Hybrid-3D-Convolutional-Transformer Model for Detecting Stereotypical Motor Movements in Autistic Children During Pre-Meltdown Crisis

Salma Kammoun Jarraya and Marwa Masmoudi *

Posted Date: 18 October 2024

doi: 10.20944/preprints202410.1451.v1

Keywords: Autism; Stereotypical Motor Movements; Pre-meltdown Crisis; Hybrid 3D Convolutional and Transformer Model; Deep Learning Approaches

Article

Hybrid-3D-Convolutional-Transformer Model for Detecting Stereotypical Motor Movements in Autistic Children During Pre-Meltdown Crisis

Salma Kammoun Jarraya [1,2] and Marwa Masmoudi [2,*]

[1] CS Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, KSA, Saudi Arabia
[2] Mir@cl Laboratory, University of Sfax, Sfax, Tunisia
[*] Correspondence: marwa.masmoudi19@gmail.com

Abstract: Computer Vision using Deep Learning algorithms has served numerous human activity identification applications, particularly those linked to safety and security. However, despite the fact that autistic children are frequently exposed to danger as a result of their activities, many Computer Vision experts have shown little interest in their safety. High-grade autistic children frequently experience the Meltdown Crisis condition, characterized by hostile behaviors and loss of control. This study aims to introduce a monitoring system capable of predicting the Meltdown Crisis condition early and alerting the children's parents or caregivers before entering more difficult settings. For this endeavor, the suggested system was constructed using a combination of a pre-trained Vision Transformer (ViT) model (Swin-3D-b) and a Residual Network (ResNet) architecture to extract robust features from video sequences in order to extract and learn the spatial and temporal features of the Stereotyped Motor Movements made by autistic children at the beginning of the Meltdown Crisis state, which is referred to as the Pre-Meltdown Crisis state. In order to attain a 92% recall and F1 Score, the final decisions made for data preparation, model construction, and training parameters were tweaked and established experimentally. The best loss value obtained was 0.08. The MeltdownCrisis dataset, which includes realistic scenarios of autistic children's behaviors in the Pre-Meltdown Crisis state and Normal state the Normal state data being used as a negative class was utilized for evaluation.

Keywords: Autism; stereotypical motor movements; pre-meltdown crisis; hybrid 3D convolutional and transformer model; deep learning approaches

1. Introduction

Autism is a neurodevelopmental disability that affects certain individuals and is caused by a complex combination of genetic and environmental factors [1]. This condition often occurs in the first three years of life and progresses over time [2]. According to the most recent WHO prevalence data, autism affects one in every 160 children around the world [3]. Autism is also considered a neurological disorder that affects normal brain development, resulting in overall impairment, particularly in cognitive, communication, and socializing abilities [4]. In fact, autistic people have a range of symptoms that vary in intensity and degree; hence, autism is formally designated as autism spectrum disorders (ASD). Aside from communication and socialization issues, children with ASD typically exhibit significant behavioral symptoms, which can take the form of repetitive, limited, stereotyped patterns of behavior [2]. In general, stereotyped movements or behaviors are characterized as inappropriate and abnormal according to normal people's perspectives; therefore, they restrict autistic children's social and school integration. Hand flapping, head rolling, and body rocking are the most common patterns performed by autistic children according to self-stimulatory [5]. In some cases, stereotyped behaviors exceed the normal rate and become more violent and aggressive especially when the child is exposed to external stimuli like being alone, forced to do boring things, or exposed to the noising source [5].

Accordingly, high-grade autistic children are sensitive to a wide range of internal and external stimuli, exhibiting a variety of aberrant repetitive behaviors. They occasionally have a Meltdown Crisis, in which they lose control and cannot be soothed until caregiver intervention or exhaustion. In such cases, it is crucial to monitor the children's conduct and allow caregivers to intervene quickly to prevent self-harm of harm to others. On the other hand, Pre-Meltdown Crisis stereotyped behaviors are a series of high-frequency movements that indicate the commencement of the Meltdown Crisis and can be caught early to avoid Meltdown consequences [6]. For example, before a Meltdown Crisis, some autistic children show behaviors such as finger fluttering, hand flapping, covering their ears, spinning their bodies, and more [7].

In Pre-Meltdown Crisis scenarios, the autistic children become uncomfortable in situations and, therefore, show several abnormal expressions such as fear, disgust, sadness, and anger. Sometimes, their expressions reflect a mixture of multiple emotions like nervous smiles [6]. Therefore, it is important to integrate facial expressions into stereotyped physical activities to analyze the children's behaviors in Pre-Meltdown Crisis.

Detecting SMMs in autistic children is primarily an activity identification problem with special challenges; hence, numerous activity recognition algorithms have been investigated to determine which one may best address the issue of SMM detection. In this context, videos of activities are a sequence of images concatenated based on their temporal occurrence. Practical techniques for activity recognition take into account both spatial and temporal components [4]. The spatial component is concerned with the visual appearance of each frame independently, whereas the temporal component captures the movement of objects across video frames.

Traditional CV approaches are good at extracting spatial and temporal characteristics. However, prior knowledge of the topic of interest is necessary to extract significant characteristics that may be further enhanced using Machine Learning (ML) approaches. In contrast, the Deep Learning (DL) technique outperforms ML approaches because of its ability to extract features from raw data right away [5]. Multiple rounds of forward and backward data processing are used in deep learning to update model weights or features until they achieve a greater degree of confidence. Furthermore, Deep learning is not limited to a certain data type; it can handle high-dimensional data such as photos and videos. For example, the Convolutional Neural Network (CNN) is a robust deep structure designed to detect pictures and learn related characteristics efficiently without the need for intermediary engineering [6]. In addition, using a Vision Transformer (ViT) model as Temporal Swin with a ResNet-3D pretrained model offers several advantages over traditional deep models such as; Improved Attention Mechanism, Better Handling of Long-Range Dependencies, Flexibility in Input Size, Reduced Computational Cost, Interpretability. Overall, combining a Temporal Vision Transformer model as Swin with a pretrained ResNet-3D backbone can offer a powerful and flexible approach to various computer vision tasks, providing improved performance, interpretability, and efficiency compared to traditional deep models.

In this endeavor, we aim to develop a model based on the combination of pre-trained Temporal Vision Transformer (ViT) model (Temporal Swin) and a 3D-Residual Network architecture to detect Stereotyped Motor Movements (SMMs) of autistic children during the Pre-Meltdown Crisis.

The contributions of this research are summarized as follows:

- Prepared the dataset using a number of methods, resulting in noise-free sequences that retain each child's traits and record full movement information (e.g., gestures and position).
- Developed a high-performing predictive model using Temporal Swin Transformer with Renset-3D to extract and learn spatial and temporal features of autistic children's SMMs during the Pre-Meltdown Crisis.
- Conducting an empirical investigation to optimize model structure and training parameters.

The rest of the paper is structured as follows: Section 2 provides an overview of the literature on both normal human activity recognition and autistic physical activity recognition. Section 3 addresses the MeltdownCrisis dataset. Section 4 describes our study technique for the SMMs system of autistic children during the pre-meltdown crisis. As a result, we provide a hybrid architecture based on the combination of the Temporal Swin model and the 3D-Residual Network 50 (3D-ResNet). Section 5 examines and explains the efficacy of our recommended strategy. In the final section, conclusions are drawn by summarizing key findings and emphasizing the critical implications for future research and practical applications.

2. Related Works

Human activity recognition (AR) technology provides for the automatic detection of someone's actions by taking into account his or her bodily motions and functions, utilizing various technologies such as cameras, motion sensors, and location sensors. Autistic disorders can be identified by characteristics such as stereotyped motor movements, an awkward gait, and clumsiness. In addition to these characteristics, autistic disorder is distinguished by attention deficit and hyperactivity disorder, which includes hyperkinesis and stereotyping in both normal and crisis situations. To study and distinguish atypical and stereotypical motor movements in autistic children, we must first build the framework for activity recognition research by analyzing relevant material from prior studies in this subject. The following subsection highlights relevant studies on the recognition of human activity for normal people and provides an overview of research methodologies pertaining to people with autism.

2.1. Physical Activity Recognition for Normal People

A multi-modal architecture in [8] uses sequences of 3D skeletons from a Kinect tool. Three parts of the architecture were developed by Zhao et al. A dual-stream C3D is used in the first two channels to extract space-time information from 3D depth data. The 3D skeleton sequence is incorporated into the third channel to improve the accuracy of 3D behavior detection. Recognition is improved through the fusion of SVM classification results from C3D characteristics with the human skeleton. Scale and orientation invariance are guaranteed by the skeleton representation, and experimental findings show good discriminatory power even for comparable actions.

