Generative View-Correlation Adaptation for Semi-Supervised Multi-View Learning

Yunyu Liu[1], Lichen Wang[1], Yue Bai[1], Can Qin[1], Zhengming Ding[2], and Yun Fu[1]

[1] Northeastern University, MA, USA
[2] Indiana University-Purdue University Indianapolis, IN, USA
{liu.yuny,bai.yue,qin.ca}@northeastern.edu
wanglichenxj@gmail.com zd2@iu.edu yunfu@ece.neu.edu

Abstract. Multi-view learning (MVL) explores the data extracted from multiple resources. It assumes that the complementary information between different views could be revealed to further improve the learning performance. There are two challenges. First, it is difficult to effectively combine the different view data while still fully preserve the view-specific information. Second, multi-view datasets are usually small, which means the model can be easily overfitted. To address the challenges, we propose a novel View-Correlation Adaptation (*VCA*) framework in semi-supervised fashion. A semi-supervised data augmentation me-thod is designed to generate extra features and labels based on both labeled and unlabeled samples. In addition, a cross-view adversarial training strategy is proposed to explore the structural information from one view and help the representation learning of the other view. Moreover, an effective and simple fusion network is proposed for the late fusion stage. In our model, all networks are jointly trained in an end-to-end fashion. Extensive experiments demonstrate that our approach is effective and stable compared with other state-of-the-art methods [3].

Keywords: Multi-view learning, Data Augmentation, Semi-supervised learning

1 Introduction

Multi-view data refers to the data captured from multiple resources such as RGB, depth, and infrared in visual space [38, 23, 33]. MVL methods assume that the information from different views is unique and complementary. By learning from different views, MVL methods could achieve better performance. Several researches, such as [31, 8, 4, 13, 14, 9], have studied MVL in supervised setting. However, the challenge is that labeling all the multi-view data can be extremely expensive. Therefore, semi-supervised learning setting is a more practical strategy since obtaining unlabeled data is easy.

[3] Code is available on: https://github.com/wenwen0319/GVCA

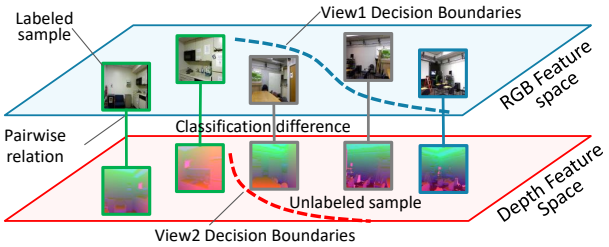

Fig. 1. The main challenge of MVL is that it is difficult to explore the latent correlations across different views due to the view heterogeneity. For instance, different view data from the same sample sometimes have the different predictions. Thus, effectively learn the cross-view information and fully explore the unlabeled data can potentially improve the learning performance.

As shown in Figure 1, the main challenge of MVL is that different views have different data formats, feature distributions, and predicted results, namely data heterogeneity. Multi-view fusion is an efficient strategy. It mainly divided into three parts: raw-data fusion, feature-level fusion, and label-level fusion. [16] utilized feature-level and label-level fusion for learning a better representation. [10] applied GAN and directly translated one view to the other view which belongs to raw-data fusion. Although these methods achieved good performance, they considered the three fusion strategies independently which cannot fully reveal the latent relations between these steps. Semi-supervised learning [7, 10, 30, 29, 32] is an effective approach which explores the unlabeled samples to increase the learning performance. [7] proposed a new co-training method for the training process. [10] used the unlabeled data to enhance the view translation process. However, these methods combined semi-supervised setting and multi-view learning in an empirical way while lacked a unified framework to jointly solve the challenges.

In this work, we propose a View-Correlation Adaptation (*VCA*) framework in semi-supervised scenario. The framework is shown in Figure 2. First, a specifically designed View-Correlation Adaptation method is proposed. It effectively aligns the feature distributions across different views. Second, a semi-supervised feature augmentation (*SeMix*) is proposed. It augments training samples by exploring both labeled and unlabeled samples. Third, we propose an effective and simple fusion network to learn both inter-view and intra-view high-level label correlations. By these ways, the label-level and feature-level fusion are considered simultaneously in an unified framework. This further helps our model obtain more distinctive representation for classification. Extensive experiments demonstrate that our method achieves the highest performance compared with state-of-the-art methods. Ablation studies illustrate the effectiveness of each module in the framework. The main novelties are listed below.

– A novel fusion strategy View-Correlation Adaptation (*VCA*) is deployed in both feature and label space. *VCA* makes one view learn from the other view by optimizing the decision discrepancy in an adversarial training manner.

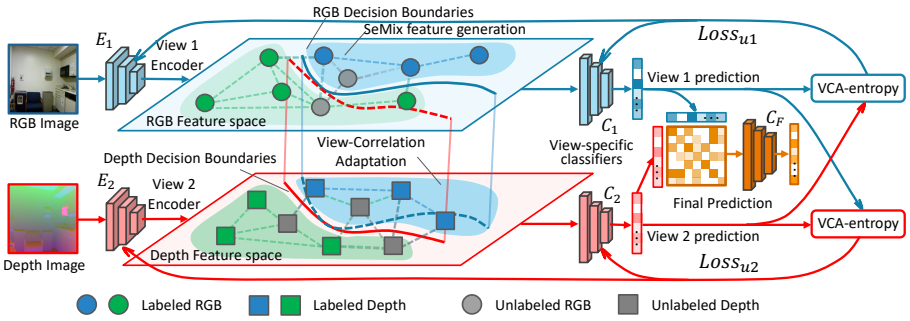

Fig. 2. Framework of our model. Multi-view samples first go through the view encoders $E_1(\cdot)$ and $E_2(\cdot)$ and get the representation features. We design *SeMix* to the feature space in order to expand the feature space and align the distribution of the labeled and unlabeled data. The representation features are sent to the view-specific classifiers $C_1(\cdot)$ and $C_2(\cdot)$. Then *VCA-entropy* is applied to learn general representations while keeping view-specific characteristics. For two single views' predictions of the unlabeled data, the one with a high entropy should learn form the other one since the entropy indicates the uncertainty of predictions. By playing an adversarial training strategy, the view encoders learn general feature distributions and the view specific classifiers keep view-specific characteristics. A cross-view fusion network $C_F(\cdot)$ is used to fuse the predictions and get a final result by capturing the inter-view and intra-view label correlation.

