Improving Video Temporal Consistency via Broad Learning System

Bin Sheng, *Member, IEEE*, Ping Li, *Member, IEEE*, Riaz Ali, and C. L. Philip Chen, *Fellow, IEEE*

***Abstract*—Applying image-based processing methods to original videos on a framewise level breaks the temporal consistency between consecutive frames. Traditional video temporal consistency methods reconstruct an original frame containing flickers from corresponding nonflickering frames, but the inaccurate correspondence realized by optical flow restricts their practical use. In this article, we propose a temporally broad learning system (TBLS), an approach that enforces temporal consistency between frames. We establish the TBLS as a flat network comprising the input data, consisting of an original frame in an original video, a corresponding frame in the temporally inconsistent video on which the image-based technique was applied, and an output frame of the last original frame, as mapped features in feature nodes. Then, we refine extracted features by enhancing the mapped features as enhancement nodes with randomly generated weights. We then connect all extracted features to the output layer with a target weight vector. With the target weight vector, we can minimize the temporal information loss between consecutive frames and the video fidelity loss in the output videos. Finally, we remove the temporal inconsistency in the processed video and output a temporally consistent video. Besides, we propose an alternative incremental learning algorithm based on the increment of the mapped feature nodes, enhancement nodes, or input data to improve learning accuracy by a broad expansion. We demonstrate the superiority of our proposed TBLS by conducting extensive experiments.**

***Index Terms*—Incremental learning, temporally broad learning system (TBLS), video temporal consistency.**

I. Introduction

WITH the constant development of computer technology, video-related applications have received extensive attention, and researchers have widely explored the video processing technology. Nevertheless, the newly developed techniques, like filters used for enhancing, restoring, editing, and analyzing static images, cannot be an appropriate choice for videos because a video is not merely a series of images. Therefore, processing each video frame individually (like applying image-based processing methods) might not be problematic in simple scenarios, such as low- and high-pass filtering. However, in some complicated situations, it might produce issues like cutting down the temporal consistency between consecutive frames that often leads to severe video quality degradation. There can be several causes behind the temporal inconsistency in videos. For example, it might be produced because of the fall of the optimization technique into local minima or the dependence of a filter on the statistics, such as average color, which may not be stable for all video frames. Without temporal consistency, flickering artifacts, such as color change or brightness fluctuation, may occur in videos. Because we always make a brightness constancy hypothesis on video matching applications, temporal inconsistency will severely impact these video matching-involved applications. Therefore, to improve the video visual aesthetic and benefit other computer vision applications, such as motion estimation, video temporal consistency has been an increasingly hot topic.

Many methods have been proposed to remove the flickering artifacts in videos. Most of them formulate the temporal consistency objective function as an energy minimization problem [1]–[4]. However, methods proposed by Bonneel *et al.* [1] and Ye *et al.* [2] are designed for applying image-based intrinsic decomposition methods to videos. To address flickering artifacts in applications besides intrinsic video decomposition, Bonneel *et al.* [3] reconstructed an original frame with flickering artifacts from its last reconstructed nonflickering output frame. However, the method of warping a frame from another frame by optical flow leads to inaccuracy. Moreover, warping-based video temporal consistency is not robust when dealing with occlusion and low-texture area. Other methods that align flickering frames to preselected nonflickering key frames are not suitable for the scenario being solved in this article because we intend to maintain temporal consistency in processed videos where nonflickering key frames are hard to choose, and aligning

Manuscript received 30 December 2019; revised 13 January 2021 and 15 April 2021; accepted 7 May 2021. Date of publication 2 June 2021; date of current version 1 July 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61872241 and Grant 61572316; in part by the National Key Research and Development Program of China under Grant 2019YFB1703600; and in part by the Hong Kong Polytechnic University under Grant P0030419, Grant P0030929, and Grant P0035358. This article was recommended by Associate Editor Y. M. Cheung. *(Corresponding author: Bin Sheng.)*

Bin Sheng is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: shengbin@sjtu.edu.cn).

Ping Li is with the Department of Computing, Hong Kong Polytechnic University, Hong Kong (e-mail: p.li@polyu.edu.hk).

Riaz Ali is with the Department of Computer Science, Sukkur IBA University, Sukkur 65200, Pakistan (e-mail: riaz.khp@iba-suk.edu.pk).

C. L. Philip Chen is with the School of Computer Science and Engineering, South China University of Technology, Guangzhou 510006, China, also with the Navigation College, Dalian Maritime University, Dalian 116026, China, and also with the Faculty of Science and Technology, University of Macau, Macau, China (e-mail: philip.chen@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2021.3079311.

Digital Object Identifier 10.1109/TCYB.2021.3079311

flickering frames to nonflickering frames is impractical and infeasible.

In this article, we propose a learning-based video temporal consistency approach. We resort to the inspiration within the deep learning domain and propose a temporally broad learning system (TBLS) to reduce the flickering artifacts generated due to applying image-based processing methods to original videos. Our novel broad learning-based system adopts a flat network layout with the recurrence in time.

Before introducing our method, it is necessary to explain our related notations. We denote the original video, which has not been processed with the image-based processing method, as the *original video*, the video processed with image-based methods and having flickering artifacts as *temporally inconsistent video*, and the video processed with different flickering removal methods as *output video*. We denote the *n*th frame in the original video as an original frame I_n, the *n*th corresponding frame in the temporally inconsistent video as P_n, and the *n*th output frame in the output video as O_n. In our technique, we first take data composed of the current original frame I_n, the corresponding frame P_n, and the output frame O_{n-1} of last original frame I_{n-1} as the input of the TBLS. In the training process, which is used to solve the target weight vector in the TBLS, the output video is generated by applying the traditional video temporal consistency methods to the processed video. While in the testing process, we directly use the TBLS to maintain the video temporal consistency. Then, we transfer the input data into mapped features. Next, we enhance the mapped features as enhancement nodes with randomly generated weights. We can also expand the input data, the mapped feature nodes or the enhancement nodes with the novel incremental learning algorithms. Compared with the deep learning architecture, we skip the retraining phase in the proposed incremental learning algorithm when we expand the broad learning system to boost the learning accuracy.

Due to the lack of publicly available training video datasets, we establish a video dataset specifically for video temporal consistency. We first download a large number of original nonflickering videos from the Internet, and then we process those original videos with various image-based methods. Finally, corresponding flickering videos are obtained. Using this novel dataset, we have successfully trained our proposed broad learning-based video temporal consistency network. In summary, our proposed TBLS-based video temporal consistency approach makes the following three main contributions.

1) We first introduce the learning-based approach to the field of video temporal consistency.
2) We propose a broad learning-based video temporal consistency approach, where we extract the features of input data and enhance these features in a flat network. In this way, we skip the time-consuming retraining step that is often used in deep learning methods.
3) We establish a big video dataset that enables the learning for video temporal consistency.

II. Related Work

To the best of our knowledge, there is no previous research proposed specifically at addressing video temporal consistency using learning-based methods. Nevertheless, various relative studies, including video temporal consistency and broad learning methods, have been researched extensively in the computer vision field. In this article, the video temporal consistency is combined with the broad learning system. A TBLS is proposed to improve the video temporal consistency. A broad learning system can reduce the training time complexity on a large scale compared to the deep learning architecture. Besides, the incremental learning algorithm can be used to improve the learning accuracy. On the basis of the analysis discussed above, the video temporal consistency method based on a broad learning system can effectively enhance the video temporal consistency. Now, we will introduce related studies on the video temporal consistency and broad learning methods.

Video Temporal Consistency: Conventional video temporal consistency methods can be divided into two types. On the one hand, some methods align flickering frames with nonflickering frames to maintain the temporal consistency in the output videos. On the other hand, some approaches formulate the temporal consistency objective function as an energy minimization task.

In video temporal consistency methods [5]–[8], a set of frames is selected according to certain predetermined measure standards. Then, other flickering frames are aligned with these selected frames to maintain the temporal consistency between frames. In the video temporal consistency approach proposed by Farbman and Lischinski [5], the authors first designated several frames as anchors and then aligned other frames with these selected anchors according to a tonal adjustment map. Because the establishment of the adjustment maps takes advantage of video temporal coherence and the characteristic of the tonal fluctuation, they can stabilize the tonal fluctuation in the output video. Bonneel *et al.* [7] proposed a frame-aligning-based color grading method. They first estimated a color transform between the input video and the model video, then they interpolated the transformation by minimizing the curvature. Next, the frames representing the interpolation curve are designated as key frames to refine the color grade. Huang *et al.* [8] proposed a method to maintain the temporal consistency in the video recoloring process. They first extract key colors and then interpolate the remapped frames according to the preselected key colors. However, these methods are impractical in the context of this article's problem because they require preselected key frames from the processed video, which, in the case of our context, contain flickers; hence, their produced results will not be much reliable.

Some other methods keep the video temporal consistency by minimizing the least-squares energy. HaCohen *et al.* [9] constructed an energy function composed of the color discrepancy term and appearance consistency term to edit a collection of photos consistently. Lang *et al.* [10] make a temporal smoothness assumption used to constrain the ambiguities between an objective frame and an original frame and then solve the global optimization approximation. Kalantari *et al.* [11]

and Aydin *et al.* [12] addressed the temporal artifacts in the tone mapping of HDR video by formulating an energy function. Multiple motion models composed of optical flow and patch-based method are proposed in [11] to enforce temporal consistency between consecutive frames. The optimization framework of tone mapping in [12] consists of two parts: 1) the base layer used to compress the dynamic range and 2) the detail layer that can preserve the local contrast. Both [3] and [4] reconstruct flickering frame with nonflickering frames, while the difference is that [3] warps the last nonflickering output frame to obtain the current output frame by optical flow, and [4] chooses the enhancement curve of the corresponding nonflickering frames to replace the original enhancement curve of the flickering frame. The method in [4] is used when we apply image-based enhancement techniques to videos, but the work of [3] can be applied in more cases.

Broad Learning System: With the immense progress and breakthrough made by deep learning architecture in a lot of applications [13]–[16], many networks like convolutional neural networks and deep belief networks have been proposed [17]–[21]. Although deep learning architecture has been proven to be useful in the field of computer vision, it still suffers from the indispensable requirement of big data and the time-consuming retraining process. The random vector functional link neural network (RVFLNN) is proposed to overcome these shortcomings [22]–[24]. Though RVFLNN has been proposed to play a vital role in lots of applications, it cannot deal with the time-variety problems [25], [26]. To address this shortcoming, a broad learning system is proposed, which can update the input data or feature nodes with time [25], [27]–[29]. Due to its effectiveness in providing competitive results and taking significantly less training time as compared to the deep-learning-based techniques, BLS has been used by many researchers to solve a variety of problems [30]–[37]. In this article, we use the BLS to create our TBLS that preserves video temporal consistency. In order to improve the learning accuracy in output videos, we also propose an incremental learning algorithm to update any of the input data, mapped feature nodes, or enhancement nodes conveniently, without the need for retraining the whole network once again. In this way, we reduce the time complexity on a large scale compared to the deep learning architecture.

