AL-SAR: Active Learning for Skeleton-based Action Recognition

Jingyuan Li, Trung Le and Eli Shlizerman

***Abstract*—Action recognition from temporal multi-variate sequences of features, such as identifying human actions, is typically approached by supervised training as it requires many ground truth annotations to reach high recognition accuracy. Unsupervised methods for the organization of sequences into clusters have been introduced, however, such methods continue to require annotations to associate clusters with actions. The challenges in annotation necessitate an effective classification methodology that minimizes the required number of labels. Active Learning (AL) approaches have been proposed to address these challenges and were able to establish robust results on image classification. Such approaches are not directly applicable to sequences, since for sequences, the variations are in both spatial and temporal domains. In this paper, we introduce a novel method for active learning for sequences, called "AL-SAR", which combines unsupervised training with sparsely supervised annotation. In particular, AL-SAR employs a multi-head mechanism for robust uncertainty evaluation of the latent space learned by an encoder-decoder framework. It aims to iteratively select a sparse set of samples, which annotation contributes the most to the disentanglement of the latent space. We evaluate our system on common benchmark datasets with multiple sequences and actions, such as NW-UCLA, NTU RGB+D 60, and UWA3D. Our results indicate that AL-SAR coupled with encoder-decoder network outperforms other AL methods coupled with the same network structure.**

***Index Terms*—Skeleton-based Action Recognition, Active Learning, Uncertainty Sampling, Human Action Recognition**

I. Introduction

ACTION recognition from spatio-temporal sequences is a key component in ubiquitous applications such as action recognition of human movements, understanding subject interaction, and robotic control. Unlike video-based action recognition approaches, which perform classification on image frames, skeleton-based methods operate on pose estimation features, such as skeleton joints or contours, and offer a more concise representation of the action by filtering unnecessary information from the scenes.

Several systems have been introduced for action recognition from body-skeleton keypoints. Such systems attempt to learn spatial and temporal relations for sequences and translate each of them to an association of an action. The majority of the methods require a fully supervised approach [1], [2], [3], [4]. While the recognition accuracy of these methods has been shown to be effective, such approaches rely on the availability of a large number of annotations in the training set. Acquiring ground truth annotations for sequences is a time-consuming process and requires human expertise. These challenges hinder the possibility to scale up the approaches and apply them to novel scenarios of actions and subjects. To address these challenges, unsupervised methods have been proposed. They demonstrate the potential of a framework that includes two network components, an encoder, and a decoder, cooperating to reconstruct spatiotemporal sequences of keypoints [5], [6], [7]. These networks were found to self-organize the latent space shared between the encoder and the decoder to form clusters that correspond to actions. While these methods appear to be promising, the association of clusters with actions continues to require a large number of labels and underperform supervised methods.

Few-shots learning and semi-supervised learning for skeleton-based action recognition were proposed to reduce the required number of annotations while not compromising the action recognition accuracy. The few-shot learning methods propose to "meta-train" a base model on auxiliary action classes, for which there is a large amount of annotated samples. Such training aims to learn a set of general model parameters such that the base model can be efficiently adapted to unseen classes which are not included in auxiliary action classes, via further training on a few examples of these classes [8], [9], [10], [11]. However, the auxiliary classes with abundant annotated samples are not always on deck. Besides, the auxiliary and unseen classes are expected to follow similar data distribution. This expectation limits the application of few-shot learning to various practical problems, since in practice data quality, style, the number of skeleton keypoints, or even subjects, e.g., human subjects vs. animal subjects, significantly vary from one dataset to another. Alternatively, semi-supervised methods have been proposed for situations where well-annotated auxiliary classes are unavailable. Examples of semi-supervised methods for skeleton based action recognition include ASSL [12], MS^{2}L [13], and SC3D [14], which learn the informative representations for both labeled and unlabeled samples in addition to correctly classifying labeled samples.

Importantly, these semi-supervised learning methods do not consider that not all annotated samples contribute equally to the training of a classifier, and therefore, it is advantageous to select for annotation the samples that dominantly represent their classes. Such selection could improve the effectiveness of the classifier and at the same time minimize the number of annotations needed for training. Active Learning (AL) algorithms [15], [16], [17] have been established based on this principle and showed promising results when applied to image classification tasks [18], [19], [20]. Indeed, a vari-

Jingyuan Li and Trung Le is with Graduate School of Electrical and Computer Engineering, University of Washington, Seattle, WA 98195 USA, (e-mail: jingyli6@uw.edu; tle45@uw.edu)

Eli Shlizerman is with the Faculty of Applied Mathematics and Electrical and Computer Engineering, University of Washington, Seattle, WA 98195 USA. (e-mail:shlizee@uw.edu)

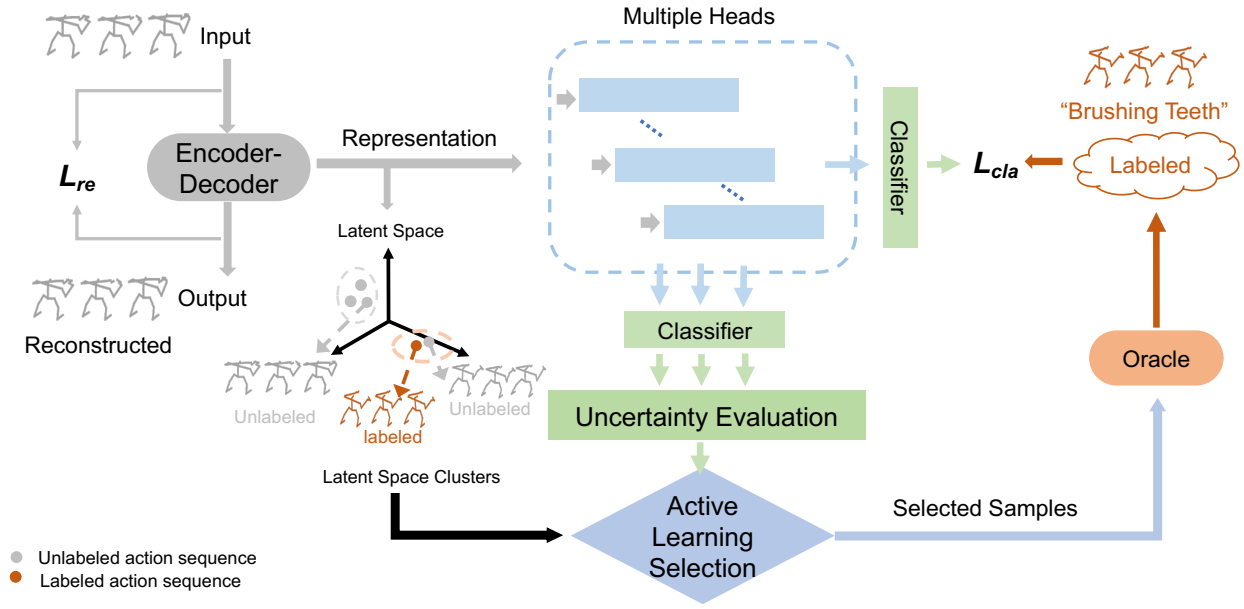

Fig. 1. AL-SAR system overview. A multi-head mechanism (middle) operates on a learned hidden representation of an encoder-decoder (left) network, which extracts the latent representation using the reconstruction task. Consequently, these latent representations are grouped into Latent Space Clusters. The multi-head mechanism (middle) computes the uncertainty of samples through the estimation of the collective confidence of the heads on the prediction output. By incorporating the latent space information and the uncertainty evaluation, the active learning algorithm ensures that the selected samples are diversified and informative.

ety of AL selection strategies have been developed. These include approaches based on principles of diversity [18], uncertainty [20], [21], [22], and model decision [23]. While AL is widely adopted for image data, only a handful of studies have investigated the applicability of AL to sequential data that is spatio-temporal [24], [25], [26], and application of these methods show no superiority over the random selection of samples.

