STSP: Spatial-Temporal Subspace Projection for Video Class-incremental Learning

Hao Cheng[1,2], Siyuan Yang[1], Chong Wang[1], Joey Tianyi Zhou[3,4], Alex C. Kot[1], and Bihan Wen[1⋆]

[1] Nanyang Technological University, Singapore
{siyuan.yang,wang1711,eackot,bihan.wen}@ntu.edu.sg
[2] Hebei University of Technology, Tianjin, China
2024025@hebut.edu.cn
[3] CFAR, Agency for Science, Technology and Research, Singapore
[4] IHPC, Agency for Science, Technology and Research, Singapore
zhouty@cfar.a-star.edu.sg

Abstract. Video class-incremental learning (VCIL) aims to learn discriminative and generalized feature representations for video frames to mitigate catastrophic forgetting. Conventional VCIL methods often retain a subset of frames or features from prior tasks as exemplars for subsequent incremental learning stages. However, these strategies overlook the connection between base and novel classes, sometimes even leading to privacy leakage. To address this challenge, we introduce a Spatial-Temporal Subspace Projection (STSP) scheme for VCIL. Specifically, we propose a discriminative Temporal-based Subspace Classifier (TSC) that represents each class with an orthogonal subspace basis and adopts subspace projection loss for classification. Unlike typical classification methods relying on fully connected layers, our TSC discerns the spatial-temporal dynamics in video content, thereby enhancing the representation of each video sample. Additionally, we implement inter- and intra-class orthogonal constraints into TSC, ensuring each class occupies a unique orthogonal subspace, defined by its basis. To prevent catastrophic forgetting, we further employ a Spatial-based Gradient Projection (SGP) strategy, adjusting network gradients to align with the null space of the spatial feature set from previous tasks. Extensive experiments conducted on three benchmarks, namely HMDB51, UCF101, and Something-Something V2, demonstrate that our STSP method outperforms state-of-the-art comparison methods, evidencing its efficacy in VCIL.

Keywords: Video action recognition · Class incremental learning · Subspace Projection · Gradient Projection

1 Introduction

Deep neural networks have demonstrated remarkable performance in modern computer vision tasks [7, 17, 32, 37, 46], particularly when trained on large, annotated datasets. However, the real-world challenge of incrementally incorporating

⋆ Corresponding author.

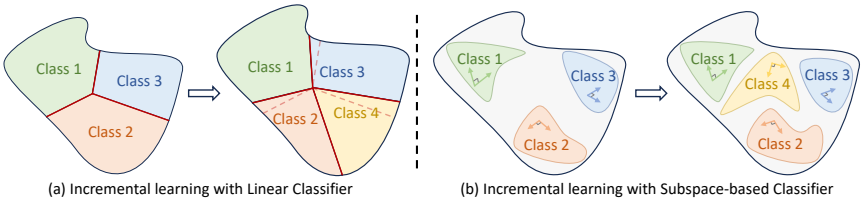


Fig. 1: Class incremental learning with (a) linear classifier and (b) subspace-based classifier. Incorporating new classes alters the decision hyperspace and modifies the representation of prior classes, leading to misalignment. In contrast, subspace-based classifiers, by constructing an orthogonal basis for each class, establish a unique geometric structure for each class, thereby reducing class misalignment.

data from new classes into a model initially trained on existing classes presents significant hurdles, known as class-incremental learning (CIL) [26]. Unlike typical classification tasks, CIL involves accessing novel classes and limited memories of prior tasks in incremental sessions, often due to privacy concerns or memory constraints. Consequently, merely updating the model with data from new classes can result in overfitting substantial performance degradation on previous classes, a phenomenon referred to as catastrophic forgetting [19, 20].

Recent advancements in CIL methods [1,4,26,44] have shown remarkable performance in image classification tasks. However, recent CIL studies [1, 21, 25, 42] demonstrate that straightforward fine-tuning of the network for novel classes may lead the subspace spanned by the novel classifier to deviate from the base one, as illustrated in Fig. 1 (a). The classification performance can be significantly affected by misalignment, particularly in cases where a notable distribution gap between novel and base classes. Moreover, when extending this task to Video Class-Incremental Learning (VCIL), unique challenges due to the inherent information redundancy within video samples, such as continuous actions with a static background. This necessitates methods to model temporal-spatial relationships between video frames to extract critical information for action recognition properly. However, prevalent approaches [22–24, 35, 36] in VCIL predominantly involve a learning pipeline that employs simple operations (*e.g.*, mean) on the feature vectors of sampled frames to generate video representations for action recognition. While this methodology is effective for standard action recognition with sufficient data, it may struggle to ensure robust generalization in scenarios of incremental learning. The aforementioned observations prompt us to consider acquiring a more suitable representation for both the classifier and video features while also exploring their relationship. To address these challenges, several subspace learning-based methods [28,33,38,45] have shown effectiveness in reducing dimensionality and capturing intrinsic structure, thereby offering a potential solution to the issues associated with simplistic video representation methods.

In this paper, we propose a spatial-temporal subspace projection method for VCIL including a temporal-based subspace classifier (TSC) and a spatial-based gradient projection (SGP) strategy. Specifically, we extend the existing linear

classifiers based on fully connected (FC) layers to a TSC in which a subspace basis is adopted to model each class. Differing from linear classifiers that seek to distinguish classes in feature space through a linear decision boundary, the proposed orthogonality constraints offer a distinct geometric structure for each class. This characteristic not only facilitates a more comprehensive representation of each class but also allows for straightforward extension to novel classes. Furthermore, to avoid catastrophic forgetting, we utilize the SGP strategy to constrain the process of gradient update by projecting the gradient of network parameters in each layer onto the subspace of corresponding input features so that losses for the old task will not increase during the incremental sessions. Compared with existing VCIL methods [22–24,35,36] that rely on feature distillation or prompt-based fine-tuning strategies, SGP offers the advantage of fine-tuning all network parameters without necessitating regularizers. Furthermore, SGP obviates the need to explicitly retain class-relevant information, such as frames or prototypes, thereby contributing to privacy protection.