Saha et al. [9] suggested an action detection framework for recognizing action start and finish points in a video series, as well as classification. Frame-level detection and classification are performed utilizing two streams: RGB for appearance and optical flow for motion. These detections are combined based on spatial overlap and softmax probability values. Action cubes are then built from modified video frames, collecting detections with spatial overlap and class-specific scores. Action cubes are cut to maintain label integrity. Despite its great performance, the framework is not appropriate for real-time applications due to the non-incremental creation of action tubes with each new frame capture.

Singh et al. [[10] offered modifications to allow for early action prediction as video frames were streamed. The newer version is structurally similar to [9], although it contains optimized components. For starters, real-time optical flow minimizes the time it takes to compute motion streams. Second, a Single Shot Multi-Box Detector (SSD) is employed for regression and classification. Third, an efficient, greedy algorithm creates action tubes frame by frame, detecting actions based on prior frames. Following the observation of k consecutive frames, action tubes are clipped. This version is seven times faster and has improved accuracy. Performance in [10] is presented as the video advances, with 48% accuracy after seeing the first 10% of movies in the J-HMDB-21 dataset.

Continuing efforts aimed at action recognition in real-time streams and untrimmed sequences, Carrara et al. [11] proposed a lightweight architecture that uses an LSTM-based network to annotate untrimmed human skeleton sequences in real time. Frame-level annotation is used to reduce delay while correctly defining activity boundaries. LSTM hidden states assign probabilities to each class for each incoming stream pose, which are then modified by a learning threshold to ensure exact action demarcation. The suggested method annotates approximately 7700 frames in one second and has higher accuracy than online detection approaches.

In the context of finding more advanced deep learning approaches to handle the action recognition problem, we found that Deep Reinforcement Learning (DRL) is one of the most promising methods in this domain. In [12], DRL was utilized, like in many other computer vision video-based efforts, to select the most useful frames while eliminating ambiguous ones in sequences for accurate action recognition. In brief, the AR suggested technique consists of two sub-networks: (1) FDNet, which selects key-frames from the action sequence using a progressive DRL, and (2) a modified CNN model dubbed graph-based CNN (GCCN), which recognizes the action label. The point is that FDNet-generated key-frames are turned into graph-based data before being fed into GCCN to learn spatial connections between joints, as illustrated in Figure 6. During the training stage, GCNN and FDNet (key-frame selector) complement each other. GCCN results are utilized as rewards for FDNet, which then selects keyframes to refine the GCCN in return.

Another study adopted DRL to optimize action detection steps in untrimmed videos instead of using extensive proposal generation or an expensive sliding window [13]. LI et al. [13] proposed using the Markov Decision Process (MDP) to detect actions. An agent modifies its location and scale via seven transformations within a temporal window, with the goal of covering actual action zones. Through training, the agent learns the best policy based on the negative and positive rewards for each action. This policy directs the search while testing. Deep Q-Network (DQN) learns the best agent policy for producing accurate action proposals. LSTM and CNN models encode action-deep features for classification. The framework's generality and efficiency in testing runtime are noted, despite the fact that it is not suited for real-time applications.

[14] suggested a novel self-supervised learning approach for video transformers that uses unlabeled video data. In this study, the authors presented a method for generating local and global spatiotemporal views from movies of varied spatial sizes and frame rates. The goal is to make the properties of these features invariant to spatiotemporal differences in activities. This technique, which does not require negative samples or specific memory banks, allows the model to interpret films with slow-fast dynamics while still supporting long-term relationship modeling across spatiotemporal dimensions. The technique outperforms action recognition benchmarks such as Kinetics-400, UCF-101, HMDB-51, and SSv2, and converges quickly even with small batch sizes. This ground-breaking methodology represents a big step toward fast and effective self-supervised learning for video analysis.

[15] developed a new framework for HAR that tackles the constraints of previous CNN-based approaches. These constraints include the difficulty of capturing long-range temporal relationships due to limited receptive fields and the high processing demands. The suggested method employs a convolution-free technique that uses Vision Transformer (ViT) for frame-level spatial feature extraction, followed by multilayer Long Short-Term Memory (LSTM) networks to capture long-range relationships. Extensive trials on the UCF50 and HMDB51 datasets revealed higher performance, with accuracy gains of 0.944% and 1.414% above state-of-the-art approaches. The findings demonstrate the framework's ability to handle problematic characteristics of HAR such as crowded backdrops, shifting views, poor resolution, and partial occlusion in surveillance videos.

[16] introduced a novel framework for HAR that uses transformers. Traditional HAR systems use convolutional neural networks (CNNs) to extract features and recurrent neural networks (RNNs) to forecast temporal sequences. These approaches, however, are computationally demanding and frequently inefficient on resource-constrained systems. The proposed ViT-ReT system uses a Vision Transformer (ViT) to extract frame-level features and a Recurrent Transformer (ReT) to forecast sequences. The ViT-ReT model is intended to increase both speed and accuracy, providing a considerable advantage over conventional techniques. Extensive studies on four benchmark datasets (YouTube Action, HMDB51, UCF50, and UCF101) show that the ViT-ReT framework outperforms the baseline ResNet50-LSTM method in terms of both accuracy and speedup. This makes the ViT-ReT framework an excellent choice for real-time HAR applications in resource-constrained contexts.

[17] presented a model that combines CNN spatial feature extraction with ViTs contextual understanding. This hybrid model, known as ConViT, improves the capacity to understand human activities by differentiating significant bits of a picture from irrelevant ones. The model is further improved by including a human categorization branch that focuses on certain portions of pictures

to help in action prediction. On the Stanford40 and PASCAL VOC 2012 action datasets, ConViT outperformed classic CNN models and other state-of-the-art approaches, with mean Average Precision (mAP) ratings of 95.5% and 91.5%, respectively.

[18] proposed a unique technique for semi-supervised action recognition utilizing transformer models, solving the issues given by the high cost of video annotations. To manage unlabeled video samples, the suggested technique, SVFormer, uses a stable pseudo-labeling architecture known as EMA-Teacher. Unlike typical convolutional neural network (CNN) techniques, SVFormer uses transformers to improve performance. In this study, the authors introduced two innovative augmentation algorithms designed for video data: Tube TokenMix, which mixes video clips with a consistent temporal mask, and Temporal Warping Augmentation, which changes the temporal length of chosen frames. Experiments on datasets like Kinetics-400, UCF-101, and HMDB-51 show that SVFormer outperforms the state-of-the-art by 31.5% with fewer training epochs than Kinetics-400 at a labeling rate of 1%.

2.2. Physical Activity Recognition for Autistic People

In this area, we include various works that investigate the stereotyped behaviors of autistic children. In fact, we have yet to find a vision-based deep framework for detecting autistic people's actions. Rad and Furlanello introduced an automatic method in [5] for recognizing and quantifying stereotyped behaviors in autistic children, allowing for early detection and, if necessary, behavioral intervention. Despite the strong system performance, the data used was collected wireless accelerometer sensors worn on children's wrists and chests, which may have made the children uncomfortable. Furthermore, this evidence was collected in laboratory and classroom settings and only capturing a subset of behavior categories, such as hand flapping and body rocking. To take a quick look at the feature extraction, this study utilized a three-layer CNN to turn the time-series of several accelerometer sensors into a new feature space that was then prepared for the classification.

On the other side, there are few studies that propose vision-based but not deep solutions for stereotyped behavior identification. For example, ([2], [19]) identify stereotyped actions by analyzing their temporal patterns and comparing them to accessible templates of movements previously registered using Kinect camera and skeleton data. To identify stereotyped behaviors, the Dynamic Time Warping (DTW) technique is utilized in [19]. In contrast, [2] uses $P - Point$ Recognizer to match temporal patterns using the Euclidean distance function. Both works ([2], [19]) have numerous traits that contradict our goal. (1) They aim to provide a supportive tool to help doctors and clinicians in diagnosing and therapists sessions, (2) training data was collected from non-autistic subjects that imitate the autistic behaviors, (3) stereotypical movements considered are limited to a small set of gestures like hand flapping, whereas there are no realistic activities and interactions; and (4) the proposed algorithms (template-based) may be exposed to failure in the case of background noise, occlusions, or other activities.

[20] suggested a new method for identifying and evaluating arm-flapping stereotypic behavior in autistic children using computer vision and machine learning approaches. The approach entails collecting frames from the input video and utilizing MediaPipe to locate and label body landmarks. The landmark data is then gathered and processed in a number of processes to identify stereotypic behavior. Finally, the system assesses the severity and frequency of the arm flapping activity. Using characteristics like human position estimation and skeleton modeling, the system can precisely track and evaluate the frequency and intensity of arm flapping, offering useful data for clinical assessment and intervention planning.

