Multi-View Time Series Classification via Global-Local Correlative Channel-Aware Fusion Mechanism

Yue Bai¹, Lichen Wang¹, Zhiqiang Tao¹, Sheng Li², Yun Fu¹

¹Department of ECE, Northeastern University, ²Department of Computer Science, University of Georgia
bai.yue@husky.neu.edu, wang.lich@husky.neu.edu, zqtao@ece.neu.edu, sheng.li@uga.edu, yunfu@ece.neu.edu

Abstract
Multi-view time series classification aims to fuse the distinctive temporal information from different views to further enhance the classification performance. Existing methods mainly focus on fusing multi-view features at an early stage (e.g., learning a common representation shared by multiple views). However, these early fusion methods may not fully exploit the view-specific distinctive patterns in high-dimensional time series data. Moreover, the intra-view and inter-view label correlations, which are critical for multi-view classification, are usually ignored in previous works. In this paper, we propose a Global-Local Correlative Channel-Aware Fusion (GLCCF) model to address the aforementioned issues. Particularly, our model extracts global and local temporal patterns by a two-stream structure encoder, captures the intra-view and inter-view label correlations by constructing a graph based correlation matrix, and extracts the cross-view global patterns via a learnable channel-aware late fusion mechanism, which could be effectively implemented with a convolutional neural network. Extensive experiments on two real-world datasets demonstrate the superiority of our approach over the state-of-the-art methods. An ablation study is further provided to show the effectiveness of each model component.

Introduction
Time series classification has attracted increasing attention recently since temporal data contains more dynamic patterns which cannot be discovered easily. Many algorithms have been proposed for modeling time series data in different application domains, such as transportation (Yao et al. 2018) and healthcare (Harutyunyan et al. 2017). Nowadays, owing to the advanced sensing techniques, objects or events can be observed through multiple modalities, which results in tons of multi-view time series data containing complementary temporal information. For example, RGB, depth, and skeleton are three common modalities for video action recognition. They provide more comprehensive information to depict human actions than an individual view. For another example, several types of signals of human body can be monitored as different modalities in health-care applications, such as magnetic resonance imaging (MRI) and electrocardiograph (ECG). These multi-view signals record the same physical state simultaneously, and thus provide view-specific information to facilitate with each other.

Seeing the great potential of utilizing the complementary information from different views, multi-view learning has drawn significant attention in recent years and has been successfully applied in several application scenarios (Xu, Tao, and Xu 2013; Nie et al. 2016; Nie, Cai, and Li 2017). Existing algorithms could be roughly classified into three groups (Xu, Tao, and Xu 2013): 1) co-training, 2) multiple kernel learning, and 3) subspace learning. Generally, the co-training related methods integrate multi-view data by maximizing the common mutual information of different views, the multiple kernel learning methods design specific learning kernel for each view and then combine them together, and the subspace learning methods seek for the common latent subspace shared by multiple views. Although these methods have achieved promising results, it is not straightforward to directly employ them for modeling temporal data.

On another side, the single-view time series classification task is widely explored (Zheng et al. 2014; H¨usken and Stagge 2003) under two cases, i.e., univariate and multivariate. The univariate time series classification mainly studies the distance measurement between two sequential data such as (Marteau and Gibet 2014). Many research attempts have also been made in handling the multivariate time series. To name a few, (Bank ´o and Abonyi 2012) revises the dynamic temporal wrapping (DTW) method, and (Cui, Chen,
and Chen 2016) utilizes the convolutional neural networks (CNN). However, only a few methods have been proposed for solving multi-view and multivariate time series classification. (Li, Li, and Fu 2016) proposes a discriminative bilinear projection framework to build a shared latent subspace for multi-view temporal data. (Zadeh et al. 2018) designs a fusion strategy based on long short-term memory (LSTM) networks. (Yuan et al. 2018) proposes an attention mechanism for multi-view temporal data. It is worth noting that all these methods focus on the early fusion strategy, which may not fully exploit the view-specific distinctive patterns and ignore the intra-view and inter-view label correlations.

To solve the above issues, we propose a Global-Local Correlative Channel-aware Fusion (GLCCF) mechanism for the multi-view time series classification, which jointly leverages the view-specific distinctive global/local temporal patterns existing in feature space and the intra-view/inter-view correlations in label space (see Figure 1). Specifically, our model first applies a two-stream temporal encoder to extract global/local temporal features, followed by a classifier for each view. Thus, the raw label information is obtained. Then, the intra-view/inter-view label correlations are captured by a concise but effective graph based correlation matrix. Finally, a learnable fusion mechanism is designed to globally integrate the label correlations and tune the entire network. The main contributions of our work are summarized as follows:

- We propose a learnable late fusion mechanism for solving multi-view time series classification, which is underexplored by previous works.
- We develop an end-to-end network to jointly capture view-specific representation by global-local temporal encoder and fuse the cross-view correlated information by channel-aware fusion layer.
- We conduct extensive experiments on two datasets compared with state-of-the-art methods to show the effectiveness of our model, and provide a detailed ablation study to further demonstrate the indispensability of each component in our proposed model.