In summary, we make the following contributions: 1) We propose a spatial-temporal subspace projection (STSP) method, containing a temporal-based subspace classifier and spatial-based gradient projection strategy to better model feature and class representation for discriminative classification. 2) We employ a spatial-based gradient projection strategy to constrain the gradient update process without the need to store frames or prototypes from previous tasks. This not only enhances performance but also mitigates the risk of privacy leakage. 3) The proposed STSP method achieves superior results on three video action recognition incremental benchmarks over state-of-the-art methods.

2 Related Work

2.1 Video Action Recognition

With significant advances in deep learning, current video action recognition approaches aim to learn effective spatial-temporal representation by well-designed network architectures. A notable advancement by Simonyan and Zisserman [29] introduces a dual-stream architecture that analyzes RGB frames in one stream and optical flow in another, laying the foundation for subsequent enhancements. Recently, Lin *et al.* [17] proposed a Temporal Shift Module (TSM), which shifts some channels along the temporal dimension to perform temporal interaction between the features from adjacent frames. Concurrently, significant research [3, 13,14,31,32,41] has evolved 2D Convolutional Neural Networks (CNNs) into 3D structures to capture both spatial and temporal contexts within videos, a critical aspect of human action recognition. Notably, Tran *et al.* [31] introduced the C3D model, a 3D CNN designed to learn spatial-temporal features through an end-to-end learning approach. Carreira *et al.* [3] developed the Inflated 3D CNN, which extends a 2D CNN into the temporal domain by inflating its convolutional and pooling layers. Additionally, Feichtenhofer *et al.* [9] crafted a dual-pathway 3D CNN, featuring both slow and fast pathways that process RGB frames at varying frame rates to simultaneously capture semantic content and motion details.

Despite significant progress in action recognition, catastrophic forgetting has received relatively limited attention. This work aims to address this gap by developing a spatial-temporal subspace projection method, which aims to offer a robust solution to the challenges of incremental learning in action recognition.

2.2 Class-Incremental Learning

CIL addresses the challenge of adapting machine learning models to recognize new classes without forgetting previously learned information, a phenomenon known as catastrophic forgetting. Pioneering in this domain, Rebuffi *et al.* [26] introduced iCaRL, which combines representation learning with a memory mechanism to selectively retain the knowledge of previous classes. Following this, Hou *et al.* [12] proposed a UCIR strategy, which addresses the class imbalance in CIL by adjusting the classification layer to focus more on newer classes while still remembering old ones. Moreover, Gao *et al.* [10] introduced R-DFCIL, employing relation-guided representation learning to overcome catastrophic forgetting and enable the assimilation of new classes in a data-free scenario.

While CIL has been well-studied in the image domain, its application to the video domain remains largely unexplored. Park *et al.* [22] introduced Time-Channel Distillation (TCD), leveraging knowledge distillation loss with time-channel importance masks. Pei *et al.* [23] proposed FrameMaker, a memory-efficient VCIL strategy that develops a condensed frame for each video to facilitate future incremental tasks with less memory usage. Pei *et al.* [24] introduced ST-Prompt, a novel space-time prompting framework tailored for VCIL, enhancing the model's adaptability to new classes with minimal memory increase.

Existing VCIL approaches often rely on preserving a subset of frames or features from previous tasks as exemplars for subsequent incremental stages to prevent catastrophic forgetting. However, these strategies overlook the connection between base and novel classes and potentially lead to privacy leakage. In contrast, we employ a spatial-based gradient projection to regulate gradient updates, thereby mitigating catastrophic forgetting without these drawbacks.

2.3 Subspace Representation

Subspace learning, by learning a suitable representation for high dimensional data, has shed light on various vision tasks [2, 18, 34, 40, 43]. Existing subspace methods such as Principal Component Analysis (PCA) [2, 18] and Linear Discriminant Analysis (LDA) [2, 40] adaptively transfer the original input data to low dimensional subspace for dimensionality reduction. For example, Tzimiropoulos *et al.* [34] formulated PCA on image gradient orientations, which demonstrated robust performance in face recognition. Recently, subspace learning has been further extended to regularize the implicit features or learned classifiers within deep networks [1,28,38,45]. Notably, Cheraghian *et al.* [5] constructed a mixture of subspaces to represent the feature clusters formed by visually similar samples. Such subspace representation enables a better approximation of the overall data distribution, consequently leading to superior performance in

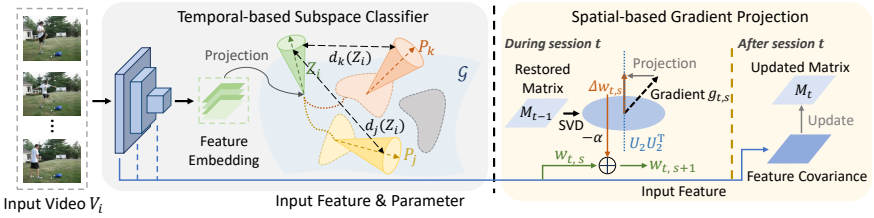


Fig. 2: The overall framework of the proposed STSP method, including a *temporal-based subspace classifier (TSC)* and a *spatial-based gradient projection (SGP)*. For each input video, spatial-temporal features are extracted using a backbone network. Subsequently, The TSC classifies by projecting temporal features into a subspace, where classification decisions are based on the distance to each class's subspace basis within this subspace. During parameter updates for the novel tasks, the SGP constrains gradient modifications for each specific layer, drawing on the mean covariance matrix M_{t-1} of previous tasks' input features. Upon the current session completion, the mean covariance matrix is updated to M_t for the gradient projection in the subsequent session.

