# AUTISM SPECTRUM SELF-STIMULATORY BEHAVIOURS CLASSIFICATION USING EXPLAINABLE TEMPORAL COHERENCY DEEP NETWORKS AND SVM CLASSIFIER

LIANG SHUAIBING

FACULTY OF COMPUTER SCIENCE AND
INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

# AUTISM SPECTRUM SELF-STIMULATORY BEHAVIOURS CLASSIFICATION USING EXPLAINABLE TEMPORAL COHERENCY DEEP NETWORKS AND SVM CLASSIFIER

LIANG SHUAIBING

DISSERTATION SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF COMPUTER SCIENCE

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITI MALAYA
KUALA LUMPUR

2022

## UNIVERSITY OF MALAYA

## ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: Liang Shuaibing

Matric No.:17199016/1

Name of Degree: Master of Computer Science

Title of Dissertation ("this work"):

AUTISM SPECTRUM SELF-STIMULATORY BEHAVIOURS CLASSIFICATION USING EXPLAINABLE TEMPORAL COHERENCY DEEP NETWORKS AND SVM CLASSIFIER

Field of Study: Image processing(computer science)

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this work;
(2) This work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every right in the copyright to this work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature: Date: 22th March, 2022

Subscribed and solemnly declared before,

Witness's Signature: Date: 22th March, 2022

Name: D

Designation:

# AUTISM SPECTRUM SELF-STIMULATORY BEHAVIOURS CLASSIFICATION USING EXPLAINABLE TEMPORAL COHERENCY DEEP NETWORKS AND SVM CLASSIFIER

## ABSTRACT

Autism spectrum disorder is a very common disorder. An early diagnosis of autism is essential for the prognosis of this disorder. The common method for diagnosis utilizes behavioural cues of autistic children. Doctors require years of clinical training to acquire the ability to capture these behavioural cues such as self–stimulatory behaviours. In recent years, the advancement of deep learning algorithms and hardware enabled the use of artificial intelligence technology to automatically capture self-stimulatory behaviours. Using this technique, the work efficacy of doctors can be improved. However, the field of self-stimulatory behaviours research still lacks large, annotated data to train the model. Therefore, the application of unsupervised machine learning methods is adopted. Meanwhile, it is often difficult to obtain good classification results using unlabelled data, further research to train a model that can obtain good classification results and at the same time being practical will be valuable. Nevertheless, in machine learning, the interpretability of the created model is also important. Hence, we have utilized the Layer-wise Relevance Propagation (LRP) method to explain the proposed model. In this work, the major innovation is utilizing the spatio-temporal continuity of close frames as a free form of supervision and setting a global discriminative margin to extract slow-changing discriminative self-stimulatory behaviours features. Extensive evaluation of the extracted features has proven the effectiveness of those features. Firstly, the extracted features are classified by the k-means method to demonstrate the classification of self-stimulatory behaviours in a completely unsupervised way. The conditional entropy method is used to evaluate the effectiveness of features. Secondly, we have obtained the state-of-the-art results by combining the unsupervised TCDN method with optimised supervised learning methods (such as SVM, k-NN, Linear Discriminant Analysis). These state-of-the-art results prove the effectiveness of the slow-changing discriminative self-stimulatory behaviours features.

Keywords: Autism Spectrum Disorder, Computational behavioural analysis, Machine Learning, Temporal coherency, Unsupervised Deep Learning

# PENGELASAN PERILAKU STIMULATORI DIRI SPEKTRUM AUTISME DENGAN MENGGUNAKAN RANGKAIAN NEURAL KOHEREN TEMPORAL YANG BOLEH DIJELASKAN BERSAMA PENGELASAN SVM

## ABSTRAK

Kecelaruan spektrum autisme adalah satu jenis gangguan yang sering berlaku. Diagnosis awal autisme adalah sangat penting untuk prognosis kepada kecelaruan ini. Kaedah diagnosis biasa menggunakan isyarat tingkah laku kanak-kanak autisme. Doktor memerlukan latihan klinikal selama bertahun-tahun untuk memperoleh keupayaan untuk mengenal pasti petunjuk kepada tingkah laku ini (seperti tingkah laku perangsangan diri). Dalam beberapa tahun kebelakangan ini, kemajuan algoritma dan perkakasan pembelajaran mendalam membolehkan penggunaan teknologi kecerdasan buatan untuk mengenal pasti tingkah laku perangsangan diri secara automatik. Dengan menggunakan teknik ini, keberkesanan tugas doktor dapat ditingkatkan. Walau bagaimanapun, bidang penyelidikan dalam tingkah laku perangsangan diri masih mengalami kekurangan data teranotasi yang besar, untuk melatih model tersebut. Oleh itu, penerapan kaedah pembelajaran mesin tanpa diselia diguna pakai. Sementara itu, seringkali sukar untuk memperoleh hasil klasifikasi yang baik menggunakan data yang tidak berlabel. Justeru, penyelidikan lebih lanjut untuk melatih model yang dapat memperoleh hasil klasifikasi yang baik dan pada masa yang sama praktikal adalah sangat berharga. Walaupun begitu, dalam bidang pembelajaran mesin, kebolehtafsiran model yang dibuat adalah penting. Oleh itu, kami telah menggunakan kaedah *Layer-wise Relevance Propagation* (LRP) untuk menjelaskan model yang dicadangkan. Dalam penyelidikan ini, inovasi utama adalah menggunakan kesinambungan spatio-temporal bingkai dekat sebagai satu bentuk pelabelan bebas dan menetapkan margin diskriminasi global untuk mengekstrak ciri-ciri tingkah laku perangsang diri yang lambat berubah. Penilaian menyeluruh terhadap ciri yang diekstrak telah membuktikan keberkesanan ciri tersebut. Pertama, ciri yang diekstrak diklasifikasikan dengan kaedah *k-means* untuk menunjukkan klasifikasi tingkah laku rangsangan diri dengan cara yang tidak diawasi sepenuhnya. Sementara itu, kaedah entropi bersyarat digunakan untuk menilai keberkesanan ciri. Kemudian, kami telah memperoleh hasil yang baik dengan menggabungkan kaedah TCDN yang tidak diselia dengan kaedah pembelajaran yang diselia yang dioptimumkan (seperti SVM, k-NN, Linear Discriminant Analysis). Hasil canggih ini membuktikan keberkesanan ciri tingkah laku perangsang diri yang diskriminatif yang perlahan.

Keywords: Gangguan Spektrum Autisme, Analisis tingkah laku komputasi, Pembelajaran Mesin, Koherensi Temporal, Pembelajaran Dalam Tidak Diselia

# ACKNOWLEDGEMENTS

Throughout the journey of my master study and research, I have found the true essence and spirit of research itself. Looking back at the journey, the challenges that I have faced simply made things more interesting and rewarding. It has been a fruitful journey so far and there are two people I should give credits to: Prof. Dr. Loo Chu Kiong and Dr. Aznul Qalid Md Sabri. They are two of my very well-respected supervisors. Whenever my research study reaches a bottleneck or when I face challenges along the way, they have been there to guide and help me get through them. Without their help, I would not have been able to go so far in the field of academic research.

Secondly, I would like to give thanks to my significant other and my family for their encouragement as well as financial support. I am grateful that I was given the opportunity of academic exploration because of them.

Finally, I thank the staff of the department and university for their assistance whenever I need them. University of Malaya will be my alma mater and I will forever be proud of that.

## TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF SYMBOLS AND ABBREVIATIONS

| | | |
|---|---|---|
| TCDN | : | Temporal Coherency Deep Networks |
| LRP | : | Layer-wise Relevance Propagation |
| ASD | : | Autism Spectrum Disorder |
| ADOS | : | Autism Diagnostic Observation Schedule |
| SVM | : | Support-Vector Machines |
| AOSI | : | Autism Observation Scale for Infants |
| TCDN-SVM | : | Temporal Coherency Deep Networks and Support-Vector Machines |
| LSTM | : | Long short-term memory |
| SSBD | : | Self-Stimulatory Behaviour Dataset dataset |
| STIP | : | Space-Time Interest Points |
| BOW | : | Bag Of Words |
| HDM | : | Histogram of Dominant Motions |
| K-NN | : | K-Nearest Neighbors Algorithm |
| SFA | : | Slow Feature Analysis |
| CNN | : | Convolutional Neural Network |
| CE | : | Conditional Entropy |
| SGD | : | Stochastic Gradient Descent |

## LIST OF APPENDICES

# CHAPTER 1: INTRODUCTION

Autism Spectrum Disorder is a prevalent disorder across the globe. A recent paper published in March 2020 revealed that 1 in 54 children are identified with Autism Spectrum Disorder (ASD) according to the estimates from CDC's Autism and Developmental Disabilities Monitoring Network (Maenner et al., 2021) Studies have also proven that early diagnosis of ASD is associated with significant gains in intellectual ability, adaptive behaviour as well as reduction of symptom severity in children with ASD (Estes et al., 2015; Shattuck et al., 2009; Zwaigenbaum et al., 2013). Using behavioural cues of autistic children is a common method of diagnosis for ASD (Rehg, 2013). Some of the exercising instruments which use those behavioural cues to diagnose ASD include the Autism Diagnostic Observation Schedule (ADOS) (Lord et al., 2000) and the Autism Observation Scale for Infants (AOSI) (Bryson et al., 2008). Moreover, self-stimulatory behaviours are atypical behavioural cues that are assessed in these instruments for diagnosis. Autism diagnosis requires clinicians to interact with the child over multiple extensive sessions to identify the behavioural cues (Rajagopalan & Goecke, 2014). However, professionally trained clinicians may be unavailable and its healthcare cost might be high in some places. Therefore, using a computer to automatically analyse characteristics of children with autism such as their self-stimulatory behaviour can help doctors to infer further diagnosis (Alnajjar et al., 2021; Alnajjar et al., 2019). Some self-stimulatory behaviours are also classified as self-injurious behaviour, such as head banging (Cantin-Garside et al., 2020), as these behaviours can cause damages to the children. Considering the random occurrence of self-stimulatory behaviour, it is impractical to observe autistic children at all times during the day. An automatic self-stimulatory behaviour analysis system can help doctors and parents to identify and provide the best of care for children with autism. For the moment, the existing research on self-stimulatory behaviours is mainly divided into two categories, namely based on accelerometers (Min, 2017; Rad et al., 2018) and based on computer vision (Hashemi et al., 2012; Rajagopalan & Goecke, 2014) respectively. Since 2D cameras are cheaper and more easily accessible, we decided to develop a self-stimulatory behaviour classification algorithm based on video data.

## 1.1 Research Motivation

First of all, autism is a prevalent disorder all over the world. The diagnosis of autism requires the clinician to interact with the child over multiple long sessions to identify their behavioural cues. However, well-trained clinicians are not available in certain places. Healthcare cost is another concern.

Secondly, some self-stimulatory behaviours are also classified as self-injurious behaviour (such as head banging), which means that these behaviours may injure the children. However, for clinicians and parents of autistic patients, it is impractical to observe children at all times during the day. An automated analysis system can save their time and detect their characteristic behaviour accurately.

Thirdly, with the development of algorithms and hardware, it is possible to use artificial intelligence and simple 2D cameras as the tools to recognize the behaviour of children with autism.

## 1.2 Problem Statement and Hypothesis

Deep learning has made great achievements in the field of human action recognition (Baccouche et al., 2011; Ji et al., 2012; Sargano et al., 2017). Although there is a large amount of unlabelled video data on the public website (such as YouTube) on self-stimulatory behaviours research, we still lack large annotation real-word datasets to train an artificial neural network. Using unlabelled video data recorded in an uncontrolled environment to train unsupervised models is often difficult to obtain good classification performance. Hence, choosing and optimising the model to achieve good classification performance remains a challenge. Furthermore, understanding the internal classification mechanism is often difficult due to the nonlinear structure of artificial neural networks. This prevents our model from providing intuitive references and suggestions for researchers and doctors.

In this article, utilising the large amount of unlabelled video data obtained from the public website, we have decided to use an unsupervised method to automatically extract the features from the video data to save time and effort. From the published paper written by Wiskott and Sejnowski, we understand that the input of a camera is a quick-changing matrix (Wiskott & Sejnowski, 2002). A slight change of the characters in a video will drastically affect the input matrix. Thus, if we can obtain a slow-changing or even steady feature of each autism self-stimulatory behaviour, we can classify those behaviours easily (Dawood & Loo, 2016, 2018). Other than that, the ability of these slow-changing features to discriminate different types of self-stimulatory behaviours is also crucial. To date, obtaining a slow-changing discriminative self-stimulatory behaviours feature remains a challenging problem.

In order to understand the model's internal classification mechanism, the Layer-wise Relevance Propagation (LRP) (Bach et al., 2015) algorithm will be used to explain the Temporal Coherency Deep Networks and Support-Vector Machines (TCDN-SVM) hybrid algorithm that was created.

