applied sciences

MDPI

Article

Spatio-Temporal Behavior Detection in Field Manual Labor Based on Improved SlowFast Architecture

Mingxin Zou [1], Yanqing Zhou [1,2], Xinhua Jiang [1,2,*], Julin Gao [3], Xiaofang Yu [3] and Xuelei Ma [1,2]

[1] School of Computer and Information Engineering, Inner Mongolia Agricultural University, Hohhot 010011, China; zoumx33@163.com (M.Z.); zhouyq@imau.edu.cn (Y.Z.); maxuelei@imau.edu.cn (X.M.)
[2] Inner Mongolia Autonomous Region Key Laboratory of Big Data Research and Application of Agriculture and Animal Husbandry, Hohhot 750306, China
[3] School of Agriculture, Inner Mongolia Agricultural University, Hohhot 010019, China; nmgaojulin@163.com (J.G.); yuxiaofang75@163.com (X.Y.)
* Correspondence: jiangxh@imau.edu.cn

Abstract: Field manual labor behavior recognition is an important task that applies deep learning algorithms to industrial equipment for capturing and analyzing people's behavior during field labor. In this study, we propose a field manual labor behavior recognition network based on an enhanced SlowFast architecture. The main work includes the following aspects: first, we constructed a field manual labor behavior dataset containing 433,500 fast-track frames and 8670 key frames based on the captured video data, and labeled it in detail; this includes 9832 labeled frames. This dataset provides a solid foundation for subsequent studies. Second, we improved the slow branch of the SlowFast network by introducing the combined CA (Channel Attention) attention module. Third, we enhanced the fast branch of the SlowFast network by introducing the ACTION hybrid attention module. The experimental results show that the recognition accuracy of the improved SlowFast network model with the integration of the two attention modules increases by 7.08%. This implies that the improved network model can more accurately locate and identify manual labor behavior in the field, providing a more effective method for problem solving.

Keywords: field; manual labor behavior; detection and recognition; SlowFast

check for updates

Citation: Zou, M.; Zhou, Y.; Jiang, X.; Gao, J.; Yu, X.; Ma, X.. Spatio-Temporal Behavior Detection in Field Manual Labor Based on Improved SlowFast Architecture. *Appl. Sci.* **2024**, *14*, 2976. https://doi.org/10.3390/app14072976

Academic Editor: Sungho Kim

Received: 12 January 2024
Revised: 9 February 2024
Accepted: 22 February 2024
Published: 1 April 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Agricultural production is a crucial means for humans to meet the demand for food and agricultural products. It also serves as the foundation for the development of human society. Some automated equipment and technologies have been implemented in agricultural production [1]; for example, harvesters and automatic watering systems. However, due to the diversity and specificity of agricultural tasks, despite the advancements in modern technology driving agricultural automation, the level of automation varies depending on crops and regions. Manual labor still remains a crucial component of production in certain regions and for certain crops. However, with a shortage of human resources and low labor efficiency, traditional manual labor methods no longer suffice to meet the demands of modern agricultural production [2]. Hence, the imperative to integrate intelligent technologies and innovations to enhance the efficacy and quality of agricultural production has grown more pronounced. The incorporation of intelligent technologies, including artificial intelligence and the Internet of Things, into agricultural practices has significantly contributed to the optimization of production efficiency, reduction of operational costs, and augmentation of both yield and quality of agricultural products [3]. In the realm of agricultural production, areas of paramount concern encompass crop growth and yield, soil and environmental conditions, the quality of agricultural products, and the pursuit of sustainable development [4]. The physical labor behaviors carried out in the field, including activities such as sowing, fertilizing, spraying, plowing, weeding, etc., play a pivotal role in fostering

optimal crop growth and achieving high-quality agricultural output. Consequently, the establishment of relevant datasets contributes to the development and improvement of automation technologies that are gradually replacing labor-intensive operations. These datasets will support research in both academia and practical applications, promote the development of agricultural robots and intelligent systems for automatically identifying agricultural workers' labor behaviors, and provide effective information for traceability systems, allowing consumers to understand the entire process of agricultural product growth. The intelligent analysis of these physical labor behaviors holds substantial significance in enhancing and streamlining physical labor processes. Such analyses contribute to heightened production efficiency, the optimization of resource utilization [5], the implementation of sophisticated business management practices [6], and the assurance of both crop growth and product quality. This is of significant importance for driving agricultural enterprises toward digitized and standardized management of farm workers, further promoting the transformation of traditional agriculture toward automation and smart farming.

Currently, the monitoring of manual labor behavior in agricultural production predominantly depends on surveillance videos and intelligent analysis technology for the observation and analysis of labor activities conducted by farmers throughout the production process. Meanwhile, the processing of surveillance videos primarily hinges on manual operations [7]. Nevertheless, when confronted with substantial video data, this processing approach proves sluggish and inefficient, and incapable of facilitating all-weather monitoring. Consequently, enhancing the recognition capabilities of physical labor behaviors in the field and fortifying the guidance and supervision of farmers' labor [8] stand as pivotal facets in the advancement of smart agriculture aiming to attain the objective of refined management.

The evolution of deep learning within the domain of human behavior recognition can be traced back to approximately 2010. Preceding the ascent of deep learning, human behavior recognition predominantly depended on conventional computer vision techniques. Nonetheless, traditional methods exhibit constrained accuracy in intricate scenarios as they necessitate meticulous feature design, which proves challenging in capturing intricate human behavior patterns. Deep learning has progressively emerged at the forefront of research in recent years [9], establishing itself as a predominant research tool in areas such as image recognition, image classification, and target detection [10]. It also presents novel opportunities for advancing human behavior recognition. Moreover, given the prevalent use of industrial communication devices, such as cameras, in agricultural settings, there exists an opportunity to integrate deep learning algorithms into these communication devices for the purpose of recognizing agricultural behaviors across diverse geographical regions.

Scholars both at home and abroad have undertaken numerous endeavors to apply existing convolutional neural network algorithms to tasks related to video behavior recognition. Diverging from the scope of image classification, video behavior recognition necessitates the extraction of both spatial and temporal features, amalgamating them into spatio-temporal features. Some researchers [11–13] have suggested employing 2D convolution to extract spatial features frame by frame and store them in a buffer. Subsequently, new input frames are processed and their spatial features are combined with those in the buffer to construct spatio-temporal features for behavior recognition. Conversely, other researchers [14–17] have advocated for the use of 3D convolution to extract spatio-temporal features directly from video clips for behavior recognition. The classical behavior recognition method based on 2D convolution is the spatio-temporal dual-stream CNN [18]. This approach utilizes optical flow to capture temporal information between video frames, and it has several variants, including the dual-stream network, TSN network [19], and I3D network [20], among others. Despite the enhancements in accuracy achieved by optical flow methods in behavior recognition, the substantial memory consumption and computational cost associated with extracting optical flow features preclude the attainment of end-to-end recognition. Instead of relying on optical flow for learning intricate temporal features, 3D

convolution-based behavior recognition introduces an additional temporal dimension compared to 2D convolution, facilitating end-to-end feature extraction and classification. The seminal approach in this domain is the C3D network [21], which extends the convolution kernel of VGGNet from a 3×3 2D convolution to a $3 \times 3 \times 3$ 3D convolution. Despite its simplicity, the 3D convolutional kernel results in an exponential increase in the number of parameters and computational cost for the network. The algorithmic design of C3D marked a significant milestone in behavior recognition, and subsequent 3D networks have been developed, including the R3D [22] and P3D [23] algorithms.

In the domain of agricultural behavior recognition, Xu Jinbo et al. introduced an enhanced ConvLSTM model [24] and an improved (2 + 1)D model [25] for the recognition of four types of agricultural behaviors in 2021 and 2022. Additionally, Yang Xinting et al. devised a methodology for recognizing two types of agricultural behaviors using the YOLOv7-tiny model in 2023 [26]. However, the datasets used to train these models suffer from small-scale and missing behavior categories, which limits their practical applicability. Moreover, these models exhibit certain deficiencies in computational efficiency and feature extraction. These are: insufficient global feature extraction capability and inadequate spatio-temporal feature extraction when extracting agricultural labor images. This paper proposes an enhanced SlowFast network. The network incorporates a CoordAttention (CA) attention module to compensate for the shortcomings in CNN's global feature extraction ability. Additionally, it augments the model's capacity to extract temporal information about the action through an action (ACTION) attention module, resulting in superior feature extraction results. The main work of this paper can be summarized in the following three points:

1. Implementing the CA attention module and presenting the enhanced SlowFast network with improved slow branching to enhance the accuracy of video recognition for human labor actions.
2. Introducing the ACTION attention module, devising the enhanced SlowFast network with improved fast branching, and integrating the two attention modules to further enhance the accuracy of video recognition for human labor actions.
3. Collecting agricultural labor data from diverse perspectives in the field and producing the Field Work Behavior Dataset (FWBD).

Through the aforementioned efforts, the enhanced SlowFast network, as proposed in this study, attains superior outcomes in agricultural behavior recognition, demonstrating enhanced precision in the video recognition of agricultural labor actions. The primary tasks of this paper are delineated as follows. Section 1 provides an overview of the current research landscape in the domain of human behavior recognition, encompassing existing methodologies and techniques, along with an elucidation of the research approaches and contributions of this paper. Section 2 reviews pertinent works related to the methodology employed in this paper, encompassing existing research and associated techniques. Section 3 meticulously provides a detailed exposition of the research methodology and data sources employed in this study, encompassing the creation of the dataset, model architecture, feature extraction methods, and other pertinent details. Section 4 undertakes experimental endeavors, encompassing trials with varied parameter configurations, alongside a comprehensive analysis and discussion of the experimental outcomes. Section 5 synthesizes the research findings of this paper and outlines prospects for future research.

2. Related Work

2.1. SlowFast Network

The SlowFast network architecture [27] is inspired by the distribution of retinal nerve cell species in biology. The network introduces two parallel branches, each dedicated to processing fast and slow movements, aiming to enhance spatial-temporal feature extraction. This is accomplished by configuring different sampling intervals and the number of convolutional channels for the two paths, leading to distinct temporal and spatial resolutions.

Consequently, it captures the diverse characteristics of fast and slow movements in the video. The network structure is depicted in Figure 1.