III. Approach Overview

Fig. 1 is a general framework of our proposed TBLS-based video temporal consistency approach. In our approach, we denote the nth frame in the original video as an original frame I_n, the nth corresponding frame in the temporally inconsistent video as P_n, and the nth output frame in the output video as O_n. We first take data composed of current original frame I_n, the corresponding frame P_n, and the output frame O_{n-1} of last original frame I_{n-1} as the input of the TBLS. Then, we extract features from input data to generate the mapped feature nodes. To compensate for the insufficiency of extracted features, when compared with the retraining process of features in the deep learning architecture, we enhance the mapped features with random weights to generate enhancement nodes. Next, we connect all extracted features to the output layer with a target weight vector. Finally, we work out an optimal weight vector to minimize an objective function. Because the sought weight vector has an aptitude for minimizing the difference between the original video and the output video, the fidelity of the output video is guaranteed. Meanwhile, the sought weight vector is also designed to minimize the discrepancy between the consecutive frames; thus, the temporal consistency in the output video is maintained. In this way, our proposed TBLS network predicts an output frame O_n. Subsequently, we combine I_{n+1}, P_{n+1} and O_n as the input of the $(n+1)$th network to obtain the output frame O_{n+1} until we output a nonflickering video. Our approach is designed for removing the flickering artifacts by taking advantage of temporal information between consecutive frames. As a result, our approach outperforms other video temporal consistency methods without a retraining process. Algorithm 1 summarizes our proposed approach.

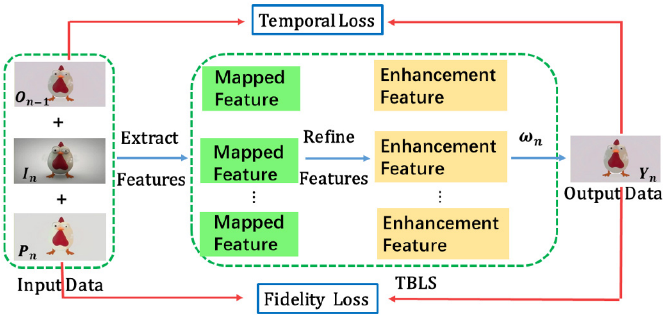


Fig. 1. Overview of our TBLS.

IV. Temporally Broad Learning System

In this section, we introduce the details of our proposed TBLS. First, the model constructed with a flat network is given. We enforce temporal consistency from the perspective of both input data and the construction of the objective function to output a frame.

A. Temporally Broad Learning System

Our proposed TBLS consists of the following steps: first, we generate the mapped features from the input data, which is composed of a current original frame I_n, a corresponding frame in the temporally inconsistent video P_n, and an output frame O_{n-1} of the last original frame I_{n-1}. Second, we enhance the extracted features with randomly generated weights to form the enhancement features. Finally, we seek a weight vector used to link the feature nodes with the output layers. With the target weight vector, which can minimize the objective function, we enforce temporal consistency in the output video.

In the proposed TBLS, we project the input data $\boldsymbol{X_n}$, where $\boldsymbol{X_n} = [I_n|P_n|O_{n-1}]$, to form the ith mapped feature nodes $\boldsymbol{Z_i}$ by a mapped function $\phi_i(\boldsymbol{X_n W_{ei}} + \boldsymbol{\beta_{ei}})$, where $\phi_i(\cdot)$ can be either a sigmoid function or a tangent function, $\boldsymbol{W_{ei}}$ and $\boldsymbol{\beta_{ei}}$ are randomly generated weights with proper dimensions. We denote all mapped features in the nth network as $\boldsymbol{Z_n^m} = [\boldsymbol{Z_1}, \ldots, \boldsymbol{Z_i}, \ldots, \boldsymbol{Z_m}]$. That is to say, utilizing $\boldsymbol{Z_n^m}$ to represent that there are m groups of mapped feature nodes

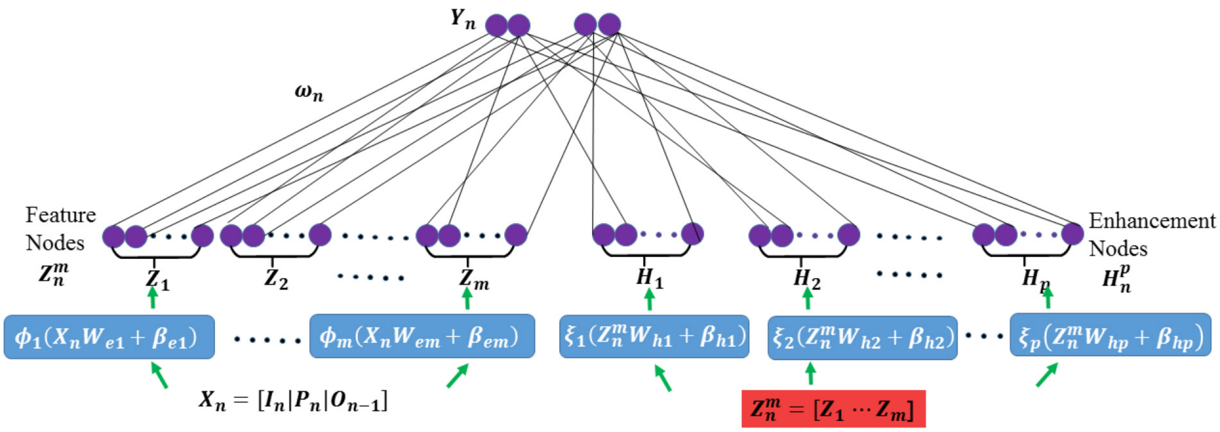

Y_n
ω_n
Feature Nodes Z_n^m
Z_1
Z_2
Z_m
H_1
H_2
H_p
Enhancement Nodes H_n^p
$\phi_1(X_nW_{e1}+\beta_{e1})$
$\phi_m(X_nW_{em}+\beta_{em})$
$\xi_1(Z_n^mW_{h1}+\beta_{h1})$
$\xi_2(Z_n^mW_{h2}+\beta_{h2})$
$\xi_p(Z_n^mW_{hp}+\beta_{hp})$
$X_n=[I_n|P_n|O_{n-1}]$
$Z_n^m=[Z_1\cdots Z_m]$

Fig. 2. Illustration of TBLS.

Algorithm 1 Broad Learning-Based Video Temporal Consistency

```
Require: Training samples $X_n$
Ensure: Weight $\omega_n$
for $i = 1; i \le m;$ do
    Random $W_{ei}, \beta_{ei}$;
    Calculate $Z_i = [\phi_i(X_n W_{ei} + \beta_{ei})]$;
end for
Set the feature mapping group $Z_n^m = [Z_1, ..., Z_m]$;
for $j = 1; j \le p;$ do
    Random $\Omega_{hj}, \beta_{hj}$;
    Calculate $H_j = [\xi_j(Z_n^m W_{hj} + \beta_{hj})]$;
end for
Set the enhancement nodes group $H_n^P = [H_1, ..., H_p]$;
Set $A_n = [Z_n^m | H_n^p]$;
Formulate the temporal consistency objective with Eq. (2);
Formulate the video fidelity objective with Eq. (3);
Obtain the target weight vector $\omega_n$ to minimize the temporal consistency loss and the video fidelity loss in the joint optimization Eq. (4);
```

in the nth TBLS. Then, we enhance all extracted mapped features, respectively, to form the enhancement nodes. Using $\xi_j(Z_n^m W_{hj} + \beta_{hj})$, where $\xi_j(\cdot)$ can be either a sigmoid function or a tangent function, W_{hj} and β_{hj} are randomly generated weights with proper dimensions. The jth enhancement nodes H_j can be obtained with randomly generated weights ξ_j, where j is on the interval $[1, p]$, all enhancement nodes in the nth network can be denoted as $H_n^p = [H_1, \ldots, H_j, \ldots, H_p]$. With H_n^p, we can represent the p groups of enhancement nodes in the nth TBLS. The values of m and p are decided through a trial-and-error method. Each mapped function $\phi_i(\cdot)$ or enhancement function $\xi_j(\cdot)$ can be different from each other and can be determined upon the practical case. We combine mapped features and enhancement nodes to construct all extracted feature nodes $A_n = [Z_n^m | H_n^p]$ in the nth network. Finally, we link all extracted features A_n to the output layer with certain target weight vector ω_n. The illustration of the TBLS is shown in Fig. 2. Hence, the predictive output frame Y_n can be computed as

$$\begin{aligned} Y_n &= \left[Z_n^m | H_n^p\right]\omega_n \\ &= A_n \omega_n \end{aligned} \tag{1}$$

where $\omega_n = [Z_n^m | H_n^p]^+ Y_n$, and we can easily solve it with the ridge regression approximation of $[Z_n^m | H_n^p]^+$. The brief illustration of our network without temporal consistency in the constraint of weight vector ω_n is shown in Fig. 3(a). However, the ridge regression approximation does not utilize the temporal consistency between consecutive frames; thus, the video fidelity in the output video is omitted. With this in mind, we improve the network as Fig. 3(b), where we enforce temporal consistency and video fidelity in the solution process of ω_n. To enforce temporal consistency in the output video, we intend to minimize the temporal inconsistency cost C_t between consecutive frames, which is computed as

$$C_t = \|A_n \cdot \omega_n - Y_{n-1}\|_2^2 \tag{2}$$

where $\|\cdot\|_2^2$ represents the L2 norm, A_n represents all extracted features in the nth TBLS network, and ω_n is a target weight vector which connects the extracted feature nodes and the output layer in the TBLS. Y_{n-1} is the $(n-1)$th output frame in the training set, which is used to train the $(n-1)$th TBLS and obtain the target weight vector ω_n.

With (2), the temporal inconsistency, caused by applying the image-based methods to original video frame by frame between consecutive frames, can be minimized. In practice, the output video content should agree with the temporally inconsistent video on which image-based methods have been applied, which we call as the video fidelity. To maintain the video fidelity, we add a restriction that the content difference between the output video and the temporally inconsistent video should be minimized. We compute the video fidelity cost C_f as

$$C_f = \|A_n \cdot \omega_n - P_n\|_2^2 \tag{3}$$

where A_n is extracted features in the nth network and ω_n is the goal weight vector. P_n is the nth frame in the processed video. With (3), we reduce the content difference between the output video and the temporally inconsistent video. After incorporating temporal consistency and video fidelity in the $(n-1)$th network, we illustrate the TBLS model via Fig. 3(b).