## 1.3 Significance of Research

Firstly, in this research, we apply an unsupervised temporal coherency deep networks method, which allows us to use a large number of real-world video data on public websites to fully automatically extract dissimilar features between different types of self-stimulatory behaviours directly. This would spare us a lot of time on data labelling and designing hand-designed features.

The following Venn  Diagram has illustrated the objective and significance of this experiment.

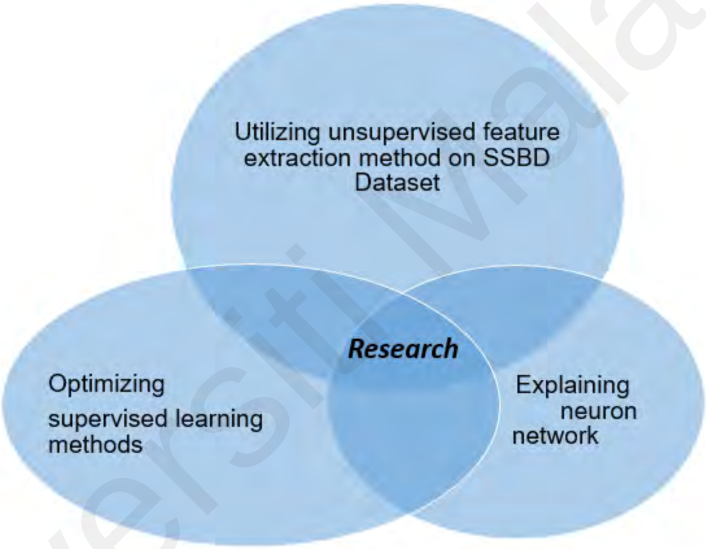

Utilizing unsupervised feature extraction method on SSBD Dataset
Research
Optimizing supervised learning methods
Explaining neuron network

**Figure 1.1: The research scope of this experiment**

Secondly, the benefits of such unsupervised learning from just unlabelled videos will be shown, which can be directly used as a prior for the supervised autism spectrum self-stimulatory behaviors classification task. Multiple supervised methods will also be used to optimize the classification.

Finally, by explaining the internal mechanism of the model's classification of autistic behaviours, references can be provided to other scientists to design models or provide evidences for doctors' diagnoses purposes.

## 1.4 Research Question

Firstly, the possibility of utilizing unlabelled video data to train a model will be looked into. Then, this model will be used to extract dissimilar features from these high-dimensional nonlinear unlabelled video data automatically.

Secondly, the trained model will be optimized to obtain a better classification performance.

Finally, the results obtained from the trained artificial neural network will be interpreted.

## 1.5 Research Objectives

The aim of this experiment is to apply the unsupervised temporal coherency deep networks method to identify self-stimulatory behaviours of autism. This will become a basis for the early diagnosis of children with autism. The following are the research objectives addressed in this thesis:

1. To develop an unsupervised automated approach to extract slow-changing discriminative self-stimulatory behaviours feature utilizing the temporal coherency between adjacent frames of a video.
2. To perform supervised learning (SVM, KNN, Decision Trees, Discriminant Analysis) methods to classify the extracted features and to optimize the resulting classifier's parameters to improve their performances.
3. To evaluate the internal mechanism of the developed model to help humankind designs models for autistic patients and to provide some valuable advices to doctors for their diagnosis proposes.

**Table 1.1: SMART Research Objectives**

| | |
|---|---|
| Specific | Using temporal coherence of contiguous video frames as a free form of supervision to train deep neural networks and to extract features. Then, using the optimized supervised method for classification purpose. Finally, the LRP algorithm is used to explain the internal mechanism of the model. |
| Measurable | The results of our research can be measured by some standard metrics. For example, the performance of the model can be evaluated by conditional entropy and confusion matrix. |
| Achievable | Use K-means to classify autism behaviour and then use conditional entropy to compare the performance with other methods to check the signature of our model. |
| Realistic | Using a video dataset recorded from an uncontrolled environment (SSBD) to train our model and the GPU and CPU for the training model is also available. |
| Time-Bound | This research will follow the schedule of the research grant funding this research work. |

## 1.6 Contribution

The main contribution of this paper is that we have innovated an approach to specifically extract slow-changing discriminative self-stimulatory behaviours feature and the experiment was able to obtain a state-of-the-art result. The proposed approach is based on the unsupervised temporal coherency deep networks (TCDN) method (Redondo-Cabrera & Lopez-Sastre, 2019).

The TCDN algorithm is based on four Alexnet with the same parameters and a loss function based on Euclidean distance (Krizhevsky et al., 2012). Next, in order to prove the efficiency of features extracted by our method, a method that combines K-means and conditional entropy are used. Thereupon, unsupervised feature extraction methods and supervised classification methods are combined to construct self-stimulatory behaviour classifiers. Multiple supervised methods are employed to improve the classification performance of our model to obtain a particularly good result. The methods yielded 98% accuracy at the frame level and 98.3% accuracy at the video level. Finally, a TCDN-SVM model is constructed and interpreted using the LRP algorithm to ascertain why this model could achieve such good results. This model allows us to provide evidence for the early diagnosis of autism.

## 1.7 Dissertation organization

The rest of this article is organised as follows: Chapter 2 discusses some existing methods that can be used for the self-stimulatory behaviour classification and its limitations. Chapter 3 briefly introduces all the methods used in this study. Chapter 4 introduces the evaluation methods and provides the result of the experiment. While Chapters 5, 6 and 7 discuss and summarise our research and propose future research directions.

# CHAPTER 2: LITERATURE REVIEW

## 2.1 Autism

Autism is defined as a disorder type of condition in the early 1940s. In Kanner's landmark paper published in 1943, he found out that some children have common and significantly different characteristics from other children, such as doing some stereotyped movements, shaking head from side to side and whispering. He believes that these abnormal movements in children can be the cues to diagnosing children with this disorder (i.e. autism) (Kanner, 1943).

In the past 50 years, the definition of autism has evolved from a narrowly defined rare childhood-only disease to a lifelong disease that has been widely publicized and studied. This disease is considered to be very common and heterogeneous, with its core features being social disorder and repetition, as well as unusual perceptual action behaviours (Lord et al., 2020).

## 2.2 Autism symptoms

Although the symptoms of autistic patients vary greatly, there are two core features of this disease: social communication problems as well as restricted, repetitive and unusual sensory-motor behaviours (Lord et al., 2020). However, the clinical records of the life course of autistic patients describe a huge heterogeneity in the development of the disease (Kanner, 1971; Tantam, 2000). Some patients lose their skills over time, while some reach their peak in adolescence, and some continue to develop their illnesses as adults.

Fong, Wilgosh, and Sobsey identified six areas of parental concern during adolescence, and underlying these parental concerns are the symptoms of autism (Fong et al., 1993). They are behavioural problems (aggressiveness, tantrums), social and communication problems (inappropriate or insufficient social skills), family-related problems (restrictions on family life and continuous supervision required), education and other related problems (choosing comprehensive services or professional services), relationships with experts (ineffective communication, criticism or accusations from professionals), and intervention in independence and future services (residence, occupation) (Fong et al., 1993).

Why do the symptoms of autistic patients vary differently at different stages of their lives? Firstly, the process of human maturation and development interacts with the characteristics of autism and affects the development of skills (Burack et al., 2001). People living with autism may only lack certain behaviours that accompany their development, such as eye contact and expressing interest. On the other spectrum, some of them might show very obvious abnormal behaviours (Seltzer et al., 2003). The second point is that the diagnostic criteria for autism may be disparate at different time. Since autism was considered a behavioural syndrome, the diagnostic criteria for autism have been very different (Fombonne, 2001). In general, the early diagnostic criteria for autism is stricter, which results in patients who are diagnosed with autism at an early stage must have more severe symptoms compared with patients who have been diagnosed with autism further in their lives (Magnússon & Sæmundsen, 2001). There are also cases where people are diagnosed with autism in adulthood only because the diagnostic criteria for autism have changed. Thirdly, the presence of autistic symptoms may be affected by the environment. There is an ecological theory of autism that states that autism is not just a simple disease, it may also reflect a disordered relationship between people and the environment.

### 2.2.1 Self-stimulatory behaviours

Self-stimulatory behaviour is an important feature of autism. Due to its diverse manifestations and long duration, self-stimulatory behaviour has been widely studied in the academic world (Bodfish et al., 2000). American special education expert Power (Powers et al., 1992) and others believe that the self-stimulatory behaviour of autistic children refers to a series of problematic behaviours without specific incentives, such as shaking the body, self-hitting the body, looking at rotating objects, etc. They think that this is a functional perceptual maladaptation. Norwegian psychologists Ekblad and Pfuhl (Ekblad & Pfuhl, 2017) believe that self-stimulation is a type of repetitive behaviour that is one of the core defects of children with autism. The self-stimulatory behaviours can seriously impair children's social interactions. Its recurrence will greatly distract children, which will hinder the learning of new social skills for children with autism. It will also result in reduction of proper behaviours and correct responses (Smith & Van Houten, 1996).

The main purpose of the self-stimulatory behaviours of children with autism is to increase and avoid perceptual stimulation (Bright et al., 1981). For example, children with autism can get tactile stimulation and escape the influence of the external noisy environment by tapping their fingers and rubbing their palms. The least complex self-stimulatory behaviour is the behaviour that can increase the sensory stimulation by stimulating the sensory organs, such as visual stimulation, auditory stimulation, and gustatory stimulation. This is mainly manifested through movements within the body or fiddling with objects. For instance, shaking the head, slapping the head, etc. Another way of expression they could do is to obtain self-stimulation through their face. For example, improper laughter, biting objects, improper tongue out, lips biting, etc. There are also children with autism who achieve self-stimulation by manipulating their hands and body. For example, clapping, shaking hands, shaking the body, and rotating the body. Other types of self-stimulation include blinking quickly, covering the eyes, and squinting at objects from the corners of the eyes. The inappropriate environmental feelings of children with autism are also regarded as a type of self-stimulating behavior. The main manifestation of this type of self-stimulatory behaviour is to obtain external environmental information in an abnormal way through various sensory inputs. For example, the sense of smell is manifested as smelling the taste, and the sense of sight and hearing is manifested as the fondness of looking at luminous objects or rotating objects and obsession with a certain type of sound, etc. The tactile sense is expressed when the person likes to touch objects with a special texture. The vestibular sense of self-stimulation is mainly expressed as shaking the body excessively like spinning, throwing, jumping up and down and doing other activities (Lovaas et al., 1987). Another self-stimulatory behaviour of autistic children is mainly meaningless echo-like language. Echoic language is also called "learning tongue" discourse. This is a higher level of self-stimulatory behaviour. There are two types of main manifestations for it. One is the timely echo. This self-stimulatory behaviour is manifested such that children with autism will repeat all or part of the words they have just heard. For example, when you ask an autistic child: "How is the weather today?" He may answer you: "How is the weather today?" Another manifestation is delayed echo. In this case, children with autism may repeat a certain word they heard in the past without any signs (Schreibman & Carr, 1978). When a child with autism is in a certain state of anxiety, delayed echoes will appear in order to relieve their anxiety or distract their attention.

Children with autism usually increase the input behaviour of perceptual stimulation by organizing the external environment. Compared with the self-stimulation obtained by manipulating one's own body, the self-stimulatory behaviour obtained by organizing the external environment is more complicated in its form and function. Children with autism usually manipulate an object in the environment to increase the input of a certain sensory stimulus. For example, they might repeatedly turn the light on and off to obtain visual stimulation, and tearing paper, rubbing plastic sheets, etc. to obtain tactile stimulation.

## 2.3 Autism diagnosis

A common diagnostic method for autism is based on the behavioural cues of autistic children (Rehg, 2013). One of the existing diagnostic instruments that is based on behavioural cues is the Autism Diagnostic Observation Schedule (ADOS) (Lord et al., 2000) . This instrument is a standardised and semi-structured evaluation method. It can assess autistic patients based on social interaction, communication, playful and imaginative use of materials. Another example of such an instrument is the AOSI for infants (Bryson et al., 2008), this algorithm was developed to detect and monitor early signs of autism in high-risk infants. Self-stimulatory behaviours are atypical behavioural cues that are assessed in these instruments for diagnosis based on accelerometers or computer vision.

## 2.4 Artificial neural network and deep learning

An artificial Neural Network is a simulation of our very own human biological neural network. Its basic idea is to form an adaptive nonlinear dynamic network through a large number of artificial neurons. In 1958, Rosenblatt et al. had proposed a multi-layer perceptron model, this is the first time that neural network research has been put into practice(Van Der Malsburg, 1986). However, this single-layer perceptron model is unable to solve the linear inseparability problem. In 1986, Rosenbaltt, Rumelhart and Hinton proposed the Back Propagation Network to solve some of the problems that single-layer perceptrons cannot solve (Rumelhart et al., 1986). However, when the number of Back Propagation Network layers increased, researchers were unable to solve the local optimal, over-fitting, and gradient diffusion problems encountered. Hence, the focus of the research had shifted to various shallow machine learning models, such as support vector machines.