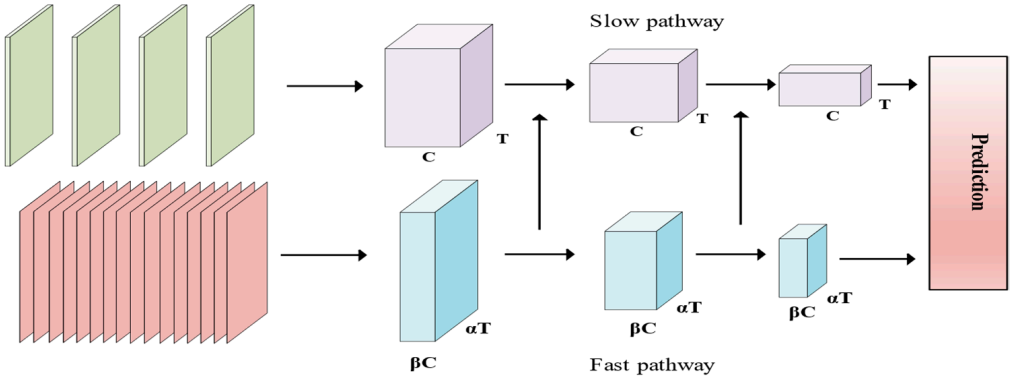

Figure 1. SlowFast network structure.

The slow path captures semantic features of spatial objects in the video with low temporal resolution and high spatial resolution. Let τ represent the temporal resolution, D denote the number of channels, and T the number of sampled frames in this branch. Consequently, the length of the original video is $\tau \times T$ frames. The 3D convolution in this branch is exclusively employed for high-level feature extraction to prevent a premature application of 3D convolution that may compromise accuracy. The temporal relationships between frames become evident only when the spatial receptive field is sufficiently large. The fast path captures the rapidly changing motion characteristics of the object in the video with high temporal resolution and low spatial resolution. Given the rapid changes in motion information within this branch, data with a high frame rate of αT (where $\alpha > 1$) frames and fewer channels, denoted as βD (where $\beta = 1/8$), are input to capture finely detailed fast-changing motion information in the temporal dimension. Due to the limited number of channels in the fast branch, it remains highly lightweight, constituting only 20% of the overall computational workload. The fast branch, to ensure temporal fidelity for fine motion information, refrains from conducting any temporal downsampling operations and consistently maintains the time dimension as αT until the ultimate global average pooling. This approach effectively preserves the temporal information of the motion and facilitates the capture of rapid motion features in the video.

The fast and slow branches exhibit strong complementarity, with unidirectional information fusion occurring from the fast branch to the slow branch. Given the disparity between the dimensions $D \times T \times S \times S$ of the slow branch and the dimensions $\beta D \times \alpha T \times S \times S$ of the fast branch, a transformed transversal connection is necessary to align the feature vectors from the outputs of both branches. Information fusion employs time-step convolution, wherein the $Pool_1$, res_2, res_3, and res_4 layers of the network undergo time-step convolution, followed by unidirectional feature fusion. Eventually, the feature vectors from the slow and fast branches are concatenated and, after the fully connected layer, category prediction is executed. The SlowFast concept represents an idea, and network implementation is specifically based on 3D ResNet50, as illustrated in Table 1.

Table 1. SlowFast network structure parameters.

Stage	Slow Pathway	Fast Pathway	S/F Output T × S2
raw clip	—	—	64×224^2
Date layer	Stride16, 1^2	Stride16, 1^2	$4/32 \times 224^2$
$conv_1$	1×7^2, 64	5×7^2, 64	$4/32 \times 112^2$
$Pool_1$	1×3^2, max	1×3^2, max	$4/32 \times 56^2$

Table 1. *Cont.*

Stage	Slow Pathway	Fast Pathway	S/F Output T × S2
res_2	$\{1 \times 1^2, 64$ $1 \times 3^2, 64$ $1 \times 1^2, 256\} \times 3$	$\{3 \times 1^2, 8$ $1 \times 3^2, 8$ $1 \times 1^2, 32\} \times 3$	$4/32 \times 56^2$
res_3	$\{1 \times 1^2, 128$ $1 \times 3^2, 128$ $1 \times 1^2, 512\} \times 4$	$\{3 \times 1^2, 64$ $1 \times 3^2, 64$ $1 \times 1^2, 256\} \times 4$	$4/32 \times 28^2$
res_4	$\{3 \times 1^2, 256$ $1 \times 3^2, 256$ $1 \times 1^2, 1024\} \times 6$	$\{3 \times 1^2, 32$ $1 \times 3^2, 32$ $1 \times 1^2, 128\} \times 6$	$4/32 \times 14^2$
res_5	$\{3 \times 1^2, 512$ $1 \times 3^2, 512$ $1 \times 1^2, 2048\} \times 3$	$\{1 \times 1^2, 64$ $1 \times 3^2, 64$ $1 \times 1^2, 256\} \times 3$	$4/32 \times 7^2$
Global average pool, concat, fc			#classes

2.2. Faster R-CNN Network

Faster R-CNN [28] represents a two-stage deep learning network designed for target detection, signifying an enhanced iteration within the R-CNN model lineage. The network achieves an end-to-end target detection process by incorporating a Region Proposal Network (RPN). The fundamental concept of Faster R-CNN involves segmenting the target detection process into two primary stages: candidate frame generation and target classification. In the initial step, RPN extracts candidate frames with varying proportions through a sliding window, generating location-sensitive bounding box offsets and corresponding confidence scores for each candidate frame. Subsequently, candidate boxes with high confidence are selected, transformed into fixed-size feature maps, and utilized for subsequent classification and location regression tasks. This transformation is executed via the RoI (Region of Interest) pooling layer. This two-stage architecture enables Faster R-CNN to significantly enhance detection speed while preserving target detection accuracy. With the integration of the RPN network, Faster R-CNN autonomously generates high-quality candidate frames, leading to more efficient and accurate performance in target detection tasks. The illustrated process is depicted in Figure 2.

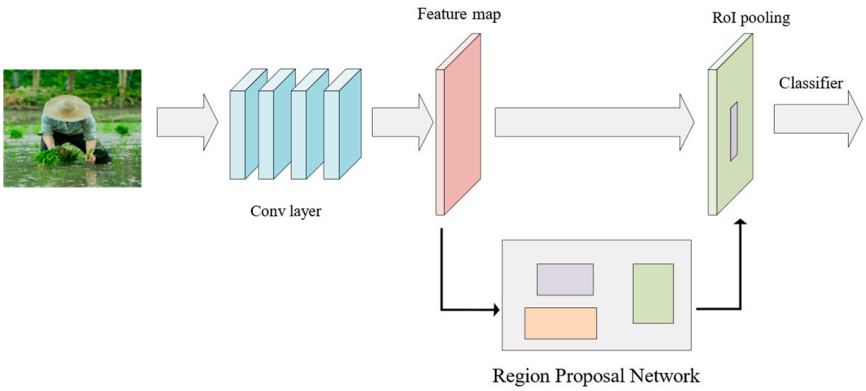

Figure 2. Faster RCNN network structure diagram.

2.3. Faster R-CNN + SlowFast Spatio-Temporal Behavior Detection Method

In the realm of human behavior recognition, depending solely on a single behavior recognition model presents several limitations. It faces challenges in capturing intricate details about the target object [29] and the surrounding contextual environment, thereby constraining the overall scene comprehension and analysis [30]. Conversely, an approach that integrates behavioral recognition with a target detection model can acquire compre-

hensive information regarding the location, scale, and orientation of the target object. This integration leads to a more precise contextual environment for behavioral recognition. This not only aids in the precise analysis of human motion but also captures the interaction between the target object and the environment, enhancing the understanding of human behavior within the entire scene. Manual labor behavior in the field is a continuous motion process, encompassing both spatial and temporal information [31]. Therefore, extracting temporal features from field manual labor behavior is highly significant for behavior recognition. Simultaneously, utilizing the spatio-temporal information from the video can enhance the extraction of video features [32]. The spatio-temporal behavior detection algorithm serves two purposes: to target localization and multi-label behavior recognition for target classification. The schematic representation of the spatio-temporal behavior detection algorithm is illustrated in Figure 3. Initially, the video input frames undergo preprocessing to identify key frames and furnish restricted temporal context information. By handling the data within a relatively concise temporal context, this not only enhances the operational speed of the spatio-temporal behavior detection algorithm but also alleviates potential ambiguities in understanding arising from the sole reliance on static frames for behavior determination. Next is the selection of the target detection model. For target localization, two mainstream methods exist: two-stage and one-stage target detection. Two-stage target detection exhibits higher accuracy compared to that of one-stage, particularly in detecting larger targets where it produces superior results. In terms of target size and the number of targets, the two-stage target detection method performs well when dealing with larger targets and a limited number of targets. Given that the focus of this work is on human behavior involving larger targets, the two-stage target detection model, Faster R-CNN, is used in the subsequent experiments to accomplish the localization task by directly acquiring the target frame size. The target frame is mapped onto the feature blocks obtained from the 3D feature extraction network to derive the corresponding feature matrix. The feature regions of interest undergo processing using the RoIAlign method to ensure uniform dimensions. Subsequently, the acquired features are uniformly pooled in the time dimension, and a fully connected layer is connected through a Sigmoid activation function for behavior prediction. The pre-trained Faster R-CNN target detector is employed directly, obviating the need for training with the SlowFast behavior detection module.

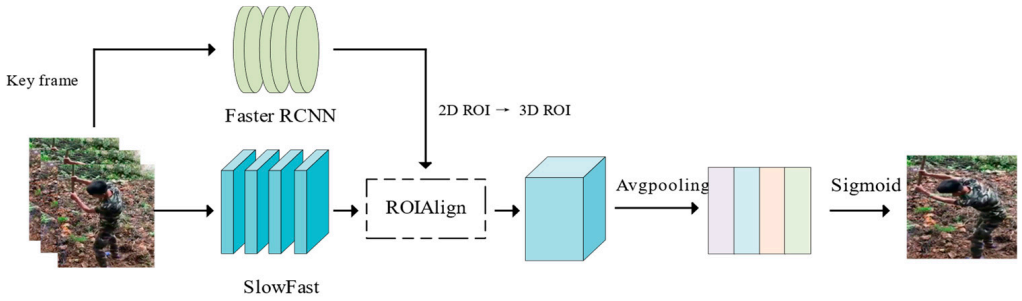

Figure 3. Network structure of spatio-temporal behavior detection algorithm.

