IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.0429000

Enhanced Industrial Action Recognition through Self-Supervised Visual Transformers

YAO XIAO, HUA XIANG,TONGXI WANG, and YIJU WANG

College of Computer Science,Yangtze University,Jingzhou, Hubei CHINA;(e-mail: 2021710604@yangtzeu.edu.cn;Xianghua@yangtzeu.edu.cn;Tongxi Wang@yangtzeu.edu.cn)

Department of Artificial Intelligence & Data Science, Guangzhou Xinhua University, Guangzhou, China (e-mail: 7120340@qq.com)

Corresponding author: HUA XIANG. Author (e-mail: Xianghua@yangtzeu.edu.cn).

This work was supported in part by the project "Research on the Construction of Management System and Mechanism of Applied Mathematics Center in Guangdong, Hong Kong and Macao" (2021A1515310004).

ABSTRACT Precise recognition of operator actions is crucial in industrial automation for enhancing production efficiency and ensuring safety standards. This study introduces a novel self-supervised pre-training framework using visual transformers to address the challenge of industrial event recognition. The framework incorporates an innovative Tube Masking strategy and leverages a comprehensive industrial dataset to effectively capture spatiotemporal features. Evaluation on our custom-built industrial dataset revealed a top-1 accuracy of 95%, demonstrating the model's practical applicability in real-world industrial environments. To further assess the model's generalization capabilities, it was tested on several public datasets, achieving top-1 accuracies of 92.8% on UCF101, 87.1% on HMDB51, and 90.2% on Kinetics400. These results highlight the robustness and versatility of our approach, paving the way for its application in diverse industrial scenarios and further research.

INDEX TERMS Self-supervised learning; Custom Industrial Dataset; Action recognition; Spatiotemporal features; Visual transformer; Pretraining strategy.

I. INTRODUCTION

IN industrial production, meticulous production procedures critically influence product quality, control over production costs, and enhancement of production efficiency [1]. Consequently, the precise recognition of operator actions is fundamental to forming an intelligent and efficient industrial environment. However, within a singular operational context, operators exhibit variable action semantics depending on the specific business procedures. Accurately discerning and differentiating these actions presents a significant research challenge.

Presently, action recognition in industrial settings include two predominant methodologies: methods leveraging multi-sensor data fusion and methods grounded in deep learning techniques. For the former, the fusion of data from diverse sources necessitates precise equipment installation or wearable devices. This introduces safety concerns, complicates operational procedures for workers, and incurs higher costs for action recognition. Within the deep learning domain, action recognition predominantly hinges on analysing image data encapsulated in video format. Methods employing by convolutional neural networks (CNNs) have historically dominated this realm. This dominance is due to the powerful inductive biases of convolution operations, which exhibit local connectivity and translational invariance in image processing [2]. While progress has been made by transitioning from 2D to 3D convolutional frameworks [3], these advancements invariably demand heightened computational resources. Moreover, these frameworks often falter when modelling dependencies that transcend the confines of their receptive fields [4], thereby revealing inherent limitations in using CNN-based strategies for action recognition.

The advent of the ViT model [5] heralded a paradigmatic shift from conventional CNNs towards visual transformers [6] in action recognition. Subsequent innovations, such as the video Swin transformer [7], have performed effectively. This can be attributed to their hierarchically structured architecture, which is reminiscent of CNNs, combined with an inductive bias favouring translational invariance and a robust capacity for handling extensive dependencies. Nonetheless, these visual transformer models often require training on expansive datasets, needing intricate feature representation learning to reach their full potential. Practical impediments to amassing and annotating large datasets, especially in spe-

cialized and intricate industrial contexts, often limit yield only modestly sized video and image datasets. This, in turn, exacerbates the challenges of training visual transformers ab initio.

In the domain of action recognition, self-supervised learning allows models to automatically learn representations of actions by observing unlabeled video data. The core of many computer vision tasks is learning discriminative features for target datasets. Collecting labeled datasets is both time-consuming and costly, thus there is an increasing trend towards learning representations in a self-supervised manner [8]–[13]. The main idea is to design surrogate tasks to serve as supervisory signals instead of manual labels, such as jigsaw puzzle prediction [14].This approach typically involves two main steps: generating prediction tasks (such as predicting the correct sequence of video frames) and training the model on these tasks to learn the intrinsic features of the data. This method significantly reduces the reliance on large amounts of labeled data, thereby lowering the cost and time associated with data preparation.

In video action recognition tasks, the Tube masking technique is a specially designed self-supervised learning method. This technique trains models to learn dynamic features and spatiotemporal relationships in videos by randomly masking parts of the data in video frame sequences and then predicting the masked parts. Unlike traditional 2D image processing, the Tube masking technique considers the temporal dimension, allowing the model to capture the temporal dynamics of video sequences, which is crucial for understanding complex actions. The masking effect used in this study is shown in Figure 5-2, which depicts the original video frame images. Through the masking strategy, the model learns to predict the content of the masked areas in the video, thereby indirectly learning the ability to recognize actions in the video.

Revolutionary pretrained architectures such as GPT [15] and BERT [16] catalysed profound shifts in natural language processing. Masked autoencoding for pretraining has permeated the image processing field. Prototypes such as ImageMAE [17] prioritize pretraining on large unlabelled image datasets, employing diverse masking paradigms (e.g., sporadic image segment masking) to train models in predicting occluded regions. This approach has subsequently been fine-tuned to cater to specific visual tasks. VideoMAE [18], for instance, applies the masking philosophy of MAE [17] o video data, offering comparative performance analyses across various masking strategies. Additionally, pioneering research, such as BEVT [19],underscores the roles of spatial-temporal cues and the importance of including intersample variations in feature extraction. Models such as TubeViT [20] have been used to repurpose the ViT [5] encoder into a versatile video-centric model that is proficient at seamlessly assimilating both image and video inputs. Collectively, these explorations corroborate the assertion that pretraining models on large-scale unlabelled datasets and subsequently fine-tuning the models can markedly amplify the action recognition performances of the models.