In 2006, Hinton et al. published a research paper proving that the Artificial Neural Network with multiple hidden layers has excellent feature learning ability and could overcome the training difficulties of deep neural networks by pre-training each layer (Hinton & Salakhutdinov, 2006). Since then, a lot of researchers have taken interest in the topic of Artificial Neural Network. This deep learning and pre-training method is able to achieve excellent performance in handwritten digit recognition and pedestrian detection tasks (LeCun et al., 2015).

In 2016, the AlphaGo developed by Google DeepMind had defeated the human European champion Lee Sedol in Go with a score of 5:0, the deep neural network technology it used was well-known since. This also illustrates the powerful potential of deep learning theory.

### 2.4.1 Neurons

The basic unit of an artificial neural network is a neuron. A neuron generally has multiple inputs and one output, as shown in the following figure:

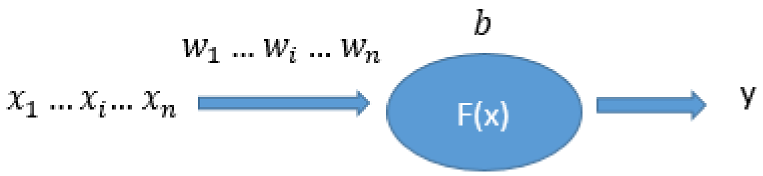


**Figure 2.1: Neurons**

In Figure 2.1, $x_1$~$x_n$ are inputs, $w_1$~$w_n$ is the weight of the connection between the input data and the neuron, $b$ is the bias term, and $y$ is the output of the neuron.

### 2.4.2 Multilayer Perceptron

As a classic neural network structure, the multilayer perceptron (MLP) is composed of the input layer, the hidden layer, and the output layer. The basic unit of each layer is a neuron. Its basic structure is shown in the following figure:

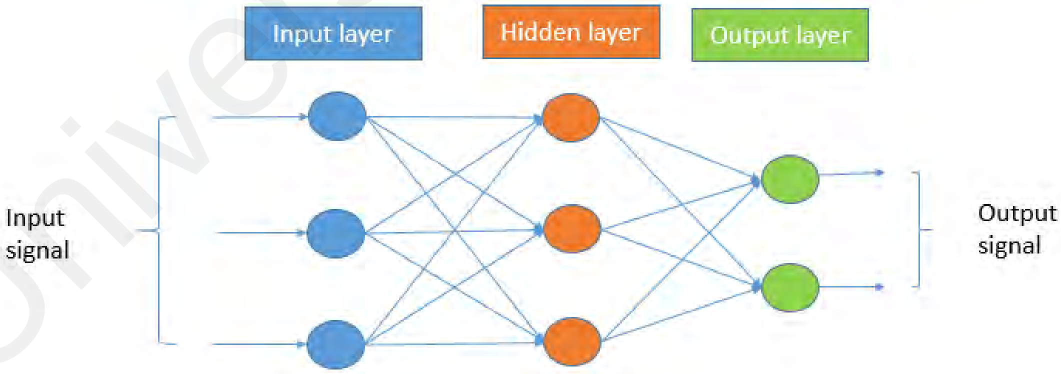


**Figure 2.2: The structure of the multilayer perceptron**

In the multilayer perceptron, the input signals of the neurons in the input layer can be pixels of the picture. Each neuron in the hidden layer is connected to all neurons in the adjacent layer, and this connection is called a full connection. When the multilayer perceptron algorithm is used for classification, the number of input neurons is the dimension of the input data, and the number of output neurons is determined by the number of categories.

### 2.4.3 Convolutional Neural Network

Biologists Hubel and Wiesel discovered through research that there are a series of complex structured cells in the visual cortex of the cat brain. These cells are called "receptive fields" because they are sensitive to the local area of the input space (Hubel & Wiesel, 1962). This receptive field can better dig out the strong local spatial correlation existing in natural images. They are divided into two types: simple cells and complex cells. Inspired by these two types of cells, the convolutional layer and pooling layer in convolutional neural networks were invented. It can accurately identify input patterns with slight displacement and deformation (Fukushima & Miyake, 1982). Subsequently, LeCun designed and trained the classic LeNet-5 CNN network using the Backpropagation algorithm, and achieved good classification results in some pattern recognition fields.

Convolutional Neural Networks (CNN) is a special kind of neural network. During the training process, the convolutional neural network will update its weights through the back propagation mechanism. A classic convolutional neural network usually includes multiple layers, such as an input layer, a convolution layer, a pooling layer, a fully connected layer, and an output layer.

The convolutional layer is composed of multiple feature maps. In the convolutional layer, each feature map is composed of multiple neurons, and each neuron is locally connected to the feature map of the previous layer through the convolution kernel. The convolution kernel is a weight matrix (such as a 5×5 two-dimensional matrix). The neuron multiplies the part of the feature map of the previous layer with the convolution kernel and the result will be passed to a nonlinear function for processing (such as the ReLU function), and then the output of the neuron can be obtained. In the convolutional layer, the size of the convolution kernel and the step size of the sliding will affect the dimensionality of the output feature map.

In the traditional CNN networks, the activation function generally uses saturating nonlinearity functions, such as sigmoid function, tanh function, etc. Compared with saturated nonlinear functions, unsaturated nonlinear functions can solve the problems of gradient explosion and gradient disappearance, and it can also speed up convergence rate (Xu et al., 2015).  Therefore, in the current CNN structure, an unsaturated nonlinear function is commonly used as the activation function of the convolutional layer, such as the ReLU function. In order to explore the impact of different factors of the convolutional neural network on the accuracy rate under the limited time complexity, He Kaiming conducted a series of experiments (He & Sun, 2015) and finally found that when the time complexity is about the same, it has a smaller volume. A CNN structure with a larger core and a deeper depth can obtain better accuracy than a convolutional neural network with a larger convolution kernel and a shallower CNN structure. At the same time, the experimental results show that the deeper the depth, the better the performance of the neural network. However, as the network depth increases, the network performance will also reach saturation gradually.

In the convolutional neural network structure, as the depth increases, so does the number of features of the neural network, as well as a larger feature space that the network can represent, and the stronger the network learning ability will become. However, this will also cause the calculation of the network to be more complicated and more prone to overfitting. Therefore, the relationship between the computational complexity and performance of the neural network should be balanced in practical applications.

The pooling layer usually follows the convolutional layer, which also consists of multiple feature faces. The main purpose of the pooling layer is to reduce the sensitivity of the output to offset and distortion. It divides the words of input image into several rectangular areas and outputs a value for each sub-area. This processing method can preserve the relative position of the feature rather than the precise position. At the same time, since the pooling layer can continuously reduce the size of the data space, the number of parameters and the amount of calculation will also decrease. Therefore, the pooling layer plays the role of the secondary feature extraction. Each of its neurons performs a pooling operation on the local receptive field. There are a variety of pooling operations in convolutional neural networks. The more commonly used methods are max-pooling and mean pooling. The neuron output of max-pooling mode usually selects the maximum value of the input area to represent the entire area. For mean pooling, the output of the neuron usually selects the mean value of the input area to represent the entire area. Boureau compared the two methods of max-pooling and mean-pooling. Through experiments he found that when the classification layer uses linear classifiers, such as SVM, the performance of the max-pooling method is better than that of mean pooling (Boureau et al., 2010). In addition, the random pooling method can assign a probability value according to the size of the input data, and then randomly select the output of the neuron according to the size of the probability value. This method provides random pooling with the advantage of maximum pooling, and at the same time, this pooling method can effectively avoid the overfitting of the convolutional neural network due to the randomness of the selection.

In a convolutional neural network, the input data is usually sent to the fully connected layer after being processed by multiple convolutional layers and pooling layers. The structure of the fully connected layer is similar to that of MLP, and each neuron is fully connected to all neurons in the previous layer. In a convolutional neural network, the output value of the last fully connected layer is usually passed to a softmax layer, which can use the softmax function for classification tasks. In the process of convolutional neural network training, in order to avoid network overfitting, dropout technology is usually used in the fully connected layer. Dropout technology can change the output value of hidden layer neurons to 0 with a certain probability, which is usually 0.5. This technology can make some hidden layer nodes to fail so that these nodes do not participate in the forward propagation process of CNN nor does it participate in the backward propagation process as well. Since each neuron cannot depend on other specific neurons, the features obtained in this way are extremely robust. At present, most of the research on CNN uses ReLU combined with the dropout technology to achieve good classification performance.

In recent years, CNN has widely been used in the field of image processing. Krizhevsky et al. achieved the best classification results in the LVSRC-12 competition by increasing the depth of the CNN network and using ReLU + dropout technology. Its network name is (AlexNet). The AlexNet model includes 5 convolutional layers and 2 fully connected layers (Krizhevsky et al., 2012). Compared with the traditional CNN network, the AlexNet network reduces the complexity of the model by using ReLU instead of the tanh function. Szegedy proposed the GoogleNet model by increasing the depth of the convolutional neural network. The main feature of this model is to reduce the amount of calculation. Compared with Alexnet, his parameter amount is greatly reduced, and the accuracy rate is improved (Szegedy et al., 2015). Simonyan et al. studied the importance of depth to convolutional neural networks. They assessed the impact of depth on model performance by continuously increasing the convolutional layer of the convolution kernel with a size of 3*3 in the network. At last, they found that when the number of layers has reached 16th to 19th, the performance of the model can be significantly enhanced. This model is known as the VGG model (Simonyan & Zisserman, 2014).

### 2.4.4 LSTM

Although the convolutional neural network can complete the task of image classification very well, it is not able to handle a large amount of data containing temporal information in the real world, such as audio, video, text datasets. In order to process this data, Recurrent Neural Network (RNN) was invented.

The main feature of the RNN network is that it could update the current network state according to the current input data and the previous network state, which enables the Recurrent Neural Network to have the ability to "remember" past information. However, when the time difference between the input data is too huge, RNN will not be able to remember the previous information very well. The ordinary RNN network is composed of standard recurrent cells. For example, sigma cell and tanh cell.

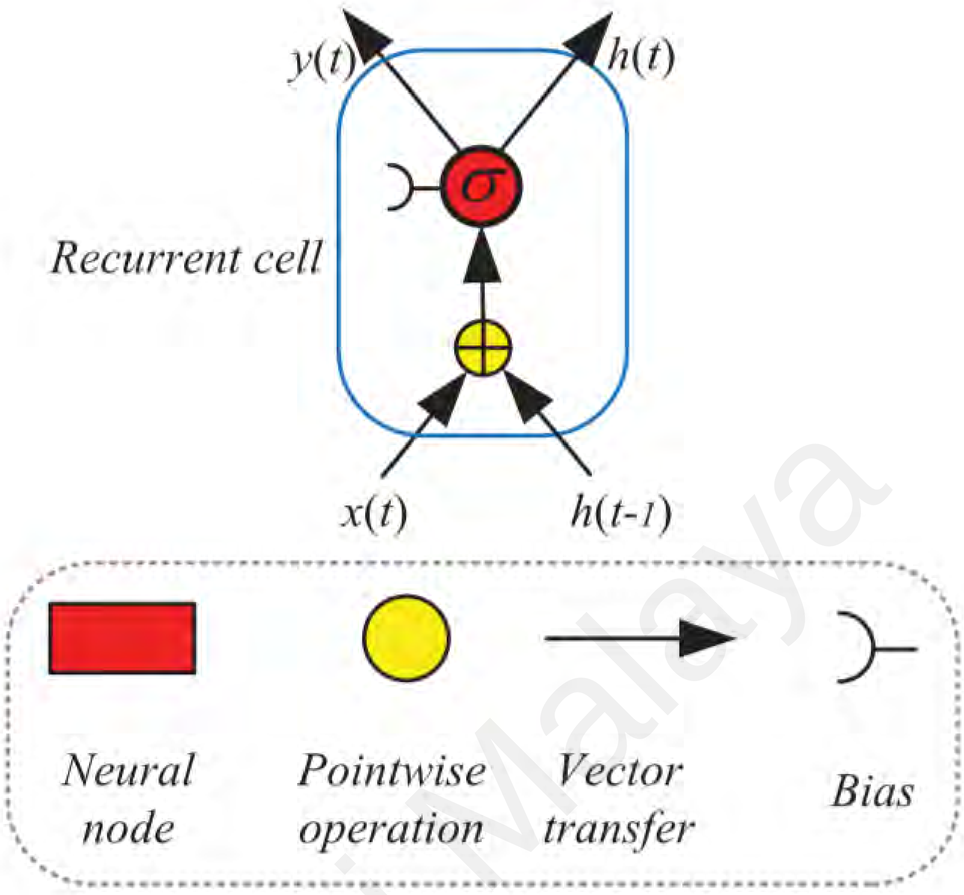


**Figure 2.3 The structure of the standard recurrent sigma cell (Yu et al., 2019)**

In the Figure 2.3, the structure of the standard recurrent sigma cell is illustrated by Yu et al..The structure of this network can also be expressed by the formula: $h_t = \sigma(W_h h_{t-1} + W_x x_t + b)$ where the input is $x(t)$ and $h(t-1)$ .The $h_t$ and $y_t$ are the recurrent information and output of this cell at time t. In this cell, $y_t = h_t$ . The $W_h$ and $W_x$ are the weights.