- A new generative *SeMix* approach is proposed to fully utilize the labeled and unlabeled data to expand the feature distributions and make the model more robust.
- An effective label-level fusion network is proposed to obtain the final classification result. This module captures both the inter-view and intra-view high-level label correlations to obtain a higher performance.

2 Related work

2.1 Multi-view Learning

Multi-view learning explores data from different views/resources for downstream tasks (*e.g.*, classification and clustering). RGB and depth are the two commonly explored views of MVL. [8, 4, 13, 14, 28] focused on learning a better representation for actions using supervised information. [12, 27, 2, 16, 7, 10] studied multi-view scene recognition. Conventional MVL methods focused on improving the multi-view features. [4] introduced a multi-view super vector to fuse the action descriptors. [13] combined optical flow with enhanced 3D motion vector fields to fuse features. [12] used contours of depth images as the local features for scene classification. [2] proposed a second-order pooling to extract the local features. Deep learning framework was proposed in the fusion procedure recently due to its

potential in a wide range of applications. [27] utilized a component-aware fusion method to fuse the features extracted from deep models. [28] introduced a generative/mapping framework which learns the view-independent and view-specific representations. [16] implemented a method to learn the distinctive embedding and correlative embedding simultaneously. [10] proposed a generative method and translated one view to the other to help classification.

However, the fusion modules are separated with other modules in most methods, which degrades the potential of the methods and make the training procedure tedious to obtain good results. Our model is a union framework with both the label-level and feature-level fusion strategies. In addition, our model is a more general framework which and can be adapted to different tasks.

2.2 Semi-supervised Learning

Semi-supervised learning deploys both unlabeled and labeled data in the training procedure. It is an effective strategy for a situation where collecting data is easy while labeling the data is extremely difficult. Semi-supervised learning explores the structural information of the labeled and unlabeled data to improve the performance. The general introduction of semi-supervised learning could be found in [6]. [35] designed co-training method for semi-supervised learning. Two-learner framework is utilized and the unlabeled samples would be assigned confident labels based on the guidance of the two learners. [3] proposed a MixMatch framework which guesses low-entropy labels for data-augmented unlabeled examples. [36] proposed a multi-modal method based on Alexnet [15] to solve the missing view issues in semi-supervised fashion. [7] introduced a co-training method combined with deep neural network. [21] implemented a graph-based method to solve the multi-view semi-supervised problem. [10] trained a translation network using both labeled and unlabeled data to obtain representations.

Although high performance is achieved by the methods. They are mainly focusing on single-view setting while ignore the sophisticated cross-view relations and the extra latent connections between labeled and unlabeled samples. Our approach is specifically designed for multi-view semi-supervised learning. It jointly explores the cross-view and the cross-instance relations. Based on our experiments, the revealed knowledge considerably improves the performance and robustness of our approach.

3 Our approach

3.1 Preliminaries & Motivation

$X_l^1 \in \mathbb{R}^{d_1 \times n_l}$ and $X_l^2 \in \mathbb{R}^{d_2 \times n_l}$ are the feature matrices of two views, and they belong to labeled samples. n_l stands for the labeled instance number. d_1, d_2 are the feature dimensions of view1 and view2, and each column is an instance. Similarly, $X_u^1 \in \mathbb{R}^{d_1 \times n_u}$ and $X_u^2 \in \mathbb{R}^{d_2 \times n_u}$ are the feature matrices of unlabeled samples obtained from view1 and view2. n_u is the unlabeled instance number.

The label matrix $Y_l \in \mathbb{R}^{d_l \times n_l}$ of labeled samples are given, where d_l is the dimension of the label space. We denote the feature and label vector of the i-th instance by x_{li}^1, x_{li}^2 and y_{li}. The goal is to recover the label of the unlabeled set given X_l^1, X_l^2, X_u^1, X_u^2 and Y_l. Generally, our framework consists view-specific encoders $E_1(\cdot)$, $E_2(\cdot)$, view-specific classifiers $C_1(\cdot)$, $C_2(\cdot)$, and a fusion network $C_F(\cdot)$. The encoders encode the original data to a subspace. The classifiers obtain classification score L_1 and L_2. Then a label-level fusion mechanism $C_F(\cdot)$ is applied to derive a final result. Details are introduced in the following sections.

3.2 Semi-supervised Mixup

Limited training samples is a common challenge for general machine learning tasks and it is significant in MVL scenario. Mixup [37] is an effective data augmentation strategy. Although various modifications are proposed [3, 26], the label information is always required. In this work, we propose a novel semi-supervised version Mixup approach *SeMix*. Different from existing methods, *SeMix* fully explores feature distributions of both labeled and unlabeled samples, and generates more general samples with confident labels. *SeMix* could be divided into two parts: 1) *Labeled augmentation* and 2) *Unlabeled augmentation*.