Related Work

Time Series Classification

Time series data, as a type of important sequence data, can be collected and applied in a wide range of domains (Xing, Pei, and Keogh 2010). Generally, the methods focusing on time series classification task can be roughly categorized into three groups: 1) feature based classification, 2) sequence distance based classification, 3) model based classification. Feature based algorithms such as (Kadous and Sammut 2005) (Ye and Keogh 2009) extract a feature vector from time series and then apply traditional classifiers such as support vector machine (SVM) (Cortes and Vapnik 1995) to make classification. Deep neural network has great ability to fit non-linear mapping and extract complicated temporal features for classification (Karim et al. 2019). Reservoir computing (Bianchi et al. 2018) is proposed based on recurrent neural network to learn the representation for multivariate time series classification. Distance based methods aim to design distance function to measure the similarity of a pair of sequences. As long as obtaining a reasonable distance metric, we apply conventional classifier such as SVM and K-Nearest neighbor (KNN) to further make the classification. For example, DTW (Xi et al. 2006) is a typical distance based algorithm which is still eligible for the different lengths of time series situation. Other distance based models also have been proposed for time series classification such as (Wei and Keogh 2006) (Ratanamahatana and Keogh 2004) (Keogh and Kasetty 2003). Model based methods assume that all sequences in one specific class are generated by a potential generative model. During the training stage, the corresponding parameters of potential model are learned and the test samples are classified based on the likelihood. To name a few, Hidden markov model (HMM) (Rabiner 1989) is widely used in time series classification for speech recognition application. Naïve bayes sequence classifier is another typical model based method which observes the feature independent assumption. In our work, we mainly focus on multi-view time series classification problem which is not fully explored by above methods.
Multi-View Learning

Multi-view learning has attracted more attention in recent decades. The distinct patterns extracted from each view can be regarded as mutual-support information to benefit multi-view learning performance. Multi-view learning has been widely used in many scenarios, such as object classification (Qi et al. 2016), clustering (Bickel and Scheffer 2004), semi-supervised learning (Hou et al. 2010), action recognition (Cai et al. 2014), face recognition (Li et al. 2002), etc. Fusing information from different views is always considered as an effective way in multi-view learning to combine distinctive patterns from each view for performance improvement (Swoger et al. 2007) (Bruno and Marchand-Maillet 2009). Fusion strategies can be roughly divided into three groups (Atrey et al. 2010): 1) Feature fusion, 2) Decision fusion, 3) Hybrid fusion. Feature fusion (early fusion) focuses on merge distinctive information from different views at the early stage to take the advantage of unusual information from each view. Decision fusion (late fusion) aims to fuse the decision for views at the late stage. Hybrid fusion is a combination of feature fusion and decision fusion.

However, most existing multi-view learning algorithms are not designed for temporal data and cannot be used for multi-view time series data directly. In our work, we propose a novel global-local correlation-aware channel-aware fusion mechanism to build an end-to-end deep framework for multi-view time series data classification task.

Methodology

Preliminary

Let $X = \{X_v\}_{v=1}^V$ be the multi-view time series training data, where $X_v \in \mathbb{R}^{N \times T \times D_v}$ refers to the $v$-th view feature matrix. For all $v$, we denote $N$, $T$, and $D_v$ as sample number, time steps, and feature dimension, respectively. Let $Y \in \mathbb{R}^{N \times K}$ be the label matrix corresponding to $X$, where $K > 0$ denotes the class number and $Y_i \in \{0, 1\}^K$, $\sum Y_i = 1$, is the one-hot label vector for the $i$-th sample. All the views in $X$ share with the sample label matrix $Y$.

In this study, we aim to train our model on the training set $\{X_{\text{train}}, Y_{\text{train}}\}$ by leveraging the complementary information of multiple views through an end-to-end learnable late-fusion way and eventually predict the class labels for all the samples in $X_{\text{test}}$. Note that, we refer to $\{X, Y\}$ as the training set by default. Our proposed framework, GLCCF, can be divided into two main parts, global-local temporal encoder and correlation-aware fusion mechanism which will be introduced concretely in the rest of this section.

Global-Local Temporal Encoder

Temporal information is the key factor to characterize the time-series data. It usually provides discriminative feature representations to classifier, and thus obtains high-quality label information for the fusing process. In our model, we propose a global-local temporal encoder to fully capture the temporal context, consisting of a global-temporal encoder $E_g$ and a local-temporal encoder $E_l$. Specifically, we encode each individual view’s feature by:

$$H^v = q(H^v_1, H^v_2)$$

$$H^v_g = E_g(X^v, \phi^v_g)$$

$$H^v_l = E_l(X^v, \phi^v_l),$$

where $H^v \in \mathbb{R}^{N \times d^v}$ is the encoded representations for $X^v$, $H^v_g/H^v_l$ represents the output given by $E_g/E_l$. $q(\cdot, \cdot)$ denotes a common fusion operation like average or concatenation, and $E_g/E_l$ are two networks parameterized by $\phi^v_g/\phi^v_l$, respectively. Generally, we could learn $\phi^v_g, \phi^v_l$ by minimizing the following loss:

$$L^v = \sum_{i=1}^N \ell(Y_i, \hat{Y}_i^v),$$

where $\ell(\cdot, \cdot)$ represents a specific loss function (e.g., $\ell_2$ or cross-entropy), $\hat{Y}_i^v = C_v(H^v_i)$ is the prediction for the $i$-th sample given by $H^v_i$, and $C_v(\cdot) : \mathbb{R}^{d^v} \rightarrow \mathbb{R}^K$ is the v-th view’s classifier usually parameterized by a linear mapping.