In the year 1997, in order to solve the problem of "long-term dependencies", Hochreiter and Schmidhuber proposed the LSTM cell (Hochreiter & Schmidhuber, 1997). Hence, LSTM was introduced to the discipline. This network enhances the memory capacity of the network by introducing "gates" into the cell. Recently there are different LSTM methods, such as LSTM without a forget gate, LSTM with a forget gate and LSTM with a peephole connection. Usually, the term LSTM cell means LSTM with a forget gate.

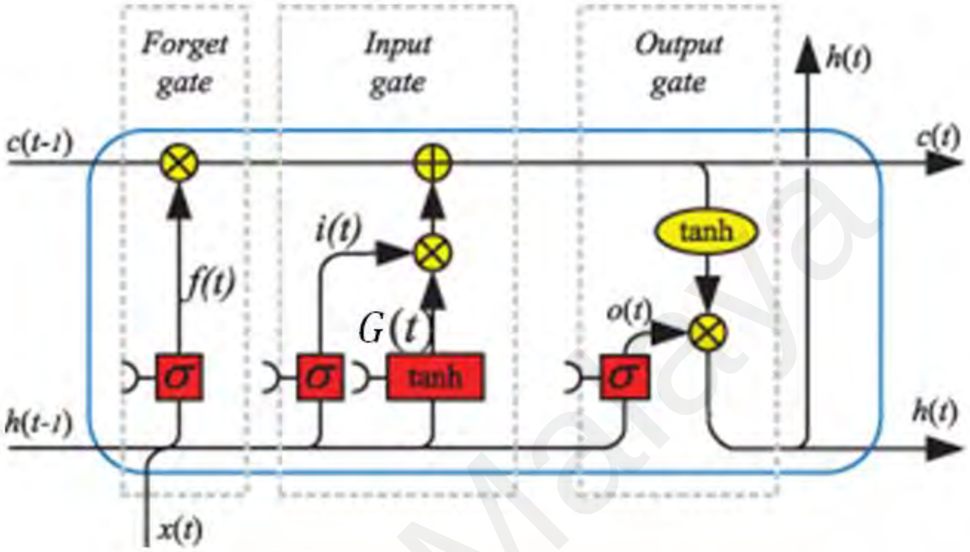


**Figure 2.4: The structure of LSTM with a forget gate (Yu et al., 2019)**

In Figure 2.4, the structure of the LSTM with a forget gate is demonstrated. The mathematical expression of this picture is as follows:

$$f_t = \sigma\left(W_{fh}h_{t-1} + W_{fx}x_t + b_f\right), \qquad (1)$$

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \qquad (2)$$

$$G_t = \tanh\,(W_{Gh}h_{t-1} + W_{Gx}x_t + b_G) \qquad (3)$$

$$c_t = f_t.c_{t-1} + i_t.G_t \qquad (4)$$

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \quad (5)$$

$$h_t = o_t.\tanh(c_t) \quad (6)$$

The $c_t$ represents the current cell state of the LSTM algorithm. $W_f, W_i, W_G, W_o$ are the weights. The function of the input gate is to determine which information can be determined to be the input to the cell state, and the output gate can determine which information can be used as the output. The function of the forget gate is to determine which information can be lost from the cell. $f_t = 1$ means to keep all the information, $f_t = 0$ means all the information will be discarded.

## 2.5 Unsupervised learning

In the field of action recognition, there are times when we cannot obtain a large amount of labelled video data. For example, in the field of self-stimulatory behavior recognition, it is difficult to obtain a large enough labelled video data. However, we will be able to source a lot of unmarked video data on websites such as Youtube and Tiktok. Therefore, figuring out ways to use these unmarked video data for analysis has become a huge problem. At this point of time, unsupervised learning has become a requisite choice.

Unsupervised learning refers to using unlabelled sample data sets as the research object and predicting the corresponding label category information by searching for the potential rules and structural information contained in the sample. Furthermore, the unlabelled sample data information is divided into clusters of several different categories according to the label category information, or the low-dimensional structure of the high-dimensional sample data is obtained. Traditional unsupervised learning includes k-means clustering, hierarchical clustering, principal component analysis and etc.

As a typical high-dimensional unstructured data, video data of self-stimulatory behaviours of autism in the real world has a lot of invalid redundant information and interruptions. Therefore, the classification algorithm must not only overcome the dimensional disaster problem but also overcome the error accumulation caused by complex interference information. In recent years, the rapid development of deep learning has made it easier to learn effective low-dimensional representations of video data. Deep learning algorithms based on unsupervised learning have also received more and more attention in the field of action recognition.

In the field of unsupervised feature learning using video data, some current unsupervised feature learning mainly focuses on dimensionality reduction techniques (Hurri & Hyvärinen, 2003; van Hateren & Ruderman, 1998). However, most of these researches that are based on video data are also contingent on the important concept of slow feature analysis (SFA) (Wiskott & Sejnowski, 2002).

### 2.5.1 Slow Feature Analysis (SFA)

Before introducing the Slow Feature Analysis, we must introduce an important concept: the slowness principle. This concept was first proposed by Geoffrey Hinton in 1989 (Hinton, 1990). Subsequently, many related algorithms have been generated around this concept (Berkes & Wiskott, 2005). In 2001, Alhazen explained the slowness principle by studying the process of human perception of external information (Smith, 2001). Alhazen believes that human perception of external information can be regarded as the behaviour produced by reconstructing external information input from the eyes. For example, when people observe a photo, they usually pay attention to the location information of the target in the photo, rather than the colour of each pixel in the picture.

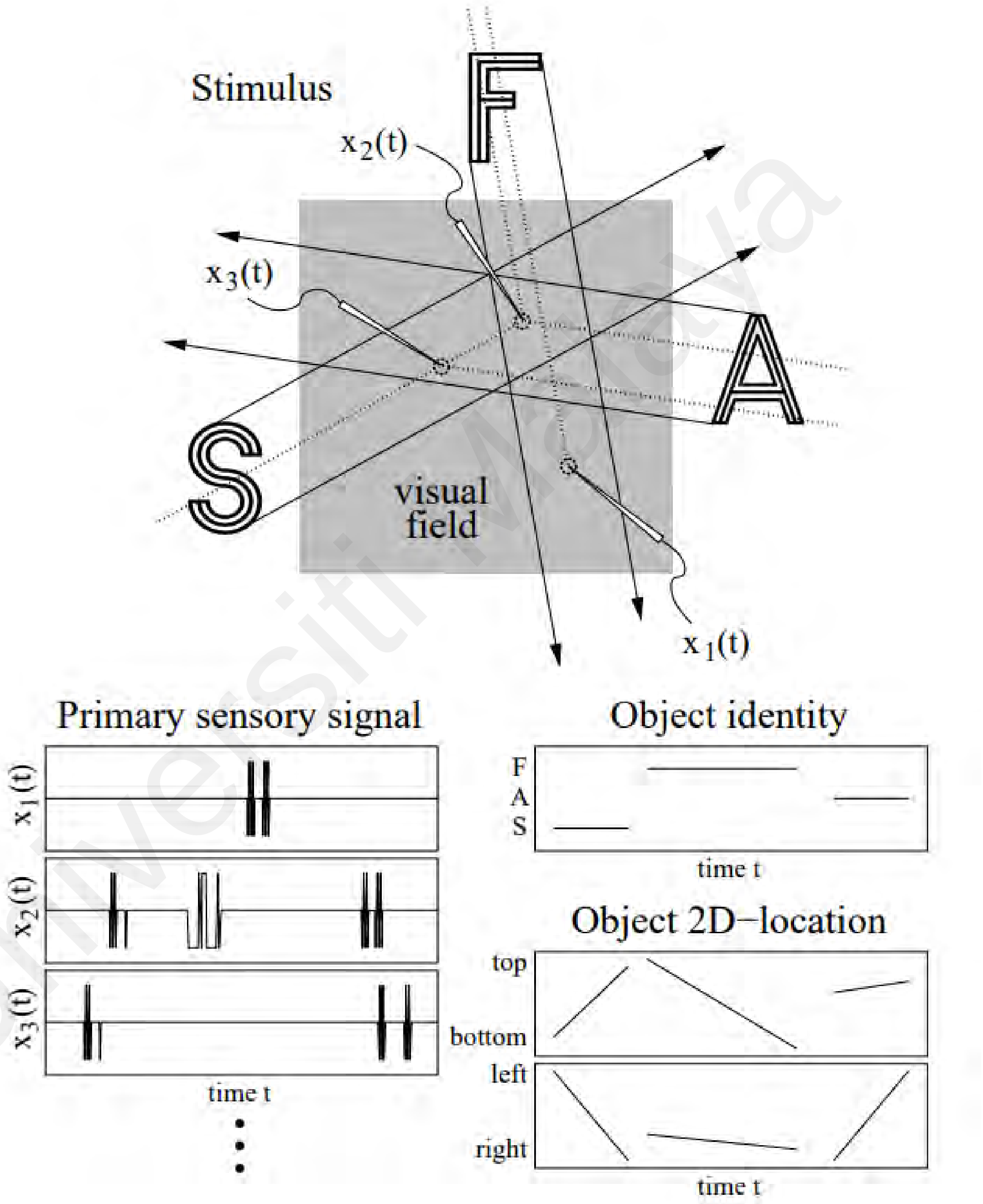


**Figure 2.5: The example for explaining the slow feature analysis (Wiskott & Sejnowski, 2002)**

In other words, if the human senses were regarded as a sensor, then the human perception of the external environment is a meaningful representation of the result constructed after reconstruction and comprehensive analysis of the signal input by the sensor.

What exactly is slow feature analysis (SFA)? Slow feature analysis (SFA) is a method that can slowly obtain changing features from input data in vector format. Wiskott and Sejnowski used a simple example to explain what slow feature analysis is (Wiskott & Sejnowski, 2002).

In Figure 2.5, the upper part of the chart shows that the three letters S, F, A pass through the visual field in a certain direction. First, the letter S, then the letter F, and finally the letter A. The lower left corner of Figure 2.5 shows the high-dimensional representation of this scene. For example, the object identifies a figure which indicates what letter is currently passing through the visual field, which can be used to indicate what-information. The object 2D-location represents the horizontal and vertical position of the letter currently in motion. This represents the where-information This kind of information can be considered high-level information. This representation is very convenient to display the movement information of the letters. The bottom left of the chart shows the primary sensory signal, which is a kind of low-level information. It represents the activation state of the photoreceptor placed in the visual field. They can be used to represent changes in the local features of the picture (such as local grey values, dots, edges and so on). This information will change very drastically, even if the letters move very slowly.

In Figure 2.6, The change of the primary sensory signal is very rapid and high-level representation. Although the primary sensory signal has a large silent period due to the blank background. However, this example only assumes a movement pattern and monitoring method of letters. In real-life situations (such as using a camera to monitor human movement), the difference between primary sensory signals and high-level representation will be more obvious.

In the object identify and object location charts in Figure 2.6, there are some gaps between the line segments representing the object. These gaps should be filled in some way. Suppose we use a certain constant value to fill these gaps, we will find that the object identity and location change on similar timescales. Compared with the drastic changes of primary sensory signals, object identity and location can still be regarded as slowly changing. When we are looking for an effective representation of the motion of an object, we usually consider that slowly changing signals can better represent effective information. In the field of computer vision, we usually use camera input data as research data. However, the data input by the camera is a matrix. When the external environment changes, even the slow change of the light and the background will cause drastic fluctuations in the input matrix. Therefore, figuring out a way to search for a function to transform rapidly changing camera input data into slowly changing output data becomes very important.

Firstly, we utilize mathematics to express how to learn slowly changing or even constant features. For input vector or matrix signal $x(t) = [x_1(t), \dots, x_i(t)]$, we need to find an input-output function $g(x) = [g_1(x), \dots, g_j(x)]$ to get j-dimension output signal $y(t) = [y_1(t), \dots, y_j(t)]$, then $y_j(t) \coloneqq g_j(x(t))$. For each $j \in \{1, \dots, J\}$:

$$minimize\ \Delta_j := \Delta(y_j) := < y_j^2 > \quad (7)$$

$$< y_j > = 0 \quad (8)$$

$$< y_j^2 > = 1 \quad (9)$$

$$\forall j' < j: \quad < y_{j'} y_j > = 0 \quad (10)$$

$$< f >: = \frac{1}{t_1 - t_0} \int_{t_0}^{t_1} f(t)\, dt \quad (11)$$

Equation (7) expresses our main goal, which is to minimize the time change of the output signal. Equation(8) and equation(9) can help us avoid the trivial solution, that is, the output is a constant term. Equation(10) ensures that the output signals are not related to each other, rather than simply copying each other, which makes it possible to reflect the different characteristics of the input signal. Equation(11) represents the meaning of angle brackets: temporal averaging.