CIL. To mitigate the catastrophic forgetting and overfitting, Simon *et al.* [28] formulated the dynamic classifiers via subspace modeling which demonstrated a significant robustness to noise in few-shot learning. Similarly, Akyurek *et al.* [1] proposed an additional subspace regularizer that pulls the weight vectors of the novel class close to the subspace spanned by the base classes.

However, existing methods lack sufficient constraints on the representation of each class in the classifier. In this paper, we model each class via an orthogonal subspace basis and leverage spatial-based gradient projection to update the entire network during incremental sessions.

3 Method

In this section, we start with an introduction of VCIL problem formulation. Then, the proposed STSP method is described and explained in detail, followed by the loss function of our method as well as the training and inference procedure.

In VCIL, a series of distinct tasks $\{\mathcal{T}_1, \mathcal{T}_2, \cdots, \mathcal{T}_k, \cdots\}$ unfolds sequentially. Each task $\mathcal{T}_k$ comprises a distinct subset of video-class pairs $\{\left(D_i^k, y_i^k\right), y_i^k \in \mathcal{C}_k\}_{i=1}^{|\mathcal{T}_k|}$. Specifically, the novel class set $\mathcal{C}_k$ of $\mathcal{T}_k$ is distinct from previous classes, *i.e.*, $(\mathcal{C}_1 \cup \cdots \cup \mathcal{C}_{k-1}) \cap \mathcal{C}_k = \emptyset$. The primary objective is to incrementally train a deep neural network $f(\cdot)$ capable of accurately classifying videos across all classes by preventing the loss of class information from previous tasks. In contrast to methods that have access to a limited exemplar set from historical tasks, our emphasis lies on exemplar-free VCIL, where the model is not permitted to store exemplars from previous tasks during the learning process of the current session.

3.1 Overall STSP Framework

Given the problem formulation, we follow the standard learning protocol of previous methods [22,23] based on knowledge distillation. During the k^{th} incremental session, the network $f(\cdot\,;\mathbf{w})$ parameterized by $\mathbf{w}$ is updated from $\mathbf{w}_{k-1}$ to adapt to the novel class set $\mathcal{C}_k$. Fig. 2 illustrates the overall framework of our method. For an input video D_i^k belonging to $\mathcal{C}_k$, we first extract T frames as V_i, which are then fed to the backbone network to generate the feature embeddings. Subsequently, we employ a TSC to differentiate between distinct classes. Moreover, our method stands out by employing an SGP strategy without restoring frames or prototypes from previous tasks. More in detail, the network parameters are optimized by projecting the gradients in each layer to the subspace corresponding to their input features. With the merit of this strategy, we can completely avoid the risk of privacy leakage, whereas existing approaches may suffer.

3.2 Temporal-based Subspace Classifier

In accordance with the standardized procedure for VCIL, given an input video-class pair denoted as (D_i, y_i), we first randomly sampled T frames V_i^T. Subsequently, these frames are fed to the backbone network, yielding the feature embeddings as $X_i \in \mathbb{R}^{T\times C\times H\times W}$, where C, H, and W represent the size of channel, height, and width, respectively. For the existing architectures [22–24] for VCIL, a prevalent technique for obtaining video representations is global average pooling (GAP), denoted as $Z_i = GAP(X_i) \in \mathbb{R}^{C\times T}$. Following this, a linear classifier L, comprising an FC layer with weights $\mathbf{w} \in \mathbb{R}^{C\times K}$, is utilized to predict the potential class of each frame. The class predictions from individual frames are subsequently averaged to produce the overall classification result.

Nevertheless, the temporal information holds the potential to encapsulate more intricate intra-frame structural details, warranting thoughtful consideration and seamless integration into the classification process. Additionally, in each incremental task, adjustments to the linear classifier may be necessary to modify the overall representation across all classes, which could influence the base classes for all previous incremental sessions. Conversely, a subspace-based classifier demonstrates heightened adaptability, enabling the modeling of each new class within its unique subspace without imposing substantial effects on the existing representations of previous classes.

To this end, to properly model each class and learn a robust classifier, we construct a TSC based on subspaces spanned by an orthogonal basis of each class. Specifically, for m-th class, we construct a subspace with orthogonal basis $P_m \in \mathbb{R}^{C\times\tau}$ with $P_m^\top P_m = I_\tau, \tau \ll C$, where C and τ indicates the feature dimension and the number of subspace basis for m-th class, respectively. Hence, for each task $\mathcal{T}_k$ in the k^{th} incremental session, we construct $|\mathcal{C}_k|$ distinct subspaces for $|\mathcal{C}_k|$ novel classes. We then adopt current subspaces to predict the class of Z_i based on the subspace distance between the subspace spanned by temporal video features Z_i and the orthogonal basis of each class as:

$$d_m(Z_i) = -\left\|Z_i Z_i^\top - P_m P_m^\top\right\|_F^2. \tag{1}$$

Then, the probability of Z_i assigned to m-th class can be calculated using the softmax function as follows:

$$p_m(Z_i) = \frac{e^{d_m(Z_i)}}{\sum_{m'=1}^{M} e^{d_{m'}(Z_i)}}, \tag{2}$$

where M is the total number of classes in the current session. Now, we can employ the cross-entropy loss on Eq. (2) to update the network as:

$$L_{CE} = -\sum_{i=1}^{J} y_i \log p_i(Z_i), \tag{3}$$

where J denotes the number of videos in each mini-batch.