In the next, we will introduce more details for each view’s encoders, i.e., $E_g(X^v; \phi^v_g)$ and $E_l(X^v; \phi^v_l)$. For convenience, we may omit the subscript $v$, and refer to $h, x$ as $\forall i$-th sample in $H, X$, when no confusion occurs.

Global-Temporal Encoder

We adopt Recurrent neural networks (RNNs) to parameterize our global-temporal encoder $E_g(\cdot; \phi_g)$ for each view, as RNNs have been well validated as an effective way to capture the long-term temporal context for time-series data. Particularly, we employ the long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997) as the RNN cell, which is given by:

$$f_t = \sigma_g(W_f x_t + U_f h_{t-1} + b_f),$$

$$i_t = \sigma_g(W_i x_t + U_i h_{t-1} + b_i),$$

$$o_t = \sigma_g(W_o x_t + U_o h_{t-1} + b_o),$$

$$c_t = f_t \circ c_{t-1} + i_t \circ \sigma_c(W_c h_t + U_c h_{t-1} + b_c),$$

$$h_t = o_t \circ \sigma_h(c_t),$$

where $f_t, i_t, o_t, c_t$, and $h_t$ represent forget gate, input gate, output gate, cell state and hidden state, at the current $t$-th time step, respectively. $1 \leq t \leq T$, $c_{t-1}$ and $h_{t-1}$ are cell and hidden states at the last time step, $\sigma_g, \sigma_c, \sigma_h$ are activation functions, and $\circ$ represents the element-wise product. In Eq (3), $W_s, U_s$ and $b_s$ are all learnable weights, $\forall s \in \{f, i, o, c\}$.

To further model the global temporal information, we leverage attention mechanism to integrate the hidden states of all the time steps. By using attention, we explicitly learn the dynamic correlation across different time steps, and obtain the global temporal representation $H_g$ by:

$$H_g = \sum_{i=1}^T \omega_i h_i,$$

where $\omega = \{\omega_i\}$ are the learnable attention weights.

By using Eqs. (3), (4), we eventually formulate our $E_g(\cdot; \phi_g)$ as LSTM with attention mechanism, and have $\phi_g = \{W_s, U_s, b_s, \omega\}, \forall s \in \{f, i, o, c\}$. 
Local-Temporal Encoder Different from the global temporal encoder, we utilize convolutional neural networks (CNNs) to formulate our local-temporal encoder $E_l(\cdot; \phi_l)$, as CNNs probe patterns from local-characterized data. Specifically, we apply a set of 1D convolutional filters to extract local patterns in $X$ as following the temporal convolutional networks (TCN) (Lea et al. 2016). Let $M$ be the number of CNN layers and $F_m \in \mathbb{R}^{N \times D_m \times D_m}$ be the output of the $m$-th layer, where $1 \leq m \leq M$, $T_m$ and $D_m$ denotes the corresponding time steps and output dimension, respectively. Given $F_0 = X$, we compute $F_m$ by

$$F_m = \text{BN}_{\{\gamma_m, \beta_m\}}(\text{ReLU}(W_m \ast F_{m-1} + b_m)), \quad (5)$$

where $W_m \in \mathbb{R}^{D_m \times D_m \times \Delta T}$ is the weight of convolutional filter, $b_m \in \mathbb{R}^{D_m}$ is the bias, $\Delta T$ represents the size of temporal sliding window, $\ast$ denotes the convolution operation. In Eq. (5), $\text{BN}_{\{\gamma_m, \beta_m\}}(\cdot)$ refers to the batch normalization block (Ioffe and Szegedy 2015) with learnable parameters $\gamma_m$ and $\beta_m$, which is used to further improve the effectiveness and stability of $E_l$.

In order to reduce the number of parameters and avoid over-fitting issue, a global average pooling layer (Lin, Chen, and Yan 2013) is added after each convolutional block. By using these techniques, we efficiently extract local temporal information and obtain high-level representation $H_l$ by

$$H_l = g(F_M), \quad (6)$$

where $g(\cdot)$ is the global average pooling layer.

Through applying Eqs. (5-6), we concretize our $E_l(\cdot; \phi_l)$ as CNN with batch normalization and global average pooling, and have $\phi_l = \{W_m, b_m, \gamma_m, \beta_m\}_{m=1}^{M}$.

Correlative Channel-Aware Late Fusion

Efficiently fusing mutual-support information from the predicted label of each view $H^v$ is the central fact of late fusion mechanism. It takes advantage of intra-view and inter-view label correlations to achieve higher performance for multi-view learning. In our model, we propose a correlative learnable fusion mechanism to sufficiently capture and fuse the label correlation information. It constructs a graph based correlation matrix to probe intra-view and inter-view label correlations and a CNN based fusion module to integrate global patterns. Then, based on the $i$-th predicted label $\hat{Y}^v_i$ of each view from $H^v$, we introduce our correlative learnable fusion strategy. For convenience, we introduce our fusion model based on Vi-th sample and omit the subscript $i$.