In response to the challenges of action recognition within industrial contexts, we present the Enhanced Visual Industrial Sequence Transformer (E-VIST). An overarching overview of our approach is depicted in Figure 1. Our main contributions are summarized as follows.

1) A self-supervised pre-training approach for action recognition has been proposed, utilizing the core architecture of a visual Transformer. This method efficiently extracts and learns spatiotemporal features of video data through masked modeling techniques.
2) The Tube masking strategy and the Video Swin Transformer architecture are employed to sample video data at intervals and predict the masked tokens using a Transformer encoder-decoder. This optimizes the model's capability to extract action features.
3) In the actual environment of an industrial production line, task execution data were collected from multiple operator stations to construct a comprehensive industrial action dataset.

The remaining paper is structured as follows. Section II, includes an analysis of prior research in action recognition. Section III begins with an extensive examination of the video Swin transformer architecture, followed by a detailed portrayal of the design and construction of the Enhanced Visual Industrial Sequence Transformer. Section IVprovides experimental outcomes and comparative evaluations with other methods. Finally, in Section V, a comprehensive summary and conclusion are provided.

II. RELATED WORK

A. EXPLORATION OF SPATIOTEMPORAL FEATURES IN ACTION RECOGNITION

Leveraging Convolutional Neural Networks: Considerable research efforts have been dedicated to utilizing CNNs for discerning spatial and temporal characteristics within video data. Preliminary methods harnessed 2D convolution to incorporate temporal dynamics, epitomized by dual-stream network paradigms such as two-stream [21] and TSN [22].These methods concurrently interpret spatial aspects of videos and temporal movement features by including optical flow. A variant of the dual-stream model, the slow-fast network [23], adopts an alternative approach: the slow stream captures spatial traits at a high-resolution, reduced frame rate, which is complemented by the fast stream's grasp of timing nuances between frames at a coarser resolution but higher frame rate. The introduction of 3D convolution initiated a paradigm shift in action recognition techniques. Nonetheless, training challenges led to a preference for pseudo3D and (2+1)D convolutional models. Notable examples include I3D [24], C3D [3], and R(2+1)D [25], which partition 3D convolution into distinct spatial and temporal phases, thereby capturing intertwined spatiotemporal attributes.

Transitioning to Visual Transformers: Action recognition strategies employing transformer frameworks parallel the principles of 2D convolution in their emphasis on delineating

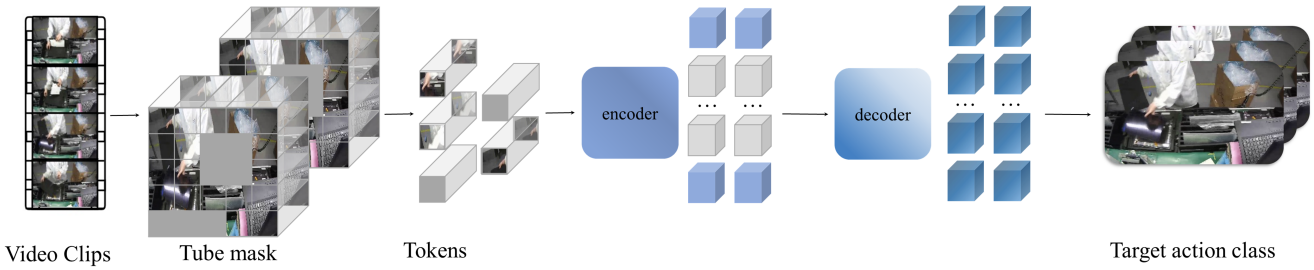

FIGURE 1. Overview of the Enhanced Visual Industrial Sequence Transformer.

spatiotemporal nuances within video sequences. For instance, TimeSformer [26]experiments include five iterations of spatiotemporal attention. However, the resultant spatiotemporal descriptors are superfluous, and the computational demands are prohibitive. To circumvent these limitations, transformer derivations rooted in ViT [5], such as DeiT [27], use distillation loss to expedite training. Models such as Mvit [28] aim to perceive dynamism across multiple scales by incorporating hierarchical structures and attention pooling, albeit with potential pitfalls in discerning intricate sequences. In contrast, the Video Swin Transformer model [7]combines global and local attention mechanisms to enhance recognition capabilities while optimizing computational efficiency.

B. SELF-SUPERVISED REPRESENTATION LEARNING

In the field of action recognition, self-supervised learning enables models to autonomously learn representations of actions by analyzing unlabeled video data. This method primarily involves two steps: firstly, designing predictive tasks to generate learning signals; secondly, using these tasks to train the model in order to capture the inherent features of the data. This learning paradigm significantly reduces reliance on large-scale annotated datasets, thereby decreasing the time and cost associated with data preparation.

A key aspect of training models in self-supervised learning is the ability to predict content that has been masked based on its context. In Masked Language Modeling (MLM), BERT and its derivatives achieve leading performance in many downstream natural language processing tasks by predicting tokens that have been masked during the training phase. In the visual domain,the masked image modelling (MIM) technique introduced by BEiT [29] has emerged as a novel trajectory in the vision pretraining domain. Within the architecture of vision transformers, an MIM effectively discerns spatial features by aligning token-level attributes with predetermined spatial coordinates. Masking operations geared towards predicting the attributes of occluded segments are central to this paradigm. These operations notably aid the model in assimilating elevated hierarchical image representations. Recent scholarly endeavours have validated the proficiency of the aforementioned masking strategy in video and image streaming tasks. Notably, Wang et al. [30] and Sun et al. [31] have used masking on select regions within image and video datasets. Subsequent model-driven reconstructions, building upon unmasked segments, have been instrumental in differentiating both spatial and temporal disparities embedded within action sequences. The prevailing action contexts within industrial production scenarios illustrate that temporal features play a more pivotal role in distinguishing operator actions than spatial features. Accordingly, in this study, a masking technique is used on video streams with the primary aim of highlighting crucial temporal variations and subsequently enhancing action recognition accuracy in industrial settings.