In 2016, Jayaraman and Grauman used the slow and steady feature to obtain good classification results from rapidly changing video data (Jayaraman & Grauman, 2016). In

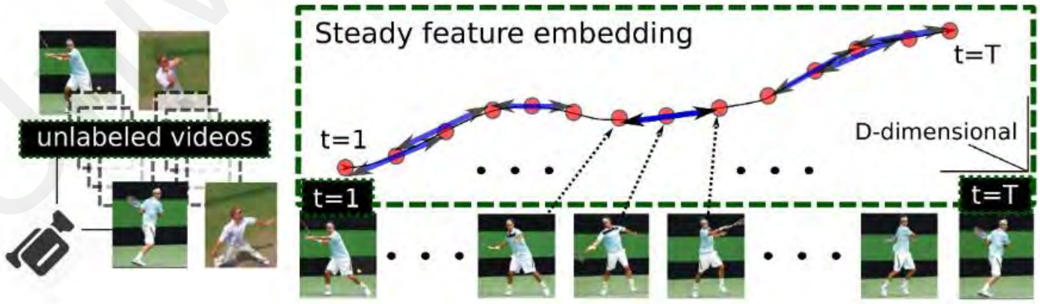


**Figure 2.6: The example of extract steady feature extract from video (Jayaraman & Grauman, 2016)**

Figure 2.6, Jayaraman and Grauman show how to extract steady features from fast-changing video motion data for recognition tasks. Their goal is to improve visual feature learning through unlabelled video data that is easily available on the Internet. Especially in visual recognition tasks to improve the performance of the learned video features. In the unsupervised video analysis based on the SFA principle, most of the algorithms are based on the use of temporal coherence between two adjacent frames of the video as free supervision. In the SFA algorithm, the features extracted from two adjacent frames should be approximately equal. This is because high-level semantic vision concepts related to video usually only slowly change according to changes in the pixels that make up the frame. Therefore, the features useful for recognizing high-level semantic concepts usually also have the characteristic of slowly changing over time. In the algorithm before this article, usually, people just keep the feature between two adjacent frames consistent. The basic idea is: for a learned representation z and adjacent video frames a and b, the constraint is $z(a) \approx z(b)$.

However, Jayaraman and Grauman believe that capturing how visual content changes over time is more important. Therefore, they hope to be able to capture steady visual dynamics from the video. This means they aim to learn a feature space in which frames from a novel video about human action would follow a smooth, predictable trajectory. Therefore, their algorithm not only encourages temporal coherency between two adjacent frames in the video. At the same time, the feature changes on consecutive frames should also be stable. That is, for the three temporally adjacent frames a, b, and c, the newly added constraint condition is $z(b) - z(a) \approx z(c) - z(b)$. Therefore, the steady learning proposed by Jayaraman and Grauman insists that no matter what changes occur in each frame of the video over time, they will continue to develop in the same way in the near future (Jayaraman & Grauman, 2016).

### 2.6 Supervised learning

In our work, supervised learning is a part of the proposed approach. Therefore, this section presents several supervised learning algorithms that are discussed later in this thesis.

#### 2.6.1 Decision tree

The decision tree can be expressed as the process of recursively dividing the sample space (Quinlan, 1986). In the decision tree, the tree nodes represent the attribute collection of the data set, and the branch of each node represents the attribute value of an attribute in the collection. The leaf nodes of the tree represent the decision attributes in the data set. The goal of the decision tree is to generate judgment rules for decision attributes through the learning of data sets. In 1986, Quanlian used a simple example to explain what a decision tree is (Quinlan, 1986). Table 2 is a simple training set used in the article. In this data set, every object is defined and described by a collection of attributes. Each attribute represents some important aspect of the object. The value of an attribute must be discrete and mutually exclusive. For example, in the simple data set of Table 2, the attributes of the Saturday morning object include:

outlook, with values {sunny, overcast, rain}
temperature, with values {cool, mild, hot}
humidity, with values {high, normal}
windy, with values { true, false }

**Table 2: The small training set for introducing decision tree (Quinlan, 1986)**

| No. | Attributes | | | | Class |
|---|---|---|---|---|---|
| | Outlook | Temperature | Humidity | Windy | |
| 1 | sunny | hot | high | false | N |
| 2 | sunny | hot | high | true | N |
| 3 | overcast | hot | high | false | P |
| 4 | rain | mild | high | false | P |
| 5 | rain | cool | normal | false | P |
| 6 | rain | cool | normal | true | N |
| 7 | overcast | cool | normal | true | P |
| 8 | sunny | mild | high | false | N |
| 9 | sunny | cool | normal | false | P |
| 10 | rain | mild | normal | false | P |
| 11 | sunny | mild | normal | true | P |
| 12 | overcast | mild | high | true | P |
| 13 | overcast | hot | normal | false | P |
| 14 | rain | mild | high | true | N |

Therefore, a Saturday morning might be described as:

Outlook: overcast

temperature: cool

humidity: normal

windy: false

In real life, every object is belong to one of a set of mutually exclusive classes. In the simple training set of Table 2, the author assumes that there are only two classes, namely P and Q. Positive instances are considered to be class P, and negative instances are considered to be class Q. In the training set, the class of each object is already known. In the classification task, the main task is to develop a classification rule to determine which class it belongs to based on the value of the object attribute. However, an important question is whether the value provided by the object's attributes contained enough information to help determine which category the object belongs to.

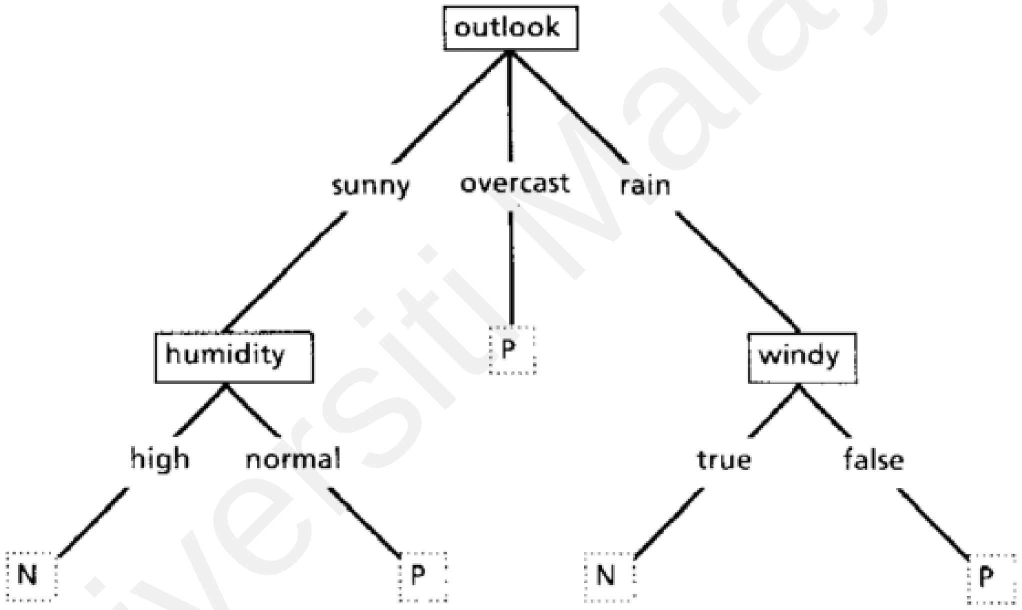


**Figure 2.7： A simple decision tree (Quinlan, 1986)**

Therefore, in the training set, if two objects have the same attribute value, they should belong to the same class, not different classes. As mentioned above, in the classification task, the classification rule can be expressed as a decision tree. In Table 2, a small data set using the attribute of "Saturday morning" is shown, and its category P indicates that it is suitable for certain unspecified activities this morning. A decision tree that can classify each object correctly is shown in Figure 2.7.

In Figure 2.7, leaves of the decision tree are the class name of the object, the other nodes represent the attribute-based tests. When we start to classify an object, we start from the root node of the decision tree, and then evaluate and choose the appropriate branch as the outcome. When this process reaches a leaf node (class), the object is classified into a certain category, and the classification of it is over.

In this example, we found that some objects can be identified only by evaluating a few attributes. For example, when the outlook attribute is overcast, it can be directly judged that the category of the object is P. This shows that sometimes we don't need to use all the attributes to judge the object. When we have enough attributes that contain a lot of useful information, we can always build a decision tree to classify all the objects correctly, and there are often multiple decision trees to achieve this goal. However, an important principle is that the constructed decision tree should not only classify the test set correctly, but also classify the training set correctly as well.

There are many kinds of decision tree algorithms, the root of which is the CLS algorithm. The basic idea of it is to use the data source as the starting point of the decision tree, and then divide it according to a certain attribute. The divided sub-data sources will be evaluated using the same method until the subset has no attributes that can be divided or belong to the same decision category. In real life, there are usually problems such as missing data attribute values, reduced necessary data, and noise. This usually leads to problems such as overfitting of the decision tree algorithm, decreased accuracy of the algorithm, and the too-complex structure of the decision tree. Therefore, the trained decision tree usually needs to be pruned.

**Table 3：ID3 Algorithm**

| Introduction to ID3 algorithm |
|---|
| 1. The attribute with the largest information gain value in the attribute set is selected as the root node, and the current node is divided according to this attribute.<br>2. Child nodes with the same attribute category are converted into leaf nodes.<br>3. Sub-nodes with different attribute categories will be divided according to the first step until the attributes in the node are of the same category.<br>4. When the samples in all nodes are of the same type, the decision tree establishment is completed. |

We can usually get different decision trees for the same data set. These decision trees can classify the data set correctly. Therefore, one way to select the appropriate decision tree is to calculate all the correctly classified decision trees at first and then select the simplest decision tree among them. However, due to the huge computing resources required, this method is only suitable for relatively small data sets. In order to adapt to a larger data set, ID3 was invented(Quinlan, 1986).

The ID3 algorithm introduces the concept of information entropy in information theory into the decision tree algorithm. The core of this algorithm is by using the concept of information gain to select the attributes on each node in the decision tree. For all the attributes in the attribute set, the ID3 algorithm will calculate the information gain, and select the attribute with the largest information gain to divide the decision tree. The size of the information gain reflects the uncertainty of the attribute selection. The larger the attribute gain, the smaller the uncertainty. Therefore, the attribute with the largest information gain should be selected in the selection of test attributes.

Although the ID3 algorithm could quickly construct the correct decision tree, it still has some shortcomings due to the test attributes selected by the ID3 algorithm using information gain. This algorithm is more effective when the number of samples in the branch is not so much different.

However, if there is a large gap in the number of samples in the child nodes, the algorithm usually selects attributes with a larger number of samples, which causes some important attributes with a small sample size to not be selected. In the process of building a tree, although the attributes of continuous values can be processed, the processing is very cumbersome. At the same time, since the algorithm is a top-down tree-building strategy, it will consider the training set one by one, which easily causes the decision tree to converge to a local optimal solution and thus cannot obtain a global optimal solution.

The C4.5 algorithm and the C5.0 algorithm improve the ID3 algorithm, which introduces the concept of gain ratio to overcome the problem that the maximum gain in the ID3 algorithm is biased towards multi-valued attributes and they can handle continuous attributes. However, the C4.5 classification algorithm still requires repeated scanning and sorting of sample sets, which results in a very inefficient algorithm.

In 1984, Breiman, Friedman, and Olshen et al. proposed a decision tree classification method(Breiman et al., 2017). The CART algorithm uses a method different from that in the ID3 series. It does not use the information estimation function, instead, it uses the Gini index standard based on the minimum distance. When using CART,and when the data set on a branch node in a decision tree roughly belongs to a certain category, we can use voting to decide which category this branch belongs to. Therefore, the continued expansion of the node is stopped and the node becomes a leaf node until the CART algorithm can generate a decision tree that can finally meet the requirements.

### 2.6.2 Linear Discriminant Analysis

Linear Discriminant Analysis is a method used to reduce dimensionality, and it is also an important method of supervised data sample classification. Another name is Fisher Discriminant Analysis. It has an important position in many supervised machine learning algorithms. The basic idea is to ensure that in a feature space, the distance between data samples of the same category is the smallest, and the distance between different data samples is the largest. This can ensure that the data sample points of the same category are as close together as possible, and at the same time, the data sample points of different categories are as far away as possible.

In a statistical sense, linear discriminant analysis is an analysis method that determines the classification of a data sample by selecting a criterion. The algorithm principle of Linear Discriminant Analysis is mapping the labelled high-dimensional data sample points to a normal vector $w$ in a lower-dimensional space through a mapping function. On this vector w, the distance between data sample points of the same category should be as small as possible, and at the same time, the distance between data sample points of different categories should be as large as possible, which could ensure that we get a better classification effect. Since the vector $w$ is the normal vector of the classification hyperplane in the mapping space, we can transform a binary classification problem into an optimization problem of the normal vector $w$ of the classification hyperplane.