Labeled Augmentation. Given labeled samples, *SeMix* augments labeled data based on the follow steps. A random variable $\lambda' \sim Beta(\alpha, \alpha)$ is sampled where $Beta(\alpha, \alpha)$ is the Beta distribution. And the weight parameter λ is obtained via $\lambda = \max(\lambda', 1-\lambda')$, where $\max(\cdot, \cdot)$ is the max value of the inputs. Then, the augmented representations, $\tilde{r^i}$, and the corresponding labels, $\tilde{y}$, could be obtained from the equations below:

$$\begin{aligned} \tilde{r}_{nm}^i &= \lambda E_i(x_{ln}^i) + (1-\lambda) E_i(x_{lm}^i), i = \{1, 2\}, \\ \tilde{y}_{nm} &= \lambda y_{ln} + (1-\lambda) y_{lm}, \end{aligned} \tag{1}$$

where x_{ln}^i and x_{lm}^i are two randomly selected labeled samples from the i-th view, y_{ln} and y_{lm} are the corresponding label vectors. The obtained $\tilde{r}_{nm}^i$ and $\tilde{y}_{nm}$ could be directly involved in the training procedure. In Eq. (1), *SeMix* augments the samples in the encoded spaces instead of the original feature space since we observed that the performance is higher and more stable.

Eq. (1) is the redesigned version of Mixup for labeled samples. There are two-fold improvements compared with conventional Mixup methods [37]. First, *SeMix* achieves augmentations in the low-dimensional subspace instead of the original feature space. The representation of each sample in the subspace becomes more distinctive with lower noise. By this way, *SeMix* could effectively and accurately explore the structural knowledge and generate clear and high quality samples. The decision boundaries of the learned classifiers would become smoother [26] which improves the generalization and the model. Second, we define λ' following a Beta distribution instead of a uniform distribution and make sure λ is bigger than 0.5 by the equation $\lambda = \max(\lambda', 1-\lambda')$. This strategy promises $E_i(x_{ln}^i)$ is in a dominate place compared with $E_i(x_{lm}^i)$. This constraint is also useful for the unlabeled data and will be discusses in the following section.

Unlabeled Augmentation. The above mentioned approach could only work when the labels of both the pairwise samples are known. To fully explore the distribution knowledge from the unlabeled samples, an effective algorithm is designed which bypass the label missing issue while still guarantee the quality of the augmented samples. Randomly select an unlabeled sample x_u^i and a labeled sample x_{lm}^i, where $i = \{1, 2\}$ means both of them are from the i-th view. Assuming the label of x_u^i is y'_u. According to Eq. (1), replacing x_{ln}^i and y_{ln} with x_u^i and y'_u, we can obtain the representations and labels of x_u^i and x_{lm}^i pair as:

$$\begin{aligned} \tilde{r}_{um}^i &= \lambda E_i(x_u^i) + (1-\lambda)E_i(x_{lm}^i), i = \{1, 2\}, \\ \tilde{y}_{um} &= \lambda y'_u + (1-\lambda)y_{lm}. \end{aligned} \tag{2}$$

Similarly, when x_u^i is paired with another labeled sample x_{ln}^i, we can obtain another set of representation and label as below:

$$\begin{aligned} \tilde{r}_{un}^i &= \lambda E_i(x_u^i) + (1-\lambda)E_i(x_{ln}^i), i = \{1, 2\}, \\ \tilde{y}_{un} &= \lambda y'_u + (1-\lambda)y_{ln}. \end{aligned} \tag{3}$$

Since y'_u is unknown, Eq. (2) and Eq. (3) cannot be directly deployed in the training process. Therefore, we calculate the difference of the obtained labels, $\tilde{y}_{um}$ and $\tilde{y}_{un}$, and we can obtain the result as follow:

$$\tilde{y}_{un} - \tilde{y}_{um} = (1-\lambda)(y_{ln} - y_{lm}). \tag{4}$$

Therefore, Eq. (4) effectively bypass the missing label issue. When training, we randomly selected an unlabeled sample x_u^i and two labeled samples x_{lm}^i and x_{ln}^i to obtain the label difference. We also deploy the prediction differences instead of exact prediction results in the objective function which is shown below:

$$L_{semi} = \sum_{i=1}^{2} \|(C_i(\tilde{r}_{um}^i) - C_i(\tilde{r}_{un}^i)) - (1-\lambda)(y_{ln} - y_{lm})\|_{\mathrm{F}}^2, \tag{5}$$

where L_{semi} is the objective value based on the label differences and we minimize L_{semi} by jointly optimizing both the view-specific encoders and classifiers (*i.e.*, $E_1(\cdot)$, $E_2(\cdot)$, $C_1(\cdot)$, and $C_2(\cdot)$). In this way, *SeMix* fully explores the distribution knowledge across the labeled and unlabeled samples which enhance the diversity and generality of the augmented samples.

We give an intuitive explanation of *SeMix* in word embedding scenario for easy understanding. For instance, we want the learned encoders and classifiers are robust which could obtain accurate representation such as *man-woman = king-queen*. This also explains why we set λ in the larger side (*i.e.*, $\max(\lambda', 1-\lambda')$). If λ is small, the unlabeled data will be regarded as a noise since its scale is small. It is hard for the model to find useful information in the unlabeled data.

3.3 Dual-level View-Correlation Adaptation

In MVL, classification discrepancy is that different views have different classification results. This commonly exists that because different views have their unique characteristics. It can be a valuable clue for MVL.

Here, we propose a novel way to explore and align the discrepancy in both feature space and label space, and we name this mechanism as Dual-level View-Correlation Adaptation(VCA). For the labeled data, the classification results should be the same as the ground-truth. The objective function is as follow:

$$L_{labeled} = \sum_{i=1}^{2} \|\tilde{y}^i - y\|_{\mathrm{F}}^2, \tag{6}$$

where $\tilde{y}_i$ is the classification result from the i-th view and y is the ground truth. The training samples are labeled samples and the generated samples via Eq. (1).