III. PROPOSED METHOD

In this section, we will intricately delineate the content and fabrication process of the industrial assembly line dataset. Additionally, we will revisit Video Swin transformer [7]. Then we show how we explore MAE in the video data by presenting our Enhanced Visual Industrial Sequence Transformer.

A. DATASET AND EVALUATION CRITERIA

The effectiveness of the proposed method was comprehensively evaluated using a custom-built industrial dataset, as well as public datasets including UCF101 [32], HMDB51 [33], and Kinetics400 [34].

1) Custom-built dataset

The custom dataset used in this study was created from recorded videos captured on-site in an actual industrial electronics production line. To ensure minimal disruption to the work of the operators, the cameras were adjusted and securely positioned relative to the operators. This positioning aimed to maximize the amount of relevant information within the video frames. Each video from different production stations was required to capture error-free operation sequences and exclusively feature single-person activity without any outside interference.

The dataset creation process involved randomly selecting video sequences from fixed-point surveillance footage at each workstation. Each distinct action category was treated as a fundamental action unit, encompassing activities such as card pin retrieval and right-sided object placement. These action unit videos, grouped by class name, were extracted from the

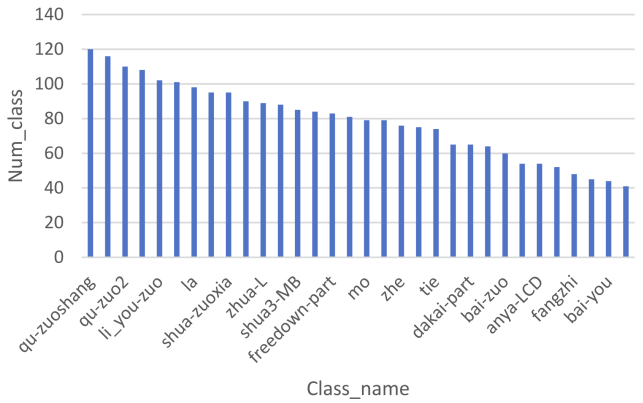

FIGURE 2.
Graphical Representation of Category Distribution in a Tailored Dataset.

recorded operation videos. The final dataset comprised approximately 1764 training videos and 756 validation videos, all with a resolution of 224 × 224 pixels. It encompassed 32 action categories, each with varying quantities of action videos, as illustrated in Figure 2 and detailed in Figure 3.

(a) bai-you (b) fangzhi (c) li_you-zuo

(d) la (e) qu-zuo2 (f) suo-song

justification=raggedright, singlelinecheck=false

FIGURE 3. Visual Overview of Selected Categories from the Action Classification Dataset

2) Public dataset

The public dataset employed in our study is UCF101 [32], which was released by Google in 2016. It comprises 101 action categories and includes approximately 9.5k training videos and 3.5k validation videos.This dataset encompasses a wide range of action types, providing rich scenarios for validating the generalization ability of our proposed method. The HMDB51 dataset, sourced from platforms like YouTube, consists of video classification data for 51 action classes, with at least 51 videos per action. The public dataset Kinetics400, released in 2017, is an extension of the action recognition datasets UCF101 and HMDB51, derived from non-professional videos on YouTube. This dataset comprises 400 action categories, each with a minimum of 400 video

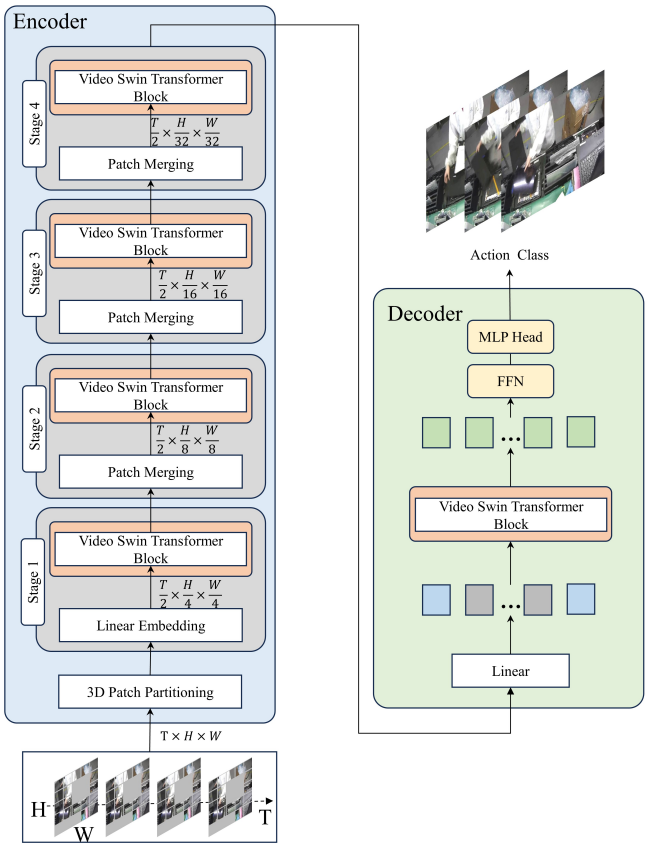

FIGURE 4.
Architectural Overview of the Enhanced Visual Industrial Sequence Transformer.

clips, with each clip approximately 10 seconds long, totaling over 300k samples with a relatively balanced distribution.

B. VIDEO SWIN TRANSFORMER

The video Swin transformer [7] is a visual model designed for processing and analysing video data. It is an extension of the Swin Transformer model, as illustrated in Figure 4. This model inherits the windowed self-attention mechanism of the Swin Transformer, enabling it to capture correlations between local and global features within video data. It also employs a hierarchical and translation-invariant structure that effectively extracts spatial features and captures temporal dynamic information within video data. The video Swin transformer model [7] comprises three main components: a visual sequence encoding curation layer, a temporal coding layer, and an output layer. This architectural design is tailored to the requirements of video data analysis, facilitating comprehensive feature extraction and temporal context modelling while maintaining the model's robustness and effectiveness in video data tasks.