Label Correlation Matrix We adopt a concise but effective strategy to capture the intra-view and inter-view label correlations separately. The intra-view label correlation matrix for each view $v$ is introduce by

$$G^{v,v} = \hat{Y}^v \cdot \hat{Y}^{v\top}, \quad (7)$$

where $G^{v,v} \in \mathbb{R}^{K \times K}$ is the correlation matrix derived by multiplying the predicted label $\hat{Y}^v \in \mathbb{R}^{K \times 1}$ and its transpose $\hat{Y}^{v\top} \in \mathbb{R}^{1 \times K}$ for $1 \leq v \leq V$. Each element of $G^{v,v}$ represents the relationship of corresponding two predicted labels from view $v$. $r^{\text{intra}}$ is obtained by concatenating $V$ intra-view matrices together introduced by

$$r^{\text{intra}} = [G^{1,1}, G^{2,2}, \ldots, G^{V,V}], \quad (8)$$

where $r^{\text{intra}} \in \mathbb{R}^{K \times K \times V}$.

In a similar way, the inter-view label correlation matrix for each pair of views is introduced by

$$G^{u,w} = \hat{Y}^u \cdot \hat{Y}^{w\top}, \quad (9)$$

where $G^{u,w} \in \mathbb{R}^{K \times K}$ is the correlation matrix derived by multiplying the predicted label $\hat{Y}^v \in \mathbb{R}^{K \times 1}$ from view $u$ and the transpose of predicted label $\hat{Y}^w \in \mathbb{R}^{1 \times K}$ from view $w$ for $\forall u, w \in V, u \neq w$. Each element of $G^{u,w}$ represents the inter-view relationship of corresponding two predicted labels from view $u$ and view $w$. Considering all the possible combinations of pair-view, $r^{\text{inter}}$ is obtained by concatenating $(V^2)$ inter-view matrices together introduced by

$$r^{\text{inter}} = [G^{1,2}, G^{1,3}, \ldots, G^{V-1,V}], \quad (10)$$

where $r^{\text{inter}} \in \mathbb{R}^{K \times K \times (V^2)}$.

By using Eqs. (7-8) and Eqs. (9-10), we formalize the stacked intra-view and inter-view correlation matrices as multi-channel tensors.

Channel-Aware Global Late Fusion Mechanism Multi-view label correlation information is extracted and represented by label correlation matrices. The informative patterns of label correlations are reserved in each element instead of a local area of matrix, but still contained in the same place across different channels. Hence, we employ CNN structure with $1 \times 1$ kernels as “pixel-wise” channel-aware pattern extractor to integrate cross-view correlative information which can be given by

$$r = E_f([r^{\text{intra}}, r^{\text{inter}}], \phi_f), \quad (11)$$

where $r \in \mathbb{R}^{K \times K \times N_r}$ is the fusion matrix, $r^{\text{intra}}$/$r^{\text{inter}}$ denotes the stacked correlation matrix of intra-view/inter-view, and $E_f(\cdot, \phi_f)$ is the CNN based fusion encoder parameterized by $\phi_f$, with $N_r$ being the number of kernels in $E_f$. We parameterize the fusion encoder $E_f(\cdot, \phi_f)$ by

$$r^{(o)}_{p,q} = f(b^{(o)} + \langle W^{(o)}, [r^{\text{intra}}, r^{\text{inter}}]_{p,q} \rangle), \quad (12)$$

where $r^{(o)}_{p,q}$ is the $(p, q)$ element of $r^{(o)} \in \mathbb{R}^{K \times K \times 1}$ which is the $o$-th component of $r$ for $1 \leq o \leq N_k$. $W^{(o)} \in \mathbb{R}^{1 \times 1 \times (V^2)}$ and $b^{(o)} \in \mathbb{R}^{1 \times 1 \times (V^2)}$ are the parameterized weights and bias of $1 \times 1$ filters, $r^{\text{intra}}, r^{\text{inter}}_{p,q}$ represents the $(p, q)$ element of cross-view correlation tensor concatenated by $r^{\text{intra}}$ and $r^{\text{inter}}$. $f(\cdot)$ is the activation function.

By using Eqs. (11-12), we formulate our fusion encoder $E_f(\cdot, \phi_f)$, and eventually have $\phi_f = \{W, b\}$. We could update $\phi_f$ by minimizing the following loss:

$$L_f = \sum_{i=1}^{N} \ell(Y_i, \hat{Y}^f_i), \quad (13)$$

where $\ell(Y_i, \hat{Y}^f_i)$ is the pairwise prediction error.
Algorithm 1: The procedure of training GLCCF algorithm.