To encourage the network to capture discriminative information for each class and construct distinct subspaces corresponding to different classes, we impose orthogonal constraints on both the temporal features of the video and the weights of the TSC. Specifically, we employ orthogonal loss on the temporal features Z_i of each video D_i in the final layer as:

$$L_{OF} = \sum_{i=1}^{J} \left\| Z_i^\top Z_i - I_T \right\|_F^2, \tag{4}$$

where $I_T \in \mathbb{R}^{T\times T}$ is an identity matrix. This approach guarantees that every frame independently captures distinct, class-aware information, setting it apart uniquely from the rest. Furthermore, it facilitates the construction of a discriminative subspace for each corresponding class.

For TSC, we impose intra- and inter-class orthogonal constraints on the subspace basis of each class, which is denoted by L_{OW} as:

$$L_{OW} = \sum_{m=1}^{M} (\underbrace{\left\| P_m^\top P_m - I_\tau \right\|_F^2}_{\text{intra-class}} + \sum_{q=m+1}^{M} \underbrace{\left\| P_m^\top P_q \right\|_F^2}_{\text{inter-class}}). \tag{5}$$

The intra-class orthogonal constraint is introduced to ensure independence among the basis vectors of subspaces, which is crucial for learning distinct and discriminative feature representations. Furthermore, the imposition of inter-class orthogonal constraint ensures that the basis vectors of distinct classes are orthogonal, signifying their mutual independence. This orthogonality enforcement serves to prevent confusion and safeguard the uniqueness of each basis vector, thus preserving the integrity of class information previously learned. Moreover, these orthogonal constraints enhance discriminability, guaranteeing that the basis vectors of subspaces related to different classes are clearly distinguishable.

Therefore, the overall objective function for TSC is defined as:

$$L_{total} = \alpha L_{CE} + \beta L_{OF} + \gamma L_{OW}, \tag{6}$$

where α, β, and γ denote the weighting parameters of each losses, respectively.

3.3 Spatial-based Gradient Projection

In contrast to prior approaches [22–24] that rely on retaining exemplars or prompts from previous classes during incremental sessions, our focus is on addressing the challenging exemplar-free VCIL problems by imposing constraints on the network during parameter updates. To avoid catastrophic forgetting on previous classes in the incremental sessions, the network's parameter update should satisfy the following proposition:

Proposition 1. *Given the data x_k from the previous task $\mathcal{T}_k$, the current network $f(\cdot;\mathbf{w}_t)$ could preserve the historical information if it satisfies the following equation for all sessions:*

$$f\left(x_k;\mathbf{w}_t\right)=f\left(x_k;\mathbf{w}_k\right),\qquad \forall\ k<t. \tag{7}$$

Here, $\mathbf{w}_t$ denotes the network parameters at task $\mathcal{T}_t$.

To realize the objective presented in Proposition 1, it is imperative to confine the gradient update of network parameters to a specific direction—precisely, the gradient update should lie in the subspace spanned by the corresponding input spatial features. Specifically, the gradients in the linear layers, including FC and Convolution layers [5] should satisfy Proposition 2 [27, 39].

Proposition 2. *Let $f^l(\cdot;\mathbf{w}_t)$ denotes the l-th linear layer of the network trained on current task $\mathcal{T}_t$ and its corresponding input feature A_k^l from any previous task $\mathcal{T}_k$ $(k<t)$. During each training step s, if the linear layer's parameter update $\Delta\mathbf{w}_{t,s}^l$ lies in the null space of A_k^l,* i.e.*,*

$$A_k^l\Delta\mathbf{w}_{t,s}^l=0, \tag{8}$$

we could preserve the historical information as $f^l\left(A_k^l;\mathbf{w}_t^l\right)=f^l\left(A_k^l;\mathbf{w}_k^l\right)$.

Building upon Proposition 2, we further extend it to our proposed TSC with Eq. (1), as detailed in Proposition 3.

Proposition 3. *Let $f^{TSC}(\cdot;\mathbf{w}_t)$ denotes the TSC of the network trained on current task $\mathcal{T}_t$ and its corresponding input feature Z_k from any previous task $\mathcal{T}_k$ $(k<t)$. During each training step s, if the TSC parameter update $\Delta\mathbf{w}_{t,s}^{TSC}$ lies in the null space of the subspace spanned by Z_k,* i.e.*,*

$$Z_kZ_k^\top\Delta\mathbf{w}_{t,s}^{TSC}=0, \tag{9}$$

similarly we have $f^{TSC}\left(Z_k;\mathbf{w}_t^{TSC}\right)=f^{TSC}\left(Z_k;\mathbf{w}_k^{TSC}\right)$.

Proof. Please refer to the ***supplemental material****.*

[5] The convolution operation can be regarded as a specific type of linear operation.

Inspired by previous work [27, 39], we calculate the mean covariance matrix, denoted as M_{t-1}, of the input feature for each specific layer in each mini-batch to enhance network stability and optimize memory efficiency. Here, let Q represent the collective term for the current input feature at each specific layer, with A_t^l for the l-th linear layer and $Z_t Z_t^\top$ for the TSC, respectively. To initialize, M_0 is set to $\mathbf{0}$ for the first session. At the end of each session t, M_t is updated as:

$$M_t = M_{t-1} + \sum_{b=1}^{\mathcal{B}} \frac{1}{J} Q_b^\top Q_b, \tag{10}$$

where $\mathcal{B}$ represents the number of mini-batches in the current task, and J denotes the number of videos in each mini-batch.