3. Theorical Study: Stereotyped Motor Movements (SMMs) of Autistic Children

People with ASD exhibit a series of nonfunctional, constrained, repetitive, and stereotyped patterns of behavior described as Stereotypical Motor Movements (SMM) [4]. SMMs affect autistic children's functioning, impede skill learning, and challenge social integration in school and community contexts [21]. The intensity and frequency of SMMs vary from kid to child according to the severity level of ASD, and the most severe degree may result in self-injurious conduct in particular circumstances. Furthermore, they do not remain consistent over time in the same individual [4]. Table 2 categorizes distinct forms of stereotypic behaviors according to their origins [22].

According to ([7], [21]), SMMs have a role in ASD diagnosis in addition to speech and socialization evaluation. The fifth version of the Diagnostic and Statistical Manual of Mental Disorders (DSM-V) includes the following factors when diagnosing SMMs: (1) severity level (mild, moderate, or severe), and (2) link to another medical problem [7]. Mild SMMs can be easily suppressed by distraction or sensory input, while moderate behaviors require specific protective measures and behavioral adjustments due to their frequency. Both mild and moderate behaviors are non-self-injurious. In contrast, severe autistic activities need ongoing monitoring and preventative measures to avoid serious damage [7].

Children with severe autism are sensitive to a variety of internal and external stimuli and situations, including sensory isolation, a loud environment, stress, and the actions of others [5]. As a result, individuals engage in several severe SMMs that can escalate into Meltdown events in which they lose control of their activities and cannot be calmed down until external behavioral intervention is provided or they grow sleepy [6]. As seen in Table 1, the autistic child in the Meltdown Crisis exhibits violent and intense actions such as striking himself, punching others, smashing his hands against the ground, placing himself on the floor, moving swiftly while rolling, and so on.

In clinical practice, several approaches for evaluating SMMs in ASD are employed, including caregiver interviews, caregiver surveys, and observational methods. (1) The Child Autism Rating Scale (CARS) [5] is a paper-and-pencil rating scale designed to differentiate autistic children's behaviors based on their severity and frequency from other problems based on general impressions and non-specific observations. (2) The Repetitive Behavior Scale-Revised (RBS-R) [23] is a questionnaire created for child caregivers to score child behavior based on a restricted number of pre-defined categories such as body rocking, hand flapping, and so on. (3) Video-based techniques [24] include analyzing the child's activities by watching video recordings of the encounter and manually offline coding them by an expert. Despite its dependability, this procedure is laborious and time-consuming; hence, it is not generally utilized. (4) Automatic detection systems ([7], [25]) are designed to catch SMMs using wireless accelerometers and analyze the signal information through pattern recognition algorithms in order to automate the measuring procedure. These devices are designed to rate autistic conduct for therapy and diagnosis purposes; however, they cannot be used for continuous monitoring since sensors on the wrist put children in unpleasant circumstances. (5) A few automated vision-based systems ([2], [19]) were created to produce a diagnostic tool utilizing the Kinect sensor because of its capacity to follow the human body with certain feature points. So, in this context, we propose to detect the SMMs automatically in a pre-meltdown crisis to alert the caregivers to the beginning of the outbreak of a meltdown crisis and protect the child from passing through a dangerous crisis.

Table 1. SMMs: descriptions and samples from public images

SMM	Description	Sample from public
Face	Grimacing, lips or tongue movements, opening the mouth, mouth stretching, sucking objects	
Head and neck	Head tilting, shaking, nodding, hair twirling, headbanging, neck stretching	
Trunk	Body rocking, spin, spinning or rotation of the entire body	
Shoulders	Bending, arching the back, shrugging the shoulders	
Arm OR leg	Arms flapping, bilateral repetitive movements involving the arms and hands such as crossing the arms on the chest, and tapping one's feet	
Hand OR finger	Hand flapping, slapping, nail-biting, finger wiggling, Shaking, tapping, waving, clapping, opening-closing, rotating or twirling the hand or fingers, thumb-sucking, pointing, fanning fingers, fluttering fingers in front of the face, picking skin, and scratch self.	
Hand OR finger with object	Shaking, tapping, banging, twirling an object, rubbing, repetitive ordering, arranging toys in patterns, adding objects to a line, manipulating objects.	

Table 1. *Cont.*

SMM	Description	Sample from public
Gait	Pacing, jumping, running, skipping, spinning.	
Self-directed	Covering the ears, mouthing, smelling, rubbing the eyes, tapping the chin, slapping self or an object or surface, and self-mutilating behavior.	

4. "MeltdownCrisis" Dataset Collection and Preprocessing

Dataset collection is quite challenging, especially when the videos are for autistic children. The previously analyzed literature shows inadequate dataset collection in terms of quality and quantity. The majority of the provided datasets are private and contain a limited number of stereotyped behaviors.

In this research, we hope to create a monitoring system that can continually collect autistic child's behavior in realistic situations and predict if any child will be in a Meltdown Crisis stage. After scanning numerous vision-based datasets, the "MeltdownCrisis" dataset [6] was chosen to create the suggested system because it contains realistic scenarios that match our goals.

The "MeltdownCrisis" dataset contains a significant number of video clips from thirteen severely autistic children aged five to nine. The documented scenarios range from a normal state to extremely active and aggressive symptoms in the Meltdown Crisis state. An expert analyzed the films based on the children's emotions and their behavior patterns. The video clips in the "MeltdownCrisis" dataset are divided into three categories: normal, pre-meltdown crisis, and meltdown crisis. This study focuses on the pre-Meltdown Crisis state; hence, it is separated into two portions. The pre-meltdown crisis state is distinguished by stereotyped actions that reveal the autistic child's distress and internal instability (cf. Figure 1).

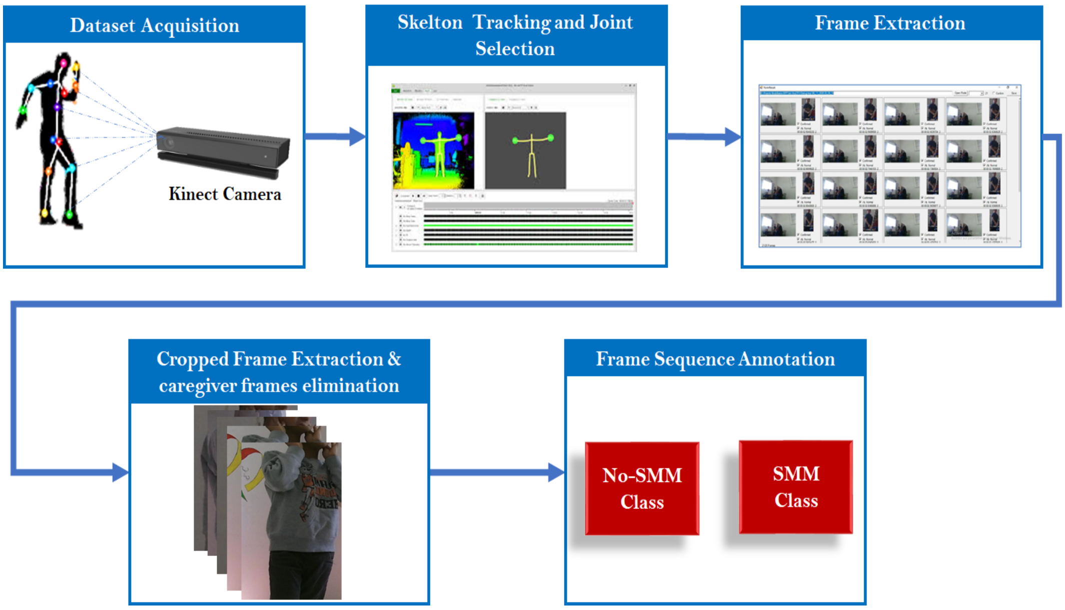
Dataset Acquisition
Kinect Camera
Skelton Tracking and Joint Selection
Frame Extraction
Cropped Frame Extraction & caregiver frames elimination
Frame Sequence Annotation
No-SMM Class
SMM Class

Figure 1. MeltdownCrisis Dataset Collection and Preprocessing

4.1. Dataset Collection

4.2. Dataset Preprocessing

After collecting the "MeltdownCrisis" dataset, we need to perform a preprocessing step to extract (cf. Figure 1;) (1) frame sequences, (2) cropped frames of each person participating in the video, (3) eliminate the cropped frames of the caregiver, and (4) annotate frame sequences as 0 for No-SMM and 1 for SMM. Hence, to develop our "Activity Form" application, we employed both the SDK V2.0 of Kinect and the API Body Basics. Our application allowed us to detect and track autistic bodies based on skeleton joints extracted from "MeltdownCrisis" videos. Thus, the cropped frames are extracted by plotting a bounding box based on the minimum and maximum values of skeleton joint coordinates that reflect the end points of person's skeleton because participant sizes vary. Then, the caregiver's body frames were deleted. Each frame can be defined based on "Time Stamp" (TS), which is used to synchronize the extracted frame sequences. Moreover, each frame sequence is relevant to one child which is identified by the child's body index. Finally, each sequence is annotated as (Class: 0) for No-SMM and (Class:1) for SMM.