**Input:** batches of \( \{X, Y\} \), number of view \( V \), number of training steps \( S \).

**Output:** prediction result of each view \( \hat{Y}^v \) and final fusion result \( \hat{Y}^f \).

1: for each \( i \in [1, S] \) do
2: for each \( v \in [1, V] \) do
3: sample a batch data \( X^v \) from view \( v \);
4: forward \( X^v \) into \( E_g(\cdot, \phi_g^v) \) and \( E_l(\cdot, \phi_l^v) \);
5: compute \( H^v \) and \( Y^v \) through \( C_v() \) and Eq. (1);
6: update \( \phi_g^v, \phi_l^v \) and \( C_v \) using Eq. (2);
7: end for
8: forward \( \hat{Y}^v, v \in 1, 2, ..., V \) into \( E_f(\cdot, \phi_f) \);
9: compute \( \hat{Y}^f \) through \( C^f \) and Eq. (11);
10: update \( \phi_f \) and \( C^f \) using Eq. (13);
11: end for
12: return \( \hat{Y}^v \) and \( \hat{Y}^f \).

where \( l(\cdot, \cdot) \) denotes the same loss function in Eq. (2). \( \hat{Y}^f = C^f(T_{\text{flatten}}(r_i)) \) is the prediction for the \( i \)-th sample given by \( r_i \) which is the corresponding fusion matrix \( r \) for \( i \)-th sample. \( T_{\text{flatten}} \) is a flatten operation to transfer feature matrix \( r_i \) into a feature vector, and \( C_f(\cdot) : \mathbb{R}^{D_f} \rightarrow \mathbb{R}^K \) is the final classifier usually parameterized by a linear mapping with \( D_f = K \times K \times N_b \). During the training process, we alternatively optimize the set of loss \( L^v \) for each view and \( L^f \) for final classifier. The entire procedure of training our proposed GLCCF algorithm is summarized in Algorithm 1.

**Experiments**

**Experimental Setting**

**Datasets** EV-Action dataset (Wang et al. 2019) is a large-scale multi-view human action dataset. It contains RGB, depth, skeleton and EMG views. We choose the first three views to set our multi-view time series experiments. EV-Action contains 20 human common actions including 10 actions finished by single subject like walking, sitting, jumping, etc, and the other 10 actions finished by the same subjects interacting with other objects like moving table, drinking, reading book, etc. It includes 53 subjects (the first 53 out of 70 subjects in original EV-Action dataset) performing each action 5 times so that we have 5300 samples in total and each subject performs 100 action clips for 20 classes. We choose the action clips collected from first 40 subjects as training set and the rest 13 subjects as test set.

UCI daily sport dataset (Asuncion and Newman 2007) (Altun, Barshan, and Tuncel 2010) is a multivariate time series dataset which contains motion sensor data of 19 human daily and sports actions including walking, running, standing, sitting, etc. There are 45 sensors placed on subject’s body in 5 different units: torso, right arm, right leg, left arm, left leg. Each unit has 9 sensors on it with 25 Hz sampling frequency to collect the time series signal. Each class of activity is performed by 8 subjects for 5 minutes. Each 5 minutes time series signal is divided into several 5 second segments. Each activity has 480 samples and the feature dimension and the number of time step for each sample are 45 and 125 respectively. We design a multi-view experimental setting which follows the same setting from (Li, Li, and Fu 2016) on the UCI dataset (Asuncion and Newman 2007). We manually split the whole feature space into two views where the first view represents the upper part of human body and the second view represents the lower part of the body. The View1 contains 27 sensors put on torso, right arm and left arm while the View2 contains 18 sensors put on right leg and left leg. We follow the experimental setting from (Li, Li, and Fu 2016) to randomly choose 10 out of 480 samples of each activity as training set so that there are 190 samples for training and 8930 samples for test.

**Baseline Methods** We test our model in multi-view time series scenarios. Several baseline methods including the state-of-the-art frameworks are deployed to demonstrate the effectiveness of our proposed model. Comparison baselines are introduced as below. MLSTM-FCN (Karim et al. 2019) is novel deep framework proposed to handle multivariate time series data which contains a two pathways structure to encode temporal data. RC framework (Bianchi et al. 2018) proposes an RC approach to encode the time series data as vectorial representations in an unsupervised fashion. MFN (Zadeh et al. 2018) designs a memory fusion mechanism to tackle with multi-view temporal data. We fuse multi-view temporal data by concatenating them together as input of LSTM and CNN model for early fusion comparison referring to Concat-LSTM and Concat-CNN. We employ three different late fusion strategies to further prove the effectiveness of our proposed late fusion module by concatenating, computing the average value and choose the largest label score from different views referring to Label-Concat, Label-Average and Label-Max.

For all the baseline methods, we use exactly the same experimental settings to split datasets and evaluate the performance as our model. For the MLSTM-FCN and RC framework baseline methods, we concatenate time series data from different views together as a multivariate time series input to adopt them for multi-view time series scenario. Since MFN is designed for multi-view learning, we directly input data from different views to evaluate the model. All the state-of-the-art baseline methods cannot provide single-view results and multi-view results simultaneously so that we report the performances separately. For the Concat-LSTM and
Concats-CNN, they cannot provide single-view performance. For the other three self-designed baseline methods Label-Concat, Label-Average and Label-Max which contain view-specific classifier for each view and final classifier for cross-view fusion so that we can report the single-view and multi-view performances simultaneously.