1) **Video to token:**

First, the input video X is divided into multiple nonoverlapping 3D blocks. These blocks are then transformed into vector representations through linear mapping, and these representations are combined with positional encoding to create

a continuous spatiote-mporal sequence Z. Z is calculated as defined in Eq. 1.

$$Z = W_P X + P \tag{1}$$

where W_p represents the vector after linear mapping, X represents the input video data, and P represents positional encoding.

2) Model stages:

The model is then enhanced via the temporal coding layer, which bolsters its ability to extract spatiotemporal features through utilizing self-attention mechanisms and feed-forward neural networks.

Queries, keys, and values are calculated as outlined in. Eq. 2, while attention computation is executed as Eq. 3.

$$Q = ZW_Q, K = ZW_K, V = ZW_V, \tag{2}$$

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{3}$$

where d_k represents the dimensionality of the key K . The attention weights W are computed by taking the dot product between the query Q and the key K , and scaling the result using the SoftMax function.

Eq. 4 defines the residual connections and regularization.

$$Output = LayerNorm(Attention(Q, K, V) + Z) \tag{4}$$

Residual connections facilitate the learning of an identity mapping by adding the input Z to the output of the self-attention mechanism,thereby enhancing the convergence speed during training. The feed-forward neural network defined in Eq. 5 introduces nonlinear mappings to better capture abstract features within the data, thereby improving the model's representational capacity.

$$FFN(Output) = ReLU(OutputW_1 + b_1)W_2 + b_2 \tag{5}$$

where W_1, W_2, b_1, and b_2 denote the weights and biases, respectively, within the feed-forward neural network.

These operations collectively constitute the encoder layer,which is employed to capture intricate patterns and relationships within the input data. Stacking multiple encoder layers yields a richer feature representation.

3) Shifted Window Partitioning Strategy in Consecutive Blocks:

The shifted window partitioning strategy (as shown in the Figure8) is used to connect consecutive Transformer blocks. Specifically, it is alternately applied between different blocks within each stage. By alternately applying standard window self-attention and shifted window self-attention between blocks, the model can exchange information across windows. This strategy enhances the model's ability to capture long-range dependencies while avoiding the limitations of window partitioning.

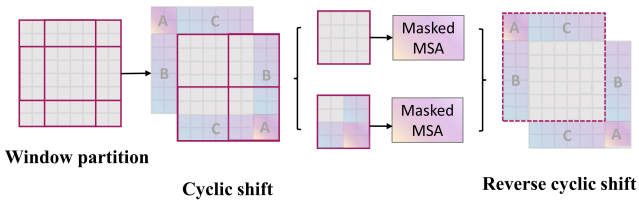

FIGURE 5.
Efficient Batch Computation for Self-Attention in Shifted Window Partition.

When processing video data, the shifted window partitioning not only moves the windows spatially but also shifts them along the temporal dimension. This enhances the model's ability to capture information across both spatial and temporal dimensions. The calculation method is as follows:

$$z^l = W - MSA(LN(z^{l-1})) + z^{l-1} \tag{6}$$

$$z^l = MLP(LN(z^l)) + z^l \tag{7}$$

$$z^{l+1} = SW - MSA(LN(z^l)) + z^l \tag{8}$$

$$z^{l+1} = MLP(LN(z^{l+1})) + z^{l+1} \tag{9}$$

where z^{l-1} is the output feature of the previous layer..

4) Head

Finally, the output layer is responsible for generating classification results. In the context of action recognition, the model produces a probability distribution for each action category:

$$Prob = softmax(ZW_H + b_H) \tag{10}$$

where W_H and b_H represent the parameters of the fully connected layer.

Using these three stages, the Video Swin Transformer [7] effectively extracts intricate spatiotemporal features from the raw video input; these features are subsequently harnessed for action recognition tasks.

C. ENHANCED VISUAL INDUSTRIAL SEQUENCE TRANSFORMER ARCHITECTURE

The proposed approach adopts the Video Swin Transformer as the backbone network and enhances it by introducing masking strategies and improving the decoder, thus optimizing the original model. A diagram of the framework of the Enhanced Visual Industrial Sequence Transformer is shown in Figure 6.

1) Input clip

Enhanced Visual Industrial Sequence Transformer images are taken as input video clips $X \in R^{(T \times H \times W \times 3)}$, which represent real-world industrial environments and consist of T frames with dimensions $H \times W$ in RGB format.

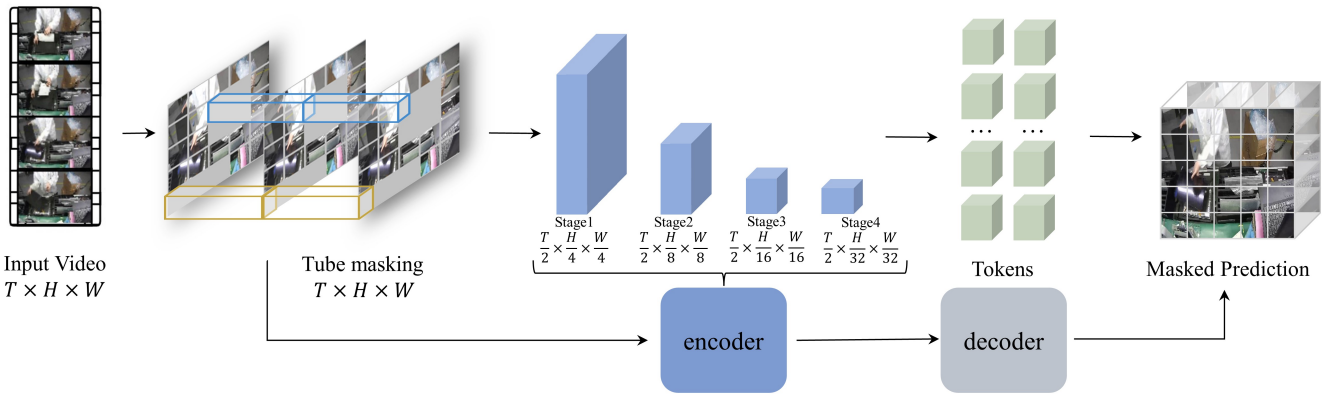

FIGURE 6. The framework diagram of Enhanced Visual Industrial Sequence Transformer.This diagram depicts the segmentation of input video data into frames and the subsequent application of tube masking strategies. After encoding, the resulting feature sequence is fed into the decoder for training, leading to the prediction of a reconstructed token sequence, which is ultimately fused to generate action categories.