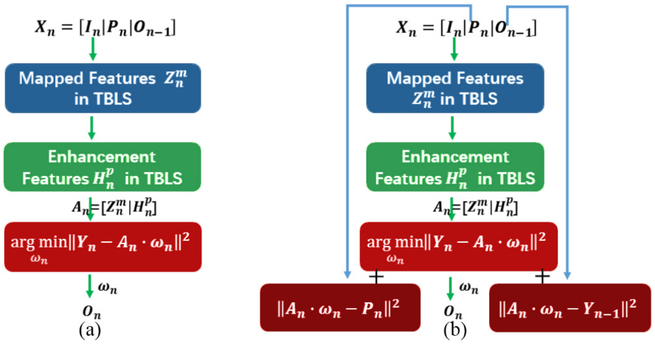

$X_n = [I_n|P_n|O_{n-1}]$
Mapped Features Z_n^m in TBLS
Enhancement Features H_n^p in TBLS
$A_n=[Z_n^m|H_n^p]$
$\arg\min_{\omega_n}\|Y_n - A_n \cdot \omega_n\|^2$
ω_n
O_n
(a)
$X_n = [I_n|P_n|O_{n-1}]$
Mapped Features Z_n^m in TBLS
Enhancement Features H_n^p in TBLS
$A_n=[Z_n^m|H_n^p]$
$\arg\min_{\omega_n}\|Y_n - A_n \cdot \omega_n\|^2$
+
$\|A_n \cdot \omega_n - P_n\|^2$
ω_n
O_n
(b)
+
$\|A_n \cdot \omega_n - Y_{n-1}\|^2$

Fig. 3. Illustration of different TBLS models. (a) We only enforce temporal consistency at the input data level. The input data in this model is composed of the current input frame and the output frame of the last frame. (b) We enforce temporal consistency and video fidelity when we solve the target weight vector $\boldsymbol{\omega_n}$ for the TBLS.

Algorithm 2 Alternative Broad Learning-Based Video Temporal Consistency

```
Require: Training samples $X_n$
Ensure: Weight $\omega_n$
for $i = 1; i \leq m;$ do
    Random $W_{ei}, \beta_{ei}$;
    Calculate $Z_i = [\phi_i(X_n W_{ei} + \beta_{ei})]$;
    Random $\omega_{hi}, \beta_{hi}$;
    Calculate $H_i = [\xi_i(Z_i W_{hi} + \beta_{hi})]$;
end for
Set the feature mapping group $Z_n^m = [Z_1, \ldots, Z_m]$;
Set the enhancement nodes group $H_n^m = [H_1, \ldots, H_m]$;
Set $A_n = [Z_n^m|H_n^m]$;
Formulate the temporal consistency objective with Eq. (2);
Formulate the video fidelity objective with Eq. (3);
Obtain the target weight vector $\omega_n$ to minimize the temporal consistency loss and the video fidelity loss in the joint optimization Eq. (4);
```

Equation (4) shows how to incorporate the above two consistencies to output a nonflickering video. We adopt a similar elastic net to [38], which utilizes temporal consistency to predict the number of pedestrians in a video accurately

$$\arg\min_{\boldsymbol{\omega_n}} \|\boldsymbol{Y_n} - \boldsymbol{A_n} \cdot \boldsymbol{\omega_n}\|_2^2 + \lambda_1 \cdot \|\boldsymbol{\omega_n}\|_1 + \lambda_2 \cdot \|\boldsymbol{\omega_n}\|_2^2 + \lambda_t \cdot C_t + \lambda_f \cdot C_f \tag{4}$$

where we have used the refined least squares method to solve the optimization problem of minimizing the first term $\|\boldsymbol{Y_n} - \boldsymbol{A_n} \cdot \boldsymbol{\omega_n}\|_2^2$ to generate a minimal error for the output video. The second term and the third term are regularization terms. λ_1 and λ_2 are regularization coefficients for the L1 norm $\|\boldsymbol{\omega_n}\|_1$ and the L2 norm $\|\boldsymbol{\omega_n}\|_2^2$, respectively. The two regularization terms are used to prevent the overfitting of our TBLS network. λ_t in the fourth term and λ_f in the last term are regularization coefficients for the temporal consistency term and the video fidelity term, respectively. Similar to [38], we can solve (4) by the least angle regression [39].

B. Alternative Temporally Broad Learning System

When the task required to be solved is simple, or the desired accuracy requirement is easy to achieve, we introduce another alternative TBLS (see Algorithm 2). In the alternative TBLS, we assume that the expansion of mapped features is relevant to the augmentation of enhancement nodes. The process of the alternative TBLS is shown in Fig. 4. In the alternative TBLS, after projecting the input data to the ith mapped features $\boldsymbol{Z_i}$ with random weight, we can obtain the corresponding ith enhancement nodes $\boldsymbol{H_i}$ with the enhancement function $\boldsymbol{H_i} = \boldsymbol{\xi}(\boldsymbol{Z_i W_{hi}} + \boldsymbol{\beta_{hi}})$, where i is on the interval $[1, m]$, m is the number of enhancement node groups. Next, we improve each enhancement node with the corresponding mapped feature nodes to generate enhancement nodes $\boldsymbol{H_n^m}$. Then, we combine all mapped feature nodes $\boldsymbol{Z_n^m}$ with corresponding enhancement nodes $\boldsymbol{H_n^m}$ to form all extracted features $\boldsymbol{A_n}$. As has been proven in [26], the TBLS in Fig. 2 is equivalent to the alternative TBLS in Fig. 4 if and only if that the dimension of enhancement nodes is equal to that of the mapped features in the network.

V. Incremental Learning Algorithm

Because we can reduce the flickering artifacts in our experiments and achieve satisfying results with the proposed TBLS, we have no plan for improving the learning results with additional methods. However, we might not reach the preset learning accuracy with existing extracted features in complicated practical applications. The issue mentioned above may be caused by the lack of extracted features that can reflect the input data structure. To improve the broad learning system's accuracy, we propose an incremental learning approach to expand the flat network at the level of input data, mapped features, or enhancement features.

A. Increment of Enhancement Nodes and Mapped Features

When the task to be solved is complicated, or we intend to improve the broad learning system's accuracy, we can increase the enhancement nodes. As we propose two TBLS models, we introduce all our following incremental learning approaches with the TBLS model. As shown in Fig. 5, when we add b enhancement nodes to original enhancement nodes $\boldsymbol{H_n^p}$, we can represent the additional b enhancement nodes as $\boldsymbol{H_{p+1}}$ by using $\boldsymbol{H_{p+1}} = \boldsymbol{\xi_e}(\boldsymbol{Z_n^m W_{hp+1}} + \boldsymbol{\beta_{hp+1}})$, where $\boldsymbol{H_{p+1}} \in \boldsymbol{R^b}$ and $\boldsymbol{\xi_e} \in \boldsymbol{R^b}$. $\boldsymbol{W_{hp+1}}$ and $\boldsymbol{\beta_{hp+1}}$ are randomly generated. With the augmentation of enhancement nodes $\boldsymbol{H_{p+1}}$, all extracted feature nodes $\boldsymbol{A_n} = [\boldsymbol{Z_n^m}|\boldsymbol{H_n^p}]$ can be rewritten as $\boldsymbol{A_n} = [\boldsymbol{Z_n^m}|\boldsymbol{H_n^p}|\boldsymbol{H_{p+1}}]$.

After reaching a specific point, the results start saturating with the increase in the number of enhancement nodes in the network and our broad learning system's accuracy cannot be further enhanced. In this situation, increasing the number of enhancement nodes does not change the convergence of output video. It can be noted that the fixed convergence of output video is caused by the insufficiency of the mapped features. Therefore, in our incremental mapped features learning approach, with the corresponding illustration in Fig. 6, we add

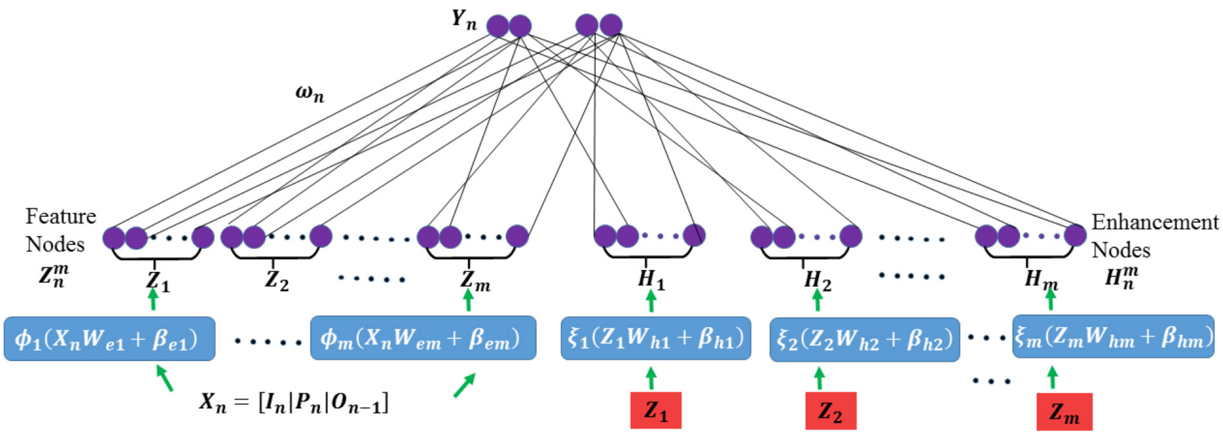

Y_n
ω_n
Feature Nodes
Z_n^m
Z_1
Z_2
Z_m
H_1
H_2
H_m
Enhancement Nodes
H_n^m
$\phi_1(X_nW_{e1}+\beta_{e1})$
$\phi_m(X_nW_{em}+\beta_{em})$
$\xi_1(Z_1W_{h1}+\beta_{h1})$
$\xi_2(Z_2W_{h2}+\beta_{h2})$
$\xi_m(Z_mW_{hm}+\beta_{hm})$
$X_n = [I_n|P_n|O_{n-1}]$
Z_1
Z_2
Z_m

Fig. 4. Illustration of alternative TBLS.

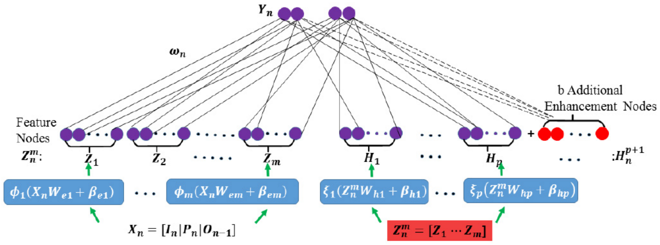

Y_n
ω_n
b Additional Enhancement Nodes
Feature Nodes
Z_n^m:
Z_1
Z_2
Z_m
H_1
H_p
$:H_n^{p+1}$
$\phi_1(X_nW_{e1}+\beta_{e1})$
$\phi_m(X_nW_{em}+\beta_{em})$
$\xi_1(Z_n^mW_{h1}+\beta_{h1})$
$\xi_p(Z_n^mW_{hp}+\beta_{hp})$
$X_n = [I_n|P_n|O_{n-1}]$
$Z_n^m = [Z_1 \cdots Z_m]$

Fig. 5. Illustration of incremental enhancement nodes.