For the unlabeled data, instead of forcibly eliminating the discrepancy of $C_1(\cdot)$ and $C_2(\cdot)$, we optimize the encoders $E_1(\cdot)$ and $E_2(\cdot)$ to minimize the discrepancy while the classifiers $C_1(\cdot)$ and $C_2(\cdot)$ to maximize the discrepancy. As shown in Figure 2, this adversarial training strategy is the crucial module for cross-view structure learning process. We define the objective function:

$$L_{unlabeled} = W(C_1(\tilde{r}_n^1), C_2(\tilde{r}_n^2)), \tag{7}$$

where $W(\cdot,\cdot)$ is the Wasserstein Distance, the inputs are the classification results from two views. We assume the prediction of the unlabeled samples are noisy and uncertain, using Wasserstein distance can be more effective and stable.

Both of Eq. (6) and Eq. (7) are used to calculate the classification discrepancies. We empirically tested different evaluation setups and we found out this setup achieves the best performance.

The four networks are alternately optimized. Firstly, we use the labeled data to train $E_1(\cdot)$, $E_2(\cdot)$ and $C_1(\cdot)$, $C_2(\cdot)$. The loss can be defined as follows.

$$\min_{C_1,C_2,E_1,E_2} L_{labeled}. \tag{8}$$

The second step is letting the encoders minimize the discrepancy.

$$\min_{E_1,E_2} L_{labeled} + L_{unlabeled}. \tag{9}$$

The third step is letting the classifiers to enlarge the discrepancy.

$$\min_{C_1,C_2} L_{labeled} - L_{unlabeled}. \tag{10}$$

Labeled loss $L_{labeled}$ to Eq. (9) and Eq. (10) are added to stabilize the optimization. The empirical experiments also show that the performance decreases significantly without $L_{labeled}$.

By adversarial training strategy, $E_i(\cdot)$ could borrow the structural information from the other view and obtain more distinctive feature representations in their subspaces. In addition, $C_i(\cdot)$ would obtain robust classification boundaries. Overall, the adversarial manner promises: 1) the training step is more stable and 2) the model is more robust and has less possibility to be over-fitting.

Entropy-based modification. For Eq. (9) and Eq. (10), ideally, we want the encoders to learn a more robust representations based on the guidance of the

other view. However, in some cases, this process may decrease the capacity of the encoders and make the final prediction results worse. For example, the view has a wrong result affects the view with the right result by pulling them together.

Therefore, we introduce an entropy based module to handle this challenge. Our goal is to make the view with the confident prediction to guide the other and avoid the reverse effect. To achieve this goal, the entropy is used to evaluate the confidence of a classification. The higher the entropy, the more uncertainty(lower confidence) of the classification results. Specifically, if view 1 has a higher entropy than view 2, the loss of view 1 should be larger than the loss of view 2. This will encourage view 1 learns from view 2 while view 2 tries to preserve its own knowledge. Therefore, Eq. (6) and Eq. (7) can be revised as below,

$$L^1_{unlabeled} = \frac{H(C_1(\tilde{r}^1))}{H(C_2(\tilde{r}^2))} W(C_1(\tilde{r}^1), C_2(\tilde{r}^2)), \tag{11}$$

$$L^2_{unlabeled} = \frac{H(C_2(\tilde{r}^2))}{H(C_1(\tilde{r}^1))} W(C_1(\tilde{r}^1), C_2(\tilde{r}^2)), \tag{12}$$

where $H(\cdot)$ stands for the entropy. The inputs of $H(\cdot)$ are two classification results from two views. For $W(\cdot,\cdot)$, we have $W(C_1(\tilde{r}^1), C_2(\tilde{r}^2)) = W(C_2(\tilde{r}^2), C_1(\tilde{r}^1))$. Meanwhile, Eq. (9) and Eq. (10) are changed correspondingly as follows.

$$\min_{E_i} L_{labeled} + L^i_{unlabeled}, i = \{1, 2\}, \tag{13}$$

$$\min_{C_i} L_{labeled} - L^i_{unlabeled}, i = \{1, 2\}. \tag{14}$$

3.4 Label-level Fusion

The discrepancy still exists even after the alignment procedure. We assume different views have the unique characteristics and some categories could achieve more accurate/reliable classification results from a specific view. An intuitive example, if there are "Police" and "Doctor" views, "Police" view is good at distinguishing good/bad people, while "Doctor" view is good at distinguishing healthy/sick bodies. Conventional methods utilize naive mechanisms (*e.g.*, mean or max pooling) which lost valuable information. Based on this assumption, we deploy a novel fusion strategy in label space. It automatically learns the prediction confidences between different views for each category. Then, the best prediction is learned as the final classification result.

In our model, we apply a cross-view fusion network $C_F(\cdot)$ to handle this challenge. For two predictions from different views $\tilde{y}_1$ and $\tilde{y}_2$, we multiply the predictions with the transpose to obtain a matrix. There are totally three matrices(*i.e.*, $\tilde{y}_1^\top \tilde{y}_1$, $\tilde{y}_2^\top \tilde{y}_2$ and $\tilde{y_1}^\top \tilde{y}_2$) In the matrix, each element is the multiplication of the pairwise prediction scores. Then, we sum up three matrices and reshape the matrix to a vector and forward to the fusion network $C_F(\cdot)$. The framework is illustrated in Figure 2. The objective function is shown below:

$$L_F = \|C_F\big(reshape(\tilde{y}_1^\top \tilde{y}_1 + \tilde{y}_2^\top \tilde{y}_2 + \tilde{y_1}^\top \tilde{y}_2)\big) - y\|^2_{\mathrm{F}}, \tag{15}$$

where y is the ground truth labels. By this way, $C_F(\cdot)$ would hopefully discover latent relations between different views and categories.