During the feature update in task $\mathcal{T}_t$, we perform Singular Value Decomposition (SVD) to find the approximate null space of the given feature matrix M_{t-1} corresponding to Proposition 2 and Proposition 3 as:

$$\begin{bmatrix} U_1, U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} U_1^\top \\ U_2^\top \end{bmatrix} = \mathrm{SVD}(M_{t-1}), \tag{11}$$

where U_1, U_2 represent unitary matrices that comprise orthonormal bases associated with the singular values in Σ_1, Σ_2, respectively, with all zero singular values relegated to Σ_2. However, it is unrealistic to guarantee that there always exists zero singular values. Previous work [27, 39] constructs $U_2^\top$ by selecting singular values restricted to a specific proportion. However, this strategy might not always provide the most compact representation, particularly when there is a significant difference in the scale of singular values. Hence, we employ a threshold strategy based on the energy retained in the matrix M for the construction $U_2^\top$, corresponding to a segment of smallest singular values that satisfy $\|\Sigma_2\|_F^2 \leq (1-\eta)\left(\|\Sigma_1\|_F^2 + \|\Sigma_2\|_F^2\right)$, with the proportion of retained energy denoted by η. When $\eta \approx 1$, it is reasonable to approximate the null space of the matrix M with $U_2^\top$. This enables the parameter update $\Delta\mathbf{w}$ from the gradient $\mathbf{g}$ concerning the parameters $\mathbf{w}$ through subspace projection based on U_2. That is, during the training step s within any incremental session t, we have:

$$\Delta\mathbf{w}_{t,s} = U_2 U_2^\top \mathbf{g}_{t,s}. \tag{12}$$

3.4 Model Optimization and Inference

Our STSP method is summarized in Algorithm 1. In the training phase, the network parameters are updated conventionally during the initial session. After this initial session, considering parameter updates, the mean covariance of input features from the base classes is retained for each layer. These retained features are then utilized to update parameters according to Eq. (12) at each step in every subsequent incremental session. Additionally, after each incremental session, these parameters are further updated with features extracted from novel classes.

For the testing phase, no additional memories or operations are necessary. We simply feed the video into the network and obtain classification results using the trained model.

Algorithm 1 STSP for class incremental learning

Input: Task series$\{\mathcal{T}_1, \mathcal{T}_2, \cdots\}$, network $f(\,\cdot\,; \mathbf{w})$ with parameters $\mathbf{w}$ for linear layers and TSC, learning rate α and energy threshold η.
Initialize: $M_0 = \mathbf{0}$ for each linear layer and TSC.
for $\mathcal{T}_t \in \{\mathcal{T}_1, \mathcal{T}_2, \cdots\}$ **do**
  $s = 0$
  while not converge **do**
    Sample a mini-batch $\{D^t, y^t\} \in \mathcal{T}_t$ with videos $\{D_i^t, y_i^t\}_{j=1}^J$
    Compute $f(\{D^t, y^t\}; \mathbf{w}_{t,s})$ and obtain the gradient $\mathbf{g}_{t,s}$
    if $t > 1$ **then**
      for $\mathbf{w}_{t,s}$ in linear layers and TSC **do**
        Compute U_2 using M_{t-1} via Eq. (11) with a threshold η
        Update $\Delta\mathbf{w}_{t,s}$ using Eq. (12)
        Update $\mathbf{w}_{t,s+1} = \mathbf{w}_{t,s} - \alpha\Delta\mathbf{w}_{t,s}$
      end for
    else
      Update $\Delta\mathbf{w}_{t,s} = \mathbf{g}_{t,s}$
      Update $\mathbf{w}_{t,s+1} = \mathbf{w}_{t,s} - \alpha\Delta\mathbf{w}_{t,s}$
    end if
    $s = s + 1$
  end while
  Get current input features Q at each linear layer and TSC
  Update M_t using Eq. (10) for later incremental session
end for

4 Experiments

In this section, we compare our results against state-of-the-art techniques in class-incremental action recognition benchmarks. Additionally, we perform ablation studies to demonstrate the efficacy of our proposed method.

4.1 Experimental Setup

Datasets. The evaluation of our proposed framework is conducted using the UCF101 [30], HMDB51 [15], and Something-Something V2 [11] datasets, which are recognized benchmarks in video action recognition. The UCF-101 dataset comprises 13,320 instances across 101 action categories, with the data divided into three splits for training (70%) and testing (30%). Similarly, the HMDB-51 dataset features 6,766 video clips distributed among 51 action categories, also with three splits for training and testing in the same proportions. For our evaluations, we selected the first split from both datasets. Distinct from UCF101 and HMDB51, the Something-Something V2 dataset, a crowd-sourced collection, includes 220,000 videos across 174 classes. This dataset uniquely maintains consistent objects and backgrounds across its action categories, thereby emphasizing the importance of a model's capability to discern subtle motion cues.
Evaluation Protocol. Our methodology adheres to the class-incremental learning benchmarks frequently utilized in video analysis, as outlined in TCD [22].

Table 1: Video class incremental learning results on UCF101 and HMDB51. The best result for each setting is emphasized in bold.

Method	UCF101			HMDB51	
	10×5 stages	5×10 stages	2×25 stages	5×5 stages	1×25 stages
LwFMC [16]	42.14	25.59	11.68	26.82	16.49
LwM [6]	43.39	26.07	12.08	26.97	16.50
UCIR [12]	74.31	70.42	63.22	44.90	37.04
PODNet [8]	73.26	71.58	70.28	44.32	38.76
TCD [22]	74.89	73.43	72.19	45.34	40.47
FrameMaker [23]	78.13	76.38	75.77	47.54	42.65
STSP (Ours)	**81.15**	**82.84**	**79.25**	**56.99**	**49.19**

This approach begins with a model pre-trained on half of the total number of classes, with the remaining classes introduced sequentially at each incremental step. Specifically, for UCF101, the initial training involves 51 classes, after which the subsequent 50 classes are structured into incremental learning tasks of 5, 10, and 25 classes. In the case of HMDB51, the model begins with training on 26 classes, followed by the distribution of the remaining classes into groups of 5 and 25 for subsequent learning phases. For Something-Something V2, the initial training phase covers 84 classes, with the remaining classes organized into incremental learning groups of 10 and 5 classes.