In a standard binary classification task, our common practice is to map all the sample points of the data to a straight line in an one-dimensional space. For the data point $x_i$ , in a sample data set, we can regard it as an n-dimensional vector. Then, the mapping of this sample point on the normal vector w can be regarded as $w^T x_i$, if the category label $y_i \in \{0.1\}$, assuming that the number of data samples of the c-th type is $N_c$, the data points of the c-th type samples are the set is $X_c$, the mean vector of the c-th sample point is $u_c$, and the covariance matrix is $\sum_c$. Then, according to the basic theory of linear discriminant analysis, we need to maximize the distance between the centre points of different types of data sample points, then we need to maximize $||w^T u_0 \; - \; w^T u_1||_2^2$ to ensure that data sample points of different categories would not overlap and could be classified.

$$u_c = \frac{1}{N_c} \sum_{\mathrm{x} \in X_c} x \qquad c = (0,1) \tag{12}$$

$$\Sigma = \sum_{\mathrm{x} \in X_c} (x - u_c)(x - u_c)^T \, , \qquad c = (0,1) \tag{13}$$

While ensuring the maximum distance between sample points of different categories, we also need to ensure that the distance between sample points of the same category is the smallest. In a statistical sense, we need to ensure that the covariance of data sample points of the same category is minimized. Then, for a data set with only two categories, we need to minimize $w^T \sum_0 w \; + \; w^T \sum_1 w$. Finally, in theory, our optimization goal is as follows:

$$\mathrm{Max\ J(w)} \; = \frac{||w^T u_0 \; - \; w^T u_1||_2^2}{w^T \sum_0 w \; + \; w^T \sum_1 w} \tag{14}$$

By optimizing the objective function, we can transform the standard linear binary classification problem into the problem of solving the mapping vector w. According to the basic principle of linear discriminant analysis, we know that the best mapping direction ,*w,* can separate the two types of data samples.

In real life, we will not only encounter a large number of binary classification problems, but also a large number of multi-classification problems. When faced with multi-classification problems, linear discriminant analysis can still classify samples. When classifying multi-class data, the low-latitude space mapped is no longer a straight line, but a hyperplane. Its dimension is d-dimensional. W is an n*d matrix composed of all basis vectors. Multi-classification tasks can be completed by solving W.

### 2.6.3 SVM

Support vector machine (SVM) was proposed by Vapnik in 1995 when discussing the linear inseparability problem encountered in classification problems. It has a greater advantage in solving classification problems when the sample size is small. At the same time, it can also solve the problems of overfitting and dimensionality disasters in learning, and the performance is also very good. In view of its good adaptability and easy promotion, it is the most commonly used machine learning method so far. The SVM algorithm transforms the linearly inseparable problem of low-latitude space into high-latitude space by constructing a kernel function, thereby transforming the problem into a linearly separable problem and constructing a hyperplane to maximize the interval between samples.

Support vector machine algorithms usually achieve good classification results in the field of machine learning and pattern recognition. Thanks to the characteristics of SVM based on supervised learning, it can be applied to many statistical problems, such as classification problems, regression problems, pattern recognition problems. Since the SVM algorithm is based on statistical learning theory, it can improve the generalization ability of the model by optimizing the Structural Risk Minimization. The implementation process is to find the optimal classification hyperplane of the SVM algorithm, and then we can control the performance of the classifier by controlling the interval on both sides of the hyperplane. In the SVM algorithm, the optimal classification hyperplane is the hyperplane that can obtain the maximum interval. In the development of the SVM algorithm, by constructing various kernel functions, the mapping changes can be effectively used to transform the low-latitude linear inseparable problem into the linearly separable problem. Due to the above advantages, support vector machines can effectively solve some common problems in machine learning such as local optima, dimensionality catastrophe and overfitting. During the training process, the support vector machine selects a set of feature subsets (also called support vectors) in the training set to make the division of the support vector set equivalent to the division of the entire data set. This strategy can reduce the complexity of the calculation process while ensuring the accuracy of classification, so it is often used to solve classification problems and regression problems. In the process of comparing traditional machine learning methods, we found that it has obvious advantages in preventing overfitting, training speed, and accuracy.

The SVM algorithm is mainly based on the principle of structural risk minimization in the training process, while some traditional machine learning methods in the past were usually based on the principle of empirical risk minimization. When the sample size is large enough, the principle of empirical risk minimization is successful. However, under the condition of small sample size, this principle usually fails to get a good result. This shows that when the sample size is small, the principle of empirical risk minimization is unreasonable. Under normal circumstances, it is necessary to minimize both the empirical risk and the confidence range at the same time. This is the principle of structural risk minimization.

The basic structure of a support vector machine is similar to a neural network, and its output is a linear combination between intermediate nodes. Each intermediate node corresponds to a support vector. The intermediate nodes and connection weights of the model are automatically generated, and there is no need to make judgments based on the human experience. The development process of the SVM algorithm can be roughly summarized as, from low latitude to high latitude, from linear to nonlinear, from separable to inseparable, from classification to regression, which follows the cognitive law from special to general and from simple to complex. The initial goal of SVM is to solve the problem of pattern recognition (classification), and its goal is to minimize structural risks. Compared with the infinite sample size required by experience risk, structural risk seeks a compromise between experience risk and confidence range. It defines a compromise between the accuracy of the given data and the complexity of the approximation function, which makes the SVM algorithm more suitable for small sample data. In the SVM algorithm training process, the choice of kernel function is also very important. Commonly used kernel functions include Polynomial Kernel Functions, Radial Basis kernel Functions (RBF) and so on.

Although the basic theory of the SVM algorithm is aimed at two classification problems and there are many two classification algorithms, some scenarios that require the application of multiple classification algorithms, such as face recognition, are still an important development direction of the SVM algorithm. Next, some popular multi-class SVM algorithms will be introduced.

The standard algorithm introduced first, also known as the one-against-rest algorithm, constructs k-number of SVM classifiers for a multi-classification problem with a classification category of k. Its i-th SVM classifier uses the training samples in the i-th class as positive training samples and other samples as negative training samples. The final output classification result is the one with the largest output among the two types of samples. The disadvantage of this method is that the classification results are prone to produce points belonging to multiple categories and points that have not been classified. Similar to this algorithm is the one-against-one algorithm. The idea of this algorithm is to construct all possible two-class SVM algorithms in the classification task of k categories. This means that for k-class classification problems, we need to construct k(k-1)/2 two-class SVM classifiers. The classification result of each sample is naturally produced using the voting method. The class with the most votes is the classification result of the sample points. This algorithm has huge shortcomings as well. For a single binary classifier, there may be a problem of non-standard classification, and from the perspective of the entire multi-classification algorithm, there may be a problem of over-learning.

At the same time, as with the first algorithm, it is also possible that some sample points appear in multiple categories and some sample points are not classified. Finally, considering that we need to construct a classifier for any two categories, this means that we need to construct a large number of classifiers, which results in a very large amount of calculation and ultimately a slow decision-making speed. The hierarchical classification method improves the one-against-one algorithm. It first merges k categories into two categories and then re-divides them into two categories until the most basic category. In this way, different levels are formed, and each level uses SVM for classification.

As a classic classification algorithm, the SVM algorithm has obvious advantages and disadvantages. Its main advantage is that the SVM algorithm is specific for the limited sample situation, and its goal is to obtain the optimal solution under the existing information, rather than the optimal solution when the number of samples tends to be infinite. The SVM algorithm will theoretically obtain a global optimal point, which solves the local extremum problem that cannot be avoided in neural networks.

The SVM algorithm can transform the actual problem into a high-dimensional feature space through nonlinear transformation. Subsequently, it can construct a discriminant function in high-dimensional space to achieve the division of sample points in the original space. This makes the dimensionality of the sample being ineffectual to affect the complexity of the algorithm, thus cleverly solving the dimensionality problem. However, its shortcomings are also very obvious: In the SVM algorithm, the selection of the kernel function during the training process is very important. A good kernel function often plays a decisive role in the success of the classification task. However, question on how to choose the correct kernel function is still a huge challenge for researchers. Secondly, since the support vector machine is based on a small number of support vectors for algorithm training and design, its performance is often affected by some noises, thus challenges on how to build a more robust support vector machine are still a key issue. Finally, the calculation of the support vector machine algorithm is very large, which causes the algorithm to consume a lot of time for training when the training sample is large.

### 2.6.4 k-NN

The KNN algorithm (Chen et al., 1995; Soucy & Mineau, 2001) is a supervised classification algorithm, which is a non-parametric classification technique based on category learning (Kuncheva, 1997). It was originally proposed by Cover and Hart in 1968. It does not need to generate additional feature data of substitute samples for analysis, and its rule is the sample data itself (Cover, 1968). The KNN algorithm does not even require the consistency of the data, which means that the data can be noisy.

The basic principle of the KNN algorithm is very simple. First, the sample *y* to be classified is expressed as a feature vector with the same dimension as the sample in the training sample library, and then a distance function is selected to calculate the distance between the sample *y* to be classified and each sample in the training sample library. Then the K samples are selected with the smallest distance from the sample to be classified as the K nearest neighbours of *y*, and finally the category of the sample *y* to be classified based on the K nearest neighbours of *y* is determined.

Since the basic idea of the KNN algorithm is to predict the class of the sample based on the K nearest neighbour samples around the sample to be classified, in the KNN algorithm, the two factors that must be determined are the value of K and the selection of the distance function that measures the similarity of the samples. K represents the number of samples to be selected for reference, and the distance function is a non-negative function used to indicate the degree of similarity between different data. In the KNN algorithm, the selection of the model often requires a large number of independent data sets to verify the best choice.

The KNN algorithm has many advantages. For example, because the KNN algorithm is a non-parametric classification method, it can obtain a relatively good classification accuracy for unknown data and non-normally distributed data. And its principle is clear and simple, and easy to be implemented. Secondly, since the KNN algorithm is only related to a very small number of adjacent samples in class decision-making, the use of this algorithm can avoid the problem of imbalance in the number of samples. Finally, the classification method of the KNN algorithm is to directly utilize the relationship between the sample, which reduces the adverse impact on the classification result caused by the improper selection of category features, thereby minimizing the error in the classification process. This is conducive for us to achieve better performance in some classification tasks with unobvious data features.

However, the traditional KNN method also has many shortcomings. First of all, its classification is very slow. The KNN algorithm is a lazy learning method based on examples. According to its basic principles, we can know that it does not actually construct a classifier based on the training samples. It first stores all the training samples and temporarily performs calculations when it is about to start classifying. This requires the algorithm to calculate the similarity between the sample to be classified and each sample in the training library to get the K nearest samples and obtain a classification result. If the sample dimension is too high or the training sample is too large, the algorithm may not be able to obtain an acceptable time and space complexity. Secondly, the KNN algorithm has a greater dependence on the capacity of the sample library. In practical applications, we can often find that certain categories of some classification tasks cannot provide enough training samples so that there is no way to meet the condition of a relatively uniform feature space required by the KNN algorithm. In the end, the error in the recognition of the sample becomes higher. Finally, the KNN algorithm usually considers that each attribute of the sample contributes the same to the classification since the distance between samples is calculated based on all the attributes of the sample. However, among these attributes, some attributes (features) may be strongly correlated with the classification result, and some attributes may be weakly correlated with the classification result. Therefore, when calculating the similarity, if all the features were considered to have the same effect, it may mislead the classification results. At the same time, in the KNN algorithm, the choice of K value is also very important. Improper selection of K value will seriously affect the performance of classification.

Based on the shortcomings of the KNN algorithm, a large number of improved methods have been proposed. It mainly includes four types: accelerating the classification speed, maintaining the training sample library, optimizing the distance formula and determining the K value. First of all, as far as the training process is concerned, the lazy learning method is actually faster than the active learning method. However, in the testing phase or classification phase, due to the need to calculate the distance to all sample points, the lazy learning method is much slower than the active learning method. In response to this problem, most of the current methods are considered from the two aspects of reducing the sample size and accelerating the learning speed. When the sample size of the training sample is too large, in order to reduce the amount of calculation, we can condense the training sample, that is, process the training sample set. The basic principle is to search for K-nearest neighbours by selecting the optimal reference subset from the original training sample set, thereby reducing the number of training samples and improving computational efficiency. Another idea is to find the K nearest neighbours of the sample to be classified in the shortest possible time through a fast search algorithm. This method does not blindly calculate the distance between the sample to be classified and all the training samples when searching but uses certain methods to speed up the search, such as using the concept of network or hierarchy to organize training samples to improve search efficiency.

In order to avoid the disadvantage of different features (attributes) that have the same effect in the traditional KNN algorithm, we can assign different weights to different features in the distance function that measures the similarity. The weight of the feature is set according to the role of each feature in the classification. In order to ensure the performance of the KNN algorithm, we should also maintain the training sample library. This maintenance includes adding and deleting samples in the training sample library. However, this process is not a simple addition and deletion. The goal should ensure that the samples in the training sample library of the KNN algorithm can provide a relatively uniform feature space (Chen et al., 2005). Therefore, it is necessary to establish a certain criterion for the addition and deletion of training library samples. However, sometimes the sample library cannot provide enough training samples for each class. This makes the KNN algorithm unable to obtain a relatively uniform feature space, and simply increasing the training samples will create the problem of excessive calculation. In the KNN algorithm, the choice of K value is also very important, and its choice usually needs to be verified by a large amount of independent test data and multiple models. The choice of K value can be determined in advance or dynamic. In some cases, the selection of K value is very difficult due to the extremely unbalanced samples between categories. Hand and Vinciotti proposed a new solution to this situation (Hand & Vinciotti, 2003).

