, and Richard Zhang. Few-shot image generation via cross-domain correspondence. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 10738–10747, June 2021. 3, 4

[33] Mathis Petrovich, Michael J. Black, and Gul Varol. Action-conditioned 3D human motion synthesis with Transformer VAE. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 10965–10975, Oct. 2021. 2, 3, 6

[34] Lorenzo Pitto, Hans Kainz, Antoine Falisse, Mariska Wesseling, Sam Van Rossom, Hoa Hoang, Eirini Papageorgiou, Ann Hallemans, Kaat Desloovere, Guy Molenaers, Anja Van Campenhout, Friedl De Groote, and Ilse Jonkers. SimCP: A simulation platform to predict gait performance following orthopedic intervention in children with cerebral palsy. *Frontiers in Neurorobotics*, 13, 2019. 2

[35] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 2849–2858, Oct. 2017. 2

[36] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+D: A large scale dataset for 3D human activity analysis. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 1010–1019, June 2016. 2, 5

[37] Khurram Soomro and Amir R. Zamir. *Action Recognition in Realistic Sports Videos*, pages 181–208. Springer International Publishing, 2014. 2

[38] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 5686–5696, June 2019. 2

[39] Ximeng Sun, Huijuan Xu, and Kate Saenko. TwoStream-VAN: Improving motion modeling in video generation. In *IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, pages 2733–2742, Mar. 2020. 3

[40] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 2, 3

[41] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 4489–4497, Dec. 2015. 2

[42] Juanhui Tu, Hong Liu, Fanyang Meng, Mengyuan Liu, and Runwei Ding. Spatial-temporal data augmentation based on LSTM autoencoder network for skeleton-based human action recognition. In *IEEE Int. Conf. Image Process. (ICIP)*, pages 3478–3482, Oct. 2018. 2, 3

[43] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 1526–1535, June 2018. 2

[44] Yaxing Wang, Abel Gonzalez-Garcia, David Berga, Luis Herranz, Fahad Shahbaz Khan, and Joost van de Weijer. MineGAN: Effective knowledge transfer from GANs to target domains with few images. In *IEEE/CVF Conf. Comput. Vis. Pattern Recog. (CVPR)*, pages 9329–9338, June 2020. 3

[45] Yaxing Wang, Chenshen Wu, Luis Herranz, Joost van de Weijer, Abel Gonzalez-Garcia, and Bogdan Raducanu. Transferring GANs: Generating images from limited data. In *European Conf. Comput. Vis. (ECCV)*, pages 220–236, 2018. 3

[46] Sijie Yan, Zhizhong Li, Yuanjun Xiong, Huahan Yan, and Dahua Lin. Convolutional sequence generation for skeleton-based action synthesis. In *IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, pages 4393–4401, Oct. 2019. 2, 3

[47] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI Conf. Artif. Intell.*, pages 7444–7452, New Orleans, Louisiana, USA, Feb. 2018. 2, 6