Implementation Details. Our framework leverages the official implementation of TSM [17], utilizing the PyTorch Library for development. We train a ResNet-34 TSM model on UCF101 for 50 epochs with a batch size of 32. For both HMDB51 and Something-Something V2, ResNet-50 TSM models are employed, undergoing training for 50 epochs with a batch size of 32.

4.2 Main Results

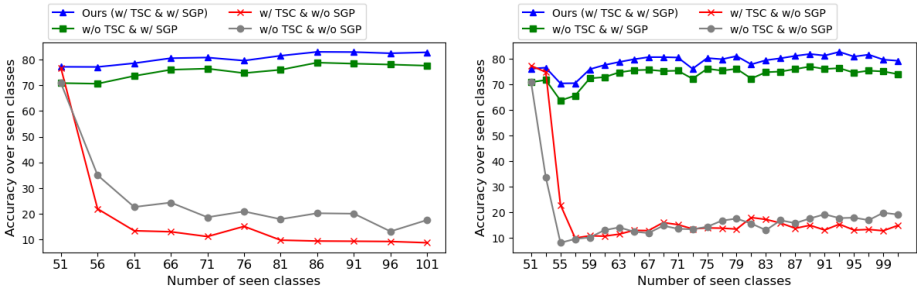
For a fair comparison, we employ the same model structure with the same pre-training initialization as the existing methods [22, 23]. Unlike prior approaches that incorporate exemplars for incremental learning and assess performance using both the standard classification protocol and Nearest Mean of Exemplars (NME) metrics [26], our method does not retain any exemplars. Consequently, we only report the standard classification results. We compare our proposed STSP method with the existing video-class incremental learning methods over three benchmarks, as shown in Tab. 1 and Tab. 2. From both tables, we observe that the proposed STSP method significantly outperforms all competing methods across all settings on each dataset. Notably, unlike other approaches that rely on retaining exemplars for incremental learning sessions, our method utilizes gradient projection to constrain model updates, effectively safeguarding against performance drops on previously learned classes.

Table 2: Video class incremental learning results on Something-Something V2. The best result for each setting is emphasized in bold.

Method	10×9 stages	5×18 stages
UCIR [12]	26.84	20.69
PODNet [8]	34.97	26.95
TCD [22]	35.78	29.60
FrameMaker [23]	37.25	30.98
STSP (Ours)	**69.68**	**70.87**

Table 3: Ablation studies for our proposed STSP method on UCF101 with 5×10 stages and 2×25 stages settings, respectively.

	TSC	SGP	Exemplars	UCF101	
				5×10 stages	2×25 stages
(i)	✗	✗	5/class	68.28	57.39
(ii)	✓	✗	5/class	71.50	61.78
(iii)	✗	✗	0	17.63	19.22
(iv)	✓	✗	0	8.78	14.99
(v)	✗	✓	0	77.66	74.02
(vi)	✓	✓	0	**82.84**	**79.25**



(a) UCF101 (5×10 stages) **(b)** UCF101 (2×25 stages)

Fig. 3: Plots of accuracy over seen classes along with the incremental steps on UCF101.

4.3 Ablation Study and Analysis

In this section, we perform ablation studies to investigate the effectiveness of the key components of the proposed STSP on the UCF101 dataset under both 2×25 stages and 5×10 stages settings.

Effectiveness of gradient projection To evaluate the SGP's capability to mitigate catastrophic forgetting, we designed experiments with various configurations: utilizing the method with SGP, the method without SGP, and the method without SGP but incorporating exemplars. In the last configuration, we preserved 5 videos, each containing 8 frames, as exemplars for prior classes to be used in future training sessions. Furthermore, we extended these comparisons to assess performance by employing a linear classifier and the proposed TSC. The comparison results are shown in Tab. 3, where (i), (iii), and (v) utilize the linear classifier, while (ii), (iv), and (vi) implement the TSC. In cooperation with the comparative analysis of the network's performance as the number of seen classes increases, as shown in Fig. 3, it can be found that the network tends to forget most knowledge of previous classes when no strategy is adopted.

Moreover, when comparing our SGP strategy against the exemplar-based distillation method, particularly through the comparison of (i) *v.s.* (v) and the comparison of (ii) *v.s.* (vi) in Tab. 3, we can find that our proposed SGP strategy

Table 4: Ablation study of the subspace basis size τ for TSC on UCF101 with 5×10 stages setting.

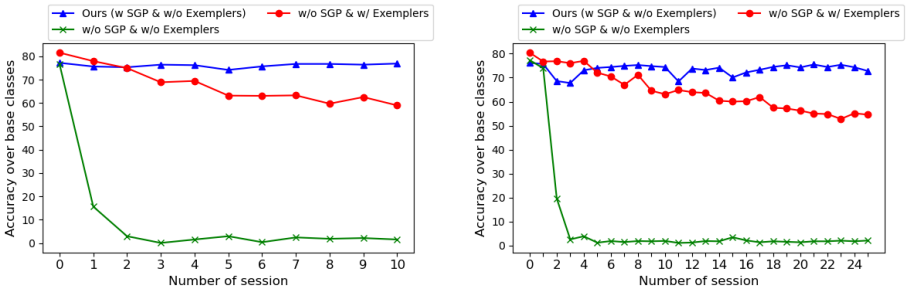
	$\tau = 1$	$\tau = 2$	$\tau = 4$	$\tau = 8$	$\tau = 16$
STSP (Ours)	66.75	67.88	75.39	**82.84**	63.02

significantly outperforms the exemplar-based distillation method. Specifically, regarding the performance of base classes during incremental learning sessions, as shown in Fig. 4, it is evident that our proposed SGP has minimal impact on the classification performance for the base classes. These findings indicate that our proposed SGP can uphold accurate classification across all classes without compromising performance on the base classes, emphasizing the effectiveness of SGP in mitigating knowledge degradation.