RoIAlign (Region of Interest Align) [33], as mentioned in the preceding paragraph, represents a technique employed in target detection tasks that primarily addresses the quantization error issue inherent in RoIPooling [34]. In the realm of deep learning, target detection stands as a critical task, revolving around the precise localization and identification of targets within an image. The introduction of RoIAlign seeks to enhance the performance of target detection by refining the accuracy of features within the region of interest (RoI). The primary enhancement offered by RoIAlign over RoIPooling resides in the interpolation of feature points within the Region of Interest (RoI). Conventional RoIPooling involves segmenting the RoI into fixed-size grids and subsequently applying operations such as maximum pooling to the feature points within each grid. This methodology encounters an issue where quantizing the coordinates inside the RoI may result in information loss, particularly when handling small targets or requiring finer positional information. RoIAlign

addresses this challenge by intricately handling the feature points within the RoI through the utilization of bilinear interpolation. In detail, for every sampling point within the Region of Interest (RoI), RoIAlign computes its precise position on the feature map through interpolation, as opposed to approximating to the nearest integer position. This approach enables RoIAlign to extract features within the RoI more accurately, mitigating quantization errors and enhancing the precision of target detection. In practical applications, RoIAlign finds extensive use in advanced target detection models, including Faster R-CNN [35]. By enhancing the precision of object localization within the region of interest, RoIAlign has emerged as a pivotal technique in the realm of target detection, demonstrating efficacy particularly in addressing challenges posed by intricate scenes and diminutive targets. The incorporation of this technique furnishes a potent method for augmenting the efficacy of deep learning models in tasks related to target detection.

3. Dataset and Methods

In this section, we will intricately delineate the content and fabrication process of the Field Work Behavior Dataset. Additionally, we will elucidate the principal enhancements made to the SlowFast model in comparison to its original iteration, focusing on the key modules that underwent improvement.

3.1. Dateset

In the past decade, despite the proliferation of numerous large-scale video datasets for action recognition, there has been a notable absence of dedicated datasets specifically designed for agricultural behavior recognition. This deficiency in datasets is apparent. In this study, we addressed this gap by constructing a dataset named FWBD (If data is needed, interested parties may contact the corresponding author) (Field Work Behavior Dataset). While certain portions of this dataset were sourced from public databases such as Tencent Video and Youku, the quality of the videos was suboptimal, and they were limited in both quantity and coverage of labor types. To address these limitations, we adopted a multi-angle shooting approach in the field to augment the dataset with a greater number of videos and increased diversity in labor types. During the data preprocessing stage, we employed editing techniques to eliminate dirty data. Dirty data refers to video data capturing field labor activities that are challenging for human eyes to recognize. Dirty data were generated for two main reasons: 1. people were backlit or obscured by crop weeds, making it difficult to clearly visualize human labor practices and labor tools; and 2. most videos were edited during the web video collection process, resulting in a prolonged lack of people appearing in the frame. For videos in which it was challenging to discern agricultural labor due to the above-mentioned circumstances and other reasons, we opted for their removal. The subsequent stage involved data augmentation. The prior steps of data cleaning and video editing led to a decrease in both the quantity and duration of videos within each behavioral category, and this consequently impacted the overall dataset's size. Notably, certain behavioral categories experienced more pronounced reductions, particularly those derived from public web databases. To establish equilibrium in data distribution among behavioral categories within the database, our objective was to ensure a comparable number of actions within each behavioral segment. This strategy aimed to enhance the model's stability and accuracy in real-world applications by mitigating overfitting to the constructed dataset. To achieve this objective, we conducted data augmentation procedures on the videos for each behavioral category. The employed data augmentation methods encompass horizontal flipping, clipping, and color adjustment. Horizontal flipping is employed to generate visually distinct data points by flipping the video data horizontally. The clipping operation is applied to subtly modify the background content of the image. Additionally, adjustments to brightness and saturation were made during the model training process. We incorporated various strategies to address changes in illumination, including data in which lighting naturally dims over time, data with altered brightness achieved through changes in shooting angles, and data augmented to simulate variations in lighting conditions. These

measures were implemented to ensure that the trained model can effectively handle images under different lighting and color conditions. Ultimately, we successfully compiled a dataset comprising 200 original videos depicting manual labor behaviors in the field. These encompass hoeing, fertilizing, seeding, transplanting, spraying pesticides, watering, and weeding—seven of the most prevalent and conventional categories. The videos maintain a frame rate of approximately 30 frames per second (fps).

The steps of data processing and labeling are as follows: Step 1: Utilize the ffmpeg command to read the video, segmenting it into short 10 s videos. Subsequently, these videos are renamed based on their behavioral type and stored in the respective folders. Step 2: Construct the FWBD dataset following the AVA dataset format. Crop the video into frames with two requirements. Firstly, employ target detection to annotate key frames with the bounding box of the person in the image. These key frames are extracted at a rate of one frame per second, resulting in 10 images spanning from 0 s to 9 s. Secondly, capture 30 frames per second, equating to 300 images. Step 3: Initially, employ the Faster R-CNN target detection network to frame the target box for the human body's position in the key frame image. Throughout this process, set the recognition threshold for people by the Faster R-CNN network to 0.5, ensuring comprehensive framing of individuals in the image. Step 4: To comply with the CSV file requirements in the AVA dataset format for character IDs in the images, the outcomes obtained in Step 3 are input into Deep Sort to generate the IDs for characters in the key frames of the images. As the operational principle of Deep Sort [36] necessitates a minimum of 3 video frames for judgment, the first 2 frames and the last 2 frames are excluded. Only the key frames from the 2nd to the 7th second of each short video are utilized as images that require labeling with behavioral categories, ensuring error-free detection by Deep Sort. Consequently, each short video comprises 6 images necessitating behavioral category labeling. Step 5: After roughly framing the position of the characters using the Faster R-CNN target detection network, the target frames are further manually corrected using the VIA data labeling tool. The reasons for the correction include that the Faster R-CNN did not successfully recognize the characters in the video without framing them and that the target frames of some labor behaviors did not frame the tools used by the characters together. In certain cases, agricultural tools can be considered as an aspect of recognition because they are closely associated with specific labor behaviors. For instance, in agricultural settings, farmers may employ various tools for different farming activities, and the presence and manner of their usage can provide crucial cues for identifying particular labor behaviors. Consequently, the recognition of agricultural tools may be regarded as an aspect of labor behavior recognition that aims to enhance accuracy and comprehensiveness in identifying labor behaviors. For all these reasons, they need to be corrected manually. Following correction, corresponding action categories are labeled with categories. The correspondence of the action categories (label names) is as follows: hoeing (farm), sowing (seed), fertilizing (manure), transplanting (sow), spraying pesticides (spray), weeding (weed), and watering (water). Following this, the final labeled file is generated. The final output comprises four types of files: detection frame annotation files, spatio-temporal behavior annotation files, non-participating frame annotation files, and label annotation files. The semi-automated data annotation and FWBD dataset production process is illustrated in Figure 4.

The dataset composition is outlined in Table 2, and it comprises a total of 1445 videos, 8670 key frames, and 9832 labeled frames. Partial images for each behavioral category are illustrated in Figure 5. This dataset exhibits a broad range of personnel targets, diverse scenarios, and one or more targets engaging in the same type of action within the same video. To create the training and test sets, 80% of the data combinations in each category of the dataset are randomly selected for the training set, while the remaining 20% constitute the test set. This division ensures a relatively even distribution of data in the training and test sets, facilitating a more comprehensive evaluation of the model's performance.

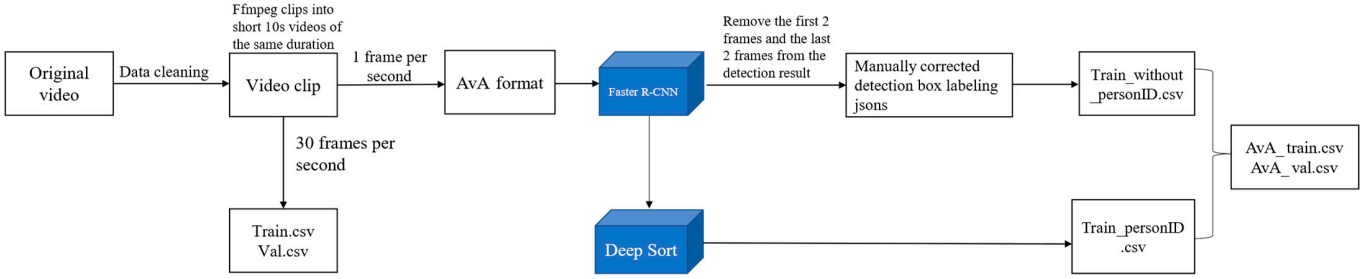

Figure 4. Dataset production process.

Table 2. FWBD statistics.

Category	Videos	Key Frames	Labels
Farm	215	1290	1546
Manure	192	1152	1152
Seed	245	1470	1478
Sow	202	1212	1939
Spray	185	1110	1193
Water	205	1230	1233
Weed	201	1206	1291
Total	1445	8670	9832

Figure 5. FWBD example demonstration.

3.2. Improvements to SlowFast Network Structure

In tasks related to video behavior understanding, human behavior exhibits an array of rich and diverse features, encompassing variations in the same behavior under different execution goals, and the impact of multiple factors such as shooting angle and lighting conditions. While the 3D Convolutional Neural Network (3DCNN) proves effective in spatio-temporal modeling, it faces challenges in capturing the wealth of information embedded in videos and requires the acquisition of more fine-grained features to enhance classification accuracy. To elevate the detection accuracy of the network, this study enhances the SlowFast network in two key aspects. Firstly, for the extraction of spatial features from the video, the CoordAttention (CA) [37] attention model was incorporated into the slow branch. This enhancement strengthens the network's capacity, particularly in extracting critical information related to spatial features of behavior, thereby contributing

to an overall improvement in performance. Secondly, to augment the network's proficiency in learning temporal features of behavioral motion, we introduced the action attention module in the fast branch. This module is primarily tasked with extracting such features, and its integration aims to enhance the network's capability to learn fine-grained features, thereby refining the architecture of the SlowFast network. With these two enhancements, the SlowFast network demonstrates enhanced performance in video behavior-understanding tasks. Specifically, the introduction of the CA (CoordAttention) and ACTION attention modules enables the network to more effectively capture crucial spatial and temporal motion features, thereby improving the comprehension and classification accuracy of intricate behaviors in videos. The enhanced version of SlowFast, denoted as CA-ACTION-SlowFast, is introduced herein, with its network architecture depicted in Figure 6.