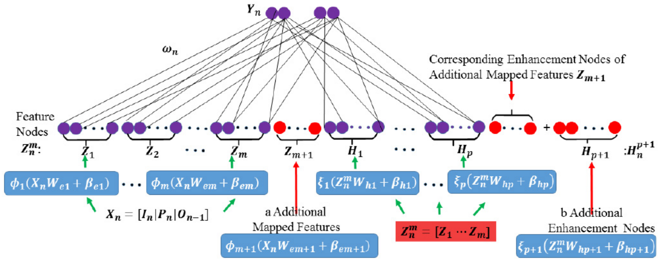

Y_n
ω_n
Corresponding Enhancement Nodes of Additional Mapped Features Z_{m+1}
Feature Nodes
Z_n^m:
Z_1
Z_2
Z_m
Z_{m+1}
H_1
H_p
H_{p+1}
$:H_n^{p+1}$
$\phi_1(X_nW_{e1}+\beta_{e1})$
$\phi_m(X_nW_{em}+\beta_{em})$
$\xi_1(Z_n^mW_{h1}+\beta_{h1})$
$\xi_p(Z_n^mW_{hp}+\beta_{hp})$
$X_n = [I_n|P_n|O_{n-1}]$
a Additional Mapped Features
$\phi_{m+1}(X_nW_{em+1}+\beta_{em+1})$
$Z_n^m = [Z_1 \cdots Z_m]$
b Additional Enhancement Nodes
$\xi_{p+1}(Z_n^mW_{hp+1}+\beta_{hp+1})$

Fig. 6. Illustration of incremental mapped features.

a mapped features, which can be denoted by $\boldsymbol{Z}_{m+1}$ as a whole, to the TBLS in Fig. 5. The additional a mapped features can be obtained through $\boldsymbol{Z}_{m+1} = \boldsymbol{\phi}_{m+1}(\boldsymbol{X}_n\boldsymbol{W}_{em+1}+\boldsymbol{\beta}_{em+1})$, where $\boldsymbol{Z}_{m+1} \in \boldsymbol{R}^a$ and $\boldsymbol{\phi}_{m+1} \in \boldsymbol{R}^a$, $\boldsymbol{W}_{em+1}$ and $\boldsymbol{\beta}_{em+1}$ are randomly generated. Similar to calculating the number of original/initial mapped feature nodes m and enhancement nodes p, we compute the number of additional mapped features a through trial-and-error to maintain the balance in the training time and accuracy of the results. After increasing the number of mapped feature nodes, we can rewrite the mapped feature nodes as $\boldsymbol{Z}_n^m = [\boldsymbol{Z}_n^m|\boldsymbol{Z}_{m+1}]$. The corresponding enhancement nodes of the a additional mapped features $\boldsymbol{Z}_{m+1}$ can be denoted as $\boldsymbol{H}_{m+1}$, which can be computed as $\boldsymbol{H}_{m+1} = \boldsymbol{\xi}_{m+1}(\boldsymbol{Z}_n^m\boldsymbol{W}_{hm+1}+\boldsymbol{\beta}_{hm+1})$. Similar to $\boldsymbol{Z}_{m+1}$ and $\boldsymbol{\phi}_{m+1}$, $\boldsymbol{H}_{m+1} \in \boldsymbol{R}^a$ and $\boldsymbol{\xi}_{m+1} \in \boldsymbol{R}^a$. Corresponding weight vectors $\boldsymbol{W}_{hm+1}$ and $\boldsymbol{\beta}_{hm+1}$ are randomly generated. With the augmentation of mapped features, we can rewrite the feature $\boldsymbol{A}_n = [\boldsymbol{Z}_n^m|\boldsymbol{H}_n^p|\boldsymbol{H}_{p+1}]$ in Fig. 5 as $\boldsymbol{A}_n = [\boldsymbol{Z}_n^m|\boldsymbol{H}_n^p|\boldsymbol{H}_{p+1}|\boldsymbol{Z}_{m+1}|\boldsymbol{H}_{m+1}]$ in Fig. 6.

B. Increment of Input Data

In the models mentioned above, the input data only consists of original frame I_n, corresponding frame P_n in temporally inconsistent video, and output frame O_{n-1} of last original frame, so we only take the temporal consistency between consecutive frames I_n and I_{n-1} into account. In this work, for an original frame I_n, we suppose the previous frames of I_n have already been processed. However, the method of enforcing temporal consistency between consecutive frames is not always feasible. For instance, when there are drastic appearance variation or brightness changes between consecutive frames, only combining features from adjacent frames is inaccurate. There is a similar drawback in the solution process of target weight vector ω_n. To get rid of the inaccuracy caused by reconstructing an output frame from only an adjacent frame, some traditional video temporal consistency methods, such as the methods proposed by Dong *et al.* [4] and Bonneel *et al.* [7], take no less than one frame into account. With a similar method, we propose an incremental input data learning approach and increase the input data by finding the matching frames of I_n besides its adjacent frames.

Here, we introduce our proposed approach to find the matching frames of I_n. First, we adopt SIFT flow [40] to find the dense correspondence between consecutive frames and then link the dense corresponding pixels to form the motion path. Any two pixels having the same motion path are seen as corresponding pixels. Next, we compute the number of corresponding pixels denoted as c in other frames. Finally, corresponding frames for I_n can be determined according to the value of c. For instance, we intend to find the corresponding frames of I_n, compute the number of corresponding pixels between I_{n-1} and I_n, which can be denoted as $c(I_{n-1}, I_n)$. After computing $c(I_m, I_n)$, m is on the interval $[1, (n-1)]$, the frames with highest $c(I_m, I_n)$ can be denoted as corresponding frames of I_n. The corresponding output frame O_m of I_m can be added into our proposed BLS system; then, we extract the feature of O_m to refine the extracted features in the nth network. Finally, we utilize the incremental learning algorithm of input data to improve the learning accuracy.

Denoting X_c as the newly increasing input data, we can formulate the corresponding mapped features as

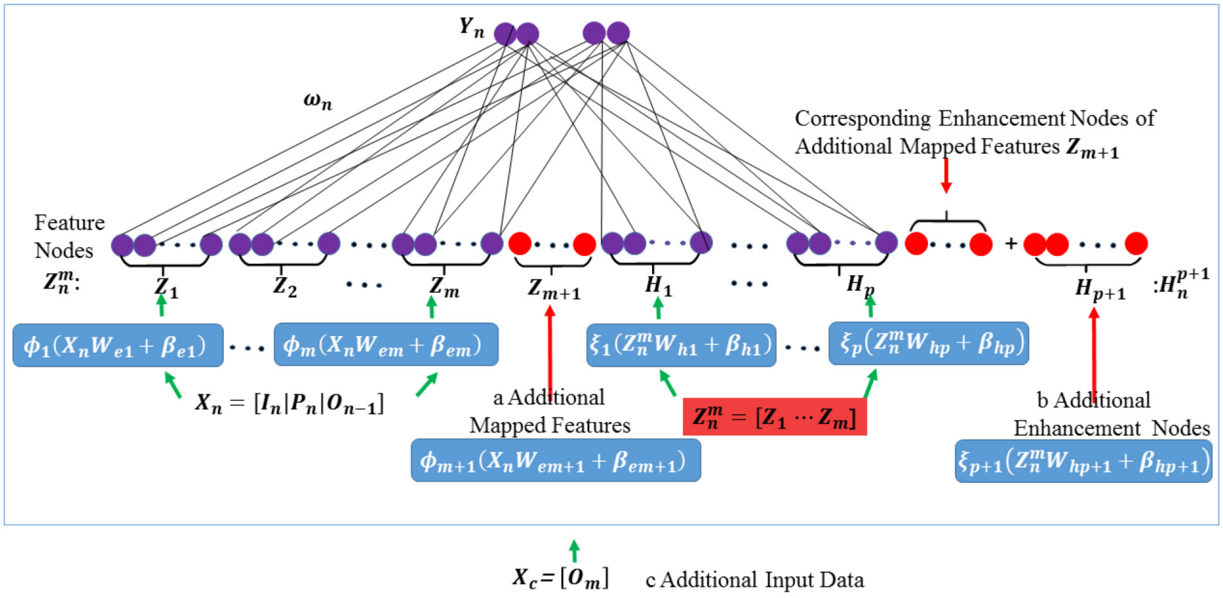

Y_n
ω_n
Corresponding Enhancement Nodes of Additional Mapped Features Z_{m+1}
Feature Nodes Z_n^m:
Z_1
Z_2
Z_m
Z_{m+1}
H_1
H_p
H_{p+1}
$:H_n^{p+1}$
$\phi_1(X_nW_{e1}+\beta_{e1})$
$\phi_m(X_nW_{em}+\beta_{em})$
$\xi_1(Z_n^mW_{h1}+\beta_{h1})$
$\xi_p(Z_n^mW_{hp}+\beta_{hp})$
$X_n = [I_n|P_n|O_{n-1}]$
a Additional Mapped Features
$Z_n^m = [Z_1 \cdots Z_m]$
b Additional Enhancement Nodes
$\phi_{m+1}(X_nW_{em+1}+\beta_{em+1})$
$\xi_{p+1}(Z_n^mW_{hp+1}+\beta_{hp+1})$
$X_c = [O_m]$
c Additional Input Data

Fig. 7. Illustration of incremental input data.

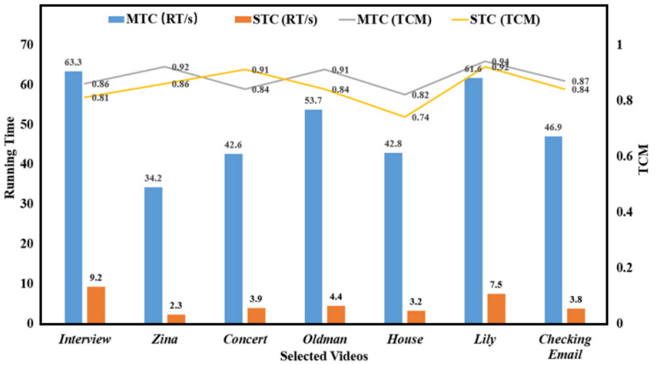


Fig. 8. Comparison results of running time and temporal consistency between MTC and STC.

$Z_c = \phi_c(X_cW_{ec} + \beta_{ec})$, where W_{ec} and β_{ec} are randomly generated. The corresponding enhancement nodes can be denoted as $H_c = \xi_c(Z_cW_{hc} + \beta_{hc})$, where W_{hc} and β_{hc} are randomly generated, then all extracted features in Fig. 7 can be rewritten as $A_n = [Z_n^m|H_n^p|H_{p+1}|Z_{m+1}|H_{m+1}|Z_c|H_c]$.

To see the performance of multiple frames-based video temporal consistency method and single frame-based video temporal consistency method, we make comparison experiments on our established dataset. The result of comparison experiments is shown in Fig. 8, where MTC and STC represent "multiframe-based temporal consistency method" and "single-frame-based temporal consistency method," respectively. RT/s in Fig. 8 describes the running times (comprising training time + inference time) of different video temporal consistency methods. TCM in Fig. 8 represents the temporal consistency metric proposed by Yao *et al.* [41], and the values of TCM can be applied to describe the quality of video temporal consistency. From Fig. 8, it can be noted that the running time of multiframe-based video temporal consistency method (MTC) is slower than the single-frame-based video temporal consistency method (STC). However, the TCM value of MTC is higher than the TCM value of STC. Since we need more time to detect the temporally corresponding frames when the task is easy and prefer the time complexity rather than the quality, using one previous frame can meet the requirements of removing flickering artifacts. However, if we intend to improve the videos' temporal consistency, the incremental model of input data can meet the requirement and produce nonflickering output videos. In this way, whether we choose MTC or STC depends on the practical application to be solved.