**Implementation** We arrange the EV-Action dataset and extract the frame-level feature for RGB, depth, skeleton views respectively to set the multi-view time series scenario. We align all the action clip of three views into 60 frames using cutting and repeating strategies for longer and shorter clips. We use TSN (Wang et al. 2016) to extract frame-level feature for RGB view with pre-trained BNInception network as backbone. Each RGB action clip is extracted as a 60x1024 feature matrix where 60 is the time steps and 1024 is the frame-level feature dimension. The depth view is transferred into RGB form firstly using HHA algorithm (Gupta et al. 2014) and extracted feature by exactly the same TSN framework used for RGB view. Each depth action clip is also extracted as 60x1024 feature matrix. For skeleton view, since skeleton data contains accurate position information of each joint on human body, we simply concatenate 3D coordinates of 25 joints and obtain 75 dimensional frame-level feature. As a summary, RGB, depth and skeleton views are represented as multi-view time series data with 60 time steps and 1024, 75 feature dimensions for them separately in EV-Action dataset. UCI Daily and Sports Activity data is collected by sensors placed on human body and we follow the self-designed multi-view experimental setting in ?Dataset section? to set the multi-view scenario directly.

For EV-Action dataset, the results are shown in Table 1, RGB, Depth and Skeleton represent the classification accuracy of each single-view respectively, while the Three-view indicates the fused multi-view classification results. The skeleton view can be regarded as the most informative view which always achieves the best performance using different methods on single-view while the RGB and depth obtain lower performance. The baseline methods can obtain comparable even better performance on single-view, however, our proposed model achieves the best performance for multi-view scenario and outperforms each single-view result. MFN cannot make early fusion efficiently to improve multi-view performance on EV-Action dataset which indicates the early fusion of MFN is not capable of handling high dimensional temporal data with a large difference between the dimension of features for different views. However, our proposed method will not suffer from this issue since we focus on extracting label correlations for multi-view fusion strategy. RC classifier and MLSTM-FCN achieve competitive results on skeleton view, however they cannot fully fuse multi-view information for better performance while our fusion strategy still obtain better results. Three simple label prediction through optimizing $L_f$. We set the batch size to 128. The Adam optimizer (Kingma and Ba 2014) is applied for optimization and the learning rates are set 0.0001 for all the view-specific classifier synchronously and global late fusion. During the training process, the classifiers of all different views $C_v$ are trained firstly to obtain the initial classification result of each view which makes a concrete foundation for the late fusion learning. Then the final classifier $C_f$ is trained based on the initial predicted labels. The set of $C_v$ and $C_f$ are trained alternatively during the whole training process and we report the performance of single-view and cross-view fusion simultaneously. Our model is implemented using Tensorflow with GPU acceleration.

**Performance Analysis**

For EV-Action dataset, the results are shown in Table 1. RGB, Depth and Skeleton represent the classification accuracy of each single-view respectively, while the Three-view indicates the fused multi-view classification results. The skeleton view can be regarded as the most informative view which always achieves the best performance using different methods on single-view while the RGB and depth obtain lower performance. The baseline methods can obtain comparable even better performance on single-view, however, our proposed model achieves the best performance for multi-view scenario and outperforms each single-view result. MFN cannot make early fusion efficiently to improve multi-view performance on EV-Action dataset which indicates the early fusion of MFN is not capable of handling high dimensional temporal data with a large difference between the dimension of features for different views. However, our proposed method will not suffer from this issue since we focus on extracting label correlations for multi-view fusion strategy. RC classifier and MLSTM-FCN achieve competitive results on skeleton view, however they cannot fully fuse multi-view information for better performance while our fusion strategy still obtain better results. Three simple
late fusion strategies are implemented to prove our learnable late fusion mechanism is more effective than simple fusion strategies. The comparisons between different late fusion strategies for EV-Action dataset are shown in Figure 3 which illustrates the performance variations along with batch steps.

For UCI dataset, the results are shown in Table 2. View1 and View2 represent the two single-view and Two-view indicates the fused multi-view. View2 always obtains better results for single-view compared with View1. The baseline methods can achieve competitive results for single-view but cannot outperform our fusion strategy. MFN improves the multi-view performance compared with single-view, however, it is still lower than our model which denotes the temporal feature encoder of MFN is not effective enough to provide foundation for its early fusion strategy. MLSTM-FCN obtains high performance for both single-view and multi-view, however, it cannot utilize multi-view data sufficiently for further improvement. Our proposed model achieves the best multi-view performance including comparing with three simple late fusion strategies.

Ablation Study

We further design detailed ablation study to prove the necessity of each component in our model. First, we use global-temporal encoder and local-temporal encoder individually to extract feature vector for view-specific classification on two datasets as shown in Table 3. For EV-Action dataset, global encoder always achieves higher at least comparable performance than local encoder for each view. However, for UCI dataset, the local encoder outperforms global encoder for both two views. Our two-stream temporal encoder takes advantage of these two encoders to capture global and local temporal patterns simultaneously which indicates it is indispensable to handle diverse time series data.