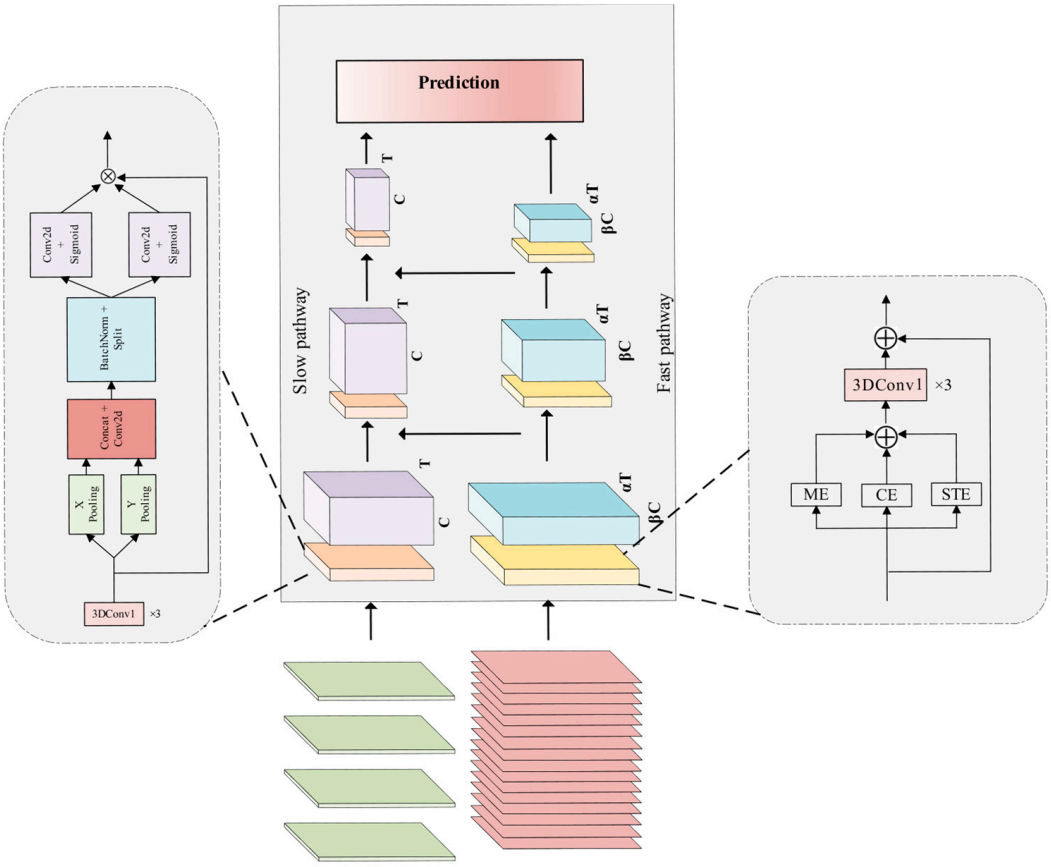

Figure 6. The overall structure of CA-ACTION-SlowFast.

3.2.1. CA Attention Modules

The CA module encodes channel relationships and long-range dependencies through precise location information. It comprises two steps: coordinate information embedding and coordinate attention generation. The specific structure is illustrated in Figure 7. Examining the network structure diagram of the CA attention module, we observe that the input features undergo two distinct pooling operations to generate spatial dimension features. These features are then inputted into a convolutional layer, and this is followed by processing through Batch Normalization and a Sigmoid activation function to produce the attention graph. This process can be outlined in the following steps:

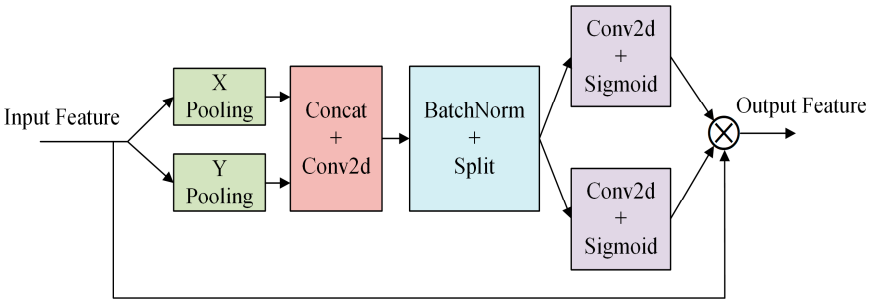

Figure 7. The structure of CA attention module.

1. A coordinate information embedding operation is conducted. To enable the attention module to capture spatial long-range dependencies with precise location information, global pooling is decomposed into a pair of one-dimensional feature encoding operations. Spatial information is extracted from the input features through pooling operations, reflecting the statistical information of different locations in the feature map. For input X, it undergoes pooling (Pooling) using pooling kernels with dimensions (H, 1) and (1, W). This process encodes each channel along the horizontal and vertical coordinate directions, resulting in the output of the cth channel with height h and the output of the cth channel with width *yes*, given as follows:

$$z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} X_c(h, i) \tag{1}$$

$$z_c^W(w) = \frac{1}{H} \sum_{0 \le j < H} X_c(j, w) \tag{2}$$

2. The aforementioned two transformations conduct feature aggregation along two spatial directions, yielding a pair of direction-aware attention maps that signify the regions to which SlowFast should "pay attention". These transformations enable the attention module to capture long-range dependencies along one spatial direction and preserve precise location information along the other, enhancing the network's ability to accurately localize target regions of interest with deep features. Coordinate attention generation is subsequently performed by concatenating the two feature maps generated by the preceding module and then transforming them into features using a 1×1 convolution, Batch Normalization (BatchNorm), and nonlinear activation. The output of this process can be expressed as follows:

$$f = \delta\left(F_1([z^h, z^w])\right) \tag{3}$$

The variable $f \in R^{C/r \times (H+W)}$ represents an intermediate feature that incorporates both horizontal and vertical spatial information, where r is a constant regulating the channel count, and δ denotes batch normalization (BatchNorm). Subsequently, f undergoes a segmentation into two distinct feature maps, $f^h \in R^{C/r \times H}$ and $f^w \in R^{C/r \times W}$, along the spatial dimension. Two 1×1 convolutions, denoted as F_h and F_w, along with Sigmoid functions are employed for feature transformation. This ensures that the dimensions of the feature maps f^h and f^w align with the input X, producing outputs j^h and j^w. The output of this process can be expressed as follows:

$$j^h = \sigma(F_h(f^h)) \tag{4}$$

$$j^w = \sigma(F_w(f^w)) \tag{5}$$

Here, δ signifies the Sigmoid activation function, $F_w(\cdot)$ and $F_h(\cdot)$ denote the convolution operation.

3. Feature recalibration: The original feature map undergoes recalibration through the generated attention map. As shown in Equation (6), the original features are weighted based on the generated attention map to accentuate specific regions that demand the model's focus. Here, i and j represent spatial location coordinates on the feature map, and $X_c(i,j)$ signifies the original feature map. The outputs $j_c^h(i)$ and $j_c^w(j)$ are amalgamated into a weighting matrix for computing the recalibration weights. These weights are then multiplied with the original feature map, yielding $Y_c(i,j)$, which signifies the output feature values at position (i,j) post the processing by the coordinate attention block. The objective of this step is to optimize the feature representation to align with the attentional requirements of the model by accentuating pivotal regions:

$$Y_c(i,j) = X_c(i,j) \times j_c^h(i) \times j_c^w(j) \tag{6}$$

In SlowFast networks, the slow branching network is designed to learn spatial information, with the number of channels set to β times that of the fast branches. The increase in the number of channels implies richer features. However, treating these features equally during the learning process may lead the network to learn features that are not beneficial for behavior classification tasks. To more effectively extract spatial information related to behavior in the video, this paper introduces a CA module, called CA residual block (Figure 8b), which is introduced in each original residual block (Figure 8a) of the slow branch.

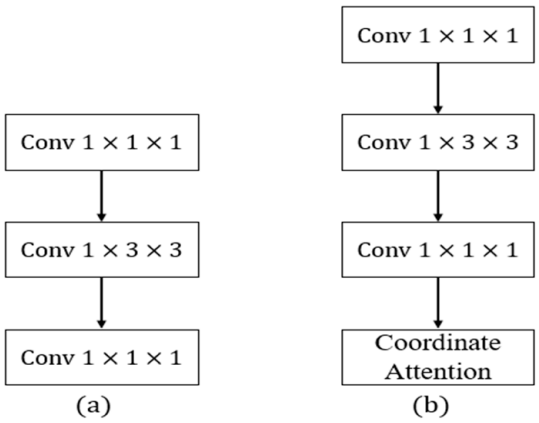

Figure 8. Original and CA residual block network structure. (**a**) Original block network structure, (**b**) CA residual block network structure.

3.2.2. Action Attention Modules

The ACTION [38] module comprises three complementary attention modules: STE (Spatio-temporal Excitation), CE (Channel Excitation), and ME (Motion Excitation). These consider three crucial aspects of temporal action: (a) spatio-temporal information, i.e., the temporal and spatial relationship of the action; (b) weighting of the temporal information of the action across different channels; and (c) the trajectory of the action between each pair of neighboring frames. Three paths—the spatio-temporal excitation path, channel excitation path, and motion excitation path—are utilized to extract the key spatio-temporal features of the video, the weights of the temporal features of the action across different channels, and the trajectory features of the action's changes between two adjacent frames. The network structure is illustrated in Figure 9.

The STE module extracts spatio-temporal features from the video by generating a spatio-temporal mask M, which results in a spatio-temporal attention map. The structure of the STE module is illustrated in Figure 10.