2) Division into 3D patches

Like in the approach employed in the Video Swin Transformer [7], the video clips are divided into 3D patches of size $\frac{T}{2} \times \frac{H}{4} \times \frac{W}{4}$. Each 3D patch, conceptualized as an ensemble of tokens, has dimensions of $2 \times 4 \times 4 \times 3$,consequently yielding a token feature representation with a dimensionality of 96.Each token then undergoes projection mapping through a linear embedding layer with dimensionality C, yielding a token embedding sequence with dimensions $\frac{T}{2} \times \frac{H}{16} \times \frac{W}{16} \times C$.

3) Mask reconstruction

After a portion of the token embedding sequence is masked in preparation for input into the transformer encoder, the tube masking strategy is employed. This strategy involves randomly selecting an initial frame and determining the number of frames to be masked, with a masking rate set at 0.5. For the tokens that have been masked, as described in Eq. 11, the model aims to make predictions based on contextual information.

$$X^{'}_{t,i,j,k} = X_{t,i,j,k} \times M_{i,j,k} \tag{11}$$

where M denotes the spatial mask $M \in \{0,1\}^{H \times W \times C}$, $X^{'}$ denotes the masked tokens, i and j denote the height and width of the pixels, respectively, $k$denotes the channel dimension, and t denotes the temporal dimension, which corresponds to the length of the video frame. The ability to predict masked tokens helps the model learn profound spatiotemporal features.

The masked tokens are then converted into a learnable vector, which is input into the transformer architecture along with the remaining token sequence to train the model to predict the masked tokens.

4) Decoder

The decoder in Enhanced Visual Industrial Sequence Transformer(E-VIST) is primarily employed to predict and generate the masked tokens based on the input token sequence. Following the typical design principles of decoders in visual transformer architectures [18], [35], the decoder's channel dimension is set to $C/2$, which is half of the encoder's dimension. As described in the Model stages of the video Swin transformer architecture [7] presented in Sec III-B, the processing of video data by the encoder yields four distinct dimensional features.Therefore,the decoder is tasked with merging features from different dimensions through concatenation and linear layer fusion. These acquired features are ultimately utilized to reconstruct the masked token information, denoted as$Y$.

$$Y = g(concat(f(X_{vis}), X^{'}_{t,i,j,k})) \tag{12}$$

where $X_{vis} = X - X^{'}$ represents tokens that remain unmasked and observable. The function$f(X)$ is employed for feature extraction from these visible tokens, and $concat(X)$ denotes the concatenation of sequences.

5) Loss Function Specification

For each masked token, the associated reconstruction target is denoted as a token feature represented by$h(T(X^{'}))$, with $h$representing the function employed for generating the target feature. Afterwards, to train both the encoder and decoder and compute the distance D between the actual features of the masked tokens and their reconstructed counterparts, the loss function L is established in Eq. 13:

$$L_{mfm}(h) = \frac{1}{|M|} \sum_{p \in M} D(Y(p), h(T(X^{'}_{p}))) \tag{13}$$

where p represents the index within the token sequence and Msignifies the collection of masked tokens.By minimizing this loss function, the model continually enhances its capacity to predict the masked tokens, thereby effectively extracting spatiotemporal features from the video data.

Finally, after model training, the acquired features are integrated to derive the class representation of the input video data.

IV. EXPERIMENTS AND RESULTS

A. EVALUATION CRITERIA

In this research, we adopt the $Top-k$ metric as the primary evaluation criterion to assess the action recognition accuracy.

$$Top-1-acc = \frac{k}{N} \times 100\%, \tag{14}$$

$$Top-5-acc = \frac{m}{N} \times 100\% \tag{15}$$

where N denotes the total number of samples,k represents the number of samples for which the first prediction is correct, and$m$signifies the number of samples among the first five predictions that contain the correct category. Thus, $Top-1$ signifies the likelihood that the model's predictions precisely match the true action category for each sample, while$Top-5$ indicates the probability that the$Top-5$most likely categories predicted by the model encompass the true action category.A higher$Top-1$ value indicates the superior action recognition performance of the model.

B. EXPERIMENTAL SETUP

The experiments involving Enhanced Visual Industrial Sequence Transformer were conducted utilizing the PyTorch deep learning framework on a single RTX-3090 GPU. In this study, the video Swin transformer architecture was employed for experimentation. This architecture involves subsampling video data, taking 16 frames of RGB images as input, and applying online data augmentation techniques such as random cropping, scaling, and flipping.

To mitigate overfitting, we utilized a learning rate adjustment strategy involving warm-up and cosine annealing. Initially, a smaller learning rate was utilized to ensure stability, and once the training process reached stability, the learning rate was adjusted to 0.003. The network was optimized using the AdamW optimizer, enhancing the model's generalizability.

Two distinct training strategies, akin to those used in [17], [19], [26], were employed in this study. The first approach involved training the model from scratch on the custom-built dataset for 150 epochs. The second approach involved pretraining the Swin transformer model on a large-scale video dataset (K400) for 150 epochs as an initialization step. The obtained weights were then fine-tuned for the video Swin transformer model.

C. RESULTS

1) Comparisons with State-of-the-art Methods

The Enhanced Visual Industrial Sequence Transformer method was compared with previous action recognition methods on both the custom-built dataset and the public dataset UCF101 [32], HMDB51 [33] and Kinetics400 [34]. The results presented in Table 1 and Table 3 indicate that our method outperforms previous approaches, particularly on the custom-built dataset, which emphasizes temporal correlation. This finding suggests that our Enhanced Visual Industrial Sequence Transformer approach, which utilizes masked self-supervised pretraining, is more effective at learning spatiotemporal features.