We divide our fusion module into several parts and make the ablation studies to prove the effectiveness for each of them. First, the whole late fusion module can be separated as two parts, label correlative matrix and CNN global fusion. Further, the label correlative matrix can be divided into intra-view and inter-view parts. The experimental results of ablation study are shown in Table 4. Only intra-view represents we only use intra-view matrices to demonstrate the necessity of inter-view matrix in our model. Only inter-view indicates the similar ablation strategy of inter-view and intra-view respectively. CNN Fusion Only means we remove all the correlative matrices and concatenate predicted label vectors together as input to CNN fusion network which proves the necessity of our whole correlative matrices. Ours without CNN indicates that instead of employing a CNN fusion module, we directly flatten all the correlation matrices into one feature vector as input to final classifier which demonstrates our late fusion is effective to capture global patterns. The results illustrate that a part of the integrated model cannot obtain the highest performance while the complete late fusion module achieves the ideal accuracy. The performance curves of ablation study is shown in Figure 4 which presents the performance variations along with batch steps.

Conclusions

We propose a novel end-to-end framework for multi-view time series classification task in this paper called Global-Local Correlative Channel-Aware Fusion mechanism called GLCCF. A global-local temporal encoder is applied to extract informative global and local temporal feature for view-specific classifier. By this way, the distinctive feature from each view is fully and independently utilized to lay a foundation for late fusion. Further, we propose a novel learnable late fusion mechanism to fuse the multi-view label information in terms of the intra-view and inter-view label correlations. Extensive evaluation results of two multi-view time series datasets demonstrate that our model is an effective end-to-end framework for multi-view time series classification. Detailed ablation study further demonstrates that our global-local temporal encoder is necessary in order to extract view-specific distinctive patterns for diverse time series data; also, each component of our learnable late fusion module is indispensable for fully utilizing multi-view label correlations. All the experimental results illustrate our proposed model is an effective end-to-end framework for a wide range of multi-view time series classification tasks.
References

[Altun, Barshan, and Tuncel 2010] Altun, K.; Barshan, B.; and Tuncel, O. 2010. Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recognition 43(10):3605–3620.

[Asuncion and Newman 2007] Asuncion, A., and Newman, D. 2007. UCI machine learning repository.

[Atrey et al. 2010] Atrey, P. K.; Hossain, M. A.; El Saddik, A.; and El Saddik, A. 2010. Modulation fusion for multimedia analysis: a survey. Multimedia systems 16(6):345–379.

[Bankó and Abonyi 2012] Bankó, Z., and Abonyi, J. 2012. Correlation based dynamic time warping of multivariate time series. Expert Systems with Applications 39(17):12814–12823.

[Bianchi et al. 2018] Bianchi, F. M.; Scardapane, S.; Løkse, S.; and Sørensen, M. 2018. Reservoir computing approaches for representation and classification of multivariate time series. arXiv preprint arXiv:1803.07870.

[Bickel and Scheffer 2004] Bickel, S., and Scheffer, T. 2004. Multi-view clustering. In Proc. ICDM, volume 4, 19–26.

[Bruno and Marchand-Maillet 2009] Bruno, E., and Marchand-Maillet, S. 2009. Multi-view clustering: a late fusion approach using latent models. In Proc. SIGIR, 736–737. ACM.

[Cai et al. 2014] Cai, Z.; Wang, L.; Peng, X.; and Qiao, Y. 2014. Multi-view super vector for action recognition. In Proc. CVPR, 596–603.

[Cortes and Vapnik 1995] Cortes, C., and Vapnik, V. 1995. Support-vector networks. Machine learning 20(3):273–297.

[Cui, Chen, and Chen 2016] Cui, Z.; Chen, W.; and Chen, Y. 2016. Multi-scale convolutional neural networks for time series classification. arXiv preprint arXiv:1603.06995.

[Gupta et al. 2014] Gupta, S.; Girshick, R.; Arbeláez, P.; and Malik, J. 2014. Learning rich features from rgb-d images for object detection and segmentation. In Proc. ECCV, 345–360. Springer.

[Hartutunyan et al. 2017] Hartutunyan, H.; Khachatryan, H.; Kale, C. D.; Steeg, G. V.; and Galstyan, A. 2017. Multitask learning and benchmarking with clinical time series data. arXiv preprint arXiv:1703.07771.

[Hochreiter and Schmidhuber 1997] Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. Neural computation 9(8):1735–1780.

[Hou et al. 2010] Hou, C.; Zhang, C.; Wu, Y.; and Nie, F. 2010. Multiple view semi-supervised dimensionality reduction. Pattern Recognition 43(3):720–730.

[Hüsken and Stagge 2003] Hüskens, M., and Stagge, P. 2003. Recurrent neural networks for time series classification. Neurocomputing 50:223–235.

[Ioffe and Szegedy 2015] Ioffe, S., and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

[Kadous and Sammut 2005] Kadous, M. W., and Sammut, C. 2005. Classification of multivariate time series and structured data using constructive induction. Machine learning 58(2):179–216.