To propose a more effective selection paradigm, we introduce a novel active learning approach, termed AL-SAR, for action recognition from spatio-temporal skeleton sequences. AL-SAR selects sequences for annotation according to the clustering information in the latent space along with robust estimation of uncertainty. Our approach is a novel extension of the margin-based uncertainty selection strategy with a multi-head mechanism. Beyond applicability to skeleton-based action recognition, our approach provides a generally effective algorithm for unsupervised sequence reconstruction and classification methods to perform learning with efficient annotations. Validation of AL-SAR on three extensive common benchmarks of skeleton-based human action recognition (UWA3D, NW-UCLA, NTU-RGB+D) shows that AL-SAR allows for significant improvement over state-of-the-art unsupervised and semi-supervised methods, especially when only a sparse number of annotations can be obtained.

II. Related Work

Various skeleton-based action recognition approaches have been proposed. These include supervised methods which analyze action related physical statistics [27], [28], [29], deep learning with CNNs [30], [31], [32], RNNs and their variants [1], [33], [34], and graph convolutional networks [4], [35], [36], [37], [38]. Unsupervised methods have been introduced as well. Such approaches build a latent representation through learning to reconstruct input sequences with an encoder-decoder network structure. The latent space of such networks is shown to self-organize into clusters enabling a simple classifier such as K-Nearest Neighbour (KNN) to identify action types [6], [7]. However, the KNN component still remains supervised and annotations are necessary to identify actions. To avoid supervision and improve action recognition accuracy, semi-supervised approaches have been proposed. These approaches learn the recognition task by leveraging annotations from a randomly selected subset of samples. Examples include methods such as ASSL [12], MS^{2}L [13], SC3D [14]. These methods do not deal with the selection of sequences for annotation and instead assume a given random annotated set. In applications, it is critical to optimize such a selection and to seek samples that are more informative for learning actively. Such a selection would need to be achieved with AL methods.

AL methods typically belong to three categories: (i) sample synthesis, (ii) stream-based selective sampling, and (iii) pool-based sampling [15], [16], [39]. Sample synthesis is based on generating additional samples of action sequences. These generated sequences are typically of lower quality, making them challenging candidates for annotation [40]. Stream-based and pool-based sampling methods work with real data. The stream-based selection considers one sample at a time, decides whether to annotate the sample at that time, and is applicable in online learning scenarios [16]. Pool-based methods select a

set of samples at each stage and are therefore expected to be more efficient for applications with a dataset already prepared. Such methods are widely used in classical machine learning techniques such as support vector machine and logistic regression, and take into consideration aspects such as diversity [41], [42], [43], and uncertainty (based on entropy [44], confidence estimation [45], and margin estimation [46], [47]). The idea of enforcing diversity or uncertainty for sample selection has been adapted to deep learning as well. For example, diversity is incorporated by selecting a sample batch that covers the whole space with a minimum covering radius for each sample [18]. Uncertainty metrics with deep learning models are computed with techniques such as dropout [22], [44], [48], the ensemble of models [19], and image augmentation [20]. These techniques compute the prediction entropy or the variance among 'multiple-outputs' for a sample. In many scenarios, it is advantageous to consider both diversity and uncertainty [49], [50], [51], since diversity reduces redundancy of selected samples and uncertainty focuses on samples where the model is less confident. In addition to these methods, a new branch of deep learning pool-based AL methods has been introduced, solving AL from a different aspect by learning a Discriminator (DIS) where samples which DIS is least confident in are selected for annotation [23], [25]. These methods rely on a network to learn the characteristics of unlabeled samples instead of measuring the uncertainty or the diversity with hand-designed features like the aforementioned approaches do.

Our method, AL-SAR, belongs to pool-based AL methods and selects samples based on diversity and uncertainty estimations. In contrast to the aforementioned methods, where the diversity is imposed by computing similarity [50], [51], in our work, the diversity is incorporated by annotating samples located in different clusters in the organized latent space formed during the training process. Several methods have exploited the preservation of similarity among samples in the latent space and demonstrated success on downstream tasks such as video-based person re-identification [52] and vehicle re-identification [43]. The representation are flourishingly generated by the encoder-decoder structure [53], [5], [6], GAN [54], [55], contrastive learning [14], [56] and multi-view learning [57], [58], [59]. Here, we use the encoder-decoder structure [6] to learn the representation. However, other methods could be used to generate representations, such as contrastive learning methods, e.g., SC3D [14]. Apart from the organized latent representation, the uncertainty is measured for the guidance of active selection. Estimation of uncertainty is related to methods that consider 'multiple-outputs' [19], [20], [48], [50]. Unlike earlier approaches, we generate multiple-outputs through a single forward pass using a novel multi-head mechanism. Multi-heads are composed of several parallel, randomly initialized, and fixed heads (fully connected layers). The heads are injected right before the last classification layer (classifier), as shown in the middle part of Fig. 1. Multiple-outputs are obtained through the alteration of the connectivity weights of the classifier with different heads. All heads take the same input computed from the earlier module. With the average prediction from multiple-outputs, we compute the margin-based uncertainty metric by measuring the difference between the predicted probabilities of two most likely classes. Entropy and variance computation proposed in earlier for the guidance of sample selection works [20], [47], [48] could be biased, e.g., there are cases where prediction variance on a sample is small but predictions are inconsistent across inference passes. In these cases, it is beneficial to annotate such samples.

To address these aspects, we implement a margin-based selection that is coupled with diversity filtering that operates on samples represented in the latent space. The margin based selection is expected to perform more optimally in cases such as the aforementioned scenario. Indeed, our experiments on standard benchmarks show that AL-SAR can achieve enhanced action recognition performance and requires fewer samples when compared with other AL methods.

III. Methods

AL-SAR works with datasets of multi-dimensional time series specifying the coordinates of body keypoints at each given time. We denote the times-series as $\mathcal{X} = \{X_u \bigcup X_l\}$, with X_u representing the sequences in the unlabeled set and X_l in the labeled set. At first, $\mathcal{X}$ has only unlabeled samples ($\mathcal{X} = X_u$). A sample $\mathbf{x}_i \in \mathcal{X}$ is represented as a sequence $\mathbf{x}_i = [x_1, x_2, .., x_t, ...x_T]$, where x_t is the vector of coordinates of the keypoints at time t, $x_t \in R^{p \times d}$. Here p is the number of keypoints, d is the dimension of the keypoints (typically $d = 3$). p is expected to vary across datasets. For datasets with keypoints obtained from video frames recorded from multiple views (e.g., NW-UCLA, UWA3D), we follow the procedure of transforming them to a view invariant representation [6], [60].

The proposed AL-SAR system includes three main components: (i) Learning the latent representation of skeleton sequences, (ii) The multi-head mechanism for robust computation of the margin-based metric, (iii) AL selection which integrates location information in the latent space and the margin-based metric to select samples for annotation. In this paper, we focus on advancing components (ii), and (iii) since, for (i), there are powerful pre-existing methods available. An overview of AL-SAR system architecture is depicted in Fig. 1.