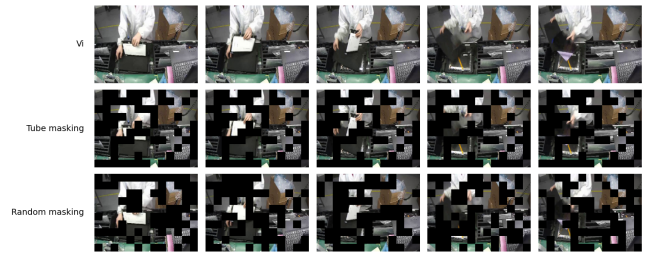

FIGURE 7. The figure demonstrates the results of different masking methods. Sequence V_i comprises a series of original image frames that sequentially depict an action within a specific category. The middle sequence illustrates the application of the tube masking method, whereas the bottom sequence demonstrates the use of the random masking method.

The experimental results in Table 1 demonstrate that the MIS (our approach) achieves a Top-1 accuracy of 95.0% and a Top-5 accuracy of 98.4%, with a moderate level of FLOPs and parameter count. This indicates that compared to other methods, our approach performs excellently in terms of accuracy while maintaining a good balance in model complexity. To assess the model's generalization ability, tests were also conducted on the public Kinetics400 [34] in Table 2,and datasets UCF101 [32] and HMDB51 [33] in Table 3.

This observation underscores the effectiveness of the Enhanced Visual Industrial Sequence Transformer (E-VIST) in capturing action sequence transformations and spatial distinctions, enabling the discrimination of diverse action categories.

D. ABLATION EXPERIMENTS

For the ablation experiments, two sets of investigations are conducted to explore the pivotal roles of masking strategies and pretraining within the Enhanced Visual Industrial Sequence Transformer framework.

1) Comparison of Masking Strategies

Using our self-built dataset and the same video Swin-base architecture, we performed a comparative analysis of two distinct masking strategies: tube masking and random 3D masking. Both strategies employed 50% masking probability and an equal number of masked frames, as summarized in Table 4.For a visualization of these masking strategies, see Figure 7. The experimental results are presented below.

As illustrated in Table 4, the tube masking strategy outperforms the random 3D masking strategy in terms of $Top-1$ accuracy. In a given video clip, denoted as V, where V_i represents the i_{th} frame within the clip, after tube masking, $M(V)$ is calculated as shown in Eq. 16.

$$M_{\text{Tube}}(V)_i = \begin{cases} 0, & \text{if } i \in [a,b] \\ V_i, & \text{otherwise} \end{cases}, \tag{16}$$

TABLE 1. Comparative Results of Various Action Recognition Methods on the Custom-Built Dataset.

Method	Pretrain	FLOPs	Param	Top-1(%)	Top-5(%)
C2D [4]	N/A	19	24.3	88.9	94.8
C3D [3]	N/A	38.5	78.4	87.3	96.0
NL-I3D [24]	ImageNet	59.3	35.4	82.5	92.1
CSN [36]	N/A	55.9	13.13	89.6	96.8
SlowFast [23]	N/A	36.3	34.5	92.0	97.4
TSN [22]	ImageNet	32.88	122	84.4	95.7
TimeSformer [26]	ImageNet	141	86.11	85.6	94.5
VideoSwin [7]	ImageNet	282	28.2	90.0	97.5
UniFormerV2 [37]	Kinetics400	59	49.8	91.4	95.3
BEVT [19]	ImageNet + Kinetics400	32	88	83.2	N/A
VideoMAE V2 [38]	Kinetics400	180	87	91.4	97.1
E-VIST(ours)	Kinetics400	282	88	95.0	98.4

TABLE 2. Results on the Kinetics400 dataset. We report the performance of our pretrained model with larger input resolution.

Method	Pretrain	FLOPs	Param	Top-1(%)	Top-5(%)
TDN [39]	N/A	66	N/A	79.4	94.4
NL-I3D [24]	ImageNet	59.3	35.4	74.7	91.8
SlowFast [23]	N/A	36.3	34.5	75.6	92.3
TSN [22]	ImageNet	102.7	24.3	72.8	90.7
MViTv2 [40]	Kinetics400	225	51.2	82.9	95.7
TimeSformer [26]	ImageNet	141	86.11	76.9	92.3
VideoSwin [7]	ImageNet	282	28.2	78.8	93.7
UniFormerV2 [37]	Kinetics400	59	49.8	90.0	98.4
VideoMAE V2 [38]	Kinetics400	180	87	88.5	98.1
MIS(ours)	Kinetics400	282	88	90.2	98.5

TABLE 3. Comparison with the state-of-the-art methods on UCF101 [32] and HMDB51.

Method	Pretrain	UCF101	HMDB51
C3D [3]	N/A	82.3	51.6
TSM [22]	N/A	88.0	70.9
NL-I3D [24]	ImageNet	93.4	74.8
CVRL [41]	Kinetics400	93.4	70.6
CoCLR [42]	N/A	74.5	54.6
RSPNet [43]	Kinetics400	93.7	64.7
VideoMAE [18]	N/A	91.3	73.3
MIS(ours)	Kinetics400	92.8	87.1

where a and b represent the consecutive masked frame intervals. In a similar vein, after Random 3D masking, $M(V)$ is calculated as follows.

$$M_{\text{Random 3D}}(V)_i = \begin{cases} 0, & \text{if } i \in \{r1, r2...\} \\ V_i, & \text{otherwise} \end{cases}, \quad (17)$$

TABLE 4. Comparison of masking strategies.