5. Methodology: Hybrid 3D Convolutional and Transformer Model

Computer vision is a field of artificial intelligence that allows robots to interpret and process visual input and has evolved significantly over the years. Traditionally, computer vision relied on handcrafted features and traditional machine learning methods. However, introducing deep learning, particularly convolutional neural networks (CNNs), caused a paradigm shift. CNNs have demonstrated considerable success in applications such as image classification, object identification, and segmentation, significantly improving computer vision system performance and capabilities. In this respect, pre-trained temporal models based on CNN layers, used in computer vision, are models trained on large video datasets to grasp and capture spatial and temporal information. These models are designed to manage the intricate nature of video data, comprising static visual content inside individual frames (spatial information) and dynamic changes between frames over time (temporal information). These models have various advantages. Firstly, they improve understanding because they can completely comprehend video information by considering both the visual aspects and their evolution over time. Second, transfer learning enables pre-trained models to be fine-tuned for certain tasks or datasets, increasing their adaptability while reducing computing resource requirements. Third, these models have demonstrated greater performance in a variety of video-related tasks as compared to those that simply deal with spatial data. Finally, efficiency is significantly boosted since pre-trained models based on large-scale datasets speed up the development process by providing a strong fundamental framework.

The next important leap in computer vision is represented by the Transformer architectures, which were originally developed for natural language processing (NLP). Then, Vision Transformer (ViT) model, has shown that it can outperform CNNs on a range of vision tasks by employing self-attention mechanisms. This allows them to better imitate long-range associations in images than CNNs, which are often limited by their receptive fields. Transformers have also enabled new applications and enhanced current ones. They excel at processing large-scale datasets and have been used for a variety of tasks, including picture production, video interpretation, and multi-modal learning, which combines visual information with other data types such as text. The shift to transformer-based models in computer vision represents a wider trend of integrating techniques across diverse disciplines of AI, resulting in more robust and adaptable systems.

The Pre-trained Temporal Vision Transformers (TimeSformer) offer a substantial leap in video interpretation tasks in computer vision. These models expand the capabilities of classic Vision Transformers (ViT) by processing the temporal information inherent in video data, allowing for a more thorough study of both spatial and temporal aspects. TimeSformers use the self-attention mechanism to detect relationships not just between individual frames but also across time. TimeSformers' capacity to represent temporal dynamics enables them to succeed in tasks like action identification, video categorization, and video segmentation. In particular, Temporal Pretrained models are extremely helpful since they have been extensively trained on large-scale video datasets such as Kinetics or Moments in Time, allowing them to perform well even when task-specific data is sparse. Pretrained Temporal Vision Transformers provide several key benefits, including improved performance through better temporal context understanding, efficient transfer learning that allows for fine-tuning on specific tasks with fewer resources, and versatility across multiple domains such as surveillance, autonomous driving, sports analytics, and healthcare. This progress offers a big step forward in linking spatial and temporal data interpretation, resulting in powerful tools for complete video analysis.

The combination of a Temporal Vision Transformer (ViT) with a pre-trained temporal model improves video analysis by harnessing the capabilities of both designs. This hybrid technique enhances spatial and temporal feature extraction, leading to greater accuracy and resilience in tasks such as action identification and video segmentation. It allows for effective transfer learning, adaptation across domains, and a better grasp of temporal dynamics. This synergy results in cutting-edge performance across a wide range of applications, including surveillance, healthcare, autonomous cars, and sports analytics, making it an effective tool for complicated video interpretation jobs.

In this context, we propose a model on the combination of a pre-trained Temporal ViT model (Temporal Swin-3D-b) and a 3D-ResNet pre-trained model to detect Stereotyped Motor Movements (SMMs) of autistic children during the Pre-Meltdown Crisis. As shown in Figure 2, our methodology is composed of several development steps: (1) Video Data Augmentation, (2) Dataset Transformation, (3) local feature extraction using 3D-ResNet Backbone, (4) global feature extraction using Swin-3D-b Transformer and (5) these features are flattened to fully connected layer to be classified into **No-SMM** (Class: 0) and **SMM** (Class:1).

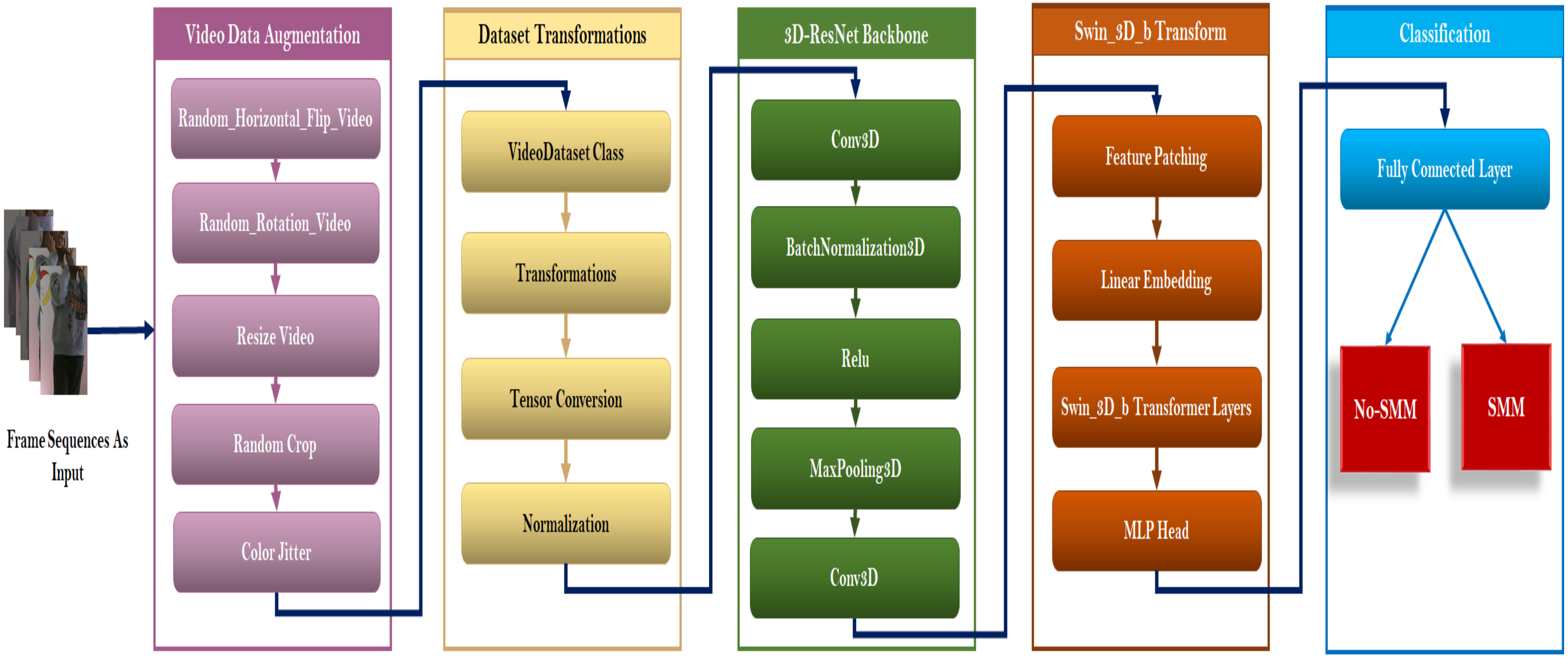

Figure 2. Evaluation Framework to detect SMM

5.1. Video Data Augmentation

Data augmentation is critical in situations where the real data is limited and obtaining additional samples of ground truth data is costly and time-consuming. These issues are frequently encountered while working with videos for autistic children, which are scarce. In such instances, data augmentation is a very effective strategy for expanding the sample size. Some of the primary causes that have led to the massive deployment of data augmentation techniques are as follows: Data unavailability, variation in data, ease of work of researchers and cost-effectiveness [26]. Hence, several approaches are proposed to handle this issue and produce image sequence samples. In this work, we employed the Random Horizontal Flip, Random Rotation, Resize Video, Random Crop, and Color Jitter approaches.