Furthermore, based on the comparison in Fig. 3 and Fig. 4, we observe that all methods perform worse under the setting of 2×25 stages, while our method with SGP still achieves the best performance. A plausible explanation is that the model may be susceptible to overfitting when dealing with a limited set of novel classes. Moreover, for our method, the computation of the null space becomes more unstable in these scenarios, especially when updating the mean covariance matrix is solely based on a few classes.

Linear or Subspace-based Classifier? To assess the effectiveness of the proposed TSC, we compare it with a linear classifier based on an FC layer. Moreover, this ablation study extends to both the exemplar-based distillation method and our proposed SGP, with the comparison results shown in Tab. 3. For the exemplar-based distillation approach, the implementation of TSC leads to a performance increase of 3.22% and 4.39% on UCF101 at the 10-stage and 25-stage settings, respectively, as evidenced by the comparison between (i) and (ii). In the context of SGP, the comparison between (v) and (vi) reveals that the proposed TSC similarly improves the overall performance by 5.18% and 5.23% for the 10-stage and 25-stage settings on UCF101, respectively. Compared with the exemplar-based distillation method, our method employs SGP to constrain the classifier update through gradient subspace projection. The above findings align with the concept of TSC, collectively constraining the subspace-based classifier and further enhancing classification performance.

From the comparison in Fig. 4, another observation is that TSC can even boost the classification performance on the base classes, demonstrating its effectiveness in modeling class representations. However, we also observe that TSC may perform poorly when there are no constraints (*i.e.* without exemplars nor SGP) on the network. One plausible explanation for this phenomenon is that, during incremental sessions, the network tends to shift the feature space toward new tasks without constraints from historical information. Such adjustments can result in the original subspace deviating from its previous configuration, leading to significant misalignment that greatly impacts the performance of TSC.



(a) UCF101 (5×10 stages) **(b)** UCF101 (2×25 stages)

Fig. 4: Plots of accuracy over base classes along with the incremental steps on UCF101.

The Size of the Subspace Basis We explore how the size of the subspace basis influences the classification accuracy of our proposed method, the comparison results are shown in Tab. 4. It can be found that large subspace basis sizes (i.e., $\tau = 4$ or $\tau = 8$) are more effective at constructing the subspace for each category, enabling the capture of more distinctive features, which in turn leads to better performance as opposed to smaller bases. This improvement, however, results in an increase in both the network's parameters and computational complexity. Additionally, we can observe that a subspace basis size of 16 yields the least favorable results. This could be attributed to the difficulty in constructing orthogonal subspaces for new classes when the subspace basis is excessively large, especially as $M \times \tau$ approaches the total number of classes ($\mathcal{C}$).

5 Conclusion

In this paper, we proposed STSP, a novel spatial-temporal subspace projection method designed for video class-incremental learning tasks. The objective is to learn discriminative and robust representations for both video and class, facilitating further incremental learning. STSP mainly consists of two key components, *i.e.*, temporal-base subspace classifier (TSC) and spatial-based gradient projection (SGP). TSC employs a subspace to model each video and its class, incorporating intra-frame temporal information. On the other hand, SGP is designed to mitigate catastrophic forgetting by projecting the gradient of network parameters in each layer onto the subspace spanned by the corresponding input spatial features. Extensive experiments on three benchmarks demonstrate the effectiveness of our proposed STSP method.

Limitations. In contrast to previous video class incremental learning methods, our proposed STSP method incorporates gradient projection to prevent the forgetting of historical information from previous classes. This strategy removes the necessity of storing exemplars and prototypes of previously learned classes, yet it requires the storage and updating of input features for both linear layers and the introduced TSC. Such a requirement could pose challenges when employing large backbones, requiring further exploration and discussion.

Acknowledgements

This research was carried out at the Rapid-Rich Object Search (ROSE) Lab at Nanyang Technological University in Singapore.

References

1. Akyürek, A.F., Akyürek, E., Wijaya, D., Andreas, J.: Subspace regularizers for few-shot class incremental learning. In: International Conference on Learning Representations (2022), https://openreview.net/forum?id=boJy41J-tnQ
2. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on pattern analysis and machine intelligence **19**(7), 711–720 (1997)
3. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
4. Chen, H., Wang, Y., Hu, Q.: Multi-granularity regularized re-balancing for class incremental learning. IEEE Transactions on Knowledge and Data Engineering **35**(7), 7263–7277 (2023)
5. Cheraghian, A., Rahman, S., Ramasinghe, S., Fang, P., Simon, C., Petersson, L., Harandi, M.: Synthesized feature based few-shot class-incremental learning on a mixture of subspaces. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 8661–8670 (2021)
6. Dhar, P., Singh, R.V., Peng, K.C., Wu, Z., Chellappa, R.: Learning without memorizing. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 5138–5146 (2019)
7. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy
8. Douillard, A., Cord, M., Ollion, C., Robert, T., Valle, E.: Podnet: Pooled outputs distillation for small-tasks incremental learning. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16. pp. 86–102. Springer (2020)
9. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 6202–6211 (2019)
10. Gao, Q., Zhao, C., Ghanem, B., Zhang, J.: R-dfcil: Relation-guided representation learning for data-free class incremental learning. In: European Conference on Computer Vision. pp. 423–439. Springer (2022)
11. Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al.: The" something something" video database for learning and evaluating visual common sense. In: Proceedings of the IEEE international conference on computer vision. pp. 5842–5850 (2017)
12. Hou, S., Pan, X., Loy, C.C., Wang, Z., Lin, D.: Learning a unified classifier incrementally via rebalancing. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 831–839 (2019)