## 2.7 Human Action Recognition

Since self-stimulatory behaviour is also a kind of human action, a simple understanding of the current field of human action recognition is essential to a better understanding of this research. Human action recognition is a multi-disciplinary cross-field, which has a very wide range of social significance and huge challenges. This has attracted a large number of scholars and research institutions to participate. Action recognition generally includes three steps. First, the camera or sensor needs to obtain the original data, such as sequence data, 3D skeleton data. Subsequently, we need to extract the action features based on the original data. Finally, we can use some recognition methods to understand the semantics of actions based on action features. In the field of self-stimulatory behaviour research, research methods can be roughly divided into two categories: using sensors to collect data and using cameras to collect data. Here, we briefly introduce the ways of human action recognition based on computer vision.

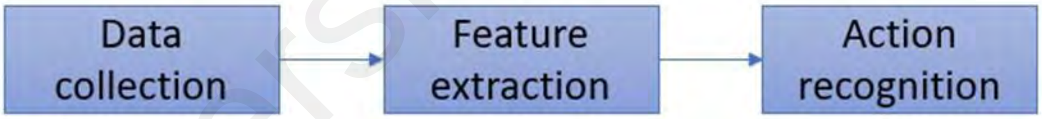


**Figure 2.8: Action Recognition Flow**

Feature extraction refers to the use of certain methods to extract the features of human actions based on the original input data. The feature extraction of human motion is a crucial step in the process of human motion recognition. The extracted features should mainly have the characteristics of low dimensionality, high clustering, and stability. The original human body motion video data contains a series of continuous frames, and each frame is composed of the RGB value of each pixel. Only the proper mapping and description of the original data can form more refined and reasonable features that contain the information. The current mainstream human motion data feature methods mainly include three feature extraction methods based on human body structure features, human appearance features, and human motion features.

The appearance feature of the human body refers to the description of the appearance of the moving human body. This method mainly locates the key frames of the human body in motion and extracts important information from them, such as silhouettes, contours, etc., as feature templates. Subsequently, by calculating the distance between the behaviour feature to be measured and the feature template, we can recognize the human body's action behaviour. Figure 2.9 shows the silhouette information of a running action.

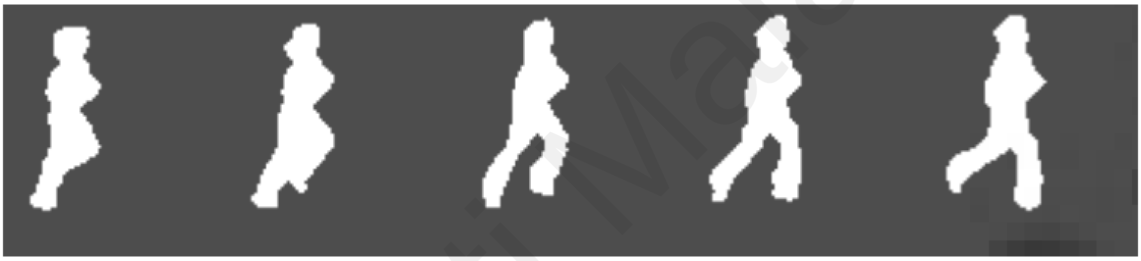


**Figure 2.9: Silhouette information of a running action**

Bobick et al. superimposed the silhouettes of human actions to obtain a motion energy image (MEI) that can express the effects of motion, and use motion functions to construct a motion history image (MHI) that can reflect the chronological order of human actions.

This has become a common feature of human behaviour recognition (Bobick & Davis, 2001) . Figure 2.10 is a frame in a video and it MEI and MHI.

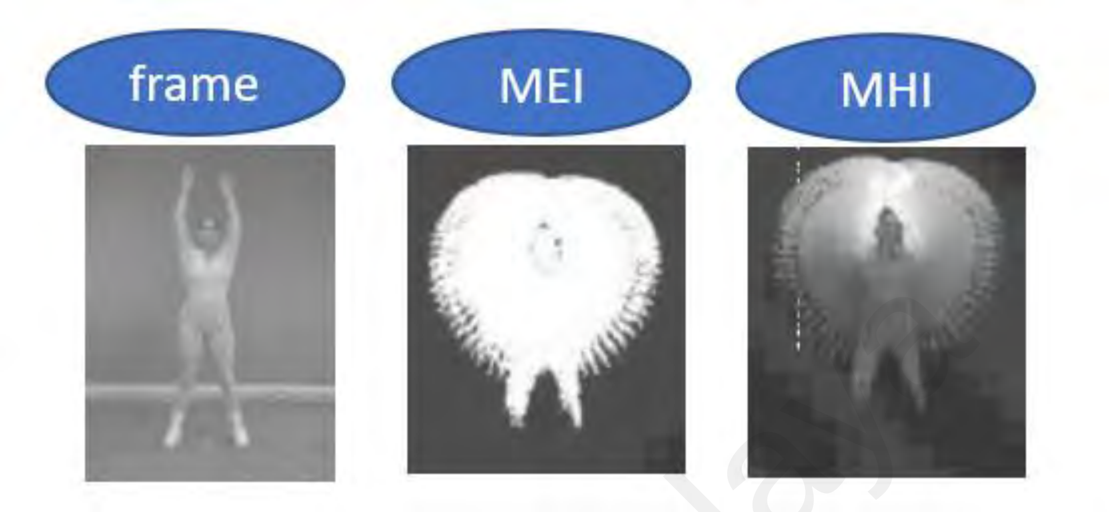


**Figure 2.10：The frame MEI and MHI of a video**

The appearance feature of human body movement includes the entire process of human body movement. Therefore, it can usually provide enough information for human body movement recognition, and its calculation process is simple and straightforward. However, the appearance of the human body is too dependent on the underlying visual operations, such as precise background subtraction, positioning of moving objects, etc. When the moving background of the human body is complicated, the viewing angle changes, the moving target is blocked, or the positioning of the moving object is not accurate, the feature extraction of the moving human body will become very difficult, which will affect the recognition of the movement.

The feature extraction method based on human motion is to express the motion process of the human body through the use of optical flow, motion trajectory, time and space interest points, etc., so as to hope to discover the motion law of different actions. Optical flow refers to the instantaneous speed of pixel motion of moving objects in a video. Polana first proposed the use of optical flow field information for behaviour recognition in 1994 (Polana & Nelson, 1994). Efros et al. divided the optical flow field into four directions and performed statistics separately to obtain more detailed movement direction information (Efros et al., 2003). The behaviour characteristics based on the optical flow field do not need to subtract the background, thus avoiding the problem of segmentation of complex environment background. However, since the optical flow field assumes that the gap between the image frame and the frame in the video is only caused by the movement of the target, it is possible to ignore the influence of external factors. Moreover, the dense optical flow with better features also requires a larger amount of calculation, which is not conducive to real-time action recognition. Spatio-temporal interest points mainly focus on and describe some unrelated points in human body motion information. The basic idea is to pay attention to certain areas of drastic changes in human movement. It does not need to detect moving targets, nor does it need to model the background. It is also not sensitive to changes in light viewing angle. However, since spatiotemporal interest points only focus on a few important areas and cannot express the overall motion state of the human body, the accuracy of their recognition is limited.

After obtaining the features of human body actions, we also need to classify the body actions according to the extracted features. Typical classification methods used in action recognition include template matching, state space, and deep learning. The template matching method is a direct classification method. In the classification, the entire action is summarized as a template representing this action. By comparing the test sample with the training sample template, we can calculate and classify the distance between the motion feature of the tested human body and the training template feature. For example, we can use motion history maps, motion energy maps, HOG, HOF, bag-of-words models and other features that can represent the global characteristics of human motion as templates, and use KNN, SVM, etc. to combine the template of the test sample and the template of the training sample by making comparisons to calculate the classification results. Although the template matching algorithm is simple to be calculated, this algorithm relies too much on the feature extraction link, and the quality of the extracted features directly affects the effect of the classifier. Another classification method is the state space classification method. This method uses each static posture or motion posture in the video sequence as a state node, and the various state nodes are connected with a certain probability. Each action sequence is determined by the final joint probability. The typical representative is Hidden Markov Model. Yamato et al. first introduced the HMM model to the human action recognition algorithm for research (Yamato et al., 1992). Caillette et al. proposed a variable-length Markov model to observe various behaviours and model 3D poses (Caillette et al., 2008). The state space method can mine the transition information between different states in the human body movement, but the calculation steps are usually very complicated. Researchers attach great importance to deep learning as a breakthrough in the field of computer vision. The deep neural network algorithm can save the extraction of artificial prior features, and it can learn features autonomously from a large amount of data. When there are enough training samples, the features learned through the deep network can usually make the classifier obtain a good classification effect. Generally, deep learning algorithms need to extract temporal and spatial features from video data. Ng, Donahue et al. used CNN network for feature extraction and then used LSTM to classify videos (Donahue et al., 2015; Yue-Hei Ng et al., 2015). In human action recognition tasks with huge amounts of data and complex environments, deep learning algorithms can often obtain better classification results.

## 2.8 Diagnosis based on accelerometers

There are some research articles that utilize the Deep learning method and Stereotypical Motor Movement behaviour to analyze autism (Rad et al., 2018; Sadouk et al., 2018). Sadouk and Gadi et al. have used a wearable inertial measurement unit to detect the Stereotypical Motor Movement of autistic patients. Stereotypical Motor Movement behaviour is very similar to self-stimulatory behaviours (Sadouk et al., 2018). These actions include repetitive mouth opening, hand waving, and complex body movements. In the development process of children, these actions greatly hinder children's learning and social interaction (Mandy & Skuse, 2008). Therefore, it is very important to monitor Stereotypical Motor Movement using advanced sensing technology throughout the screening and treatment of autism.

Mohammadian Rad et al. have used the wearable 3-axis accelerometers dataset to analyse the Stereotypical Motor Movement . First, they convert multiple dimensions of information obtained from multiple sensors into fixed-length signals, so that they can be treated as a frame to be recognized by the deep learning system. In this way, the data recorded by the accelerator at a certain moment can be expressed as a matrix, which can be analyzed by Convolutional Neural Networks (CNN). Subsequently, the CNN network will get a probability to predict if the detected movement is Stereotypical Motor Movement or not. Therefore, in their published work, they have used the convolution neuron network to extract features and used LSTM to classify them. However, since the model is fully trained using supervised learning methods, the model might not be able to adapt to the new data very well (Rad et al., 2018).

Westeyn et al. have used small 3-axis accelerometer modules to study self-stimulatory behaviour (Westeyn et al., 2005). The accelerators are placed on the right wrist, the back of the waist and on the left ankle of a non-autistic person. Then, this non-autistic person was prompted to mimic autistic patients to perform self-stimulatory behaviours to collect data. Finally, the collected accelerometer data was assessed using hidden Markov models (HMMs). Since this dataset was collected from a single, neurotypical adult, the model may not be well adapted to new data generated from people with autism. Other than that, Min et al. have also used accelerometer modules to assess self-stimulatory behavioural using the Time-Frequency methods to extract features together with the hidden Markov model to detect and label self-stimulatory behaviours (Min, 2017). When self-stimulatory behaviour occurs, the system will automatically use a webcam and microphone to store the patient's video and audio data. By using this system, doctors can view the patient's video data to diagnose and treat autism (Rad et al., 2018).

## 2.9 Diagnosis based on computer vision

A study published by Rajagopalan et al. (Rajagopalan et al., 2013) in 2013 demonstrated a standard action recognition pipeline on the new Self-Stimulatory Behaviour Dataset dataset (SSBD). This dataset was collected from public domain websites such as YouTube. Such video format datasets from an uncontrolled environment are difficult to be classified. Hence, Space-Time Interest Points (STIP) was used with the Harris3D detector in the Bag Of Words (BOW) framework to train the classifier. However, the results reported were not promising, the best accuracy was only 50.7%. Another article also published by (Rajagopalan & Goecke, 2014) used the SELECTION OF POSELET BOUNDING BOXES method to identify the positions of autistic children to create a motion model based on the Histogram of Dominant Motions (HDM) method. In this experiment, they have achieved a state-of-the-art result (73.6%) when the 5-fold cross-validation method was employed.

# CHAPTER 3: METHODOLOGY

As shown in Figure 3.1, the first part illiustrated the usage of an unsupervised TCDN method to automatically extract the features of the self-stimulatory behaviour videos of children with autism. This method is adapted from a paper published by Redondo-Cabrera et. al that has introduced Quadruplet Method, it is known to achieve unsupervised classification for human action recognition tasks (Redondo-Cabrera & Lopez-Sastre, 2019). Once the features were extracted, K-means and condition entropy methods were used to verify feature effectiveness.