- **Random Horizontal Flip:** This is a deep learning approach that increases the amount of a dataset by flipping frames horizontally. Exposing a model to several variants of the same frame can assist increase its accuracy. This method is applied with a "p=0.5" parameter, which is the probability of the image being flipped.
- **Random Rotation:** is a data augmentation technique that rotates each frame (or picture) in a dataset by a random angle within a predetermined range. This allows the model to become invariant to the orientation of the objects in the frames. In our case, we apply this method by rotating frames randomly within a range of -15 to 15 degrees. When a frame is exposed to random rotation with a degree range of 15, it can be turned at any angle between -15 and 15 degrees. For example, one frame may be by -10 degrees, another by 5 degrees, and another by 14 degrees. The rotation angle is determined at random for each frame, ensuring that each frame may have a different rotation angle within the defined range.
- **Resize Video (256x256):** Resizing video frames to 256x256 pixels entails resizing each frame in the movie to 256 pixels in width and height. This modification guarantees that all frames have the same size, which is frequently required for input into neural network models. This method guarantees: the (1) Consistency seeing that all video frames have the proportions, which is critical for batch processing in machine learning pipelines; (2) Standardization, seeing that Models frequently demand inputs of a specific size. So, resizing all frames to 256x256 pixels standardizes the input, ensuring compatibility with the model design. And (3) Efficiency: reducing the size of the frames can minimize computational load and memory utilization, which is useful when training huge datasets.
- **Random Crop (224x224):** selecting a random area of each frame in the video and cropping it to 224x224 pixels. This transformation introduces variability in the training data, which can help improve the model's robustness and generalization.
- **Color Jitter:** this includes making random adjustments to the (1) *brightness* which controls the intensity of light in a frame. Increasing brightness makes the picture brighter, while lowering brightness makes the image darker, (2) *Contrast*, which adjusts the contrast between the bright and dark regions of a picture. Higher contrast makes the shadows deeper and the highlights brighter, whilst lower contrast makes the image look more consistent, (3) *Saturation*, which controls the strength of colors in a frame. Increasing saturation makes colors more vibrant, and reducing it makes the image more grayscale and (4) *Hue* shifts the image's hues along the color spectrum. Changing the hue can affect the colors of the items in the image. Hence, this approach makes models more resilient to fluctuations in lighting conditions and color discrepancies. In our case, this method takes a video file, applies color jitter transformations to its frames, and returns the transformed frames as tensors.

5.2. Dataset Transformation

The Video Dataset Class contains the dataset and its related modifications. Within this framework, many essential modifications are applied:

- **Transformations:** Each frame is resized to 224x224 pixels. This standardizes the frame size, which is essential for effective batch processing in neural networks.

- **Tensor Conversion:** Frames are transformed from PIL images to PyTorch tensors. This change converts the picture data format from PIL's HxWxC to PyTorch's CxHxW, while also scaling pixel values from [0, 255] to [0, 1].
- **Normalization:** Tensor values are normalized based on mean and standard deviation parameters. This normalization method reduces the data's mean to 0 and standard deviation to 1, as aided by mean= [0.485, 0.456, 0.406] and std= [0.229, 0.224, 0.225]. Such standardization improves the speed and consistency of deep learning model training.

These transformations and augmentations collectively enhance the training process, improving the model's ability to learn robust features and generalizing new data.

5.3. Local Feature Extraction using 3D-ResNet pretrained model

This stage represents the local feature extraction phase within the 3D-ResNet Backbone, a deep learning model architecture commonly applied in video processing tasks to extract spatiotemporal features from video frames using a pretrained 3D-ResNet architecture. This method is useful for a variety of video analysis applications, including action detection, video categorization, and more. In this paper, we employed this pretrained model to recognize SMMs. We load a pretrained 3D-ResNet model (**r3d_18**) and replace the fully connected layer with an identity layer (‘**nn.Identity()**), essentially transforming the model into a feature extractor. Here is a detailed description of each component within this step:

- Convolutional Layer (3D): This layer performs 3D convolution on the input video frames. Unlike 2D convolution, which only functions on spatial dimensions (height and width), 3D convolution works on both spatial and temporal dimensions (height, width, and depth/time). This layer is responsible for extracting spatio-temporal characteristics from video frames. The implementation details of the Con3D layer are illustrated in Table 2.

Table 2. Implementation details of the Con3D layer

Parameters	**Description**
	3 (RGB channels)
Input Channels	
Output Channels	128
Kernel Size	(3, 7, 7), indicating the convolution kernel spans 3 frames in time and 7x7 pixels in space.
Stride	(1, 2, 2), meaning the kernel moves 1 frame at a time in the temporal dimension and 2 pixels in spatial dimensions.
Padding	(1, 3, 3), adding padding to maintain the spatial dimensions of the output.

- **Batch Normalization (3D):** This layer normalizes the Conv3D layer's output for each mini-batch. It helps to accelerate the training process while also boosting the neural network's performance and stability. Batch normalization for 3D data is applied across the feature maps while preserving temporal coherence. Here, the input channels are equal to 128, which matches the output channels of the previous Conv3D layer.
- **The Rectified Linear Unit (ReLU):** is an element-wise activation function that adds non-linearity to the model. It outputs the input straight if it is affirmative otherwise it returns zero. This enables the network to learn more complicated patterns. The Inplace is set to True in order to alter the input immediately and save memory.
- **Max Pooling (3D):** This layer down-samples in both spatial and temporal dimensions, lowering the dimensionality of the feature maps while maintaining the most important features. Max pooling reduces computing effort while controlling overfitting by offering an abstracted form of the representation (see. Table 3).

Table 3. Implementation details of the MaxPooling 3D layer

Parameters	**Description**
Kernel Size	(1, 3, 3), indicating pooling over 1 frame in time and 3x3 pixels in space.
Stride	(1, 2, 2), meaning the kernel moves 1 frame at a time in the temporal dimension and 2 pixels in spatial dimensions.
Padding	(0, 1, 1), maintaining the spatial dimensions of the output.

- **3D Convolutional Layer:** This layer reduces the number of channels from 128 to 3, preparing the output for the next stage. The Input Channels are equal to 128, the Output Channels are equal to 3 and the Kernel Size is equal to (1, 1, 1), indicating point-wise convolution.

5.4. Global Feature Extraction Using Swin_3D_b Model

The **Swin3D Transform** section is responsible for further processing the features extracted by the 3D-ResNet backbone. This section includes the following components:

- **Feature Patching:** This stage includes breaking down the 3D features retrieved from the 3D-ResNet backbone into smaller patches. Each patch is handled as a separate token for the transformer. This approach improves the efficiency with which the spatial and temporal elements of the incoming video are handled.
- **Linear Embedding:** The patches are passed through a linear embedding layer. This layer converts the patches into dimensions that the transformer can handle. It simply translates the raw patches into a higher-dimensional space, allowing the transformer to better grasp the data.
- **Swin3D_b Transformer Layers:** These are the core layers of the Swin3D transformer. The Swin3D transformer employs a hierarchical structure with shifting windows to collect both local and global characteristics in the video. It consists of numerous layers of self-attention and feed-forward networks, which enable the model to learn complicated connections between video frames. Figure 3 shows the typical architecture of a Swin-3D-based Transformer block. The Swin Transformer design, including the 3D edition, is composed of alternating layers of window-based multi-head self-attention (W-MSA) and shifting window multi-head self-attention (SW-MSA), as well as feed-forward networks (MLP layers) and normalization layers (LayerNorm). The provided diagram is consistent with these ideas. The following is a breakdown of the components depicted in the Figure 3 and how they fit into the Swin Transformer model [27]:
  - **MLP (Multi-Layer Perceptron):** The MLP is a feed-forward network made up of two completely linked layers and a GELU activation function in the middle. This component appears in both the standard transformer block and the Swin transformer block.
  - **LayerNorm (Layer Normalization):** Layer normalization is used before the MLP and self-attention layers to help stabilize and expedite training. It assures that the inputs to these layers have zero mean and unit variance, which is useful for training deep neural networks.
  - **SW-MSA (Shifted Window Multi-Head Self-Attention):** Swin Transformer computes self-attention inside local windows, which are then moved across layers to allow for cross-window connections. This shifted window method captures both local and global dependencies efficiently.
  - **W-MSA (Window-based Multi-Head Self-Attention):** This is the standard window-based self-attention method, which calculates attention within fixed-size windows. It focuses on capturing local dependencies between non-overlapping planes.

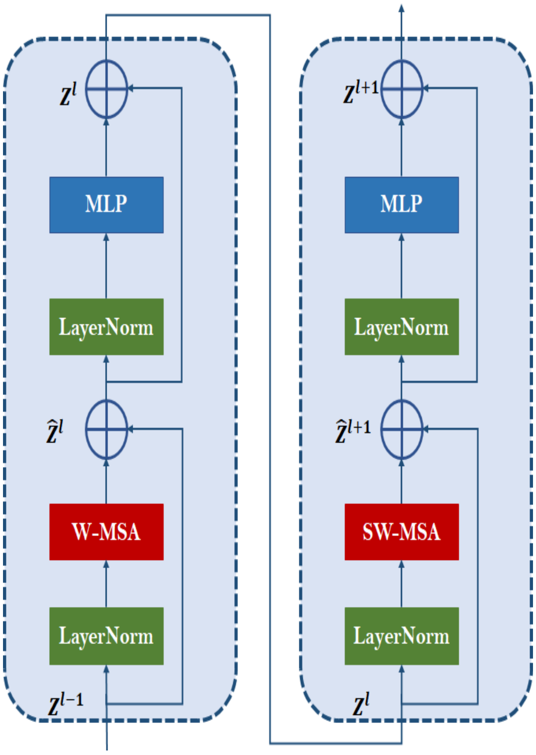

Figure 3. The base architecture of a Swin (Swin-3D-b) Transformer block

- **Multi-Layer Perceptron Head:** Following processing via the transformer layers, the resultant features are sent into a Multi-Layer Perceptron (MLP) head. The MLP head is generally made up of one or more fully connected layers with activation functions. This component serves to refine the extracted features by the transformer before they are fed into the classification layer.