4 Experiments

4.1 Dataset

Three action recognition multi-view datasets are deployed for our evaluation. **Depth-included Human Action dataset (DHA)** [17] is an RGB-D multi-view dataset. It contains 23 kinds of actions performed by 21 subjects. We randomly choose 50% as the training set and the others as the testing set. We pick the RGB and depth views for the evaluation. **UWA3D Multiview Activity (UWA)** [24] is a multi-view action dataset collected by Kinect sensors. It contains 30 kinds of actions performed by 10 subjects. 50% samples are randomly extracted as the training set and the others are assigned to the testing set. Similarly, we choose the RGB videos and depth videos for the experiments. **Berkeley Multimodal Human Action Database (MHAD)** [22] is a comprehensive multimodal human action dataset which contains 11 kinds of actions. Each action is performed by 12 subjects with 5 repetitions. We choose RGB videos and depth videos for the evaluation. 50% samples are set as the training set and the others as the testing set.

4.2 Baselines

We test our approach under the scenario of multi-view (RGB-D) action recognition. We deploy some baseline methods for comparison. **Least Square Regression (LSR)** is a linear regression model. We concatenate the features from different views together as the input. LSR learns a linear mapping from the feature space to the label spaces. **Multi-layer Neural Network (NN)** is a classical multi-layer neural network. It is deployed as a classifier with the multi-view features concatenated as the input. **Support Vector Machine (SVM)** [25] attempts to explore the hyper-planes in the high-dimensional space. The features are concatenated from multiply views and the SVM module we used is implemented by LIBSVM [5]. **Action Vector of Local Aggregated Descriptor (VLAD)** [11] is an action representation method. It is able to capture local convolutional features and spatio-temporal relationship of the videos. **Auto-Weight Multiple Graph Learning (AMGL)** [20] is a graph-based multi-view classification method designed for semi-supervised learning. Optimal weights for each graph are automatically calculated. **Multi-view Learning with Adaptive Neighbours (MLAN)** [19] deploys adaptive graph to learn the local and global structure and the weight of each view. **Adaptive MUltiview SEmi-supervised model (AMUSE)** [21] is a semi-supervised model for image classification task. It learns the parameters from the graph to obtain the classification results. **Generative Multi-View Action Recognition(GMVAR)** [31] is a multi-view action recognition method. It augments samples guided by another view and applies a label-level fusion module to get the final classification.

Table 1. Baselines and Performance. Classification accuracy (%) on DHA, UWA, and MHAD datasets.

Setting	Method	DHA	UWA	MHAD
RGB	LSR	65.02	67.59	96.46
	SVM [25]	66.11	69.77	96.09
	VLAD [11]	67.85	71.54	97.17
	TSN [34]	67.85	71.01	97.31
Depth	LSR	82.30	45.45	47.63
	SVM [25]	78.92	34.92	45.39
	WDMM [1]	81.05	46.58	66.41
RGB+D	LSR	77.36	68.77	97.17
	NN	86.01	73.70	96.88
	SVM [25]	83.47	72.72	96.80
	AMGL [20]	74.89	68.53	94.70
	MLAN [19]	76.13	66.64	96.46
	AMUSE [21]	78.12	70.32	97.23
	GMVAR [31]	88.72	76.28	98.94
	Ours	**89.31**	**77.08**	**98.94**

Table 2. Ablation study. Classification accuracy (%) on DHA [17] dataset.

Setting	RGB	Depth	RGB+D
TSN [34]	67.85	-	-
WDMM [1]	-	81.05	-
MLP	77.10	79.01	79.12
Mixup	68.51	81.43	81.48
SeMix	69.37	**82.73**	83.15
VCA	75.26	80.86	81.32
VCA-entropy	**80.86**	82.61	84.10
Ours complete	-	-	**89.31**

4.3 Implementation

We extract initial action features from the original visual spaces first. We deploy Temporal Segment Networks (TSN) [34] for RGB view and Weighted Depth Motion Maps (WDMM) [1] for depth view respectively. The implementation and feature extraction details are introduced below.

Temporal Segment Networks (TSN) [34] divides the video into several segments. Then it randomly samples and classifies the frames in each segment. The fusion of all the classification results(*e.g.*mean) is the final result. We sample 5 frames for training and 3 frames for testing. Each video is described by a concatenation of features from 3 frames.

Weighted Depth Motion Maps (WDMM) [1] is designed for human gestures recognition in depth view. It proposed a new sampling method to do a linear aggregation for spatio-temporal information from three projections views.

For LSR, NN and SVM, the original RGB features and depth features are extracted by TSN and WDMM respectively. We apply a normalization to both features since directly concatenating features from different views is not reasonable due to the the view heterogeneity. Then we concatenate these two features as a single feature. We use LSR, NN and SVM to classify the single feature. For AMGL, MLAN, AMUSE and GMVAR, we consider RGB features extracted by TSN and depth features extracted by WDMM as two views for multi-view setting. We do not include extra preprocessing steps for fair comparison.

4.4 Performance

The experimental result is shown in Table 1. From the results, we observe that our approach outperforms other state-of-the-art methods. It illustrates the effectiveness of our approach. Our model achieves similar performance in MHAD

Table 3. Classification accuracy(%) given different ratios of labeled training samples.

Dataset	Ratio	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
DHA	GMVAR [31]	44.86	62.55	69.14	73.25	77.37	80.66	84.36	84.36	86.01	88.72
	Ours	48.15	69.14	74.90	79.42	83.54	83.56	85.60	86.83	87.24	89.31
UWA	GMVAR [31]	35.18	46.64	54.15	58.57	65.61	69.17	73.91	75.49	76.28	76.28
	Ours	36.36	57.71	62.85	64.03	67.98	70.75	73.12	76.56	76.66	77.08
MHAD	GMVAR [31]	53.36	72.79	90.11	92.64	93.41	94.76	95.91	95.49	96.28	98.94
	Ours	52.30	75.83	91.17	92.23	92.93	93.11	95.05	96.82	97.88	98.94

dataset compared with the best performance, we assume the classification accuracy in MHAD dataset is already high (*i.e.*, 98.94%) and is too challenging to obtain considerable improvements. For DHA and UWA datasets, our performance is slightly higher than GMVAR [31]. We assume the datasets are well-explored and hard to achieve great improvement. Meanwhile, Table 3 shows that our improvements are higher when less labeled samples($< 50\%$) are given, which reveals the potential and advantages of our model in semi-supervised setting.