(i) Preliminary: Learning Meaningful Latent Representations. In AL-SAR, we embed the spatial-temporal body keypoints into latent representation space with the encoder-decoder framework introduced by [6], [7], which has been shown to achieve meaningful latent representation. The encoder uses bidirectional Gated Recurrent Units (GRU) and receives $\mathbf{x_i} \in \mathcal{X}$ as input. The vector $\mathbf{h}_i^T$ is the latent code transferred from the encoder at the last time step T to the decoder. It encodes the dynamic properties of the whole sequence $\mathbf{x_i}$ and lies in the latent space V, where $V = \{\mathbf{h}_i^T | \mathbf{h}_i^T = encoder(\mathbf{x_i}), \mathbf{x_i} \in \mathcal{X}\}$, i.e., the space spanned by the latent codes of all sequences. The unidirectional GRU-based decoder receives h_i^T and reconstructs the original input sequences by minimizing the reconstruction loss

$$\mathcal{L}_{re} = |\hat{\mathbf{x}}_i - \mathbf{x}_i|. \quad (1)$$

Notably, AL-SAR is not limited to the encoder-decoder framework and training strategy of P&C [6], and as we demonstrate in Experiments & Results, AL-SAR could achieve successful

sample selection and classification with latent representation learned by other encoders.

(ii) Multi-head Mechanism. Pool-based AL with uncertainty strategies is susceptible to biased classification predictions, especially when access to annotated instances is limited. The prediction of the classifier is affected by many factors including network initialization and training strategy. Furthermore, as shown in earlier work, the classifier can be over-confident about specific samples [61]. Therefore, direct measurement of the uncertainty from classification predictions could be misleading. The multi-head mechanism is designed to reduce these effects to provide a robust measurement of uncertainty. Notably, the multi-head mechanism introduced here differs from those widely used in Transformer networks for computing a set of attention weights [62] or ones used in CNN for spectrogram inversion [63]. The differences are in how multi-heads are structured, their training scheme and purpose. We keep the name due to the similarity of introducing parallel blocks. In particular, the proposed multi-heads are randomly initialized as a fully connected layer, fixed, and parallelly inserted right before the last classification layer. We train the classifier to correctly classify labeled samples from the output of any head it is connected to. In the sample selection phase, we average the classifier's outputs as it connects to each head for robust uncertainty estimation.

We describe the detailed the multi-head network as follows. We consider the classifier $\mathcal{C}$ as a single fully connected layer with weights W_θ. The multi-head mechanism is constructed as multiple additional heads receiving $\mathbf{h}_i^T$ as input and sending outputs to the classifier. Each head is a single fully connected layer with weights W_δ initialized according to the uniform distribution, i.e., $W_\delta \sim \mathcal{U}\left(-\frac{1}{\dim(\mathbf{h}_i^T)}, \frac{1}{\dim(\mathbf{h}_i^T)}\right)$, and kept fixed throughout the training process. Here $\dim(\mathbf{h}_i^T)$ denotes the dimension of the latent code. During training and testing, a single head is randomly chosen to be activated at each time. During operation, i.e., at the sampling phase, all heads are activated and average predictions are used to evaluate the uncertainty of the input samples. We show in Results (Section IV) that by generating multiple transformations of $\mathbf{h}_i^T$ and collectively contributing their different confidence to the knowledge of the unlabeled samples, the heads effectively reduce artifacts in estimation of uncertainty and improve the performance of the overall system. We use the margin-based uncertainty metric [46], [47], i.e., **marginal index** (***MI***), which evaluates the probability prediction p_z from each head z

$$p_z^l(\mathbf{x_i}) = p_z^l(\hat{y_i} = l|W_\theta, W_\delta) = \mathcal{C}(\mathcal{H}_z(\mathbf{x_i}))$$
$$z \in [1, N_h]; \text{and } l \in [1, C],$$

where p_z^l denotes the probability of a sample which belongs to class l predicted by the classifier $\mathcal{C}$ when the head z is activated. For C possible classes, $\mathcal{C}$ and $\mathcal{H}_z$ indicate the transformation with the classifier and the z^{th} head respectively. N_h denotes the number of heads. Given probability predictions, *MI* is computed as the measure of the confidence difference between the most probable class and the second most probable class, using the average probability prediction of the heads, i.e.,

$$MI = \max_{l\in[1:C]}\left(\frac{1}{N_h}\sum_{z=1}^{N_h}\mathbf{p}_z\right) - \max_{l\in([1:C]\setminus l^*)}\left(\frac{1}{N_h}\sum_{z=1}^{N_h}\mathbf{p}_z\right), \quad (2)$$

$$\text{Where, } \mathbf{p}_z = [p_z^1, .., p_z^l, .., p_z^C],$$
$$l^* = \underset{l\in[1:C]}{\text{argmax}}\left(\frac{1}{N_h}\sum_{z=1}^{N_h}\mathbf{p}_z\right).$$

Given the classifier output, the classification loss is computed as

$$\mathcal{L}_{cla}^i = \sum_{l=1}^{C} -y_i^l \log(p_z^l(\mathbf{x}_i)), \;\; z = \text{rand}(1 : N_h), \quad (3)$$

where $y_i^l = 1$ if $\mathbf{x}_i$ belongs to class l, and $y_i^l = 0$ otherwise. The loss, incurred for each sample $\mathbf{x}_i$, is composed from the reconstruction loss $\mathcal{L}_{re}^i$ and the classification loss $\mathcal{L}_{cla}^i$ for the labeled samples. The full model is trained according to the total loss

$$\mathcal{L} = \sum_{\mathbf{x}_i\in\mathcal{X}_l} \mathcal{L}_{cla}^i + \frac{1}{|\mathcal{X}|}\sum_{\mathbf{x}_i\in\mathcal{X}} \mathcal{L}_{re}^i, \quad (4)$$

where $|\mathcal{X}|$ is the total number of samples in the dataset.

(iii) Active Selection. In addition to the margin-based uncertainty measure (*MI*), we incorporate diversity filtering by leveraging clustering information in the latent space to enhance coverage and effectiveness of selected samples. We indeed observe that the inclusion of clustering information in AL boosts the overall classification. It presumably brings closer samples in the latent space that belong to the same class, while increasing the distances between distinct classes, as shown in Fig. 2. In each iteration, a new set of unlabeled samples is selected for annotation and then all samples (labeled and unlabeled) are used to refine the latent code and enhance the classifier. The selection can be subdivided into two scenarios: i) Initial Selection, ii) Subsequent Selection.

Initial Selection is regarded as the 'cold-start' problem, where neither ground truth annotation nor the predictions of the classifier are available [64], [65]. For effective initial selection, we form samples into clusters according to latent representation $\mathbf{h}_i^T$ generated by the encoder and then the samples in each cluster center are selected for annotation based on the assumption that these samples represent the clusters. Specifically, we use *K-Means* clustering to transform the latent representation into a collection of clusters $\mathcal{K}$. The number of clusters k

$$k = \frac{1}{N_{iter}} \times \; percentage \; \times |\mathcal{X}|, \quad (5)$$

is chosen based on the total number of selection iterations N_{iter} and the selection budget (the $percentage$ of data we want to annotate). k is fixed across selection stages and is the budget for each selection. With the labeled samples, the classifier is trained until the classification accuracy on these labeled samples converges.