5.5. Classification

The classification step is responsible for making the final predictions based on the features extracted and processed by the previous layers. The Fully Connected Layer translates the refined features from the Swin3D transformer's MLP head to the output space with the same number of classes as input. In this scenario, a basic dense layer generates logits for each class. The number of neurons in this layer is equal to the number of classes=2, (for our case, No-SMM and SMM). The final output classes are: No-SMM and SMM. The model returns probabilities for each class (**No-SMM** and **SMM**). Typically, the probabilities for each class are obtained by applying a softmax activation function to the logits from the fully connected layer. Using a fully connected layer, it converts refined features into class probabilities.

This architecture uses the benefits of both 3D-ResNet and Swin3D transformer to collect spatiotemporal features and produce good video categorization predictions. In the following section, we will present the experimental results achieved with our proposed methodology.

6. Experimental Study

In this part, an evaluation was performed to determine the most effective model for identifying stereotypical motor movements (SMMs) in autistic children.

We trained several temporal-based CNN models on the "MeltdownCrisis" dataset, which contains 623 frame sequences. Our dataset contains autistic children's SMMs acquired in uncontrolled environments and crisis situations. Body frames encompass various environmental restrictions such as luminosity, children of different sizes, occultation, etc. The networks presented in Table 3 are trained with several hyper-parameters values. Our evaluation focuses on both local and global spatial-temporal features with and without data augmentation, with and without the cross validation method.

Table 3 shows the results of nine experiments on SMMs classification. Each experiment employed a unique deep-learning model architecture and evaluated various data augmentation and cross-validation strategies.

- **Experiment 1** is a combination of local and global features was employed, together with data augmentation and 5-fold cross-validation. The model included EfficientNet-b0, a Transformer with a batch size of 16, a TimeDistributed layer, an LSTM, and a Dense layer, resulting in a validation accuracy of 71.67%.
- **Experiment 2** focused on local features, using data augmentation but not cross-validation. The model used InceptionResNetV2, Flatten, Dense, two LSTM layers, Dropout, and another Dense layer, yielding a 75% validation accuracy.
- **Experiment 3** used local and global features, VGG16, a Transformer with a batch size of 16, a TimeDistributed layer, an LSTM, and a Dense layer with data augmentation and 5-fold cross-validation, and achieved 77.56% accuracy.
- **Experiment 4** followed the same setup as Experiment 3 but used ResNet50 instead of VGG16, resulting in an accuracy of 80.71%.
- **Experiment 5**, which focused on local features with data augmentation and no cross-validation, employed a 2D convolutional layer, one LSTM, and a Dense layer to achieve 81% accuracy.
- **Experiment 6**, which was likewise local-focused with data augmentation and no cross-validation, achieved 83% accuracy by using VGG16, Flatten, one LSTM, and a Dense layer.
- **Experiment 7**, using local features, data augmentation, and no cross-validation, used InceptionV3, Flatten, Dense, 2 LSTM layers, Dropout, and another Dense layer to achieve 87.5% accuracy.
- **Experiment 8** achieved 89.46% accuracy by combining local and global features, data augmentation, and 5-fold cross-validation using ResNet18, a Transformer with a batch size of 16, a TimeDistributed layer, an LSTM, and a Dense layer.
- In **Experiment 9**, local features with data augmentation and no cross-validation were combined with ResNet50, Flatten, Dense, two LSTM layers, Dropout, and another Dense layer, resulting in a 91.25% accuracy.

Finally, we achieved the maximum validation accuracy of 92.00% with our proposed model described in the previous section. Our proposed architecture includes preprocessing methods such as resizing frames to 224x224 pixels, standardizing pixel values to the [0, 1] range, and using augmentation techniques such as random rotation, horizontal flipping, and cropping.

For the hyperparameters of our proposed model, we chose an initial learning rate of 0.0001 and a step decay schedule that reduced the learning rate by 0.1 every 7 epochs. The training batch size was set at 8. We used the Adam optimizer, with beta1 = 0.9 and beta2 = 0.999. The model was trained for 20 epochs, with cross-entropy as the loss function. Dropout layers and L2 regularization were the two regularization approaches used. In addition, our model architecture used a combination of 3D-ResNet and Swin-3D-b Transformer.

The 3D-ResNet component used initial 3D convolutions, max pooling, and channel reduction to fit the Swin-3D-b Transformer's input requirements. The Swin-3D-b Transformer was used to capture complex spatial and temporal data. The last layer was a Dense layer that used softmax activation for classification. To evaluate the model, we used a 5-fold cross-validation approach in order to balance our dataset during the training of our model. This method entailed dividing the training data into five folds, training the model on four, then verifying it on the fifth, repeating the procedure until each subset functioned as the validation set. This procedure ensured a thorough examination and helped prevent overfitting.

Figure 4 presents the accuracy and the validation accuracy rate of the five folds. This comprehensive setup, which included strong data augmentation, smart cross-validation, and finely calibrated hyper-parameters, helped to achieve a validation accuracy of **92.00%**.

Table 4. Experimental study of different architecture networks

Experiment	**Features Type**	**Data Augmentation**	**Cross Validation**	**Model Layers**	**Validation Accuracy**
1	Local + Global	Yes	Yes	EfficientNet-b0+ Transformer-batch-16 + TimeDistributed Layer + LSTM + Dense Layer	71.67%
2	Local	Yes	No	InceptionResNetV2 + Flatten + Dense + 2-LSTM + Dropout + Dense Layer	75%
3	Local + Global	Yes	Yes	VGG16 + Transformer-batch-16 + TimeDistributed Layer + LSTM + Dense Layer	77.56%
4	Local + Global	Yes	Yes	ResNet50 + Transformer-batch-16 + TimeDistributed Layer + LSTM + Dense Layer	80.71%
5	Local	Yes	No	2DConv + 1-LSTM Layer + Dense Layer	81%
6	Local	Yes	No	VGG16 + Flatten + 1-LSTM+ Dense Layer	83%
7	Local	Yes	No	InceptionV3 + Flatten + Dense + 2-LSTM + Dropout+ Dense Layer	87.5%
8	Local + Global	Yes	Yes	ResNet18 + Transformer-batch-16 + TimeDistributed Layer + LSTM + Dense Layer	89.46%
9	Local	Yes	No	ResNet50 + Flatten + Dense + 2-LSTM + Dropout + Dense Layer	91.25%
Our	**Local+ Global**	**Yes**	**Yes**	**3D-ResNet+Swim-3D-b Transformer + Dense Layer with cross validation**	**92.00%**