4.5 Ablation Study

To prove the effectiveness of all the components in our approach, we intentionally remove some components and evaluate the performance based on partial of our model. We also evaluated the efficiency of our approach utilizing labeled samples. The effectiveness of the generated samples are also explored. In Table 2, the performance of *TSN* and *WDMM* are listed as baselines. The baseline model only has an encoder and a classifier for each view and only optimized by Eq. (8).

Mixup and SeMix. We conduct two experiments to prove the effectiveness of *SeMix* compared with *Mixup*. The first experiment is *Mixup* with labeled data. We apply the data augmentation method from [37] with Eq. (1) as the baseline model. The second experiment is *SeMix* with both labeled and unlabeled data. We apply Eq. (1) and Eq. (5) simultaneously to the baseline model. We concatenate the obtained representations of the two views and send it to a two-layer neural network to get the RGB+D multi-view performance. The results are shown in Table 2 and Figure 3 (a) and (b). In Table 2, we observe that both *Mixup* and *SeMix* achieve considerable improvements in single-view scenario. We can conclude that our framework does learned the extra structural knowledge from the other view's distribution and help the single view classification. From Figure 3 (a) and (b), we can see that *SeMix* achieves higher and more stable performance than the *Mixup* baselines. We assume *SeMix* fully utilizes the unlabeled information, which helps the classification and representation.

VCA and VCA-entropy. We conduct several experiments to prove the effectiveness of *VCA* and the entropy-based VCA, *VCA-entropy*. *VCA* and *VCA-entropy* aims to operate the Dual-level View-Correlation Adaptation and we evaluate the performance with and without the entropy algorithm. *MLP* is the baseline model. It has an encoder and a classifier for each view. A cross-view

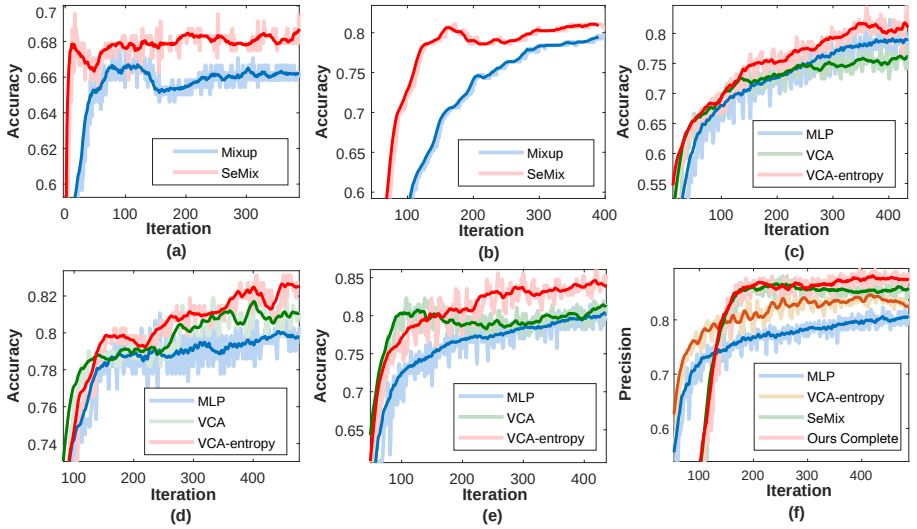

Fig. 3. Ablation study results. (a) and (b) are the performance of *Mixup* and *SeMix* in RGB view and depth view. Blue lines are the performance of *Mixup* and red lines are the performance of *SeMix*. (c), (d) and (e) are the performance of *MLP*, *VCA* and *VCA-entropy* in RGB view, depth view and after the cross-view fusion. Blue lines, green lines and red lines indicate *MLP*, *VCA* and *VCA-entropy* respectively. (f) is the performance of cross-view fusion. The whole framework is proposed to achieve a higher cross-view fusion. Blue line, yellow line, green line and red line indicate the *MLP*, *VCA-entropy*, *SeMix* and our whole framework respectively. The shadow lines are the exact performance, indicating the robustness and stability of the model.

fusion mechanism is implemented to obtain the final result. For *VCA*, we simply add the cross-view adaptation part(*i.e.* Eq. (9) and Eq. (10)) to *MLP*. For *VCA-entropy*, we change the loss functions Eq. (9) and (10) in *VCA* to Eq. (11) and (13). Our results are shown in Table 2 and Figure 3 (c), (d) and (e). They illustrate the classification performance of view1 (RGB), view2 (Depth), and the final fusion result respectively. For each graph, there are three different lines with different colors. The blue, green and red lines indicate the results of the *MLP*, *VCA* and *VCA-entropy* respectively. We observe that in most of the cases the *VCA* performs better than *MLP*, which demonstrates the effectiveness of the *VCA* component in our approach. While in Figure 3 (c), *VCA* performs lower than *MLP*. We assume that this is because the view (RGB) with the correct classification results is misled by the view (depth) with the wrong classification results. However, *VCA-entropy* always achieves the best the best performance compared with *VCA* and *MLP*. This proves that the entropy-based modification can effectively evaluate the confidence of each view and assign the view with the higher confidence to guide the training of the other view.