For subsequent selections, we combine cluster information and *MI* computed by the multi-head structure. We choose

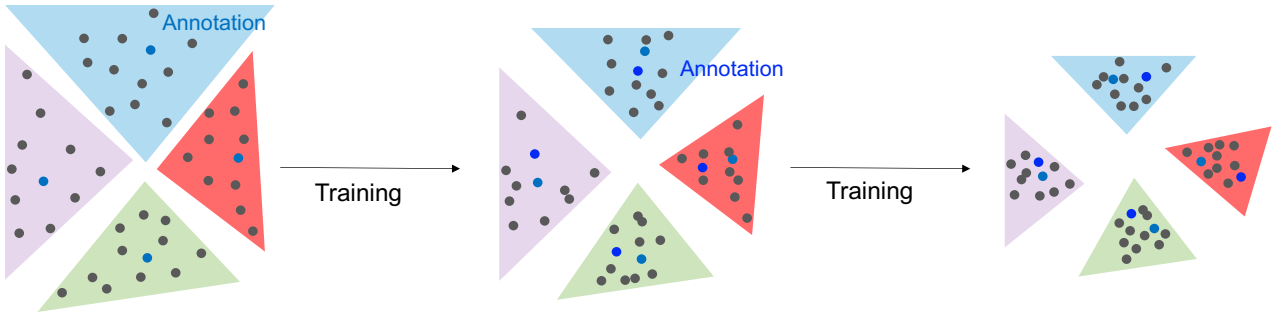

Fig. 2. Illustration of latent space organization when the encoder-decoder network is trained with AL. Three training iterations depict that clusters in the latent space self-organize and co-adapt during training and annotation process. Gray points are unlabeled points, blue points are samples selected for annotation. In each iteration, clusters are formed in the latent space. The samples closest to the center of each cluster are annotated in the initial selection. In subsequent iterations, samples for annotation are chosen from each cluster according to AL-SAR strategy. The process is repeated for multiple iterations until it reaches the maximal annotation budget.

TABLE I
COMPARISON OF FULLY SUPERVISED (FS), SEMI-SUPERVISED (SS) IN TOP SECTION, SOTA AL (MID SECTION) ACCURACY WITH AL-SAR (BOTTOM SECTION) ON THREE BENCHMARKS OF ACTION RECOGNITION.

		UWA3D VIEW3				NW-UCLA				NTU RGB+D 60 CS			
	%Labels	5%	10%	20%	50%	5%	15%	30%	40%	1%	2%	5%	10%
	#Labels	25	50	100	250	50	150	300	400	400	800	$2K$	$4K$
Full & Semi Supervised	C	20.0	23.3	37.3	50.0	42.7	57.9	70.9	68.5	21.8	37.2	49.6	56.7
	RC	20.9	32.2	38.3	48.8	55.1	50.9	72.1	77.0	33.8	41.6	47.8	60.0
	IRC	21.7	29.6	41.5	55.4	50.7	59.3	78.6	78.0	36.7	42.7	53.9	61.2
	ASSL [12]	–	–	–	–	52.6	74.8	78.0	78.4	–	–	57.3	64.3
	MS^{2}L [13]	–	–	–	–	–	–	–	–	33.1	–	–	65.2
	SC3D [14]	–	–	–	–	–	–	–	–	35.7	–	**59.6**	**65.9**
AL	DIS	19.3	28.7	40.2	53.8	47.7	71.8	76.6	80.5	34.9	39.5	53.8	60.4
	CS	21.5	29.9	40.3	52.6	57.3	69.4	77.3	80.6	17.6	23.1	37.0	49.6
	AUG	21.9	30.2	40.8	**56.2**	48.2	65.8	77.0	81.7	20.7	33.5	50.4	60.2
	U	23.4	30.5	42.1	53.4	52.3	70.4	78.4	80.8	34.6	43.8	56.3	61.2
							Ours						
	AL-SAR	**25.7**	**35.3**	**45.7**	53.9	**61.0**	**75.9**	**82.5**	**84.1**	**38.8**	**47.6**	57.9	63.7
	↑vs U	↑2.3	↑4.8	↑3.6	↑0.5	↑8.7	↑5.5	↑4.1	↑3.3	↑4.2	↑3.8	↑1.7	↑2.5

the samples (s) with the minimum *MI* within every cluster K_i ($i \in [1, k]$). The selected samples are then passed to the 'Oracle' for annotation. A new set of k annotated samples is subsequently added to the labeled set and the model is continually trained with the loss of Eq. 4. We summarize the complete procedure in Algorithm 1.

IV. EXPERIMENTS & RESULTS

Datasets. We evaluate the performance of AL-SAR on three common benchmark datasets, *UWA3D Multiview Activity II (UWA3D)* [66], *North-Western UCLA (NW-UCLA)* [67], *NTU RGB+D 60* [33]. These three datasets contain the different numbers of actions, with cross-view (CV) and cross-subject (CS) sequences. ***UWA3D*** contains 30 human action categories. Each action is performed 4 times by 10 subjects recorded from frontal, left, right, and top views. We selected the first two views as the training set and the third view as the test set, which appears to be a more challenging task according to related work [6], [68]. Results on view 4 are shown in Supplementary Material. ***NW-UCLA*** is captured by three Kinect V1 cameras containing depth and human skeleton data from three different views. The dataset includes 10 different action categories performed by 10 different subjects repeated 1-10 times. We use the first two views to form the training set, and the third view as the test set, following the same procedure as in [6], [30], [67]. ***NTU RGB+D 60*** includes both video and skeleton sequences performed by 40 different subjects recorded using 3 different cameras across different views. The dataset includes 60 different classes. We evaluate the performance in both CS and CV settings (CV results are in Supplementary Material). For the CV setting, samples from cameras 2 and 3 are used for training, and samples from camera 1 are used for testing. CS setting splits subjects into 2 groups, 20 subjects are used for training and the other 20 subjects are used for testing. It is a harder task compared to CV, especially for unsupervised and semi-supervised methods [6], [12].

Implementation Details. For experiments, we use three-layer bi-GRU cells that constitute the encoder with 1024 hidden

TABLE II
COMPARING AL-SAR WITH ITS ABLATION VERSIONS: UNIFORM WITH MULTI-HEAD (UH), NO CLUSTER WITH MULTI-HEAD (NKH), AVERAGE *MI* COMPUTED WITH DROPOUT (DROP), USING CLUSTER WITHOUT HEAD (KNH).

			UWA3D VIEW3		NW-UCLA		NTU 60 CS	
% Labels	Cluster-	Multi-	5%	20%	5%	30%	1%	5%
# Labels	ing (K)	Head (H)	25	100	50	300	400	$2K$
Random Selection								
U	✗	✗	23.4	42.1	52.3	78.4	34.6	56.3
UH	✗	✓	25.1	41.3	54.9	76.9	33.8	54.9
Margin Based Selection								
MK	✓	✗	23.8	**49.7**	55.7	81.9	38.2	57.4
MKD	✓	Dropout	23.2	45.2	57.6	81.3	37.1	57.2
MH	✗	✓	23.2	42.5	57.2	81.0	34.9	56.6
AL-SAR	✓	✓	**25.9**	45.4	**59.4**	**82.2**	**38.9**	**57.9**