[Karim et al. 2019] Karim, F.; Majumdar, S.; Darabi, H.; and Harford, S. 2019. Multivariate Istm-fcns for time series classification. Neural Networks 116:237–245.

[Keogh and Kaseti 2003] Keogh, E., and Kaseti, S. 2003. On the need for time series data mining benchmarks: a survey and empirical demonstration. Data Mining and Knowledge Discovery 7(4):349–371.

[Kingma and Ba 2014] Kingma, D. P., and Ba, J. 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[Lea et al. 2016] Lea, C.; Vidal, R.; Reiter, A.; and Hager, G. D. 2016. Temporal convolutional networks: A unified approach to action segmentation. In Proc. ECCV, 47–54.

[Li et al. 2002] Li, S. Z.; Zhu, L.; Zhang, Z.; Blake, A.; Zhang, H.; and Shum, H. 2002. Statistical learning of multi-view face detection. In Proc. ECCV, 67–81. Springer.

[Li, Li, and Fu 2016] Li, S.; Li, Y.; and Fu, Y. 2016. Multi-view time series classification: A discriminative bilinear projection approach. In Proc. International on Conference on Information and Knowledge Management, 989–998.

[Lin, Chen, and Yan 2013] Lin, M.; Chen, Q.; and Yan, S. 2013. Network in network. arXiv preprint arXiv:1312.4400.

[Marteaume and Gibet 2014] Marteaume, P.-F., and Gibet, S. 2014. On recursive edit distance kernels with application to time series classification. IEEE TNNLS 26(6):1121–1133.

[Nie et al. 2016] Nie, F.; Li, J.; Li, X.; et al. 2016. Parameter-free auto-weighted multiple graph learning: A framework for multi-view clustering and semi-supervised classification. In Proc. IJCAI, 1881–1887.

[Nie, Cai, and Li 2017] Nie, F.; Cai, G.; and Li, X. 2017. Multiview clustering and semi-supervised classification with adaptive neighbours. In Proc. AAAI.

[Qi et al. 2016] Qi, C. R.; Su, H.; Nießner, M.; Dai, A.; Yan, M.; and Guibas, L. J. 2016. Volumetric and multi-view cnns for object classification on 3d data. In Proc. CVPR, 5648–5656.

[Rabiner 1989] Rabiner, L. R. 1989. A tutorial on hidden markov models and selected applications in speech recognition. Proc. IEEE 77(2):257–286.

[Ratanamahatana and Keogh 2004] Ratanamahatana, C. A., and Keogh, E. 2004. Making time-series classification more accurate using learned constraints. In Proc. ICDM, 11–22.

[Swoger et al. 2007] Swoger, J.; Verveer, P.; Greger, K.; Huisken, J.; and Stelzer, E. H. 2007. Multi-view image fusion improves resolution in three-dimensional microscopy. Optics express 15(13):8029–8042.

[Wang et al. 2016] Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Van Gool, L. 2016. Temporal segment networks: Towards good practices for deep action recognition. In Proc. ECCV, 26–36. Springer.

[Wang et al. 2019] Wang, L.; Sun, B.; Robinson, J.; Jing, T.; and Fu, Y. 2019. EV-Action: Electromyography-vision multi-modal action dataset. arXiv preprint arXiv:1904.12602.

[Wei and Keogh 2006] Wei, L., and Keogh, E. 2006. Semi-supervised time series classification. In Proc. ACM SIGKDD, 748–753.

[Xi et al. 2006] Xi, X.; Keogh, E.; Shelton, C.; Wei, L.; and Ratanamahatana, C. A. 2006. Fast time series classification using numerosity reduction. In Proc. ICMIL, 1033–1040. ACM.

[Xing, Pei, and Keogh 2010] Xing, Z.; Pei, J.; and Keogh, E. 2010. A brief survey on sequence classification. ACM SIGKDD Explorations Newsletter 12(1):40–48.

[Xu, Tao, and Xu 2013] Xu, C.; Tao, D.; and Xu, C. 2013. A survey on multi-view learning. arXiv preprint arXiv:1304.5634.

[Yao et al. 2018] Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; and Li, Z. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In Thirty-Second AAAI Conference on Artificial Intelligence.
[Ye and Keogh 2009] Ye, L., and Keogh, E. 2009. Time series shapelets: a new primitive for data mining. In Proc. SIGKDD, 947–956.

[Yuan et al. 2018] Yuan, Y.; Xun, G.; Ma, F.; Wang, Y.; Du, N.; Jia, K.; Su, L.; and Zhang, A. 2018. Muvan: A multi-view attention network for multivariate temporal data. In 2018 IEEE International Conference on Data Mining (ICDM), 717–726. IEEE.

[Zadeh et al. 2018] Zadeh, A.; Liang, P. P.; Mazumder, N.; Poria, S.; Cambria, E.; and Morency, L.-P. 2018. Memory fusion network for multi-view sequential learning. In Proc. AAAI.

[Zheng et al. 2014] Zheng, Y.; Liu, Q.; Chen, E.; Ge, Y.; and Zhao, J. L. 2014. Time series classification using multi-channels deep convolutional neural networks. In Proc. International Conference on Web-Age Information Management, 298–310. Springer.