VI. Experimental Results

In this section, we introduce the experimental details, including the dataset used to train our proposed TBLS and the setting of training parameters in our proposed system. Then, to prove the effectiveness of TBLS, we will compare the flicker-removing results of our approach with the existing mainstream methods on the novel established video temporal consistency dataset. All experiments are tested on a laptop equipped with a 2.4-GHz Intel Core i3-3110M CPU and 8-GB memory.

A. Dataset

We have trained the proposed TBLS in our experiments with the training data comprising original videos, corresponding temporally inconsistent videos, and output videos. We categorize the videos obtained by applying the image-based methods to original videos as temporally inconsistent videos and the videos obtained from applying the flickering-removing methods to the temporally inconsistent videos as output videos. Because flickering artifacts mentioned in this article can be generated from applying random image-based processing methods to videos, we apply various image-based methods to original videos in a frame-by-frame manner. In this way, we can improve the generalization ability of our proposed TBLS system.

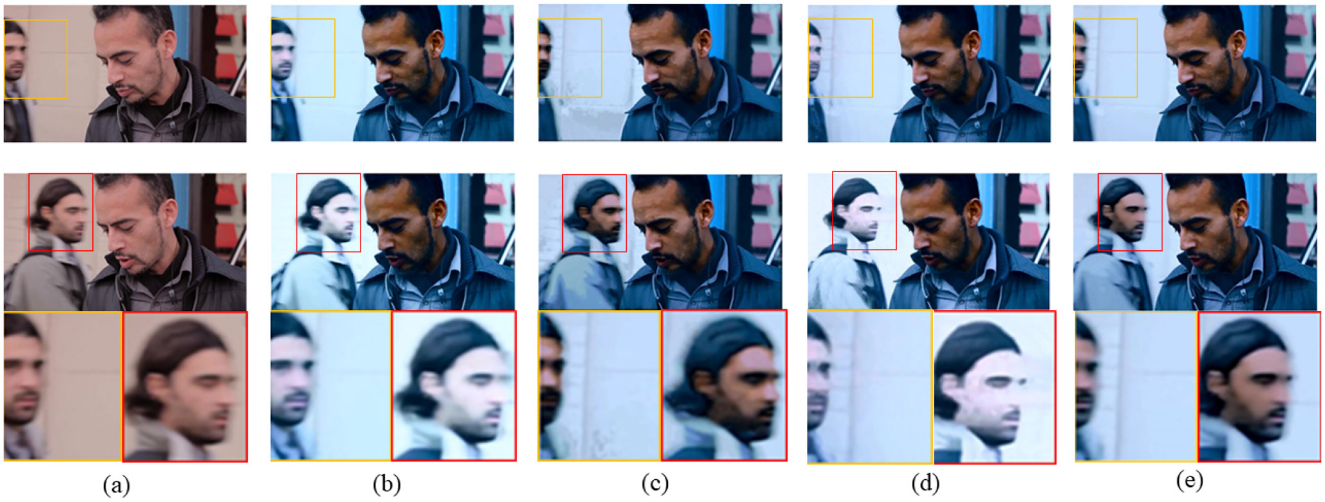


Fig. 9. Experimental results of the *interview* video. The top row is the 52nd frame, and the bottom row is the 58th frame. (a) Original frames are temporally consistent. (b) After processing the original frames with the image-based color grading method, there are temporal artifacts in the processed frames (please see the color change on the man's face). (c) There are residual temporal artifacts in the results of Bonneel *et al.* [3] (please see the blurring artifacts on the face of the man). (d) In the results of Lang *et al.* [10], there are the same residual temporal artifacts as with the results of Bonneel *et al.* [3]. (e) Temporal artifacts are reduced in the results of our proposed TBLS.

Now, we introduce our dataset comprehensively. First of all, we collect about 1000 nonflickering videos from the Internet and refer to them as the original videos. We have divided our dataset consisted of 1000 short videos into six classes: 1) male; 2) female; 3) concert; 4) building; 5) landscape; and 6) animal. The number of videos in each video class is selected arbitrarily and is as follows: 243, 182, 75, 198, 147, 155. Moreover, the average length of videos in our established dataset is 200 frames. In our experiments, we first establish a video dataset. Then, with inspiration from the work of Chen and Liu [27], we select 400 videos from the whole videos set as original videos in the training set, with the remaining 600 videos being the original videos in the testing set. The image-based methods used to process original videos in both the training set and the testing set include color grading, color constancy, spatial white balancing, style transfer, intrinsic image decomposing, color harmonization, and HDR compression. Applying the image-based methods mentioned above to the original videos in both the training and testing sets can generate corresponding temporally inconsistent videos. After utilizing one traditional video temporal consistency method [3] to the processed videos in the training set, corresponding output videos are obtained. We combine the original videos, the corresponding temporally inconsistent videos, and the training set's output videos to train our TBLS system.

B. Parameter Settings

According to our objective function in (4), we need to determine the four regularization parameters, namely: 1) λ_1; 2) λ_2; 3) λ_t; and 4) λ_f. In our experiment, we adopt a grid search to find the most befitting parameters. The determined parameters can make good performance in the leave-one-out (LOO) cross-validation. In the LOO cross-validation, one video is removed from the training set and the remaining videos are utilized to train the models. The range of λ_1, λ_t, and λ_f is from 0 to 1 with the step 0.1, 0–0.2 with the step 0.05, and 0–0.1 with the step 0.0005, respectively. And the range of λ_2 is determined by LARS automatically. In addition, with a similar method of generating random weights to [27], the randomly generated weights $\boldsymbol{W_{ei}}$ and $\boldsymbol{\beta_{ei}}$, $\boldsymbol{W_{hi}}$ and $\boldsymbol{\beta_{hi}}$, and others in our experiments are obtained from the standard normal distribution on the interval $[-1, 1]$. Transformation functions $\phi_i(\cdot)$ and $\xi_j(\cdot)$ in our experiment are set as the nonlinear sigmoid functions.

Next, we present both objective results (consisting of qualitative and quantitative results) and subjective results (comprising user study).

C. Qualitative Results

To evaluate the generalization ability of our temporally broad learning network, experiments over a number of videos have been performed. To verify the flickering-removal performance of our proposed TBLS, we also make a comparison with the traditional video temporal consistency methods, including [3] and [10]. The experimental results are shown from Figs. 9–15. In Fig. 9, which corresponds to *interview* video, we utilized the image-based color grading method to the original video [Fig. 9(a)], and then the temporally inconsistent video is obtained [Fig. 9(b)]. Then, we use the method of Bonneel *et al.* [3] to output the temporally consistent video [Fig. 9(c)]. Next, we combine the original *interview* video, temporally inconsistent video, and the output video to train our TBLS system. We demonstrate the effectiveness of the output video generated from our TBLS [Fig. 9(e)]. Our method eliminates the residual temporal inconsistency that is present in the results of Bonneel *et al.* [3] [Fig. 9(c)] and Lang *et al.* [10] [Fig. 9(d)].

The temporally inconsistent videos used to train our TBLS can not only be generated from applying the image-based color grading method to the original video but can also be obtained from transferring other image-based methods, such

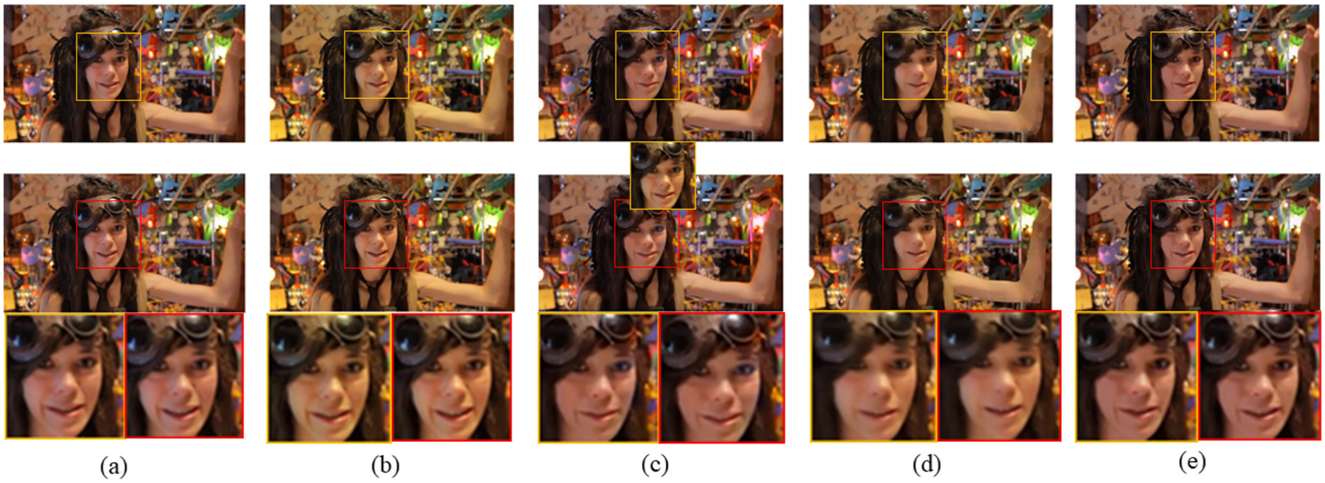


Fig. 10. Experimental results of the *zina* video. The top row is the 26th frame, and the bottom row is the 27th frame. (a) Original frames are temporally consistent. (b) After applying image-based color constancy to original frames, there is temporal inconsistency in the processed frames (please see the tonal variation on the girl's face). (c) Erratic color on the girl's left eyes in the results of Bonneel *et al.* [3] shows the temporal artifacts. (d) Blurring artifacts on the girl's face expose the drawbacks of the flickering-remove method of Lang *et al.* [10]. (e) Results of our TBLS are temporally consistent.