Algorithm 1 AL-SAR Iterative Sample Selection Procedure

procedure AL-SAR
  Number of training epochs N_{ep}, $\mathbf{h}_u^T$ latent representation for all unlabeled samples. Standard Deviationn (*std*).
  Inputs: unlabeled samples $\mathbf{X}_u$, labeled samples $\mathbf{X}_l = \emptyset$, $n_{iter} = 0$
  $\mathbf{h}_u^T \leftarrow$ *Encoder*$(\mathbf{X}_u)$
  $\mathcal{K} \leftarrow$ *K-Means*$(\mathbf{h}_u^T, k)$
  $\mathbf{s} \leftarrow Oracle([center(K_i)$ for K_i in $\mathcal{K}])$
  $\mathbf{X}_l \leftarrow \mathbf{X}_l \cup \mathbf{s}$
  $\mathbf{X}_u \leftarrow \mathbf{X}_u \backslash \mathbf{s}$
  for τ = 1:N_{ep} **do**
    $\mathbf{ACC}_\tau \leftarrow$ *Classification Accuracy*
    if $std(\mathbf{ACC}_{\tau-1}, \mathbf{ACC}_{\tau-2}, \mathbf{ACC}_{\tau-3}) < 0.01$, $\tau \geq 3$ and $n_{iter} < N_{iter}$ **then**
      $\mathbf{h}_u^T \leftarrow$ *Encoder*$(\mathbf{X}_u)$
      $\mathcal{K} \leftarrow$ *K-Means*$(\mathbf{h}_u^T, k)$
      $\mathbf{s} \leftarrow Oracle([\text{argmin}(MI(K_i)),$ for K_i in $\mathcal{K}])$
      $\mathbf{X_l} \leftarrow \mathbf{X}_l \cup \mathbf{s}$
      $\mathbf{X}_u \leftarrow \mathbf{X}_u \backslash \mathbf{s}$
      $n_{iter} \leftarrow n_{iter} + 1$

units for each direction. Hidden units from both directions are concatenated to a 2048 dimensional latent representation $\mathbf{h}_i^T$, and then $\mathbf{h}_i^T$ is sent to the decoder. The decoder is chosen to be a uni-GRU with the hidden size of 2048. Each head is instantiated as a 2048x1024 fully connected layer. The classifier $\mathcal{C}$ is a single fully connected layer and receives input that is the output from the heads and predicts class probabilities of samples. We use the Adam optimizer for optimization. The learning rate is set to 10^{-4} and then decays by 0.95 for every 10 epochs on UWA3D and NW-UCLA, and every 3 epochs on NTU RGB+D. We use 5 heads for UAW3D and NW-UCLA. For NTU RGB+D 3 heads are used with 1% and 2% annotation budgets, and 5 heads are used with 5% and 10% budgets.

Comparison with SOTA AL Methods. Since AL methods specifically designed for action recognition are limited, we compared AL-SAR with SOTA AL generic techniques and those that were proposed for other tasks. In particular, we implement and examine Uniform sampling (U), Core-Set (CS) [18], Discriminator based selection (DIS) [23], and Consistency-based AL under Augmentation (AUG) [20]. We compare these AL methods against the margin-based selection with multi-head mechanism in AL-SAR. In U, samples are randomly selected for annotation among the entire dataset. CS aims to cover the whole space using selected samples with minimized coverage range [18]. DIS leverages a discriminator to distinguish if a sample is labeled or not (as we describe in Related work) [23], [25]. With AUG, we apply the augmentation strategies for skeleton sequences introduced in [69] together with the proposed consistency-based AL technique in [20].

The evaluation is shown in Table I for UWA3D VIEW3, NW-UCLA, NTU RGB+D 60 CS. We observe that AL-SAR consistently outperforms existing AL methods (see increase vs U). Notably, when the annotation budget is small, as with 5% and 10% of UWA3D View3, AL-SAR outperforms U, the best baseline method on UWA3D dataset, by 2.3 and 4.8 in accuracy respectively. On NW-UCLA with 5% and 15% labeled samples, the performance of AL-SAR surpasses the best methods (CS and DIS respectively) by 3.7 and 4.1. Similar behavior is found for NTU RGB+D 60 CS, where the difference between AL-SAR and DIS is 3.9 with 1% labeled samples. As the annotation budget increases, the improvement vs U is still significant. For instance, with 10% (400K on NTU RGB+D 60 CS), AL-SAR surpasses U by 2.5. In general, AL-SAR consistently demonstrates high efficiency across benchmarks. Similar conclusion is drawn from Fig. 3, where the number of labels required for achieving 80% accuracy is measured for AL-SAR and other AL methods (U, CS, DIS), and baseline non-AL methods such as C: the encoder accompanied by the classifier trained with classification loss only, RC: enhancing C with the decoder and strengthening the training with the reconstruction loss, IRC: a variation of RC which initializes RC with a pre-trained encoder-decoder model [6]. As can be inspected from Fig. 3 (left), AL-SAR requires only 20% of annotated samples to achieve 80% accuracy, which is significantly less than C (which requires 70% of labels). Other baselines such as RC and IRC are more optimal than C, but still require 40%-50% labels. By inspecting the training process with a sparse set of 5% of annotated samples (Fig. 3 right), we observe that one of

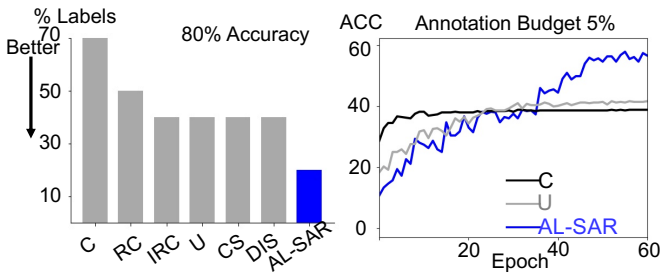

Fig. 3. Left: Annotation (% of labeled samples) required to achieve 80% percent accuracy on NW-UCLA. Comparisons are made for C, RC, IRC, U, DIS, AL-SAR. Right: Training trajectory with 5% annotated samples for C, U, and AL-SAR on NW-UCLA showing that AL-SAR continues to improve with each new set of samples being selected for annotation. Selections are performed at epoch 1, 22, 29, 36, 41.

TABLE III
EVALUATION OF ORIGINAL SPACE (OS), BETA-VARIATIONAL AUTOENCODER (β-VAE), VANILLA ENCODER-DECODER (ED) AND PREDICT&CLUSTER (P&C) IN TERMS OF ABILITY TO PRODUCE MEANINGFUL LATENT STATES FOR CLASSIFICATION.

	β-VAE	OS	ED	P&C
KNN Accuracy	41.5	66.5	82.9	83.6

the reasons that AL-SAR requires fewer labels is that AL-SAR continues to learn over selected samples in multiple iterations in comparison to methods like C or U which reach a plateau after several learning epochs. These analyses confirm that AL-SAR is significantly efficient in leveraging sparsely annotated data for the improvement of the overall action classification.

Comparison with SOTA Semi-Supervised Methods. In addition to comparing AL-SAR with other AL methods, it is of interest to compare AL-SAR with semi-supervised approaches for skeleton-based action recognition, such as ASSL [12], MS^{2}L [13], and SC3D [14], even though these methods assume a given set of annotated samples and do not deal with active sample selection. Results in Table I show that AL-SAR outperforms these methods when the number of annotated samples is small (# Labels $\leq$ 800). With more annotated samples, for instance 4000 samples (10% labels) on NTU RGB+D, ASSL and MS^{2}L perform slightly better than AL-SAR. This is not surprising since the techniques employed by ASSL and MS^{2}L specialize in accumulating the benefits obtained from each additional annotated sample. Similarly, SC3D with 2000 annotated samples exceeds AL-SAR largely due to its significantly larger and more complex network which includes two sub-networks based on GRU and an additional graph convolution neural network. However, in Ablation Study, we demonstrate that SC3D can be enhanced with AL-SAR active sample selection strategy.

This comparison elucidates that the main application of AL-SAR is when the annotation budget is small and when the annotation is performed iteratively along with inspection of the action clusters.