Method	Strategies	Top-1(%)
E-VIST	Random 3D	80.95
E-VIST	Tube	82.54

where $\{r1, r2, \dots\}$denotes the set of randomly masked frame indices. For two adjacent frames, i and $i+1$,introduced as query and key vectors into Eq. 3, the attention weights $A(i, i+1)$can be derived:

$$A(i, i+1) = softmax(\frac{Q_i K_i^T + Q_{i+1} K_{i+1}^T}{\sqrt{d_k}}) \quad (18)$$

Consequently, for n frames of video,the function $C(V)$ utilized to measure interframe continuity is defined as follows.

$$C(V) = \sum_{i=1}^{n-1} A(i, i+1) f(i, i+1) \quad (19)$$

TABLE 5. Comparison of Pretraining Strategies.

Method	pretrain	Top-1(%)
E-VIST	K400	91.27
E-VIST	None	82.68

The change in interframe continuity for tube masking can be expressed as Eq. 20.

$$\Delta C_{Tube} = C(V) - C(M_{Tube}(V)) \tag{20}$$

Similarly, the change for Random 3D masking is as follows.

$$\Delta C_{Random3D} = C(V) - C(M_{Random3D}(V)) \tag{21}$$

In tube masking, because consecutive frames are masked, the attention mechanism places more emphasis on nonmasked frames, resulting in higher weights,$A(i, i+1)$,between the frames. Consequently, continuity loss primarily occurs near the masked frames. In contrast, for random-3D masking, the attention weight distribution may be more dispersed, leading to continuity loss throughout the video.

$$\Delta C_{Tube} < \Delta C_{Random3D} \tag{22}$$

This finding underscores the advantage of the tube masking strategy over the Random 3D strategy in terms of $Top-1$ accuracy.

2) Significance of Pretraining

The proposed Enhanced Visual Industrial Sequence Transformer method employs video Swin-Base as the backbone network for experimentation. A comparison was conducted between the model trained from scratch and the model pretrained on the custom-built dataset. To ensure equitable comparisons, both methods were trained using identical architectures for 100 epochs. The experimental results align with prior findings in [18], [26], showing that employing pretrained models for initialization yields higher accuracy than training from scratch. As demonstrated in Table 5, the model initialized through pretraining achieves a 10.4% higher accuracy than the model trained from scratch. Furthermore, the two training loss progression plots, as depicted in Figure 8, illustrate that the model incorporating pretraining converges faster and exhibits better performance with increasing training iterations.

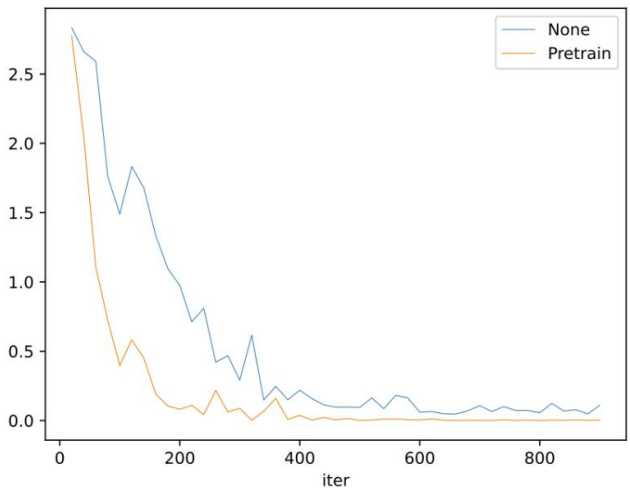

FIGURE 8. Loss Variation with and without Pre-training.

V. CONCLUSION AND DISCUSSIONS

Action recognition within industrial production processes is employed to enhance management efficiency, standardize execution protocols, and ensure the safety of employees' operations. In this paper, we build upon the visual transformer framework by employing interval-based video data sampling. The tube masking strategy is introduced to mask video data, which are then processed by an encoder, segmented, and mapped into a high-dimensional semantic space to generate a sequence of tokens. A transformer decoder is trained to predict the masked tokens based on the features of the input token sequence. Next, the loss function is minimized to reduce the gap between the predicted masked features and real features, yielding more effective spatiotemporal features. The features learned during training are subsequently fused to obtain the final action category representation. Furthermore, we employ a pretraining initialization strategy to enhance model performance. Experimental validation demonstrates that the proposed Enhanced Visual Industrial Sequence Transformer method performs exceptionally on both our custom-built dataset, and the public dataset UCF101 [32], HMDB51 [33] and Kinetics400 [34]. This underscores the potential of our action recognition approach in the industrial automation field, facilitating process analysis and optimization and thereby improving industrial production efficiency and quality.

Despite the exceptional performance of Vision Transformers in the field of video action recognition, significant challenges remain. Notably, the large-sized input feature maps required by Vision Transformers, such as images and video frames, substantially limit the model's training and inference speeds. Additionally, similar to Transformers in natural language processing, the effectiveness of Vision Transformers heavily relies on extensive pre-training, which not only complicates model deployment but also increases costs. In light of these challenges, future research will focus on model lightweighting to reduce dependence on computational resources and enhance operational efficiency. This effort aims to facilitate the widespread deployment of Vision Transformers in actual industrial settings and effectively improve the accuracy and efficiency of action recognition in industrial environments.

REFERENCES

[1] W. Li, "Process quality control and management in the production of electronic products," *Electronics Quality*, no. 4, pp. 49–52, 2011.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[3] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, Conference Proceedings, pp. 4489–4497.

[4] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, Conference Proceedings, pp. 7794–7803.

[5] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, and S. Gelly, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

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

[7] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, Conference Proceedings, pp. 3202–3211.

[8] B. Fernando, H. Bilen, E. Gavves, and S. Gould, "Self-supervised video representation learning with odd-one-out networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3636–3645.

[9] D. Kim, D. Cho, and I. S. Kweon, "Self-supervised video representation learning with space-time cubic puzzles," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 8545–8552.

[10] J. Wang, J. Jiao, L. Bao, S. He, Y. Liu, and W. Liu, "Self-supervised spatio-temporal representation learning for videos by predicting motion and appearance statistics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4006–4015.

[11] U. Buchler, B. Brattoli, and B. Ommer, "Improving spatiotemporal self-supervision by deep reinforcement learning," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 770–786.