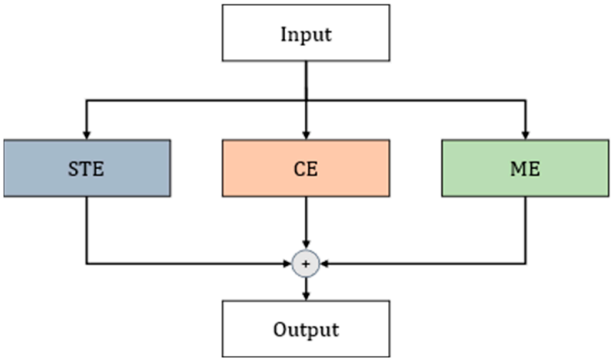

Figure 9. The structure of ACTION attention module.

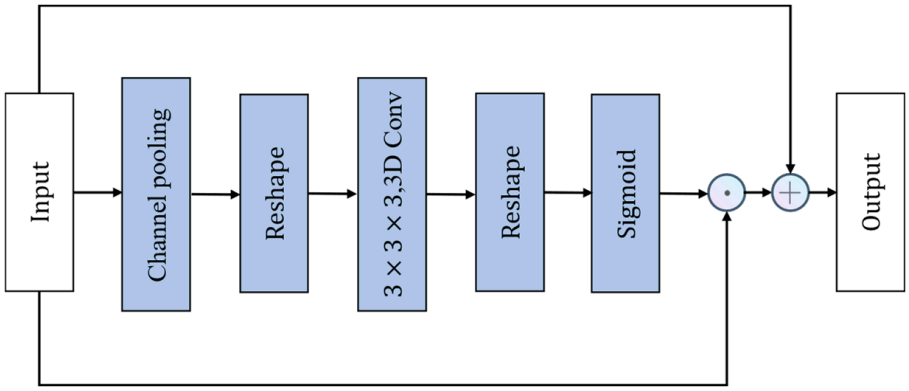

Figure 10. The structure of STE attention module.

Traditional spatio-temporal feature extraction typically involves using 3D convolution, but directly employing 3D convolution significantly increases the computational load of the model. Therefore, the Spatio-temporal Excitation (STE) module first performs channel global pooling on the input tensor $X \in R^{N\times T\times C\times H\times W}$ obtaining the corresponding channel-wise global spatio-temporal feature map $F \in R^{N\times T\times 1\times H\times W}$, Subsequently, F is reshaped into $F^* \in R^{N\times 1\times T\times H\times W}$ to make it compatible with 3D convolution operations. F^* is then convolved with a $3 \times 3 \times 3$ kernel K to produce the new spatio-temporal feature map F_o^*. The output of this process can be expressed as follows:

$$F_o^* = K * F^* \tag{7}$$

Here, N represents the batch size, T denotes the temporal dimension, represents the number of channels, H denotes the height, W denotes the width, and R represents the real domain, indicating that the elements in the feature map belong to the set of real numbers.

Then, F_o^* is reshaped to $F_o \in R^{N\times T\times 1\times H\times W}$, and the mask $M_1 \in R^{N\times T\times 1\times H\times W}$ is obtained through the Sigmoid activation function. The output of M_1 can be expressed as follows:

$$M_1 = \delta(F_o) \tag{8}$$

Finally, M_1 is multiplied element-wise with X and then summed with X to obtain the final output Y_1. The output of Y_1 can be expressed as follows:

$$Y_1 = X + X \odot M_1 \tag{9}$$

The CE module is an attention module acting on channels, adaptively recalibrating channel feature responses by explicitly modeling the interdependencies among channels in the temporal domain. The design of the CE module is similar to the SE module, with the distinction that, due to the temporal information present in video actions, the CE module

inserts a 1×1 convolutional layer in the temporal domain between two fully connected (FC) layers. This convolutional layer enhances the temporal interdependence of channels, describing the temporal information of channel features. The network structure diagram of the CE module is shown in Figure 11.

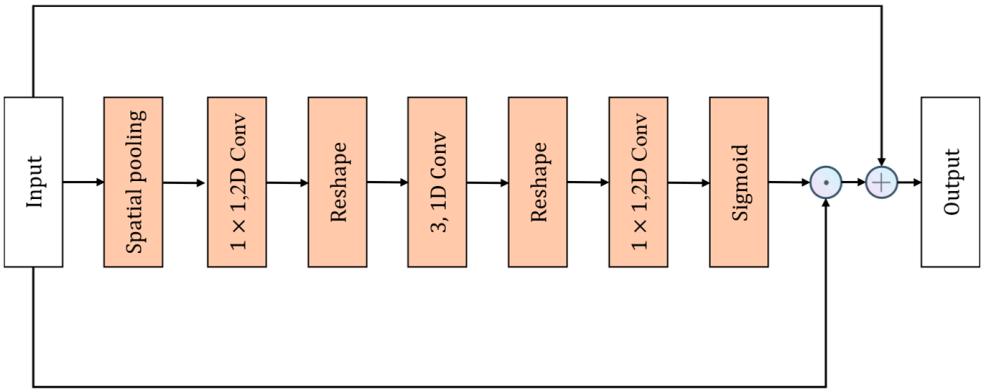

Figure 11. The structure of CE attention module.

The CE module first performs spatial average pooling on the input $X \in R^{N \times T \times C \times H \times W}$ to obtain the global spatial information tensor of the input features. The global spatial information tensor can be expressed as follows:

$$F_1 = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X[:, :, :, i, j] \tag{10}$$

The second step involves applying a 1×1 convolutional kernel K_1 to $F_1 \in R^{N \times T \times C \times 1 \times 1}$ with a channel reduction ratio r to compress the channel count of the feature map F_1, resulting in a new feature map F_r:

$$F_r = K_1 * F_1 \tag{11}$$

In the third step, $F_r \in R^{N \times T \times C/r \times 1 \times 1}$ is reshaped to obtain $F_r^* \in R^{N \times C/r \times T \times 1 \times 1}$, and in the fourth step, a one-dimensional convolutional kernel K_2 with a kernel size of 3 is applied to process F_r^*, resulting in a new feature map F_{temp}^*:

$$F_{\text{temp}}^* = K_2 * F_r^* \tag{12}$$

In the fifth step, the dimension of F_{temp}^* is modified to yield a new feature map $F_{temp} \in R^{N \times T \times C/r \times 1 \times 1}$. Subsequently, in the sixth step, a final feature map F_{o2} is derived through additional feature extraction and processing of F_{temp} using a 2D convolution kernel K_3 with a size of 1:

$$F_{\text{o2}} = K_3 * F_{temp} \tag{13}$$

and the mask $M_2 \in R^{N \times T \times C \times 1 \times 1}$ is obtained through the Sigmoid activation function. The output of M_2 can be expressed as follows:

$$M_2 = \delta(F_{\text{o2}}) \tag{14}$$

Finally, M_2 is multiplied element-wise with X and then summed with X to obtain the final output Y_2. The output of Y_2 can be expressed as follows:

$$Y_2 = X + X \odot M_2 \tag{15}$$

The Motion Excitation (ME) attention module, depicted in Figure 12, primarily captures the motion information associated with the action's movement between two adjacent

frames. This information is then concatenated as a branch with the two previously mentioned attention modules to form the ACTION module.

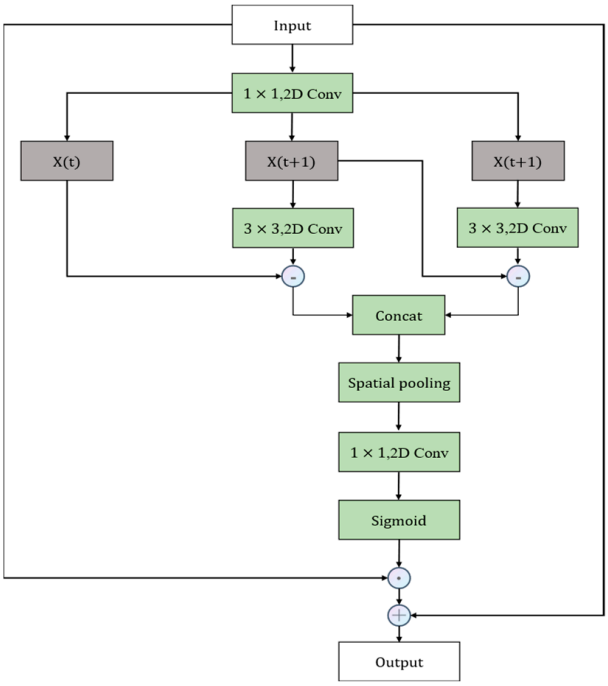

Figure 12. The structure of ME attention module.

The motion excitation (ME) attention module begins with dimension reduction through a 1×1 convolution, followed by the computation of adjacent frame features using Equation (16). Here, F_m represents the motion feature map, K_4 is a 3×3 convolutional kernel, $F_r[:, t, :, :, :]$ and $F_r[:, t+1, :, :, :]$ represent the features of the current frame and the previous frame, respectively.

$$F_m = K_4 * F_r[:, t+1, :, :, :] - F_r[:, t, :, :, :] \tag{16}$$

Subsequently, motion features are concatenated along the temporal dimension and zero-padded up to the last position. The expression is given as:

$$F_M = [F_{\mathrm{m}}(1), \cdots, F_{\mathrm{m}}(t-1), 0] \tag{17}$$

where $F_M \in R^{N \times T \times C/r \times H \times W}$ represents a series of motion feature maps concatenated along the temporal dimension, and t denotes the time dimension. Then, F_M undergoes spatial average pooling as described in Equation (10), followed by a 1×1 convolution, and subsequent unsqueeze dimensionality upscaling, as specified in Equation (13), to obtain a new feature map F_{o3}, and the mask $M_3 \in R^{N \times T \times C \times 1 \times 1}$ is obtained through the Sigmoid activation function. The output of M_3 can be expressed as follows:

$$M_3 = \delta(F_{o3}) \tag{18}$$

Finally, M_3 is multiplied element-wise with X and then summed with X to obtain the final output Y_3. The output of Y_3 can be expressed as follows:

$$Y_3 = X + X \odot M_3 \tag{19}$$

The final output Y of the ACTION module is represented as:

$$Y = Y_1 + Y_2 + Y_3 \tag{20}$$

The ACTION attention module models internal features of the network at the feature level, enhancing the representation of diverse information in videos for better action representation. The original fast branch of the SlowFast network extracts spatio-temporal features to represent actions using stacked residual blocks with $3 \times 1 \times 1$ convolution kernels. However, the extensive intra-class diversity necessitates learning more fine-grained spatio-temporal features. By introducing the ACTION module to capture diverse types of activation signals before each original residual block (Figure 13a) in the fast branch, and subsequently performing convolution, finer-grained features can be obtained. This refinement contributes to improved accuracy in multi-label behavior classification. In this study, an ACTION module is introduced into each original residual block of the fast branch, referred to as the ACTION residual block (Figure 13b).