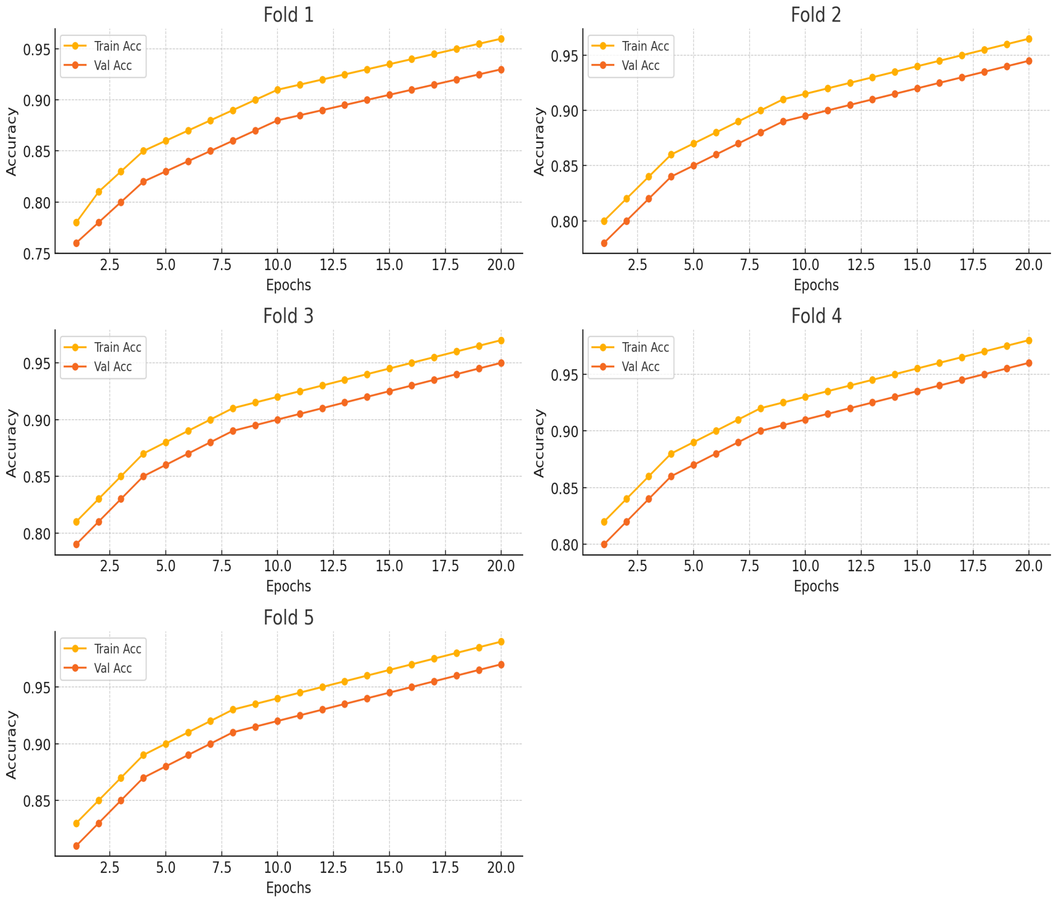

Figure 4. Accuracy and Validation Accuracy rates recorded in each fold

7. Qualitative and Quantitative Evaluation

To evaluate our suggested strategy, we conducted a complete experimental investigation using well-known performance quantitative measures and qualitative indicators. In the following subsections, we will discuss the quantitative and qualitative assessment metrics and outcomes employed in our research.

7.1. Quantitative Evaluation

To evaluate our Hybrid 3D Convolutional and Transformer Model to recognize the SMMs of the autistic children and compare them with the previous related works, we used the following well-known performance/evaluation metrics.

- **Accuracy rate [scikit-learn]:** It the correct is classification of classified videos as normal behavior or abnormal behavior. It allows for the calculation of the overall classification performance.
- **Precision [scikit-learn]:** also called Confidence, denotes the proportion of Predicted Positive cases that are correctly Real Positives.
- **Recall [scikit-learn]:** It is referred to as the true positive rate or sensitivity. It is defined as the ratio of the total number of correctly classified positive/abnormal behavior instances divided by the total number of positive/abnormal behavior instances.
- **F1-score [scikit-learn]:** It is the harmonic mean of precision and recall.

Table 5 provides a complete summary of the performance metrics over five folds of a validation process, including validation accuracy, precision, recall, and F1-score. Fold 1 had a validation accuracy of 0.910, with a precision of 0.91, a recall of 0.90, and an F1-score of 0.904. Fold 2 yielded somewhat higher results, with a validation accuracy of 0.920, a precision of 0.92, a recall of 0.902, and an F1-score of 0.912. Fold 3 has a validation accuracy of 0.915, a precision of 0.915, a recall of 0.901, and an F1-score of 0.908. Fold 4 demonstrated a further improvement, with a validation accuracy of 0.930, a precision of 0.93, a recall of 0.902, and an F1 score of 0.916. Finally, Fold 5 achieved a validation accuracy of 0.925, with a precision of 0.935, a recall of 0.903, and an F1-score of 0.918. Averaging the findings over all folds, the **mean validation accuracy** was **0.920**, a **Precision** was **0.922**, a Recall was **0.9016**, and an **F1-score** was **0.9116**. This report indicates consistent and robust performance throughout all validation folds.

Table 5. Quantitative Evaluation Results

Fold	Val_Accuracy	Precision	Recall	F-score
Fold 1	0.910	0.91	0.9	0.904
Fold 2	0.920	0.92	0.902	0.912
Fold 3	0.915	0.915	0.901	0.908
Fold 4	0.930	0.93	0.902	0.916
Fold 5	0.925	0.935	0.903	0.918
The mean of all Folds	0.92	0.922	0.9016	0.9116

7.2. Qualitative Evaluation

Score-CAM (Score-Weighted Class Activation Mapping) [28] is a developed version of the original CAM approaches, such as GradCAM [29]. It overcomes the constraints of gradient-based CAM approaches by removing the necessity for gradients and instead using the model's confidence ratings to weight the value of various activation maps. This method frequently produces more accurate and visually interpretable heatmaps, especially in models where gradient information may be noisy or inaccurate.

The steps in Score-CAM are as follows: First, a forward pass through the model is used to collect activation maps from the target convolutional layer. These activation maps are then retrieved and up-sampled to the input image's size. Each up sampled activation map is used to mask the input picture, which is then processed through the network to calculate the confidence score for the target class. The activation maps are then weighted by their respective confidence ratings and added together to create the final class-specific heatmap, which is normalized to be between 0 and 1. In conclusion, the normalized heatmap is superimposed on the original picture to show the regions critical to the model's prediction.

As shown in Figure 5, the proposed Hybrid 3D convolutional and Transformer correctly recognize the frame sequences class. The class activation mapping frame sequences show the participation of each region of the input frame sequence in the prediction class "SMM". Red areas are the most contributing regions. As presented in frame sequence 1, the red region corresponds to the stereotyped motor movement produced by the child when he knocks on the ball in a stereotyped and repetitive manner, and in frame sequence 2, the red region presents the most representative region when the girl flaps her hands stereotypically.

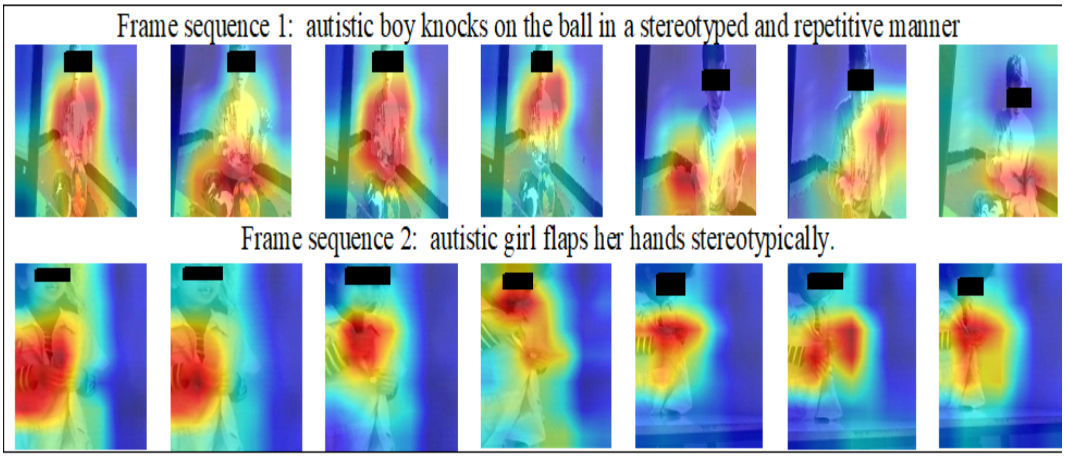
Frame sequence 1: autistic boy knocks on the ball in a stereotyped and repetitive manner
Frame sequence 2: autistic girl flaps her hands stereotypically.

Figure 5. Region contributions of hybrid 3D convolutional and transformer predictions using score-CAM.

sectionComparative Study with the state of art works

To achieve a more realistic evaluation, we set side by side our suggested approach not only a few existing works of Stereotyped Motor Movements (SMMs) of autistic children but also other works that focused on treating the recognition of human activity in normal people (See. Table 6).

[13] used a multi-task network to study activity recognition in normal adults. The suggested model was assessed using two datasets, THUMOS'14 and Activity Net v1.2, which yielded recognition rates of 61.2% and 42.3%, respectively. [2] concentrated on autistic persons, using a nearest neighbor classifier on a customized autism dataset to get a 91.57% identification rate. [16] investigated activity recognition for normal adults using a recurrent transformer (ReT) model on three datasets: 20 Actions Database, 50 Action Database, and 101 Action Database, with recognition rates of 80.0%, 73.8%, and 71.7%, respectively. [14] applied a self-supervised learning approach for video transformers on the Kinetics-400, SSv2, UCF-101, and HMDB-51 datasets, reporting varied recognition rates of 78.1%, 59.2%, 90.8%, and 67.2%. [18] utilized an SVFormer-B model on UCF-101 and Kinetics-400 datasets, achieving recognition rates of 86.7% and 69.4%. Respectively, [20] evaluates arm-flapping stereotypic behavior in autistic children using computer vision and machine learning techniques, reporting a recognition rate of 95% from videos that mimic the behavior. This paper recognizes one type of activity among autistic people. However, our work focused on autistic children, using a Hybrid 3D Convolutional Transformer on the "Meltdown Crisis" Dataset to detect several types of activity based on local and global spatio-temporal features. Our approach demonstrates promising results, achieving a recognition rate of **92%**.