13. Huang, H., Wang, Y., Hu, Q., Cheng, M.M.: Class-specific semantic reconstruction for open set recognition. IEEE transactions on pattern analysis and machine intelligence **45**(4), 4214–4228 (2023)
14. Ji, S., Xu, W., Yang, M., Yu, K.: 3d convolutional neural networks for human action recognition. IEEE transactions on pattern analysis and machine intelligence **35**(1), 221–231 (2012)
15. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: 2011 International conference on computer vision. pp. 2556–2563. IEEE (2011)
16. Li, Z., Hoiem, D.: Learning without forgetting. IEEE transactions on pattern analysis and machine intelligence **40**(12), 2935–2947 (2017)
17. Lin, J., Gan, C., Han, S.: Tsm: Temporal shift module for efficient video understanding. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 7083–7093 (2019)
18. Liu, C.: Gabor-based kernel pca with fractional power polynomial models for face recognition. IEEE transactions on pattern analysis and machine intelligence **26**(5), 572–581 (2004)
19. Masana, M., Liu, X., Twardowski, B., Menta, M., Bagdanov, A.D., Van De Weijer, J.: Class-incremental learning: survey and performance evaluation on image classification. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(5), 5513–5533 (2022)
20. McCloskey, M., Cohen, N.J.: Catastrophic interference in connectionist networks: The sequential learning problem. In: Psychology of learning and motivation, vol. 24, pp. 109–165. Elsevier (1989)
21. Papyan, V., Han, X., Donoho, D.L.: Prevalence of neural collapse during the terminal phase of deep learning training. Proceedings of the National Academy of Sciences **117**(40), 24652–24663 (2020)
22. Park, J., Kang, M., Han, B.: Class-incremental learning for action recognition in videos. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 13698–13707 (2021)
23. Pei, Y., Qing, Z., Cen, J., Wang, X., Zhang, S., Wang, Y., Tang, M., Sang, N., Qian, X.: Learning a condensed frame for memory-efficient video class-incremental learning. Advances in Neural Information Processing Systems **35**, 31002–31016 (2022)
24. Pei, Y., Qing, Z., Zhang, S., Wang, X., Zhang, Y., Zhao, D., Qian, X.: Space-time prompting for video class-incremental learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 11932–11942 (2023)
25. Peifeng, G., Xu, Q., Wen, P., Yang, Z., Shao, H., Huang, Q.: Feature directions matter: Long-tailed learning via rotated balanced representation. In: International Conference on Machine Learning. pp. 27542–27563. PMLR (2023)
26. Rebuffi, S.A., Kolesnikov, A., Sperl, G., Lampert, C.H.: icarl: Incremental classifier and representation learning. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 2001–2010 (2017)
27. Saha, G., Garg, I., Roy, K.: Gradient projection memory for continual learning. In: International Conference on Learning Representations (2021), `https://openreview.net/forum?id=3AOj0RCNC2`
28. Simon, C., Koniusz, P., Nock, R., Harandi, M.: Adaptive subspaces for few-shot learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4136–4145 (2020)

29. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: Advances in neural information processing systems. pp. 568–576 (2014)
30. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012)
31. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015)
32. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6450–6459 (2018)
33. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: Proceedings. 1991 IEEE computer society conference on computer vision and pattern recognition. pp. 586–587. IEEE Computer Society (1991)
34. Tzimiropoulos, G., Zafeiriou, S., Pantic, M.: Subspace learning from image gradient orientations. IEEE transactions on pattern analysis and machine intelligence **34**(12), 2454–2466 (2012)
35. Villa, A., Alcázar, J.L., Alfarra, M., Alhamoud, K., Hurtado, J., Heilbron, F.C., Soto, A., Ghanem, B.: Pivot: Prompting for video continual learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 24214–24223 (2023)
36. Villa, A., Alhamoud, K., Escorcia, V., Caba, F., Alcázar, J.L., Ghanem, B.: vclimb: A novel video class incremental learning benchmark. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 19035–19044 (2022)
37. Wang, C., Guo, L., Wang, Y., Cheng, H., Yu, Y., Wen, B.: Progressive divide-and-conquer via subsampling decomposition for accelerated mri. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 25128–25137 (2024)
38. Wang, J., Cherian, A.: Gods: Generalized one-class discriminative subspaces for anomaly detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8201–8211 (2019)
39. Wang, S., Li, X., Sun, J., Xu, Z.: Training networks in null space of feature covariance for continual learning. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition. pp. 184–193 (2021)
40. Yang, J., Frangi, A.F., Yang, J.y., Zhang, D., Jin, Z.: Kpca plus lda: a complete kernel fisher discriminant framework for feature extraction and recognition. IEEE Transactions on pattern analysis and machine intelligence **27**(2), 230–244 (2005)
41. Yang, S., Liu, J., Lu, S., Er, M.H., Kot, A.C.: Collaborative learning of gesture recognition and 3d hand pose estimation with multi-order feature analysis. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 769–786. Springer (2020)
42. Yang, Y., Yuan, H., Li, X., Lin, Z., Torr, P., Tao, D.: Neural collapse inspired feature-classifier alignment for few-shot class-incremental learning. In: The Eleventh International Conference on Learning Representations (2023), `https://openreview.net/forum?id=y5W8tpojhtJ`
43. Zhang, L., Fu, J., Wang, S., Zhang, D., Dong, Z., Chen, C.P.: Guide subspace learning for unsupervised domain adaptation. IEEE transactions on neural networks and learning systems **31**(9), 3374–3388 (2019)

44. Zhao, B., Xiao, X., Gan, G., Zhang, B., Xia, S.T.: Maintaining discrimination and fairness in class incremental learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 13208–13217 (2020)
45. Zhu, H., Koniusz, P.: Ease: Unsupervised discriminant subspace learning for transductive few-shot learning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9078–9088 (2022)
46. Zhu, P., Yao, X., Wang, Y., Hui, B., Du, D., Hu, Q.: Multiview deep subspace clustering networks. IEEE Transactions on Cybernetics (2024)