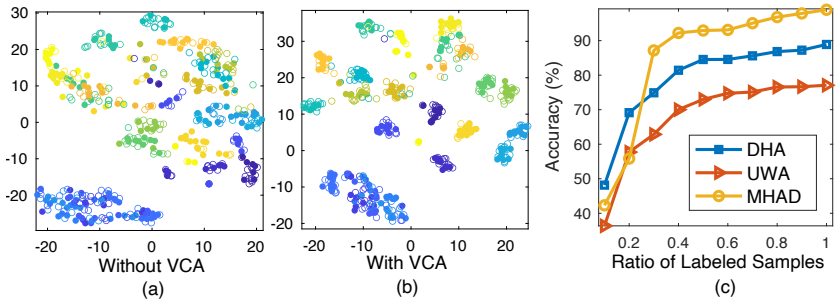

Fig. 4. (a) t-SNE Visualization of the feature extracted by TSN [34]. We use the pretrained ResNet-101 model and finetune it with raw DHA RGB features. (b) t-SNE Visualization of the output of encoder $E_1(\cdot)$. We train the whole framework for 800 epochs. (c) Performance when we use different amount of labeled data.

Complete Framework. We conduct experiments to prove that by combining these two modules, our methods can be further improved. Since our target is classification result based on two views, we show four final results of our models: 1) *MLP* with the cross-view fusion module, 2) *SeMix* with the cross-view fusion module, 3) *VCA-entropy* with cross-view fusion module, and 4) the complete model. Our results are shown in Table 2 and Figure 3 (f). Figure 3 (f) indicates that both *SeMix* or *VCA-entropy* are helpful to improve the final performance, and by adding both *SeMix* and *VCA-entropy*, the performance achieves the best. This means our framework is able to compromise the merits of both *SeMix* and *VCA-entropy* and further improve the classification performance.

Visualization. We utilize t-SNE [18] to visualize the distribution of the RGB features and its representations after the encoder. DHA dataset is deployed for this experiment. The RGB feature is extracted by TSN [34] model. The representation is the output of the encoder in our model after training 800 epochs. The solid circles are labeled data while hollow circles are unlabeled data. Different colors stand for different actions. Figure 4 (a) is the distribution without *VCA* while Figure 4 (b) is the distribution with *VCA*. We observe that the representations from the same class cluster better than the feature extracted by TSN. This can potentially help the model with the higher classification performance. The unlabeled data also clusters better after representation. This proves the effectiveness of our model. In summary, the results indicate the RGB encoder learns from the depth view.

Labeled Samples. We conduct the experiments to prove our model still works well even when fewer labeled samples are provided. We change the number of available labeled samples from 10% of the training samples to 100% of the training samples. The performance is shown in Figure 4 (c). The x-axis indicates the ratio of the labeled samples to the whole training set. The y-axis is the classification accuracy. From Figure 4 (c), we observe that the performance grows quickly in the beginning for all datasets. When the ratio is larger than 40%,

the growth trend becomes slow. This indicates our semi-supervised model can achieve a comparable result with only 50% labeled data are available. The results further prove that our model can efficiently utilize the labeled sample and practically avoid the extremely expensive data labeling procedure.

Table 4. Accuracy (%) of different generate number

Dataset	0x	0.1x	0.3x	0.5x	1x	2x	3x
DHA	84.10	87.24	88.07	88.07	89.31	88.89	89.31
UWA	75.49	76.30	77.08	76.68	77.08	77.47	76.28
MHAD	98.23	98.59	98.23	98.23	98.94	98.94	97.88

Generated Samples. We evaluate the helpfulness of the generated samples. In Table 4, different amount of the generated samples are utilized for training the model. The first row indicates the ratio of the generated samples to labeled samples. Specifically, 0x indicates there are only real samples. We observe that for DHA and UWA datasets, the performance under different ratios are higher than the results without the generative samples. The performance reaches the peak around 1x and 2x. More samples could not improve the performance and we assume it reaches the limitation of *SeMix*, and the slight performance decrease could be fluctuations. For MHAD dataset, since the 0x performance is relatively high, it is hard to significantly improve the performance. While, our model still slightly outperforms the 0x when the modal generates 1x and 2x samples.

5 Conclusion

We propose a novel Generative View-Correlation Adaptation framework for semi supervised multi-view learning. A new data augmentation mechanism, *SeMix*, is proposed which utilizes both labeled and unlabeled data to generate more diverse and robust auxiliary samples. In addition, a multi-view dual-level alignment strategy is designed. The classification results from both views are used to guide the training of the encoders and classifiers where the structural information from both feature and label space are effectively explored. Moreover, a simple yet effective view-correlation fusion network is applied which reveals the latent label relations and obtains the final result. Extensive experiments are conducted which demonstrate the effectiveness of our approach. More comprehensive ablation studies illustrate that all the modules in our approach are necessary and indispensable which considerably improve the final performance.

Acknowledgement

This research is supported by the U.S. Army Research Office Award W911NF-17-1-0367.