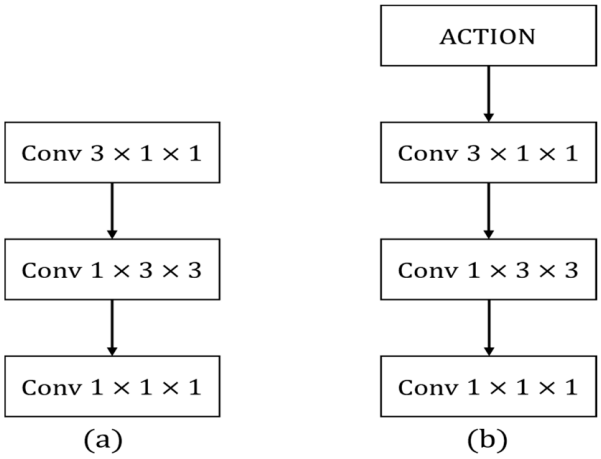

Figure 13. Original and ACTION residual block network structure. (**a**) Original residual block network structure, (**b**) ACTION residual block network structure.

3.2.3. Loss Function

This paper employs the Binary Cross-Entropy Loss (BCE Loss). Different categories of physical labor behaviors in the field are not mutually exclusive; instead, they can occur simultaneously. Therefore, normalizing the output using Softmax to probability values between [0, 1] (summing to 1) is not applicable. In the introduced multi-label classification task, the network's output data are fed into the Sigmoid function to scale the values between [0, 1]. Subsequently, they are treated as multiple binary classification problems. The Sigmoid function—as represented by Formula (21), where y denotes the output probability, z is the input, and BCE Loss, as per Formula (22)—calculates the loss for each category and obtains the total loss through summation. Here, t represents the true label values, w denotes the weight, and n is the sample count.

$$y = \frac{1}{1 + e^{-z}} \tag{21}$$

$$L(y, t) = -\frac{1}{n}\sum w \times (t \log(y) + (1 - t) \times \log(1 - y)) \tag{22}$$

4. Experiments and Results

To ascertain the accuracy of CA-ACTION-SlowFast in the realm of outdoor human labor behavior image recognition, experiments were conducted on the FWBD dataset. Comparative analyses were performed against other classical methodologies to assess its efficacy.

4.1. Training Setup

In this study, the experimental setup utilized the Ubuntu 18.04 operating system (Lenovo, Hohhot, Inner Mongolia, China), with a GPU (RTX 3090, 24 G) and a CPU (Intel(R) Xeon(R) Platinum 8338C). The software environment was configured with Python 3.8, CUDA 11.1, and PyTorch 1.8. The learning rate for network training was set to 0.1, employing the SGD optimization algorithm with a weight decay of 0.9. The learning rate was adjusted using the CosineAnnealingLR strategy, and other parameters included a Dropout of 0.5, a batch size of 8, and training for 70 epochs. The trend of learning rate reduction is illustrated in Figure 14.

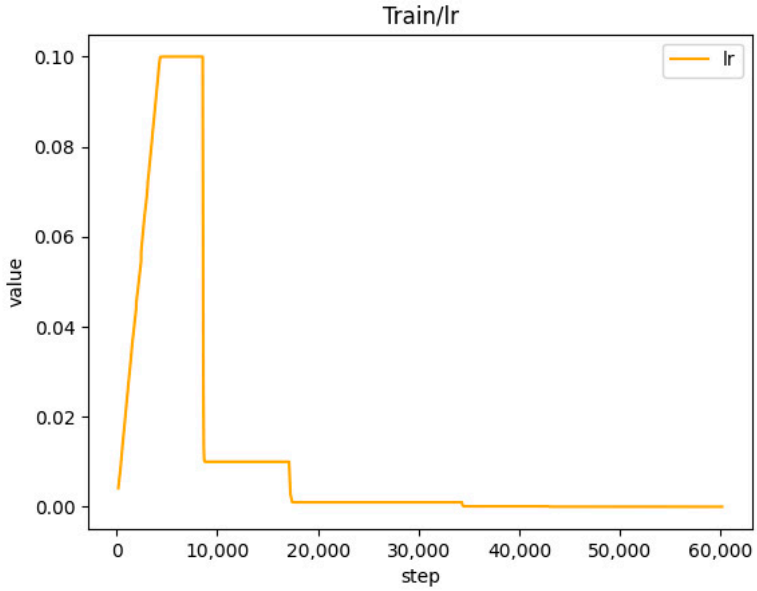

Figure 14. Learning rate change curve.

During the training process, a warm-up method was employed, initiating training with a relatively small learning rate (*lr*) in the early stages and subsequently transitioning to the pre-defined learning rate for the remainder of the training.

$$lr(t) = \frac{t}{T_{warmup}} lr_{\max}, t \leq T_{warmup} \tag{23}$$

We employ the Stochastic Gradient Descent (SGD) algorithm for model optimization, and its specific procedure is as follows. Assuming a batch of training samples, denoted as n, from which a random sample i_s is selected, let W represent the model parameters, $J(W)$ the cost function, $E(g_t)$ the gradient $\Delta(W_t)$, and φ_t the learning rate. The expression for updating model parameters using the Stochastic Gradient Descent method is given by:

$$W_{t+1} = W_t - \varphi_t g_t \tag{24}$$

Here, W_t and W_{t+1} represent the model parameters before and after the update, respectively. g_t denotes the gradient at time t. This algorithm iteratively adjusts the model parameters based on the calculated gradient and the learning rate.

4.2. Performance Assessment Metrics

We assess the performance of the enhanced network model in the behavior recognition task, where action detection accuracy serves as the paramount evaluation metric. This experiment specifically emphasizes the evaluation of the enhanced network model in this regard. Detection accuracy is typically appraised using Average Precision (AP) and mean Average Precision (mAP). These metrics offer a holistic evaluation of the model's performance across diverse categories, furnishing crucial insights into the model's proficiency in

recognizing behaviors. mAP (Mean Average Precision) is the mean accuracy calculated as the average of accuracies across all action categories. AP (Average Precision) is computed as the average accuracy for each individual category. The expressions are as follows:

$$mAP = \frac{AP}{num_{classes}} \tag{25}$$

$$AP = \int_0^1 \frac{TP^2}{(TP + FP)(TP + FN)} dR \tag{26}$$

In these formulas TP (True Positives) represents the number of instances correctly identified as positive, FP (False Positives) is the count of negative instances wrongly identified as positive, FN (False Negatives) denotes the number of negative instances incorrectly identified as negative, and $num_{classes}$ represents the total number of behavior categories being assessed.

The Intersection over Union (IoU) is a frequently employed evaluation metric in target detection. It measures the overlap between the predicted frame and the actual labeled frame; specifically, the ratio of their intersection to their union. The expression is as follows:

$$IoU = \frac{A \cap B}{A \cup B} \tag{27}$$

A represents the predicted frame, and B represents the actual frame. In practical applications, a threshold value is typically defined for IoU. When IoU exceeds this threshold, the detection is considered successful. In the context of behavioral recognition, we introduce the concept of IoU, which refers to the threshold for the overlapping area between the model's predicted bounding boxes and the actual bounding boxes. In the subsequent experiments, we compare AP and mAP when IoU is set to 0.5.

4.3. Ablation Study

To assess the roles of the CA module and the action module in SlowFast and to visually demonstrate the improvement of the enhanced model in detecting various behaviors, ablation experiments were conducted on each SlowFast model. The results for the detection of the seven behavior categories were compared. Table 3 presents the outcomes of the ablation experiments, where $\surd$ indicates that the module was used, and $\times$ indicates that the module was not used. The line graph depicted in Figure 15 illustrates the performance variations of each model throughout the experiment. In this graph, critical moments during model training are recorded every five rounds of experiments. Each folded line represents the performance curve of the corresponding model. By examining these curves, we can comprehend the model's performance in different training phases and obtain detailed timing information regarding the experimental results.

Table 3. Comparative results of ablation experiment (%).

Model	CA	ACTION	Category of AP@0.5							mAP@0.5
			Farm	Manure	Seed	Sow	Spray	Water	Weed	
1	×	×	74.19	95.92	93.37	94.17	73.80	86.99	75.25	84.78
2	$\surd$	×	76.63	98.15	96.44	96.91	81.82	91.55	80.63	88.74
3	×	$\surd$	80.28	96.83	97.39	95.05	83.13	93.47	84.14	90.04
4	$\surd$	$\surd$	84.28	98.54	97.96	97.93	84.09	94.10	86.15	91.86

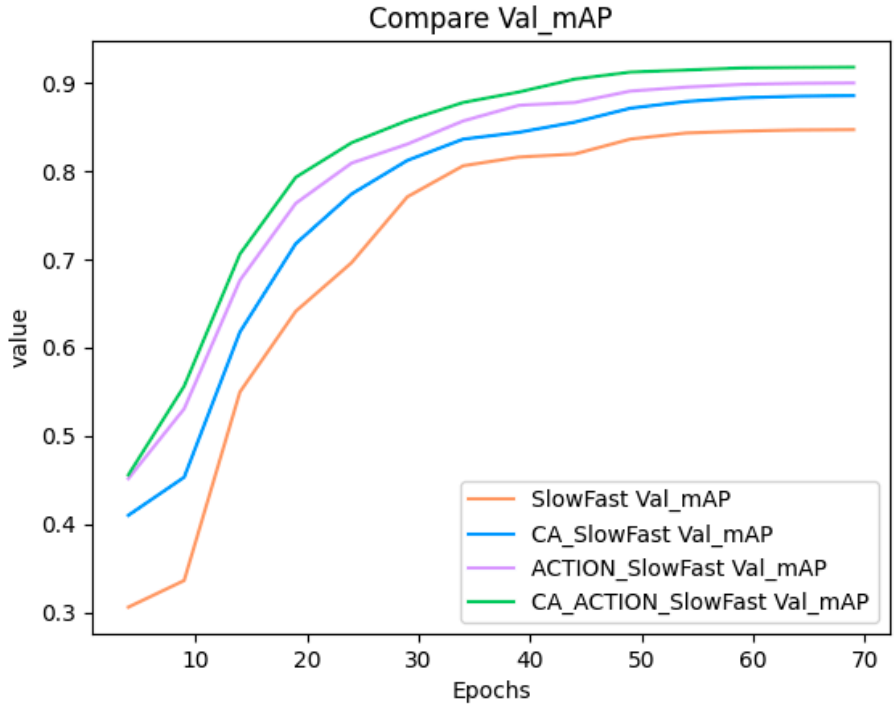

Figure 15. Comparison of Model val_mAP.