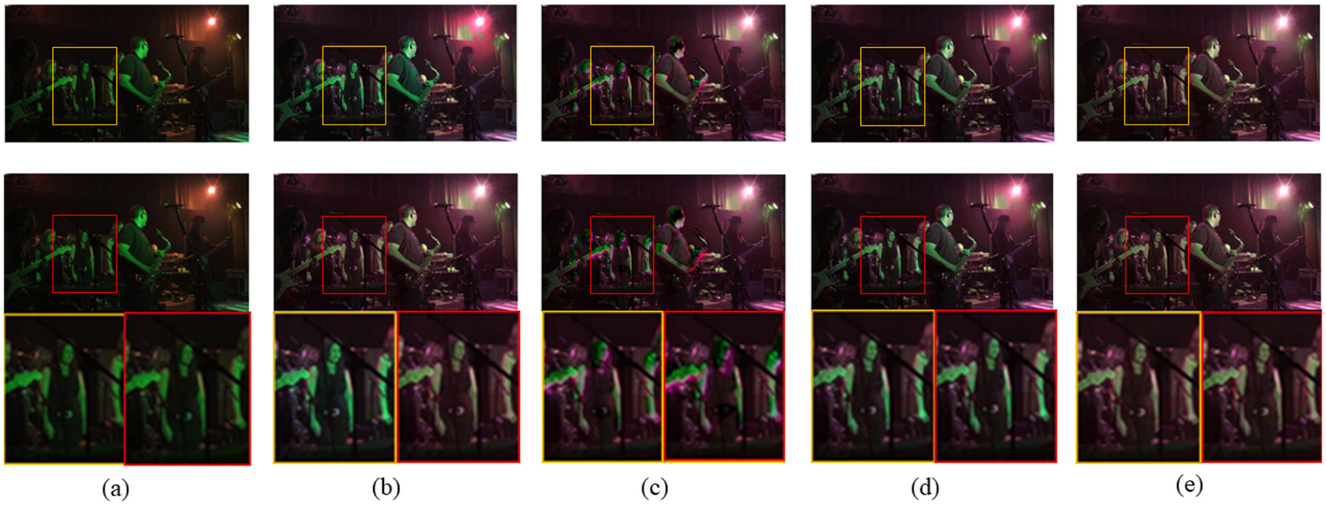


Fig. 11. Experimental results of the *concert* video. The top row is the 4th frame, and the bottom row is the 5th frame. (a) Original frames are temporally consistent. (b) After applying the image-based spatial white balancing method to the original frames, there are obvious color changes with the disappearance of the green color in the processed frames. (c) There are residual temporal artifacts in the results of Bonneel *et al.* [3]. (d) There exists residual temporal inconsistency in the results of Lang *et al.* [10] (please see the color variation on the arms of the people emphasized with the red box). (e) Temporal artifacts are reduced in the results of our proposed TBLS.

as color constancy, spatial white balancing, and style transfer to original videos. In Fig. 10, we use the temporally inconsistent video, obtained from applying the image-based color constancy method to the *zina* video, to train our TBLS system. Compared with the output videos generated from applying the temporal consistency method of Bonneel *et al.* [3] [Fig. 10(c)] and Lang *et al.* [10] [Fig. 10(d)], our TBLS reduces the temporal inconsistency [Fig. 10(e)].

In Fig. 11, we apply the image-based spatial white balancing method to the *concert* video. There are tonal variations in the temporally inconsistent video [Fig. 11(b)]. After applying the temporally consistent results of Bonneel *et al.* [3], there are residual temporal artifacts in the output videos [Fig. 11(c)]. The method of Lang *et al.* [10] can eliminate the temporal inconsistency and the tonal variation but fail to achieve the goal of spatial white balancing [Fig. 11(d)]. Our output video obtained from TBLS is temporally consistent and preserves the content of spatial white balancing in the processed video. In Fig. 12, we utilize the image-based style transferring method to the *old man* video. In Fig. 13, the temporally inconsistent video is obtained from applying the image-based intrinsic image decomposing method to the *house* video. In Fig. 14, the image-based color harmonization method is applied to the *lily* video. In Fig. 15, we apply the image-based HDR tone mapping method to the *checking e-mail* video. Experimental results demonstrate the effectiveness of our proposed TBLS.

D. Quantitative Results

To evaluate the flickering-removal performance of the proposed TBLS approach from an objective perspective, we provide quantitative results of peak signal-noise ratio (PSNR).

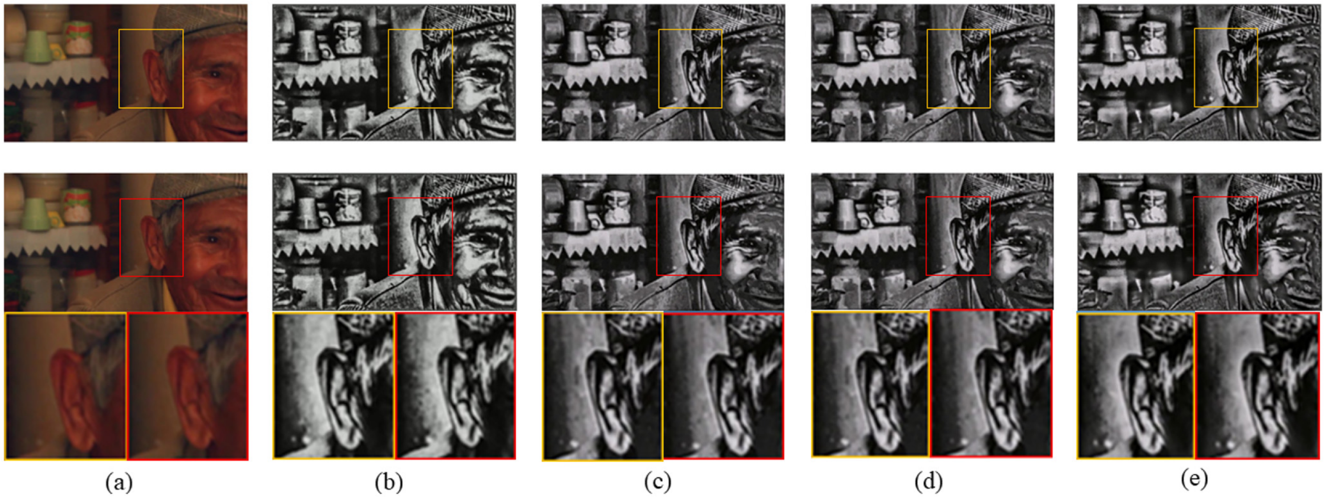


Fig. 12. Experimental results of the *old man* video. The top row is the 11th frame, and the bottom row is the 12th frame. (a) Original frames are temporally consistent. (b) After applying the image-based style transfer method to original frames, the discrepant variation of the shadow area behind the right ear of the older man shows the temporal inconsistency. (c) Applying the method of Bonneel *et al.* [3] cannot eliminate the temporal inconsistency (please see the variation of the area behind the right ear on the close-up view). (d) Results of Lang *et al.* [10] share the residual temporal inconsistency with the result of Bonneel *et al.* [3]. (e) Results of our proposed TBLS.

TABLE I
PSNR ON ENTIRE DATASETS

Video Classes	Number of Videos	Lang [10]	Bonneel [3]	Ours
Male	243	29.33	30.17	**31.48**
Female	182	32.50	34.17	**35.89**
Concert	75	26.63	26.15	**26.68**
Building	198	27.09	27.89	**28.42**
Landscape	147	30.55	29.04	**31.19**
Animal	155	27.91	28.53	**30.14**

The inputs to calculate PSNR are the corresponding temporally inconsistent video and output video. Since the value of PSNR can measure the loss of fidelity when we remove the flickering artifacts in the temporally inconsistent video, and a high value of PSNR can indicate that more dynamic scenes have been retained from the temporally inconsistent video to the output video, it can be used to evaluate the fidelity of the output video. We first show the PSNR results on seven selected videos in Fig. 16. Then, we use our dataset consisting of 1000 short videos divided into six classes: 1) Male; 2) Female; 3) Concert; 4) Building; 5) Landscape; and 6) Animal. The results of PSNR on the entire test set are shown in Table I, where the best result is written in bold for the reader's ease. The number of videos in each video class is also shown in Table I. From the result in Fig. 16 and Table I, it is evident that the output video obtained from our proposed TBLS has the highest PSNR value. Because we can weigh the quality of a video by the value of PSNR, the quality of output video obtained from our TBLS outperforms the flickering-removal methods of both Lang *et al.* [10] and Bonneel *et al.* [3].

We have also applied the value of structural similarity (SSIM) to measure the performance of output videos obtained from applying the video temporal consistency methods, including Bonneel *et al.* [3], Lang *et al.* [10], and our proposed technique, in removing the flickering artifacts. The higher values of SSIM indicate that the results of the output video are better. In the results of SSIM shown in Fig. 17, it can be noted that our proposed video temporal consistency method outperforms the methods proposed by Bonneel *et al.* [3] and Lang *et al.* [10]. To evaluate the flickering-removal effectiveness, we compute the root mean-square error (RMSE) between consecutive frames. From the result shown in Table II, when compared with either of the traditional methods of Lang *et al.* [10] or Bonneel *et al.* [3], the lowest RMSE value and the fastest runtime are obtained by our proposed TBLS. However, when comparing our TBLS with the popular deep learning system (DLS), we can realize similar RMSE values and attribute the approximate result to a similar technique of extracting features. Nonetheless, due to the retraining steps in the DLS, the runtime of our TBLS is quite faster than DLS. The value of RMSE and runtime in Table II demonstrate the effectiveness of our TBLS.

E. User Study

To further evaluate the performance of our proposed TBLS-based flickering removal approach, we invited 15 students (seven males and eight females) to participate in our user study to compare our method with traditional video temporal consistency methods, including the methods of Lang *et al.* [10] and Bonneel *et al.* [3]. All participants were aged from 16 to 22. To the best of our knowledge, they knew nothing about our video temporal consistency research. Since volunteers barely understood the flickering artifacts and nonflickering artifacts, we first displayed the original nonflickering videos that were not processed with any image-based methods. Then, we showed them the temporally inconsistent video and pointed out corresponding flickering artifacts. After the initial two steps, we intended to make volunteers aware of the distinctions between flickering artifacts and nonflickering artifacts. Moreover, we

TABLE II
RMSE AND RUNTIME IN SECONDS

Video	Number of Frames	Frame Size	Lang [10]		Bonneel [3]		DLS		TBLS	
			RMSE	Runtime	RMSE	Runtime	RMSE	Runtime	RMSE	Runtime
interview	196	960×540	2.885	198	2.935	98	2.782	83	**2.620**	**9.2**
zina	74	1024×576	3.823	88	3.857	29.6	**3.003**	27.3	3.059	**2.3**
concert	93	1024×576	2.253	112	2.273	37.2	2.252	33.5	**2.200**	**3.9**
oldman	133	1024×576	7.874	160	8.000	53.4	7.439	45.4	**6.856**	**4.4**
house	88	1024×576	4.472	106	3.606	36.3	**2.435**	29.8	2.450	**3.2**
lily	219	1024×576	4.904	265	4.840	88.7	4.211	87.3	**4.177**	**7.5**
checking email	124	1280×720	3.565	235	4.331	50.6	3.330	47.1	**3.272**	**3.8**

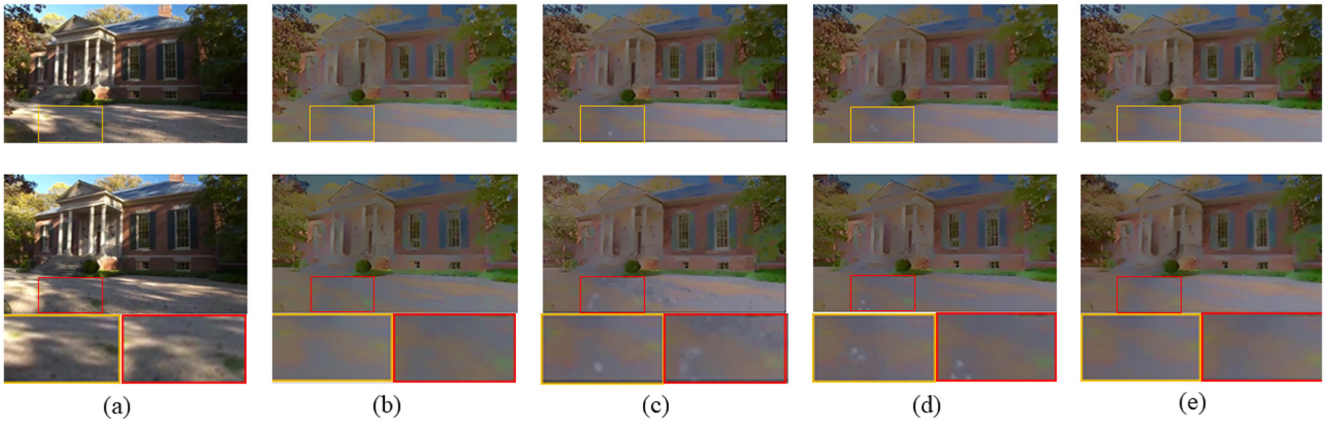


Fig. 13. Experimental results of the *house* video. The top row is the 10th frame, and the bottom row is the 35th frame. (a) Original frames are temporally consistent. (b) After applying the intrinsic decomposition method to original frames, there are temporal inconsistencies in the processed frames (see the brightness variation of the building). (c) There are temporal artifacts on the path to the building (please see the variation of gray blocks emphasized with red box) in the results of Bonneel *et al.* [3]. (d) There are similar temporal artifacts in the results of Lang *et al.* [10]. (e) Results of our proposed TBLS are temporally consistent.