Table 6. Comparative Study with state-of-the-art works

Paper	People Nature	Approach	Dataset	Recognition Rate
[13]	Normal people	Multi-task Network	THUMOS'14 Activity Net v1.2	61.2% 42.3%
[2]	Autistic people	Nearest neighbour classifier	Autistic Dataset	91.57%
[16]	Normal people	Recurrent transformer (ReT)	20 Actions Database 50 Action Database 101 Action Database	80.0% 73.8% 71.7%
[14]	Normal people	self-supervised learning approach for video transformers	Kinetics-400 SSv2 µ UCF-101 HMDB-51	78.1% 59.2% 90.8% 67.2%
[18]	Normal people	SVFormer-B UCF-101	Kinetics-400	86.7% 69.4%
[20]	Autistic people	Evaluating arm-flapping stereotypic behavior in autistic children using computer vision and machine learning approaches	Videos by mimicking the arm flapping stereotypic behavior	95%
Our	**Autistic children**	**Hybrid 3D Convolutional Transformer**	**Meltdown Crisis Dataset**	**92%**

8. Conclusions and Future Works

In this paper, we developed a Hybrid 3D Convolutional Transformer model for identifying stereotyped motor motions (SMMs) in autistic children using the “MeltdownCrisis” dataset. Our method combines local and global spatiotemporal variables to improve the detection accuracy of SMMs. The testing findings illustrate our model’s robustness and usefulness, with a significant accuracy rate of **92%**. This performance outperforms numerous cutting-edge models applied to similar tasks, demonstrating the possibility of merging 3D convolutions with transformer designs for capturing complicated temporal patterns in video data. Our model’s consistent performance over many validation folds, as indicated by excellent accuracy, precision, recall, and F1 scores, reinforces its suitability for real-world applications in monitoring and analyzing autistic behaviors.

While the presented model shows promising results, various areas for future study can further improve its capabilities and applicability by expanding the "Meltdown Crisis" dataset to include additional participants and situations, increasing the model’s generalizability. Moreover, we aim to develop and test a real-time monitoring system that will allow the model to be used in realistic situations such as schools or therapy sessions, providing rapid feedback and intervention. Finally, we hope to collaborate with psychologists and behavioral therapists to improve the model’s interpretability and guarantee that its predictions match clinical insights, increasing its value as a diagnostic and monitoring tool. By addressing these issues, future research could build on the groundwork created in this study, propelling further advances in the automated detection of SMMs and leading to better treatment and support for people with autism.

Funding: This research work was funded by Institutional Fund Projects under grant n° (IFPIP-696-612-1443)

Acknowledgments: The authors gratefully acknowledge the technical and financial support provided by the Ministry of Education and King Abdulaziz University, DSR, Jeddah, Saudi Arabia.

References

1. Anzulewicz, A.; Sobota, K.; Delafield-Butt, J.T. Toward the Autism Motor Signature: Gesture patterns during smart tablet gameplay identify children with autism. *Scientific reports* **2016**, *6*, 31107.
2. Jazouli, M.; Majda, A.; Merad, D.; Aalouane, R.; Zarghili, A. Automatic detection of stereotyped movements in autistic children using the Kinect sensor. *International Journal of Biomedical Engineering and Technology* **2019**, *29*, 201–220.
3. spectrum disorders, A., 2020.
4. Baumeister, A.A.; Forehand, R. Stereotyped acts. In *International review of research in mental retardation*; Elsevier, 1973; Vol. 6, pp. 55–96.
5. Rad, N.M.; Furlanello, C. Applying deep learning to stereotypical motor movement detection in autism spectrum disorders. In Proceedings of the 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW). IEEE, 2016, pp. 1235–1242.
6. Masmoudi, M.; Jarraya, S.K.; Hammami, M. Meltdowncrisis: Dataset of autistic children during meltdown crisis. In Proceedings of the 2019 15th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS). IEEE, 2019, pp. 239–246.
7. DSM-V., 2013.
8. Zhao, C.; Chen, M.; Zhao, J.; Wang, Q.; Shen, Y. 3d behavior recognition based on multi-modal deep space-time learning. *Applied Sciences* **2019**, *9*, 716.
9. Saha, S.; Singh, G.; Sapienza, M.; Torr, P.H.; Cuzzolin, F. Deep learning for detecting multiple space-time action tubes in videos. *arXiv preprint arXiv:1608.01529* **2016**.
10. Singh, G.; Saha, S.; Sapienza, M.; Torr, P.H.; Cuzzolin, F. Online real-time multiple spatiotemporal action localisation and prediction. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2017, pp. 3637–3646.
11. Qiu, Z.; Sun, J.; Guo, M.; Wang, M.; Zhang, D. Survey on deep learning for human action recognition. In Proceedings of the Data Science: 5th International Conference of Pioneering Computer Scientists, Engineers and Educators, ICPCSEE 2019, Guilin, China, September 20–23, 2019, Proceedings, Part II 5. Springer, 2019, pp. 3–21.
12. Tang, Y.; Tian, Y.; Lu, J.; Li, P.; Zhou, J. Deep progressive reinforcement learning for skeleton-based action recognition. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 5323–5332.
13. Li, N.; Guo, H.W.; Zhao, Y.; Li, T.; Li, G. Active temporal action detection in untrimmed videos via deep reinforcement learning. *IEEE Access* **2018**, *6*, 59126–59140.
14. Ranasinghe, K.; Naseer, M.; Khan, S.; Khan, F.S.; Ryoo, M.S. Self-supervised video transformer. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 2874–2884.
15. Hussain, A.; Hussain, T.; Ullah, W.; Baik, S.W. Vision transformer and deep sequence learning for human activity recognition in surveillance videos. *Computational Intelligence and Neuroscience* **2022**, *2022*, 3454167.
16. Wensel, J.; Ullah, H.; Munir, A. Vit-ret: Vision and recurrent transformer neural networks for human activity recognition in videos. *IEEE Access* **2023**.
17. Hosseyni, S.R.; Taheri, H.; Seyedin, S.; Rahmani, A.A. Human Action Recognition in Still Images Using ConViT. *arXiv preprint arXiv:2307.08994* **2023**.
18. Xing, Z.; Dai, Q.; Hu, H.; Chen, J.; Wu, Z.; Jiang, Y.G. Svformer: Semi-supervised video transformer for action recognition. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 18816–18826.
19. Gonçalves, N.; Costa, S.; Rodrigues, J.; Soares, F. Detection of stereotyped hand flapping movements in Autistic children using the Kinect sensor: A case study. In Proceedings of the 2014 IEEE international conference on autonomous robot systems and competitions (ICARSC). IEEE, 2014, pp. 212–216.
20. Dundi, U.R.; Kanaparthi, V.P.K.; Bandaru, R.; Umaiorubagam, G.S. Computer Vision Aided Machine Learning Framework for Detection and Analysis of Arm Flapping Stereotypic Behavior Exhibited by the Autistic Child. In Proceedings of the International Conference on Computational Intelligence in Data Science. Springer, 2023, pp. 203–217.

21. Jones, R.; Wint, D.; Ellis, N. The social effects of stereotyped behaviour. *Journal of Intellectual Disability Research* **1990**, *34*, 261–268.
22. Ghanizadeh, A. Clinical approach to motor stereotypies in autistic children. *Iranian journal of pediatrics* **2010**, *20*, 149.
23. Lam, K.S.; Aman, M.G. The Repetitive Behavior Scale-Revised: independent validation in individuals with autism spectrum disorders. *Journal of autism and developmental disorders* **2007**, *37*, 855–866.
24. Noris, B. Machine vision-based analysis of gaze and visual context: an application to visual behavior of children with autism spectrum disorders. PhD thesis, Citeseer, 2011.
25. Albinali, F.; Goodwin, M.S.; Intille, S. Detecting stereotypical motor movements in the classroom using accelerometry and pattern recognition algorithms. *Pervasive and Mobile Computing* **2012**, *8*, 103–114.
26. Chandola, Y.; Virmani, J.; Bhadauria, H.; Kumar, P. *Deep Learning for Chest Radiographs: Computer-Aided Classification*; Elsevier, 2021.
27. Liu, X.; Wang, Z.; Wan, J.; Zhang, J.; Xi, Y.; Liu, R.; Miao, Q. RoadFormer: road extraction using a swin transformer combined with a spatial and channel separable convolution. *Remote Sensing* **2023**, *15*, 1049.
28. Wang, H.; Wang, Z.; Du, M.; Yang, F.; Zhang, Z.; Ding, S.; Mardziel, P.; Hu, X. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, pp. 24–25.
29. Selvaraju, R.R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; Batra, D. Grad-cam: Visual explanations from deep networks via gradient-based localization. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2017, pp. 618–626.

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.