From Table 3, it is evident that the original model (Model 1) exhibits an average recognition performance for "farm", "spray", and "weed" behaviors, with an overall average accuracy (mAP) of 84.78%. In Model 2, which introduces the CA module, the overall recognition accuracy improves. Notably, the average accuracy of "spray" behavior increases by 8.02%, "weed" behavior improves by 5.38%, and "water" behavior improves by 4.56%. These improvements indicate that the CA attention module effectively extracts crucial information related to action and morphological features by assigning weights to the channels of the feature map. This strengthens the model's capacity to learn key features and enhances target differentiation, leading to an overall model accuracy improvement. The mAP increases by 3.96% compared to the original model, underscoring the effectiveness of the CA module in extracting key feature information. This outcome supports the efficacy of the CA module in extracting crucial feature information. In Model 3, where the ACTION module is introduced into the fast branch, there is an overall improvement in recognition accuracy, particularly for behaviors with high temporal feature requirements. Notably, the average accuracy for the "farm" behavior improves by 6.09%, and the "weed" behavior sees a significant improvement of 8.89%. This highlights the advantage of the ACTION module in integrating temporal and spatial information. The module ensures more comprehensive feature extraction of temporal information, not only emphasizing the spatial features of the image but also capturing changes in these features over time. In tasks requiring the capture of dynamic target features, such as behavior recognition or object tracking in a video, the inclusion of the ACTION module enhances the model's ability to understand and capture subtle temporal changes. Consequently, the overall mAP improves by 5.26% compared to the original model. Comparing Model 2 with Model 3, it is observed that, while Model 2 is not as effective as Model 3 overall, the CA module exhibits greater efficacy than the ACTION module in the behavioral categories of "manure" and "sow". These categories, being smaller in magnitude, may not be as reliant on timing information as the ACTION module would suggest. This underscores the effectiveness of the CA module in augmenting the model's recognition of crucial spatial features in the current frame. It suggests that the CA module may be more proficient than the ACTION module in extracting non-temporal key features. Model 4, which integrates the CA and ACTION modules, demonstrates superior overall performance by leveraging

the enhancements from both branches. The Average Precision (AP) of each category surpasses that of Model 1, Model 2, and Model 3, resulting in a 7.08% improvement in mean Average Precision (mAP) compared to the original model. The combination of these two attention mechanisms reveals complementary effects for various behavioral categories. The CA module significantly enhances performance for categories abundant in channel features or requiring fine discrimination. Conversely, the ACTION module proves more crucial for categories emphasizing action coherence and temporal information. The integration of both mechanisms enables the model not only to possess robust recognition capabilities at a specific temporal instance but also to sustain this proficiency across time series, presenting a distinct advantage for target detection in dynamic scenarios. This implies that the synergy and complementarity of the two attentional mechanisms result in cumulative improvements in model performance. The divergent outcomes observed in each model underscore the distinct roles and advantages of the attentional mechanisms in feature extraction and their impact on enhancing the accuracy of specific types of behavior recognition. Based on these experimental findings, we can deduce that both the CA and ACTION attention modules prove effective in enhancing the SlowFast model, particularly when employed in conjunction.

Table 4 serves as a model performance comparison. It consists of three columns, with the first column indicating the model names, the second column displaying the total number of model parameters (millions), and the third column showing the floating-point operations, i.e., FLOPs (G). The SlowFast model has approximately 33.646 million parameters, with FLOPs of around 40.82 G. The ACTION-SlowFast model, incorporating the ACTION attention module, has nearly the same number of parameters as the baseline model, but its FLOPs are slightly higher, at 40.87 G. The CA-SlowFast model, incorporating the CA attention module, has a parameter count of 59.46 million, indicating it is more complex than the baseline model, with FLOPs also slightly higher, at 41.48 G. The final CA-ACTION-SlowFast model contains approximately 59.456 million parameters, with total FLOPs of about 41.52 G. Although the CA-ACTION-SlowFast model has more parameters, its FLOPs increase slightly compared to the original SlowFast model, with the increase not being significant. This may be attributed to the increased computational effort during feature extraction with the introduction of the CA and ACTION modules. However, due to some degree of optimization within these modules, the overall increase in FLOPs is not substantial.

Table 4. Model performance comparison.

Comparison Model	Total Number of Parameters (Million)	FLOPs (G)
SlowFast	33.646793	40.82
CA-SlowFast	59.456073	41.48
ACTION-SlowFast	33.647206	40.87
CA-ACTION-SlowFast	59.456486	41.52

4.4. Comparison of Results on FWBD

To comprehensively assess the performance of CA-ACTION-SlowFast in detecting manual labor behavior in the field, we select Slow Only, YOWO [39], and YOWOv2-Large [40] as comparative methods for the experiments in this study. Table 5 illustrates the outcomes of each methodology applied to our proprietary FWBD dataset, where mAP@0.5 denotes the mean Average Precision at an overlap area threshold (IoU) of 0.5 during computation.

Three distinct methods were selected for comparison: Slow Only, YOWO, and YOWOv2-Large. These methods represent diverse behavior detection techniques and model architectures. Slow Only is a model centered on behavior recognition using solely slow frame sequences, emphasizing the capture of detailed features in high-quality images. However, it may not give sufficient attention to the temporal information of the action. Its mAP@0.5

value is 66.96%, indicating its performance in the absence of temporal dimension feature fusion. You Only Watch Once (YOWO) combines 2D and 3D convolutional networks to process both spatial and temporal information for behavior recognition in videos. Its mAP@0.5 value is 78.71%, showing a notable improvement over Slow Only. This suggests that combining spatial and temporal features is crucial for behavior recognition. YOWOv2-Large is an enhanced version of YOWO, utilizing a larger network to further improve performance. Its mAP@0.5 value is 83.79%, indicating that recognition accuracy can be further enhanced by increasing model complexity. Our proposed CA-ACTION-SlowFast model combines features from both slow and fast frame sequences and integrates both CA and ACTION modules. The CA module enhances the model's focus on key channel features, while the ACTION module not only emphasizes spatial features but also considers their temporal changes. This proves beneficial for behaviors in complex scenes, such as fieldwork. This combination allows the model to consider both image details and dynamic behavior features, providing more comprehensive information for behavior identification. The mAP@0.5 value of the CA-ACTION-SlowFast model is 91.86%, surpassing that of any of the classical methods mentioned above and highlighting its superior performance in capturing manual labor behavior in the field.

Table 5. Results of comparison with other methods on FWBD dataset.

Comparison Models	mAP@0.5
Slow Only	66.96
YOWO	78.71
YOWOv2-Large	83.79
CA-ACTION-SlowFast (ours)	91.86

4.5. Scenario Test

We conducted tests on realistic scenes of agricultural labor using the trained CA-ACTION-SlowFast model. Figures 16–20 illustrate selected video frames after behavior detection with CA-ACTION-SlowFast in different scenes. In Figure 16, rice seedling transplantation in a rice field is depicted, while Figure 17 captures watering on land reclaimed for crop cultivation in a farm yard. Figure 18 showcases hoeing in a weedy field, Figure 19 portrays weed removal from the edge of the field, and Figure 20 demonstrates pesticide spraying in a vegetable field. The model consistently maintains high recognition accuracy across all scenarios, achieving remarkable performance, particularly in actions such as rice planting and hoeing where the target is not occluded. Even in backgrounds with an abundance of green plants, the model exhibits high recognition accuracy for weeding behavior, distinguishing it from pesticide spraying actions. This suggests that the model emphasizes human behavioral characteristics during feature extraction. The enhanced CA-ACTION-SlowFast model, augmented with the CA and ACTION modules, demonstrates robust adaptability across diverse scenarios. These improvements are anticipated to yield superior results in various practical applications.

Figure 16. Resulting graph of "sow" behavior detection of video frames.

Figure 17. Resulting graph of "water" behavior detection of video frames.

Figure 18. Resulting graph of "farm" behavior detection of video frames.

Figure 19. Resulting graph of "weed" behavior detection of video frames.

Figure 20. Resulting graph of "spray" behavior detection of video frames.

5. Conclusions

In this study, we employ the Faster-RCNN + SlowFast spatio-temporal action detection algorithm for the purpose of detecting manual labor behavior in agricultural fields. Additionally, we curate a custom dataset named the Field Manual Labor Behavior Dataset (FWBD). The Faster-RCNN model is employed to identify personnel targets within key frame images. Subsequently, the detected targets are input into the SlowFast model's fast and slow channels based on varying frame counts, facilitating the accomplishment of spatio-temporal action recognition tasks.