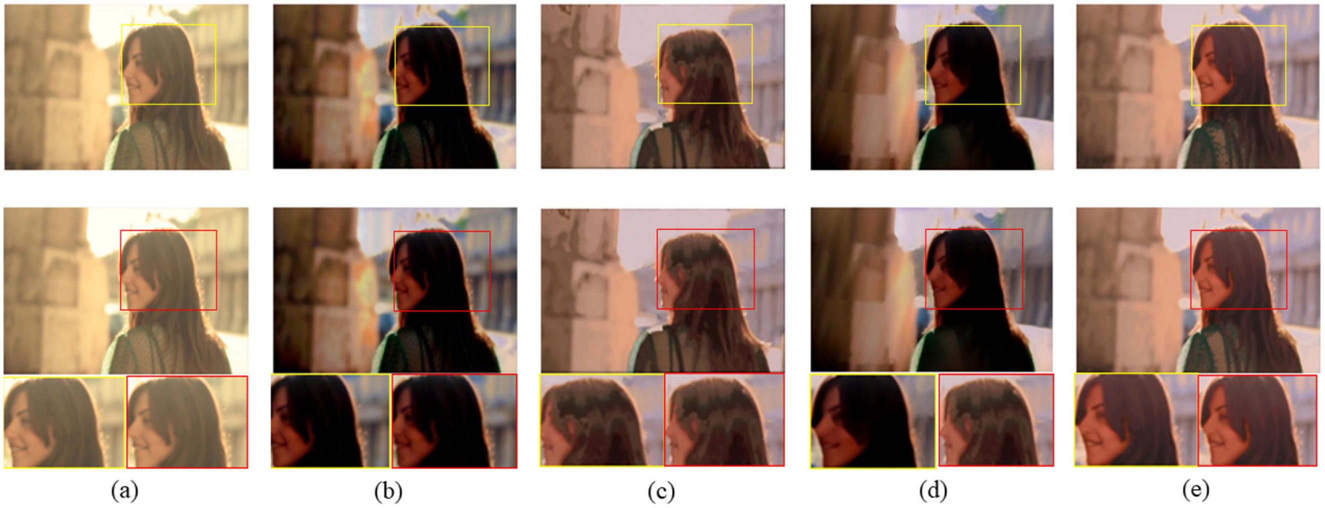


Fig. 14. Experimental results of the *lily* video. The top row is the 202nd frame, and the bottom row is the 203rd frame. (a) Original frames are temporally consistent. (b) Processed frames are temporally inconsistent, which can be seen from the change of the shining blocks on the wall (please see the fluctuation on the corresponding close-up view). (c) There are temporal artifacts in the results of Bonneel *et al.* [3] (please see the blurring blocks on the hair of the girl). (d) There are different blurring artifacts in the results of Lang *et al.* [10]. (e) Results of our proposed TBLS are temporally consistent.

intended to point out the visual effects obtained from corresponding image-based methods. Next, we showed the three output videos processed with different video temporal consistency methods (Lang *et al.* [10], Bonneel *et al.* [3], and ours). We asked the 15 volunteers to select one output video from the three output videos such that the selected output video should be nonflickering and possessed image-based processing effects, such as style transfer or spatial white balancing. To

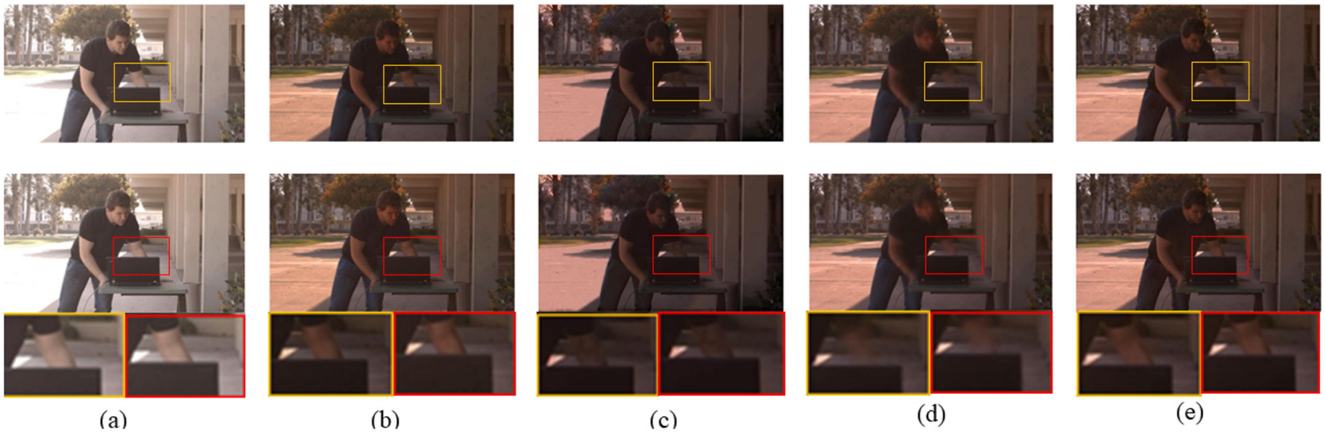


Fig. 15. Experimental results of the *checking e-mail* video. The top row is the 98th frame, and the bottom row is the 99th frame. (a) Original frames are temporally consistent. (b) After applying the method of HDR tone mapping to the original frames, there is texture variation of the ground in the processed frames (please see corresponding close-up views). (c) There are temporal artifacts in the results of Bonneel *et al.* [3] (please see the blurring artifacts on the left arm of the man). (d) Phenomenon of blurring artifacts similar to (c) appears in the results of Lang *et al.* [10]. (e) Temporally consistent results using our proposed TBLS.

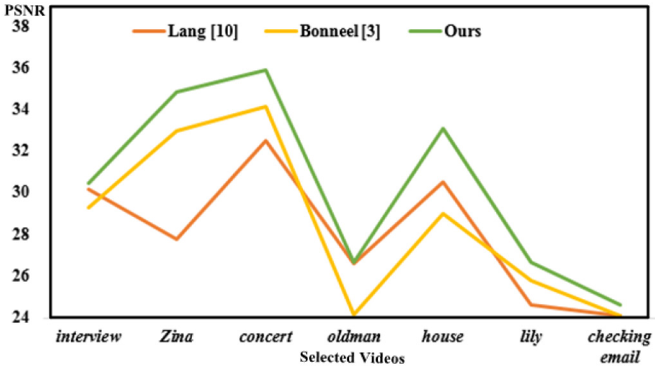


Fig. 16. Results of PSNR on seven selected videos.

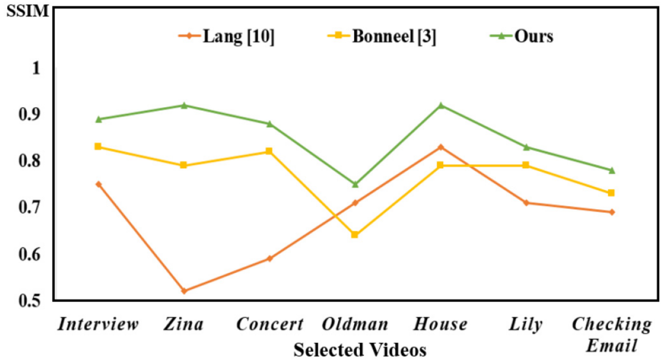


Fig. 17. Results of SSIM on seven selected videos.

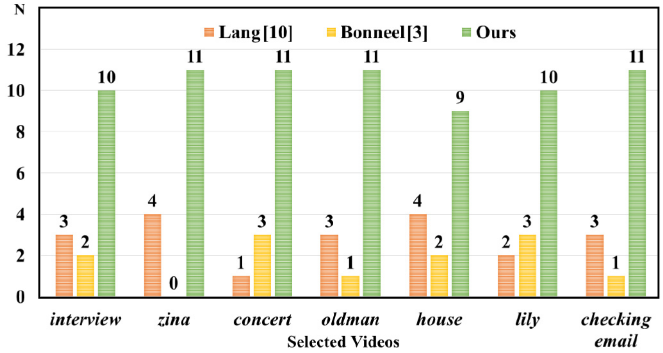


Fig. 18. Results of user study. *N* represents the number of favorable people, and the selected videos include *interview*, *zina*, *concert*, *oldman*, *house*, *lily*, and *checking e-mail*.

make it easier to choose, volunteers were allowed to replay the videos repeatedly (whether the nonflickering original videos, the temporally inconsistent videos, or the output videos). There were seven output videos including the *interview* videos, *zina* videos, *concert* videos, *old man* videos, *house* videos, *lily* videos, and *checking e-mail* videos. Each output video had three copies and was obtained from different video temporal consistency methods. Each volunteer could select one output video from the three copies and select seven output videos in total. The user study result is shown in Fig. 18, where the vertical axis N represents the number of favorable people and the horizontal axis describes the seven selected videos including *interview* video, *zina* video, *concert* video, *old man* video, *house* video, *lily* video, and *checking e-mail* video. It can be noted from Fig. 18 that our method outperforms the methods of Bonneel *et al.* [3] and Lang *et al.* [10].

VII. Conclusion

In this article, we proposed a novel broad learning system-based video temporal consistency approach and construct a TBLS. To train our proposed system, we have established a big video temporal consistency dataset. In the TBLS, we first combine the original frame, the corresponding frame in the temporally inconsistent video, and an output frame of the last original frame as the input data. Then, we map the input data as feature nodes with randomly generated weights. After that, we improve the mapped feature nodes as enhancement nodes with randomly generated weights as well. Finally, we connect the enhancement nodes and mapped features to the output layer with the target weight vector. In the solution process for the target weight vector, we enforce temporal consistency between consecutive frames in the output video. In this way, we replace the retraining process in the DLS with the broad learning system. Additional incremental learning algorithms are also proposed in this article to address the learning inaccuracy caused by the insufficient input data, mapped features, or enhancement nodes. We demonstrated the superiority of our proposed network on a large number of videos. In future work, we intend to preprocess our input data and reduce the features' dimension to accelerate the speed of our approach. We also plan to combine the spirit of both deep learning architecture and the broad learning system.