[12] D. Q. Vu, N. T. Le, and J.-C. Wang, "Self-supervised learning via multi-transformation classification for action recognition," 2021.

[13] C. Wei, H. Fan, S. Xie, C.-Y. Wu, A. Yuille, and C. Feichtenhofer, "Masked feature prediction for self-supervised visual pre-training," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 14 668–14 678.

[14] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *European conference on computer vision*, 2016, pp. 69–84.

[15] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[17] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, Conference Proceedings, pp. 16 000–16 009.

[18] Z. Tong, Y. Song, J. Wang, and L. Wang, "Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training," *Advances in neural information processing systems*, vol. 35, pp. 10 078–10 093, 2022.

[19] R. Wang, D. Chen, Z. Wu, Y. Chen, X. Dai, M. Liu, Y.-G. Jiang, L. Zhou, and L. Yuan, "Bevt: Bert pretraining of video transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, Conference Proceedings, pp. 14 733–14 743.

[20] A. Piergiovanni, W. Kuo, and A. Angelova, "Rethinking video vits: Sparse video tubes for joint image and video learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, Conference Proceedings, pp. 2214–2224.

[21] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in neural information processing systems*, vol. 27, 2014.

[22] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. Van Gool, "Temporal segment networks: Towards good practices for deep action recognition," in *European conference on computer vision*. Springer, 2016, Conference Proceedings, pp. 20–36.

[23] C. Feichtenhofer, H. Fan, J. Malik, and K. He, "Slowfast networks for video recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, Conference Proceedings, pp. 6202–6211.

[24] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, Conference Proceedings, pp. 6299–6308.

[25] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri, "A closer look at spatiotemporal convolutions for action recognition," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, Conference Proceedings, pp. 6450–6459.

[26] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" in *ICML*, vol. 2, 2021, Conference Proceedings, p. 4.

[27] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," in *International conference on machine learning*. PMLR, 2021, Conference Proceedings, pp. 10 347–10 357.

[28] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, Conference Proceedings, pp. 6824–6835.

[29] H. Bao, L. Dong, S. Piao, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

[30] R. Wang, D. Chen, Z. Wu, Y. Chen, X. Dai, M. Liu, L. Yuan, and Y.-G. Jiang, "Masked video distillation: Rethinking masked feature modeling for self-supervised video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, Conference Proceedings, pp. 6312–6322.

[31] X. Sun, P. Chen, L. Chen, C. Li, T. H. Li, M. Tan, and C. Gan, "Masked motion encoding for self-supervised video representation learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, Conference Proceedings, pp. 2235–2245.

[32] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[33] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "Hmdb: a large video database for human motion recognition," in *2011 International conference on computer vision*. IEEE, 2011, pp. 2556–2563.

[34] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[35] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, Conference Proceedings, pp. 10 012–10 022.

[36] D. Tran, H. Wang, L. Torresani, and M. Feiszli, "Video classification with channel-separated convolutional networks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, Conference Proceedings, pp. 5552–5561.

[37] K. Li, Y. Wang, Y. He, Y. Li, Y. Wang, L. Wang, and Y. Qiao, "Uniformerv2: Spatiotemporal learning by arming image vits with video uniformer," *arXiv preprint arXiv:2211.09552*, 2022.

[38] L. Wang, B. Huang, Z. Zhao, Z. Tong, Y. He, Y. Wang, Y. Wang, and Y. Qiao, "Videomae v2: Scaling video masked autoencoders with dual masking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 549–14 560.

[39] L. Wang, Z. Tong, B. Ji, and G. Wu, "Tdn: Temporal difference networks for efficient action recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 1895–1904.

[40] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer, "Multiscale vision transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 6824–6835.

[41] R. Qian, T. Meng, B. Gong, M.-H. Yang, H. Wang, S. Belongie, and Y. Cui, "Spatiotemporal contrastive video representation learning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 6964–6974.

[42] T. Han, W. Xie, and A. Zisserman, "Self-supervised co-training for video representation learning," *Advances in neural information processing systems*, vol. 33, pp. 5679–5690, 2020.

[43] P. Chen, D. Huang, D. He, X. Long, R. Zeng, S. Wen, M. Tan, and C. Gan, "Rspnet: Relative speed perception for unsupervised video representation learning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 2, 2021, pp. 1045–1053.

XIAO YAO is currently a Master's student majoring in Computer Science and Technology at Yangtze University. Her research interests mainly focus on the fields of video processing and deep learning. During her graduate studies, she has been involved in research projects related to industrial video processing and deep learning, accumulating rich theoretical knowledge and practical experience.

HUA XIANG Hua Xiang is a Master's supervisor and a young visiting scholar at the State Key Laboratory of Software Engineering, Wuhan University. He primarily engages in research and teaching in the fields of big data and artificial intelligence, with research interests including big data analysis, software architecture, and evolutionary learning.

TONGXI WANG a master's supervisor, graduated from Chengdu University of Science and Technology with a major in Computer Science and Technology in June 1994. He obtained a master's degree in Computer Application Technology from Chengdu University of Science and Technology in June 2007. Currently, he serves as the head of the first-level master's degree program in Software Engineering and the leader of the research team on comprehensive application of big data analysis and software development. Additionally, he holds positions as a technology consultant for the Economic and Information Commission of Hubei Province and as the leader of the Internet Information Expert Team of Jingzhou Science and Technology Association.

YIJU WANG holds a Ph.D. and serves as a supervisor for master's students. He graduated from the Department of Physics at Central China Normal University in 1983 with a Bachelor's degree. Subsequently, he obtained his Ph.D. degree from the Wuhan Institute of Physics, Chinese Academy of Sciences in 2001, followed by postdoctoral work completion in 2003. He participated in the 863 High-Tech Program "Research on Ionospheric TEC Nowcast and Forecast Demonstration System" and the National Defense Project "Research on Large-Scale Radio Wave Propagation Error Correction System for Beidou-1 Project". Moreover, he led the completion of projects such as the "Beidou-1 System Application Development Platform".