The SlowFast model undergoes innovative enhancement through the incorporation of the CA attention module and ACTION attention module, resulting in improved recognition

accuracy compared to the original model. Although the improved model has more parameters, the increase in FLOP compared to the original SlowFast model is not substantial. This indicates that while the CA-ACTION-SlowFast model may have increased the complexity to some extent, it did not significantly increase the computational load during the inference phase. This is beneficial for maintaining the efficiency and utility of the model in practical applications. Furthermore, experimental findings reveal that the CA-ACTION-SlowFast model enhances the mAP by 7.08% in comparison to the original model. The notable performance of the CA-ACTION-SlowFast model underscores the potential for enhancing behavioral detection accuracy through the integration of spatial and dynamic features. Leveraging channel attention and temporal attention proves effective in bolstering the recognition of key features. The substantial improvement emphasizes the significance of the CA and ACTION modules. The model's capability to prioritize more informative channels proves to be a viable strategy, particularly in real-world scenarios with disturbances. This strategy assists the model in filtering out irrelevant features, allowing it to focus on components critical to the recognition task. Ultimately, the CA-ACTION-SlowFast model offers a viable solution for practical applications, notably in the realm of detecting manual labor behaviors in agricultural settings. The CA and ACTION modules may improve the performance of existing systems by providing more granular contextual information and action recognition capabilities, and can add additional layers and depth to improve overall system understanding and responsiveness. This advancement promotes the digitization and standardization of farm workers, driving traditional agriculture toward further automation and smart farming, which is of significant importance.

Future work will focus on the following aspects. First, integrating the improved network into lightweight monitoring devices. This integration has the potential to effectively capture behavioral events, significantly enhancing the functionality and utility of monitoring devices, and providing valuable information for traceability systems. Second, further exploring the model's structure and conducting experiments on other publicly available datasets to compare results aim to improve the model's generalization capability and increase the model's evaluation metrics to improve its reliability for real-world applications. Eventually, this will enrich the quantity and diversity of datasets. Only large-scale datasets can effectively validate the model's performance. Additionally, including videos from different weather conditions and different times of day in each category can enhance the model's robustness against changes in lighting and climate.

Author Contributions: Methodology, M.Z.; software, M.Z.; validation, X.M.; formal analysis, Y.Z.; data curation, J.G.; writing—original draft preparation, M.Z.; writing—review and editing, X.J.; visualization, X.Y.; supervision, X.J.; project administration, X.J.; funding acquisition, X.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (Nos. 62061037, 31960494), Science and Technology Major of Inner Mongolia Autonomous Region of China (No. 2021ZD003), Natural Science Foundation of Inner Mongolia Autonomous Region of China (No. 2023LHMS06017).

Institutional Review Board Statement: The paper focuses on the identification of human behaviors in field manual labor. As the study does not involve intervention or manipulation of individuals, nor does it involve the collection of personal or sensitive information, it is reasonable that ethics approval was not sought. The study solely observes and analyzes characteristics and patterns of human behavior, without implications of ethical concerns or risks, thus not requiring approval from a research ethics committee.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: If data is needed, interested parties may contact the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sharma, S.; Verma, K.; Hardaha, P. Implementation of artificial intelligence in agriculture. *J. Comput. Cogn. Eng.* **2023**, *2*, 155–162. [CrossRef]
2. Huang, T.; Xiong, B. Space comparison of agricultural green growth in agricultural modernization: Scale and quality. *Agriculture* **2022**, *12*, 1067. [CrossRef]
3. Zhang, Z.; Li, P.; Zhao, S.; Lv, Z.; Du, F.; An, Y. An adaptive vision navigation algorithm in agricultural IoT system for smart agricultural robots. *Comput. Mater. Contin.* **2021**, *66*, 1043–1056. [CrossRef]
4. Deveci, M.; Brito-Parada, P.R.; Pamucar, D.; Varouchakis, E.A. Rough sets based Ordinal Priority Approach to evaluate sustainable development goals (SDGs) for sustainable mining. *Resour. Policy* **2022**, *79*, 103049. [CrossRef]
5. Liang, C.; Shah, T. IoT in Agriculture: The Future of Precision Monitoring and Data-Driven Farming. *Eig. Rev. Sci. Technol.* **2023**, *7*, 85–104.
6. Abioye, E.A.; Hensel, O.; Esau, T.J.; Elijah, O.; Abidin, M.S.Z.; Ayobami, A.S.; Yerima, O.; Nasirahmadi, A. Precision irrigation management using machine learning and digital farming solutions. *AgriEngineering* **2022**, *4*, 70–103. [CrossRef]
7. Jia, R. Attitude estimation algorithm for low cost MEMS based on quaternion EKF. *Chin. J. Sens. Actuators* **2014**, *27*, 90–95.
8. Valujeva, K.; Freed, E.K.; Nipers, A.; Jauhiainen, J.; Schulte, R.P. Pathways for governance opportunities: Social network analysis to create targeted and effective policies for agricultural and environmental development. *J. Environ. Manag.* **2023**, *325*, 116563. [CrossRef]
9. Zhang, Q.; Zhang, M.; Chen, T.; Sun, Z.; Ma, Y.; Yu, B. Recent advances in convolutional neural network acceleration. *Neurocomputing* **2019**, *323*, 37–51. [CrossRef]
10. Zhu, Y.; Li, X.; Liu, C.; Zolfaghari, M.; Xiong, Y.; Wu, C.; Zhang, Z.; Tighe, J.; Manmatha, R.; Li, M. A comprehensive study of deep video action recognition. *arXiv* **2020**, arXiv:2012.06567.
11. Kalogeiton, V.; Weinzaepfel, P.; Ferrari, V.; Schmid, C. Action tubelet detector for spatio-temporal action localization. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4405–4413.
12. Song, L.; Zhang, S.; Yu, G.; Sun, H. Tacnet: Transition-aware context network for spatio-temporal action detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 11987–11995.
13. Li, Y.; Wang, Z.; Wang, L.; Wu, G. Actions as moving points. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Part XVI 16. pp. 68–84.
14. Girdhar, R.; Carreira, J.; Doersch, C.; Zisserman, A. Video action transformer network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 244–253.
15. Wu, J.; Kuang, Z.; Wang, L.; Zhang, W.; Wu, G. Context-aware rcnn: A baseline for action detection in videos. In Proceedings of the Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Part XXV 16. pp. 440–456.
16. Liu, S.; Jiang, M.; Kong, J. Multidimensional prototype refactor enhanced network for few-shot action recognition. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 6955–6966. [CrossRef]
17. Zhao, J.; Zhang, Y.; Li, X.; Chen, H.; Shuai, B.; Xu, M.; Liu, C.; Kundu, K.; Xiong, Y.; Modolo, D. Tuber: Tubelet transformer for video action detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 13598–13607.
18. Simonyan, K.; Zisserman, A. Two-stream convolutional networks for action recognition in videos. *arXiv* **2014**, arXiv:1406.2199.
19. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Van Gool, L. Temporal segment networks: Towards good practices for deep action recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 20–36.
20. Carreira, J.; Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
21. 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, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
22. Hara, K.; Kataoka, H.; Satoh, Y. Learning spatio-temporal features with 3d residual networks for action recognition. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 3154–3160.
23. Qiu, Z.; Yao, T.; Mei, T. Learning spatio-temporal representation with pseudo-3d residual networks. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 5533–5541.
24. Zhao, W.; Chen, X.; Li, Y.; Xu, J.; Li, X. A recognition of farming behavior method based on EPCI-LSTM model. *Comput. Electron. Agric.* **2021**, *190*, 106467. [CrossRef]
25. Xu, J.; Zhao, W.; Wei, C.; Hu, X.; Li, X. A model for recognizing farming behaviors of plantation workers. *Comput. Electron. Agric.* **2022**, *202*, 107395. [CrossRef]
26. Yang, X.; Pan, L.; Wang, D.; Zeng, Y.; Zhu, W.; Jiao, D.; Sun, Z.; Sun, C.; Zhou, C. FARnet: Farming Action Recognition from Videos Based on Coordinate Attention and YOLOv7-tiny Network in Aquaculture. *J. ASABE* **2023**, *66*, 909–920. [CrossRef]
27. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. Slowfast networks for video recognition. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6202–6211.
28. Sun, X.; Wu, P.; Hoi, S.C. Face detection using deep learning: An improved faster RCNN approach. *Neurocomputing* **2018**, *299*, 42–50. [CrossRef]

29. Xian, T.; Li, Z.; Zhang, C.; Ma, H. Dual global enhanced transformer for image captioning. *Neural Netw.* **2022**, *148*, 129–141. [CrossRef]
30. Li, X.; Guo, Q.; Lin, D.; Li, P.; Feng, W.; Wang, S. MISF: Multi-level interactive Siamese filtering for high-fidelity image inpainting. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 1869–1878.
31. Guo, Y.; Du, S. Advances in the applications of deep learning technology for livestock smart farming. *Smart Agric.* **2023**, *5*, 52–65.
32. Wu, H.; Song, C.; Yue, S.; Wang, Z.; Xiao, J.; Liu, Y. Dynamic video mix-up for cross-domain action recognition. *Neurocomputing* **2022**, *471*, 358–368. [CrossRef]
33. Gong, T.; Chen, K.; Wang, X.; Chu, Q.; Zhu, F.; Lin, D.; Yu, N.; Feng, H. Temporal ROI align for video object recognition. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtually, 2–9 February 2021; pp. 1442–1450.
34. Cui, Z.; Lu, N. Feature selection convolutional neural networks for visual tracking. *arXiv* **2018**, arXiv:1811.08564. [CrossRef]
35. Yang, Y.; Sun, Q.; Zhang, D.; Shao, L.; Song, X.; Li, X. Improved Method Based on Faster R-CNN Network Optimization for Small Target Surface Defects Detection of Aluminum Profile. In Proceedings of the 2021 IEEE 15th International Conference on Electronic Measurement & Instruments (ICEMI), Nanjing, China, 2–4 November 2021; pp. 465–470.
36. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 1–17 September 2017; pp. 3645–3649.
37. Hou, Q.; Zhou, D.; Feng, J. Coordinate attention for efficient mobile network design. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13713–13722.
38. Wang, Z.; She, Q.; Smolic, A. Action-net: Multipath excitation for action recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 13214–13223.
39. Köpüklü, O.; Wei, X.; Rigoll, G. You only watch once: A unified cnn architecture for real-time spatiotemporal action localization. *arXiv* **2019**, arXiv:1911.06644.
40. Yang, J.; Dai, K. YOWOv2: A Stronger yet Efficient Multi-level Detection Framework for Real-time Spatio-temporal Action Detection. *arXiv* **2023**, arXiv:2302.06848.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.