Ablation Study. We test how aspects of AL-SAR, such as latent space clustering with *k-Means* (K), multi-head structure (H), and margin-based selection (M) contribute to its overall performance. First, we test the influence of multi-head mechanism as an additional component to U, in abbreviation UH. Results in Table II show that UH is comparable to or underperforms U in many cases, e.g., UH accuracy is 1.5 and 1.4 lower than U on 30% NW-UCLA and 5% NTU RGB+D CS, respectively. We thus observe that integration of multi-head structure without marginal selection is not optimal. We, therefore, examine variants of marginal selection and test the effectiveness of multi-head components and clustering. In AL-SAR, *MI* is used as the uncertainty measure defined in Eq. 2. When the multi-head mechanism is removed, MK variant), the equation for *MI* becomes

$$MI = \max_{l \in [1:C]} (\mathbf{p}) - \max_{l \in ([1:C] \setminus l^*)} (\mathbf{p}), \tag{6}$$

where $\mathbf{p}$ is the probability output of the classifier taking as input the latent representation $\mathbf{h}_i^T$. As shown in Table II, AL-SAR surpasses MK across datasets and annotation budgets. The Ablation study of the number of heads in the AL-SAR is presented in Supplementary Material. Another variation of multi-head mechanism is Dropout [70] which induces noise into the network and estimates the probability outputs in multiple runs. Our comparison shows that MKD does not perform as well as AL-SAR in classification accuracy. This could be due to Dropout noise induction is not systemic compared to the multi-head mechanism performing the predefined transformation.

We study the effectiveness of clustering in latent space with MH variant in Table II, where clustering is ablated from AL-SAR. MH underperforms AL-SAR by an average of 2.7. Indeed, clustering in the latent space is considered for diversity selection since the space has been shown to self-organize into meaningful structures [6]. As we show in Fig. 4, t-SNE representation of the latent space in the first and final iteration of AL indicates that training that leverages clustering succeeds to form clusters of samples that are clearly identified with action, and thus ensures selection diversity.

Different methods to construct latent space can potentially influence the final accuracy. We assess how well the latent space obtained by beta-variational autoencoder (β-VAE) [53], vanilla encoder-decoder (ED), and Predict&Cluster (P&C) [6] construct discriminative clusters. In all three variants, we evaluate the accuracy of KNN (k=1) on NW-UCLA

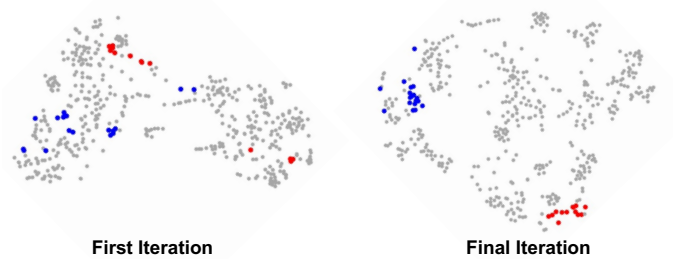

Fig. 4. T-SNE embeddings of the latent space for the first iteration (left) and the final iteration (right) on UCLA dataset. Two representative classes belonging to actions of 'two hand punching' (red) and 'squatting' (blue) are marked. The first iteration is initialized with 10% annotated samples, where in each iteration additional 10% of samples are annotated. At the final iteration, the same class samples are gathered into more enhanced clusters

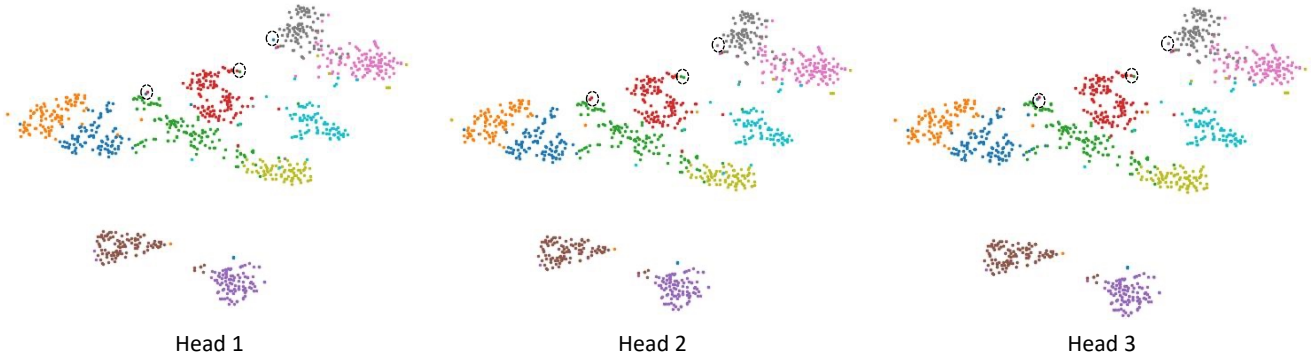

Fig. 5. Visualization of the classifier predictions in t-SNE embedded latent space. The predictions are obtained from the well-trained classifier as it is connected to different heads, and colors indicate class labels. The classifier makes consistent predictions across three heads except for a few examples (denoted by black dashed circles).

TABLE IV
COMPARISON OF VANILLA SC3D, AL-SAR WITH THE ENCODER-DECODER OF P&C (AL-SAR-PC), AND AL-SAR WITH THE ENCODER IN SC3D (AL-SAR-SC3D) ON NTU RGB+D 60 CS.

% Labels	1%	5%	10%
SC3D	35.7	59.6	65.9
AL-SAR-PC	38.8	57.9	63.7
AL-SAR-SC3D	**40.0**	**63.8**	**72.7**

and use the same encoder and decoder for comparison. We report the performance in Table III along with the baseline of KNN accuracy (k=1) on the original input space (OS). P&C architecture yields the highest accuracy and thus we choose this architecture for AL-SAR. In addition, we are interested in whether a powerful semi-supervised learning method could benefit AL-SAR. One such method is SC3D [14], which shows satisfying action recognition accuracy on NTU RGB+D dataset under 5% and 10% annotation budget (Table I). SC3D's primary relevance to our work is in learning the data representation for AL-SAR as an alternative to P&C [6]. Therefore, we substituted the encoder of SC3D for the encoder-decoder of P&C to test with whether this substitution will enhance the results of AL-SAR with P&C and/or the results of SC3D alone. The results are shown in Table IV. We denote the original AL-SAR as AL-SAR-PC to distinguish with SC3D substituted AL-SAR (AL-SAR-SC3D). AL-SAR-SC3D surpasses the performance of both vanilla SC3D and AL-SAR-PC on three annotation budgets 1%, 5%, and 10%. Specifically, at 10%, the improvement is 6.8 and 9.0 over vanilla SC3D and AL-SAR-PC, respectively. The results indicate that incorporating a more complex model in AL-SAR, such as SC3D, could improve the accuracy and demonstrates the versatility of AL-SAR in discovering informative samples from latent representations embedded with different encoders.

The Multi-head Mechanism. As aforementioned, the multi-heads is a set of randomly initialized and fixed fully connected layers inserted right before the last classification layer. It would be interesting to know the effects of these heads on the prediction of the classifier. Thus in Fig. 5, we visualize predictions of the well-trained classifier in the latent representation space as it is connected to different heads. The figure shows that the classifier makes almost the same predictions as it is connected to each of the three heads. This verifies that the classifier generalizes to outputs from different heads and makes consistent predictions. To better interpret the effectiveness of the multi-head mechanism for active selection, we further quantitatively analyze the ability of AL-SAR (Eq. 2) and MK (Eq. 6) to identify misclassified samples. Specifically, we compute the misclassification ratio within the selected samples in the four selections where *MI* is computed in Fig 6. Unlike MK, AL-SAR has a higher misclassification ratio in most iterations, except iteration 4 (I4). Overall, AL-SAR still selects more misclassified samples integrated over all four iterations (I1-I4). This indicates the advantages of AL-SAR, since, compared to correctly classified samples, these misclassified samples are supposed to be more valuable for correcting and providing additional information to the classifiers. We additionally visualized the location and correctness of selected samples in Supplementary Material.