References

[1] N. Bonneel, K. Sunkavalli, J. Tompkin, D. Sun, S. Paris, and H. Pfister, "Interactive intrinsic video editing," *ACM Trans. Graph.*, vol. 33, no. 6, p. 197, 2014.

[2] G. Ye, E. Garces, Y. Liu, Q. Dai, and D. Gutierrez, "Intrinsic video and applications," *ACM Trans. Graph.*, vol. 33, no. 4, p. 80, 2014.

[3] N. Bonneel, J. Tompkin, K. Sunkavalli, D. Sun, S. Paris, and H. Pfister, "Blind video temporal consistency," *ACM Trans. Graph.*, vol. 34, no. 6, p. 196, 2015.

[4] X. Dong, B. Bonev, Y. Zhu, and A. L. Yuille, "Region-based temporally consistent video post-processing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 714–722.

[5] Z. Farbman and D. Lischinski, "Tonal stabilization of video," *ACM Trans. Graph.*, vol. 30, no. 4, p. 89, 2011.

[6] C.-M. Wang, Y.-H. Huang, and M.-L. Huang, "An effective algorithm for image sequence color transfer," *Math. Comput. Model.*, vol. 44, nos. 7–8, pp. 608–627, 2006.

[7] N. Bonneel, K. Sunkavalli, S. Paris, and H. Pfister, "Example-based video color grading," *ACM Trans. Graph.*, vol. 32, no. 4, p. 39, 2013.

[8] C.-R. Huang, K.-C. Chiu, and C.-S. Chen, "Temporal color consistency-based video reproduction for dichromats," *IEEE Trans. Multimedia*, vol. 13, no. 5, pp. 950–960, Oct. 2011.

[9] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski, "Optimizing color consistency in photo collections," *ACM Trans. Graph.*, vol. 32, no. 4, p. 38, 2013.

[10] M. Lang, O. Wang, T. Aydin, A. Smolic, and M. H. Gross, "Practical temporal consistency for image-based graphics applications," *ACM Trans. Graph.*, vol. 31, no. 4, p. 34, 2012.

[11] N. K. Kalantari, E. Shechtman, C. Barnes, S. Darabi, D. B. Goldman, and P. Sen, "Patch-based high dynamic range video," *ACM Trans. Graph.*, vol. 32, no. 6, p. 202, 2013.

[12] T. O. Aydin, N. Stefanoski, S. Croci, M. Gross, and A. Smolic, "Temporally coherent local tone mapping of HDR video," *ACM Trans. Graph.*, vol. 33, no. 6, p. 196, 2014.

[13] S. Feng and C. L. P. Chen, "A fuzzy restricted Boltzmann machine: Novel learning algorithms based on the crisp possibilistic mean value of fuzzy numbers," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 1, pp. 117–130, Feb. 2018.

[14] C. L. P. Chen, C.-Y. Zhang, L. Chen, and M. Gan, "Fuzzy restricted Boltzmann machine for the enhancement of deep learning," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 6, pp. 2163–2173, Dec. 2015.

[15] Z. Yu, L. Li, J. Liu, and G. Han, "Hybrid adaptive classifier ensemble," *IEEE Trans. Cybern.*, vol. 45, no. 2, pp. 177–190, Feb. 2015.

[16] Z. Yu, H. Chen, J. Liu, J. You, H. Leung, and G. Han, "Hybrid *k*-nearest neighbor classifier," *IEEE Trans. Cybern.*, vol. 46, no. 6, pp. 1263–1275, Jun. 2016.

[17] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.

[18] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[19] R. Salakhutdinov and G. Hinton, "Deep Boltzmann machines," in *Proc. Int. Conf. Artif. Intell. Stat.*, vol. 5, 2009, pp. 448–455.

[20] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.

[22] Y.-H. Pao and Y. Takefuji, "Functional-link net computing: Theory, system architecture, and functionalities," *Computer*, vol. 25, no. 5, pp. 76–79, 1992.

[23] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, "Learning and generalization characteristics of the random vector functional-link net," *Neurocomputing*, vol. 6, no. 2, pp. 163–180, 1994.

[24] S. Chen, C. F. N. Cowan, and P. M. Grant, "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Trans. Neural Netw.*, vol. 2, no. 2, pp. 302–309, Mar. 1991.

[25] C. L. P. Chen and Z. Liu, "Broad learning system: An effective and efficient incremental learning system without the need for deep architecture," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.

[26] C. L. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Inf. Sci.*, vol. 275, pp. 314–347, Aug. 2014.

[27] C. L. P. Chen and Z. Liu, "Broad learning system: A new learning paradigm and system without going deep," in *Proc. Youth Acad. Annu. Conf. Chin. Assoc. Autom.*, 2017, pp. 1271–1276.

[28] Z. Liu, J. Zhou, and C. L. P. Chen, "Broad learning system: Feature extraction based on K-means clustering algorithm," in *Proc. Int. Conf. Inf. Cybern. Comput. Soc. Syst.*, 2017, pp. 683–687.

[29] Z. Liu and C. L. P. Chen, "Broad learning system: Structural extensions on single-layer and multi-layer neural networks," in *Proc. Int. Conf. Security Pattern Anal. Cybern.*, 2017, pp. 136–141.

[30] M. Xu, M. Han, C. L. P. Chen, and T. Qiu, "Recurrent broad learning systems for time series prediction," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1405–1417, Apr. 2020.

[31] B. Sheng, P. Li, Y. Zhang, L. Mao, and C. L. P. Chen, "GreenSea: Visual soccer analysis using broad learning system," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1463–1477, Mar. 2021.

[32] S. Feng and C. L. P. Chen, "Fuzzy broad learning system: A novel neuro-fuzzy model for regression and classification," *IEEE Trans. Cybern.*, vol. 50, no. 2, pp. 414–424, Feb. 2020.

[33] J. Du, C. M. Vong, and C. L. P. Chen, "Novel efficient RNN and LSTM-like architectures: Recurrent and gated broad learning systems and their applications for text classification," *IEEE Trans. Cybern.*, vol. 51, no. 3, pp. 1586–1597, Mar. 2021.

[34] H. Guo, B. Sheng, P. Li, and C. L. P. Chen, "Multiview high dynamic range image synthesis using fuzzy broad learning system," *IEEE Trans. Cybern.*, vol. 51, no. 5, pp. 2735–2747, May 2021.

[35] R. Ali *et al.*, "Optic disk and cup segmentation through fuzzy broad learning system for glaucoma screening," *IEEE Trans. Ind. Informat.*, vol. 17, no. 4, pp. 2476–2487, Apr. 2021.

[36] H. Ye, H. Li, and C. L. P. Chen, "Adaptive deep cascade broad learning system and its application in image denoising," *IEEE Trans. Cybern.*, early access, Mar. 23, 2020, doi: 10.1109/TCYB.2020.2978500.

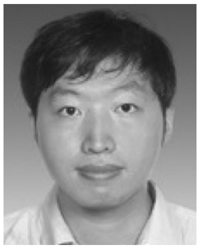
[37] J. Hu, M. Wu, L. Chen, K. Zhou, P. Zhang, and W. Pedrycz, "Weighted kernel fuzzy C-means-based broad learning model for time-series prediction of carbon efficiency in iron ore sintering process," *IEEE Trans. Cybern.*, early access, Dec. 9, 2020, doi: 10.1109/TCYB.2020.3035800.

[38] W. Xia, J. Zhang, and U. Kruger, "Semisupervised pedestrian counting with temporal and spatial consistencies," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1705–1715, Aug. 2015.

[39] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 32, no. 2, pp. 407–499, 2004.

[40] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, May 2011.

[41] C.-H. Yao, C.-Y. Chang, and S.-Y. Chien, "Occlusion-aware video temporal consistency," in *Proc. ACM Multimedia*, 2017, pp. 777–785.



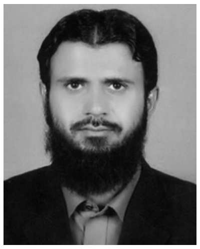
Bin Sheng (Member, IEEE) received the B.A. degree in English and the B.Eng. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China, in 2004, the M.Sc. degree in software engineering from the University of Macau, Macau, China, in 2007, and the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong, in 2011.

He is currently a Full Professor with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. His current research interests include computer vision, broad learning systems, deep learning, virtual reality, and computer graphics.

Prof. Sheng is an Associate Editor of the IEEE Transactions on Circuits and Systems for Video Technology.

Ping Li (Member, IEEE) received the Ph.D. degree in computer science and engineering from the Chinese University of Hong Kong, Hong Kong, in 2013.

He is currently a Research Assistant Professor with the Hong Kong Polytechnic University, Hong Kong. He has one image/video processing national invention patent and has excellent research project reported worldwide by *ACM TechNews*. His current research interests include image/video stylization, artistic rendering and synthesis, and creative media.



Riaz Ali received the B.Eng. degree in software engineering from the Mehran University of Engineering and Technology, Jamshoro, Pakistan, in 2010, and the M.Eng. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2020.

He is currently a Lecturer with the Department of Computer Science, Sukkur IBA University, Sukkur, Pakistan. His current research interests include video temporal consistency, broad learning systems, and computer vision.

C. L. Philip Chen (Fellow, IEEE) received the Master degree from the University of Michigan at Ann Arbor, Ann Arbor, MI, USA, in 1985, and the Ph.D. degree in electrical engineering from Purdue University, West Lafayette, IN, USA, in 1988.

He is currently a Chair Professor and the Dean with the School of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Being a Program Evaluator of the Accreditation Board of Engineering and Technology Education in the U.S., for computer engineering, electrical engineering, and software engineering programs, he successfully architects the University of Macau's Engineering and Computer Science programs receiving accreditations from Washington/Seoul Accord through Hong Kong Institute of Engineers (HKIE), of which is considered as his utmost contribution in engineering/computer science education for Macau as the former Dean of the Faculty of Science and Technology. His current research interests include systems, cybernetics, and computational intelligence.

Dr. Chen was a recipient of the 2016 Outstanding Electrical and Computer Engineers Award from his alma mater, Purdue University (in 1988). He received the IEEE Norbert Wiener Award in 2018 for his contribution in systems and cybernetics, and machine learnings. He is also a highly cited researcher by Clarivate Analytics in 2018 and 2019. He was the IEEE Systems, Man, and Cybernetics Society President from 2012 to 2013, and the Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS from 2014 to 2019. He is currently the Editor-in-Chief of the IEEE TRANSACTIONS ON CYBERNETICS, and an Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS. He was the Chair of TC 9.1 Economic and Business Systems of International Federation of Automatic Control from 2015 to 2017, and currently is a Vice President of Chinese Association of Automation. He is a Fellow of AAAS, IAPR, CAA, and HKIE; a member of Academia Europaea, European Academy of Sciences and Arts, and International Academy of Systems and Cybernetics Science.