References

1. Azad, R., Asadi-Aghbolaghi, M., Kasaei, S., Escalera, S.: Dynamic 3D hand gesture recognition by learning weighted depth motion maps. IEEE Transactions on Circuits and Systems for Video Technology (2018)
2. Banica, D., Sminchisescu, C.: Second-order constrained parametric proposals and sequential search-based structured prediction for semantic segmentation in RGB-D images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3517–3526 (2015)
3. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.: Mixmatch: A holistic approach to semi-supervised learning. arXiv preprint arXiv:1905.02249 (2019)
4. Cai, Z., Wang, L., Peng, X., Qiao, Y.: Multi-view super vector for action recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 596–603 (2014)
5. Chang, C.C., Lin, C.J.: LIBSVM: a library for support vector machines. ACM Transactions on intelligent systems and technology **2**(3), 27 (2011)
6. Chapelle, O., Scholkopf, B., Zien, A.: Semi-supervised learning. IEEE Transactions on Neural Networks **20**(3), 542–542 (2009)
7. Cheng, Y., Zhao, X., Cai, R., Li, Z., Huang, K., Rui, Y., et al.: Semi-supervised multimodal deep learning for RGB-D object recognition (2016)
8. Cheng, Z., Qin, L., Ye, Y., Huang, Q., Tian, Q.: Human daily action analysis with multi-view and color-depth data. In: Proceedings of European Conference on Computer Vision. pp. 52–61. Springer (2012)
9. Ding, Z., Shao, M., Fu, Y.: Robust multi-view representation: A unified perspective from multi-view learning to domain adaption. In: Proceedings of the International Joint Conferences on Artificial Intelligence. pp. 5434–5440 (2018)
10. Du, D., Wang, L., Wang, H., Zhao, K., Wu, G.: Translate-to-recognize networks for RGB-D scene recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11836–11845 (2019)
11. Girdhar, R., Ramanan, D., Gupta, A., Sivic, J., Russell, B.: ActionVLAD: Learning spatio-temporal aggregation for action classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. vol. 2, p. 3 (2017)
12. Gupta, S., Hoffman, J., Malik, J.: Cross modal distillation for supervision transfer. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2827–2836 (2016)
13. Holte, M.B., Moeslund, T.B., Nikolaidis, N., Pitas, I.: 3D human action recognition for multi-view camera systems. In: Proc. Inte. Conf. on 3D Imaging, Modeling, Processing, Visualization and Transmission. pp. 342–349 (2011)
14. Ji, X., Wang, C., Li, Y.: A view-invariant action recognition based on multi-view space hidden markov models. International Journal of Humanoid Robotics **11**(01), 1450011 (2014)
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Proceedings of Advances in Neural Information Processing Systems. pp. 1097–1105 (2012)
16. Li, Y., Zhang, J., Cheng, Y., Huang, K., Tan, T.: DF2Net: Discriminative feature learning and fusion network for RGB-D indoor scene classification. In: Proceedings of AAAI Conference on Artificial Intelligence (2018)
17. Lin, Y.C., Hu, M.C., Cheng, W.H., Hsieh, Y.H., Chen, H.M.: Human action recognition and retrieval using sole depth information. In: Proceedings of the ACM International Conference on Multimedia. pp. 1053–1056 (2012)

18. Maaten, L.v.d., Hinton, G.: Visualizing data using t-SNE. Journal of machine learning research **9**, 2579–2605 (2008)
19. Nie, F., Cai, G., Li, X.: Multi-view clustering and semi-supervised classification with adaptive neighbours. In: Proceedings of AAAI Conference on Artificial Intelligence (2017)
20. Nie, F., Li, J., Li, X., et al.: Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. In: Proceedings of International Joint Conferences on Artificial Intelligence. pp. 1881–1887 (2016)
21. Nie, F., Tian, L., Wang, R., Li, X.: Multiview semi-supervised learning model for image classification. IEEE Transactions on Knowledge and Data Engineering (2019)
22. Ofli, F., Chaudhry, R., Kurillo, G., Vidal, R., Bajcsy, R.: Berkeley MHAD: A comprehensive multimodal human action database. In: IEEE Workshop on Applications of Computer Vision. pp. 53–60 (2013)
23. Pagliari, D., Pinto, L.: Calibration of Kinect for Xbox One and comparison between the two generations of microsoft sensors. Sensors **15**, 27569–27589 (10 2015)
24. Rahmani, H., Mahmood, A., Huynh, D., Mian, A.: Histogram of oriented principal components for cross-view action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(12), 2430–2443 (2016)
25. Scholkopf, B., Smola, A.J.: Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press (2001)
26. Verma, V., Lamb, A., Beckham, C., Courville, A., Mitliagkis, I., Bengio, Y.: Manifold Mixup: Encouraging meaningful on-manifold interpolation as a regularizer. stat **1050**, 13 (2018)
27. Wang, A., Cai, J., Lu, J., Cham, T.J.: Modality and component aware feature fusion for RGB-D scene classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5995–6004 (2016)
28. Wang, D., Ouyang, W., Li, W., Xu, D.: Dividing and aggregating network for multi-view action recognition. In: Proceedings of European Conference on Computer Vision (September 2018)
29. Wang, L., Ding, Z., Fu, Y.: Learning transferable subspace for human motion segmentation. In: Proceedings of the AAAI Conference on Artificial Intelligence (2018)
30. Wang, L., Ding, Z., Fu, Y.: Low-rank transfer human motion segmentation. IEEE Transactions on Image Processing **28**(2), 1023–1034 (2019)
31. Wang, L., Ding, Z., Tao, Z., Liu, Y., Fu, Y.: Generative multi-view human action recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6212–6221 (2019)
32. Wang, L., Liu, Y., Qin, C., Sun, G., Fu, Y.: Dual relation semi-supervised multi-label learning. In: Proceedings of the AAAI Conference on Artificial Intelligence (2020)
33. Wang, L., Sun, B., Robinson, J., Jing, T., Fu, Y.: EV-Action: Electromyography-vision multi-modal action dataset. In: Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (2020)
34. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: Proceedings of European Conference on Machine Learning. pp. 20–36 (2016)
35. Wang, W., Zhou, Z.H.: Analyzing co-training style algorithms. In: Proceedings of European Conference on machine learning. pp. 454–465. Springer (2007)

36. Yang, Y., Zhan, D.C., Sheng, X.R., Jiang, Y.: Semi-supervised multi-modal learning with incomplete modalities. In: Proceedings of International Joint Conferences on Artificial Intelligence. pp. 2998–3004 (2018)
37. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. In: Proceedings of International Conference on Learning Representations (2018)
38. Zhang, Z.: Microsoft Kinect sensor and its effect. IEEE Multimedia **19**(2), 4–10 (2012)