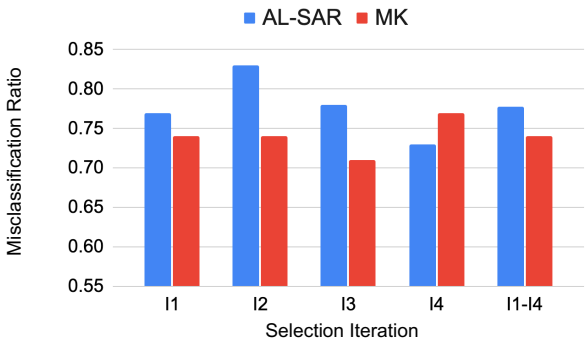

Fig. 6. Misclassification ratio within selected samples. Comparisons are made between the AL-SAR (blue bar) and MK (red bar) in four iterations: iteration 1 (I1), iteration 2 (I2), iteration 3 (I3), iteration 4 (I4), and integration of the four iterations (I1-I4). In most cases, AL-SAR has a higher misclassification ratio indicating the preference of AL-SAR for selecting more informative misclassified samples.

As previous works indicate, classifiers can be overconfident

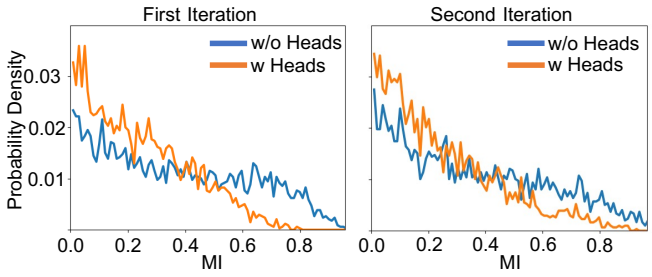

Fig. 7. *MI* distribution among the mis-classified samples in the presence (w) and absence (w/o) of multi-head structure at two stages: first iteration (left) and second iteration (right). Compared to w/o Heads, *MI* computed with heads is less likely to be larger than 0.6.

about the misclassified samples [61], [71] in which cases the computed *MI* could be misleadingly high (low uncertainty). We thus study whether the multi-head mechanism computed *MI* can mitigate the artifact caused by the overconfident classifier. Here, we estimate the *MI* distribution of misclassified samples using the multi-head mechanism and without the multi-head mechanism in two selection iterations (Fig 7). The density at the high *MI* regime could indicate the extent of over-confidence since, presumably, the *MI* of misclassified samples should be low. As shown in Fig 7, in the high MI regime, the density of *MI* generated by the multi-head mechanism is lower (orange curve) than the one without the multi-head mechanism (blue curve). This indicates that, with the multi-head mechanism that integrates the classifier outputs from all the heads to guide the sample selection, the estimated *MI* is less likely to be affected by the overconfident classifier, thus constituting a more robust uncertainty estimation.

V. Conclusion

We have introduced a novel approach for Active Learning for Skeleton-based Action Recognition (AL-SAR). The approach connects unsupervised learning with sparse active selection of sequences for annotation and boosts the action recognition performance. AL-SAR introduces a novel approach of measuring uncertainty, through multi-head mechanism, and leverages the unsupervised latent space to ensure diversity. We apply our approach to skeleton-based human action recognition benchmarks and compare the performance with current semi-supervised methods. We show that our proposed method outperforms these approaches in sparse annotation scenarios, i.e., when only a handful of samples are selected for annotation in an iterative way.

References

[1] Y. Du, W. Wang, and L. Wang, "Hierarchical recurrent neural network for skeleton based action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1110–1118.

[2] C. Li, Q. Zhong, D. Xie, and S. Pu, "Co-occurrence feature learning from skeleton data for action recognition and detection with hierarchical aggregation," *arXiv preprint arXiv:1804.06055*, 2018.

[3] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 9532–9545, 2020.

[4] B. Fu, S. Fu, L. Wang, Y. Dong, and Y. Ren, "Deep residual split directed graph convolutional neural networks for action recognition," *IEEE MultiMedia*, vol. 27, no. 4, pp. 9–17, 2020.

[5] N. Zheng, J. Wen, R. Liu, L. Long, J. Dai, and Z. Gong, "Unsupervised representation learning with long-term dynamics for skeleton based action recognition," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[6] K. Su, X. Liu, and E. Shlizerman, "Predict & cluster: Unsupervised skeleton based action recognition," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[7] K. Su and E. Shlizerman, "Clustering and recognition of spatiotemporal features through interpretable embedding of sequence to sequence recurrent neural networks," *arXiv preprint arXiv 1905.12176*, 2020.

[8] J. Wang, Y. Wang, S. Liu, and A. Li, "Few-shot fine-grained action recognition via bidirectional attention and contrastive meta-learning," in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 582–591.

[9] J. Patravali, G. Mittal, Y. Yu, F. Li, and M. Chen, "Unsupervised few-shot action recognition via action-appearance aligned meta-adaptation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 8484–8494.

[10] L. Wang, J. Liu, and P. Koniusz, "3d skeleton-based few-shot action recognition with jeanie is not so na\" ive," *arXiv preprint arXiv:2112.12668*, 2021.

[11] A. Zhu, Q. Ke, M. Gong, and J. Bailey, "Adaptive local-component-aware graph convolutional network for one-shot skeleton-based action recognition," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 6038–6047.

[12] C. Si, X. Nie, W. Wang, L. Wang, T. Tan, and J. Feng, "Adversarial self-supervised learning for semi-supervised 3d action recognition," *arXiv preprint arXiv:2007.05934*, 2020.

[13] L. Lin, S. Song, W. Yang, and J. Liu, "Ms2l: Multi-task self-supervised learning for skeleton based action recognition," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2490–2498.

[14] F. M. Thoker, H. Doughty, and C. G. Snoek, "Skeleton-contrastive 3d action representation learning," *arXiv preprint arXiv:2108.03656*, 2021.

[15] D. A. Cohn, Z. Ghahramani, and M. I. Jordan, "Active learning with statistical models," *Journal of artificial intelligence research*, vol. 4, pp. 129–145, 1996.

[16] B. Settles, "Active learning literature survey," University of Wisconsin-Madison Department of Computer Sciences, Tech. Rep., 2009.

[17] ——, "Active learning," *Synthesis lectures on artificial intelligence and machine learning*, vol. 6, no. 1, pp. 1–114, 2012.

[18] O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," *arXiv preprint arXiv:1708.00489*, 2017.

[19] W. H. Beluch, T. Genewein, A. Nürnberger, and J. M. Köhler, "The power of ensembles for active learning in image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9368–9377.

[20] M. Gao, Z. Zhang, G. Yu, S. Ö. Arık, L. S. Davis, and T. Pfister, "Consistency-based semi-supervised active learning: Towards minimizing labeling cost," in *European Conference on Computer Vision*. Springer, 2020, pp. 510–526.

[21] D. Roth and K. Small, "Margin-based active learning for structured output spaces," in *European Conference on Machine Learning*. Springer, 2006, pp. 413–424.

[22] Y. Gal, R. Islam, and Z. Ghahramani, "Deep bayesian active learning with image data," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1183–1192.

[23] S. Sinha, S. Ebrahimi, and T. Darrell, "Variational adversarial active learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 5972–5981.

[24] B. Settles and M. Craven, "An analysis of active learning strategies for sequence labeling tasks," in *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, 2008, pp. 1070–1079.

[25] Y. Deng, K. Chen, Y. Shen, and H. Jin, "Adversarial active learning for sequences labeling and generation." in *IJCAI*, 2018, pp. 4012–4018.

[26] A. Shelmanov, D. Puzyrev, L. Kupriyanova, D. Belyakov, D. Larionov, N. Khromov, O. Kozlova, E. Artemova, D. V. Dylov, and A. Panchenko, "Active learning for sequence tagging with deep pre-trained models and bayesian uncertainty estimates," *arXiv preprint arXiv:2101.08133*, 2021.

[27] L. Xia, C.-C. Chen, and J. K. Aggarwal, "View invariant human action recognition using histograms of 3d joints," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, 2012, pp. 20–27.

[28] J. Wang, Z. Liu, Y. Wu, and J. Yuan, "Learning actionlet ensemble for 3d human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 914–927, 2013.

[29] R. Vemulapalli, F. Arrate, and R. Chellappa, "Human action recognition by representing 3d skeletons as points in a lie group," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 588–595.

[30] M. Liu, H. Liu, and C. Chen, "Enhanced skeleton visualization for view invariant human action recognition," *Pattern Recognition*, vol. 68, pp. 346–362, 2017.

[31] T. S. Kim and A. Reiter, "Interpretable 3d human action analysis with temporal convolutional networks," in *2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW)*. IEEE, 2017, pp. 1623–1631.

[32] Q. Ke, M. Bennamoun, S. An, F. Sohel, and F. Boussaid, "A new representation of skeleton sequences for 3d action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3288–3297.

[33] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+ d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1010–1019.

[34] W. Zhu, C. Lan, J. Xing, W. Zeng, Y. Li, L. Shen, and X. Xie, "Co-occurrence feature learning for skeleton based action recognition using regularized deep lstm networks," *arXiv preprint arXiv:1603.07772*, 2016.

[35] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[36] X. Zhang, C. Xu, and D. Tao, "Context aware graph convolution for skeleton-based action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 333–14 342.

[37] K. Cheng, Y. Zhang, X. He, W. Chen, J. Cheng, and H. Lu, "Skeleton-based action recognition with shift graph convolutional network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 183–192.

[38] M. Li, S. Chen, Y. Zhao, Y. Zhang, Y. Wang, and Q. Tian, "Dynamic multiscale graph neural networks for 3d skeleton based human motion prediction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 214–223.

[39] S. Hanneke *et al.*, "Theory of disagreement-based active learning," *Foundations and Trends® in Machine Learning*, vol. 7, no. 2-3, pp. 131–309, 2014.

[40] K. Lang and E. Baum, "Query learning can work poorly when a human oracle is used," *IEEE Intl. JointConference on Neural Networks*, 1992.

[41] K. Brinker, "Incorporating diversity in active learning with support vector machines," in *Proceedings of the 20th international conference on machine learning (ICML-03)*, 2003, pp. 59–66.

[42] Z. Xu, R. Akella, and Y. Zhang, "Incorporating diversity and density in active learning for relevance feedback," in *European Conference on Information Retrieval*. Springer, 2007, pp. 246–257.

[43] Y. Wang, J. Peng, H. Wang, and M. Wang, "Progressive learning with multi-scale attention network for cross-domain vehicle re-identification," *Science China Information Sciences*, vol. 65, no. 6, pp. 1–15, 2022.

[44] R. Hwa, "Sample selection for statistical parsing," *Computational linguistics*, vol. 30, no. 3, pp. 253–276, 2004.

[45] A. Culotta and A. McCallum, "Reducing labeling effort for structured prediction tasks," in *AAAI*, vol. 5, 2005, pp. 746–751.

[46] M.-F. Balcan, A. Broder, and T. Zhang, "Margin based active learning," in *International Conference on Computational Learning Theory*. Springer, 2007, pp. 35–50.

[47] A. J. Joshi, F. Porikli, and N. Papanikolopoulos, "Multi-class active learning for image classification," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 2372–2379.

[48] E. Tsymbalov, M. Panov, and A. Shapeev, "Dropout-based active learning for regression," in *International conference on analysis of images, social networks and texts*. Springer, 2018, pp. 247–258.

[49] E. Elhamifar, G. Sapiro, A. Yang, and S. S. Sasrty, "A convex optimization framework for active learning," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 209–216.

[50] L. Yang, Y. Zhang, J. Chen, S. Zhang, and D. Z. Chen, "Suggestive annotation: A deep active learning framework for biomedical image segmentation," in *International conference on medical image computing and computer-assisted intervention*. Springer, 2017, pp. 399–407.

[51] L. F. Coletta, M. Ponti, E. R. Hruschka, A. Acharya, and J. Ghosh, "Combining clustering and active learning for the detection and learning of new image classes," *Neurocomputing*, vol. 358, pp. 150–165, 2019.

[52] L. Wu, Y. Wang, L. Shao, and M. Wang, "3-d personvlad: Learning deep global representations for video-based person reidentification," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 11, pp. 3347–3359, 2019.

[53] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," 2016.

[54] K. G. Dizaji, F. Zheng, N. Sadoughi, Y. Yang, C. Deng, and H. Huang, "Unsupervised deep generative adversarial hashing network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3664–3673.

[55] S. Mukherjee, H. Asnani, E. Lin, and S. Kannan, "Clustergan: Latent space clustering in generative adversarial networks," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4610–4617.

[56] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[57] Y. Wang, "Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 17, no. 1s, pp. 1–25, 2021.

[58] C. Zhang, H. Fu, Q. Hu, X. Cao, Y. Xie, D. Tao, and D. Xu, "Generalized latent multi-view subspace clustering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 1, pp. 86–99, 2018.

[59] Z. Kang, Z. Lin, X. Zhu, and W. Xu, "Structured graph learning for scalable subspace clustering: From single view to multiview," *IEEE Transactions on Cybernetics*, 2021.

[60] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *CVPR 2011*. Ieee, 2011, pp. 1297–1304.

[61] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *arXiv preprint arXiv:1610.02136*, 2016.

[62] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[63] S. Ö. Arık, H. Jun, and G. Diamos, "Fast spectrogram inversion using multi-head convolutional neural networks," *IEEE Signal Processing Letters*, vol. 26, no. 1, pp. 94–98, 2018.

[64] D. Maltz and K. Ehrlich, "Pointing the way: Active collaborative filtering," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 1995, pp. 202–209.

[65] N. Houlsby, J. M. Hernández-Lobato, and Z. Ghahramani, "Cold-start active learning with robust ordinal matrix factorization," in *International Conference on Machine Learning*. PMLR, 2014, pp. 766–774.

[66] H. Rahmani, A. Mahmood, D. Q. Huynh, and A. Mian, "Hopc: Histogram of oriented principal components of 3d pointclouds for action recognition," in *European conference on computer vision*. Springer, 2014, pp. 742–757.

[67] J. Wang, X. Nie, Y. Xia, Y. Wu, and S.-C. Zhu, "Cross-view action modeling, learning and recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 2649–2656.

[68] P. Zhang, C. Lan, J. Xing, W. Zeng, J. Xue, and N. Zheng, "View adaptive recurrent neural networks for high performance human action recognition from skeleton data," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2117–2126.

[69] H. Rao, S. Xu, X. Hu, J. Cheng, and B. Hu, "Augmented skeleton based contrastive action learning with momentum lstm for unsupervised action recognition," *Information Sciences*, vol. 569, pp. 90–109, 2021.

[70] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[71] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak, "On mixup training: Improved calibration and predictive uncertainty for deep neural networks," *Advances in Neural Information Processing Systems*, vol. 32, 2019.