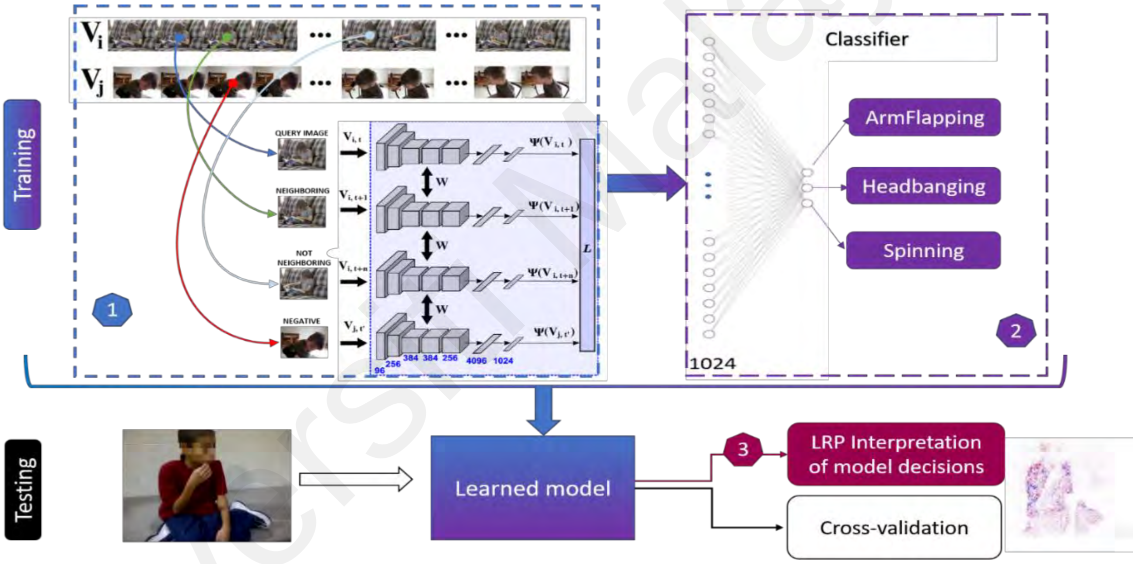


**Figure 3.1: The architecture of Temporal Coherency Deep Networks and supervised classifier (Redondo-Cabrera & Lopez-Sastre, 2019)**

In the second part, the performance of the identification of the self-stimulatory behaviour of autistic people is improvised. The features extracted using the unsupervised recognition method in the first part was used as input in this part to compare different supervised classification methods, such as Decision trees, Discriminant Analysis, Linear SVM, k-nearest neighbours algorithm (k-NN). Thereafter, the third part is to understand our model's internal mechanism to help humans design better models for the identification of the behaviour of autistic patients and to assist doctors in making diagnoses. We then selected the interpretable Linear SVM to be combined with the TCDN algorithm to produce the TCDN-SVM algorithm. The LRP algorithm was used to interpret it. The methods used in these three parts and the results obtained will be discussed in detail below.

## 3.1 Temporal Coherency Deep Networks (TCDN)

In the area of self-stimulatory behaviours research, self-stimulatory behaviours occur randomly. Hence, it is challenging to make an annotated video dataset, supervising the classification without labels becomes an issue as well. According to the slow feature analysis (SFA) method (Wiskott & Sejnowski, 2002), the image signal input by the camera, such as grayscale or point, is a low-level and rapidly changing representation of the action. Even when a child with autism moves slowly, the input signal will change quickly. If a high level, slowly changing or even unchanging features can be extracted from the input picture signal of each type of self-stimulatory behaviour, they can be used as free supervision for classification.

In this study, we proposed an input-output algorithm that utilises the temporal coherence between contiguous video frames as free supervision to extract features. In brief, our method uses unlabelled video data to train a convolutional neural network (CNN) model to extract features. Our objective function is as follows:

$$\min_{w} \frac{\delta}{2} W^2 + \sum_{i=1}^{T} L_u(W, U_i) \tag{15}$$

The input is a set of *m* unlabelled videos *V* = *V*$_1$*, V*$_2$*, V*$_3$*, ..., V*$_m$, and *W* is the parameters of our CNN network, *δ* is the weight decay constan, *L*$_u$ is the unsupervised regularization loss term, *U*$_i$ is the representation of training tuples of video frames. The key idea of this method is to keep the temporal coherence of adjacent frames in the learned feature representation. Meanwhile, the distance between two frames separated by *n* frames is shorter than the distance between frames from two different videos. The length of *n* frames is called the temporal window.

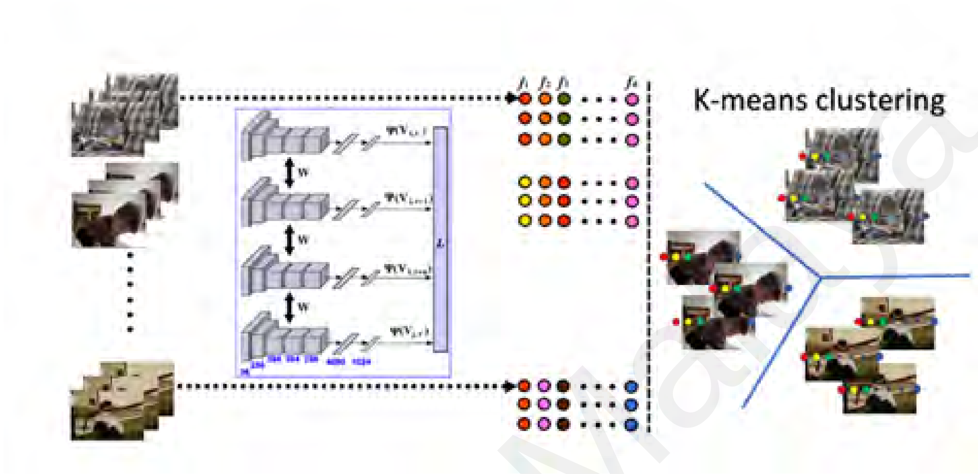


**Figure 3.2: The Architecture of Temporal Coherency Deep Networks (TCDN)**

Figure 3.2 represents the structure of TCDN. The input of this architecture includes the following four frames namely $V_{i,t}$ , $V_{i,t+1}$, $V_{i,t+n}$, and $V_{j,t}$'. These four frames are extracted from two videos $V_i$, $V_j$.

The $V_{i,t+1}$is an adjacent frame of $V_{i,t}$. There are *n* frames between $V_{i,t}$ and $V_{i,t+n}$ and the $V_{i,t}$ and $V_{j,t}$' originate from different videos. Then, there are four AlexNet networks that were used to process those four frames to four 1024 dimension representations ($\psi$). These four networks share the same parameters. We assume that the learned feature representation $\psi$ is a function of the learned AlexNet network parameters. The input of this function is a frame of a video, the output of this function is a feature representation *ψ* (*V*). In order to realize our key idea, we have designed a loss function $L$ (2) based on Euclidean distance *d* to train this network:

$$L_q\left(\psi(V_{i,t}),\psi(V_{i,t+1}),\psi(V_{i,t+n}),\psi(V_{j,t'})\right) \qquad (16)$$
$$= d\left(\psi(V_{i,t}),\psi(V_{i,t+1})\right)$$
$$+ max\left\{0, d\left(\psi(V_{i,t}),\psi(V_{i,t+n})\right)\right.$$
$$\left.- d\left(\psi(V_{i,t}),\psi(V_{j,t'})\right) + \alpha\right\}$$

This loss function tries to make the feature representation of $V_{i,t}$ similar to the feature representation of $V_{i,t+1}$. However, the distance between $V_{i,t}$ and $V_{j,t}$' must be greater than the distance between $V_{i,t}$ and $V_{i,t+n}$ by a constant α, because $V_{i,t}$ and $V_{j,t}$' originate from different videos. Therefore, the design purpose of our loss function is to hope that the distance between two adjacent frames is as small as possible and that the distance between two frames from different videos is greater than the distance of two non-adjacent frames from the same video.

## 3.2 Unsupervised Method To evaluate the unlabelled features extract by TCDN

In order the assess the efficacy of the TCDN extraction method, an unsupervised classification method and an evaluation indicator have to be introduced. The entire process ought to be unsupervised, meaning no labels from the dataset was used at all.

### 3.2.1 K means

The K-means is a traditional unsupervised classification method. In this study, the TCDN network was used to obtain 1024-dimensional features of each frame in the videos. They were then passed as inputs to the K-means algorithm for classification. Since the TCDN and K-means algorithms are both unsupervised algorithms, this method can be used to classify the behaviours of autistic patients in a completely unsupervised manner. For the K-means method, there are different types of initialising methods which could impact the performance of K-means classifier. Therefore, to obtain more credible results, three different methods were used to initialise K-means. The first method which is the K-means sample method, randomly selects k observations from the sample set. The second method is the K-means uniform, which uniformly selects k points from a range of sample sets. The last method selects k seeds by implementing the K-means++ algorithm for cluster centre initialisation. In step 1, one centre $c_1$ was chosen randomly and uniformly from the sample set. While in step 2 a new centre $c_i$ with probability was chosen as follows:

$$f(\mathrm{x})\frac{D(x)^2}{\sum_{x\in X} D(x)^2} \tag{17}$$

$D(\mathrm{x})$ represents the shortest distance from a candidate data point x to the nearest centre which we have selected. Step 2 was repeated until k centres were considered (Tuytelaars et al., 2010). Next, the impact of different K-means initialisation methods on K-means performance was compared. Considering that there were only three classes in our classification task, parameter k of K-means was set as three.

### 3.2.2 Conditional entropy

Following video classification, determining a method to evaluate the performance of our model becomes a challenge. Tuytelaars et al. compared different methods to evaluate the unsupervised model (Tuytelaars et al., 2010). Based on the result of the said evaluation (Redondo-Cabrera & Lopez-Sastre, 2019) have used unsupervised methods to perform human action identification, that is, using standard metrics named conditional entropy to evaluate models, as well as using a random method as the baseline. This method (conditional entropy) was used to evaluate our results. The conditional entropy method is as follows:

$$\mathrm{H(X|Y)} = \sum_{y \in Y} p(y) \sum_{x \in X} p(x|y) \log \left(\frac{1}{p(x|y)}\right) \tag{18}$$

## 3.3 Supervised Method To Classify Labelled Features

In the field of autism self-stimulatory behavior recognition, a large number of researchers have achieved good performance by applying supervised recognition methods. In order to compare our method with these methods, the method of supervised recognition has been introduced. Before beginning the experiment, the labels in the dataset were firstly used to label the extracted TCDN unsupervised features, and then these labelled features were used as input to train the supervised learning model.

Several supervised methods have been used to obtain better results. Firstly, the unsupervised temporal coherency deep networks (TCDN) is used to extract features as the input of the supervised method. As the TCDN is an unsupervised method, we were able to use all of the data to train this model, and then this model can be utilized to extract the feature of all frames from videos. After the features are obtained, those features were set as the input of the supervised method. Lastly, a frame-level classification was obtained. Below are few of the supervised methods that were used in this experiment:

### 3.3.1 Decision Trees

The decision tree is a straightforward and easily interpreted (De'ath & Fabricius, 2000). The parameters of this method are modified to result in three different trees. The Coarse Tree has a few leaves to make coarse distinctions, which makes the prediction more robust. However, this method is usually unable to attain high training accuracy. A

Medium Tree has a medium number of leaves, then the Fine tree has many leaves to make many fine distinctions.

### 3.3.2 Discriminant Analysis

The Linear Discriminant method creates linear boundaries between classes (Fisher, 1936). The Quadratic Discriminant creates nonlinear boundaries between classes (Srivastava et al., 2007).

### 3.3.3 Linear SVM

The idea of SVM is to find the best hyperplane that can split data points into different classes (Cristianini & Shawe-Taylor, 2000). In this article, because of the large dataset and lack of computer sources, only Linear SVM was chosen.

### 3.3.4 k-NN

This algorithm categorises points based on their distance to points (or neighbours) in a training dataset (Cover, 1968). It is a simple yet effective way of classifying new points. After evaluating the effects of using different sets of parameters (e.g. number of neighbours, distance method and distance weight) on the performance of k-NN classifiers, 5 k-NN methods were chosen to classify our SSBD dataset. Fine k-NN acquired finely detailed distinctions between classes, the number of neighbours was set to 1. The distance metric employed was the Euclidean distance while the distance weight was set to equal. Next, the Medium k-NN achieved medium distinctions between classes, the number of neighbours was set to 10. Finally, Coarse k-NN produced coarse distinctions between the classes. The number of neighbours here was set to 100. On the other hand, when the Euclidean distance was changed to cosine distance, the Cosine k-NN method yielded medium distinctions between classes. The number of neighbours was set to 10, the distance weight was set to equal. At last, when the distance weight was changed to the square inverse, the Weighted k-NN yielded medium distinctions between classes. The number of neighbours was also set to 10, the distance metric used was the Euclidean distance.

### 3.4 The Explainable Hybrid TCDN-SVM Model

Once the videos are represented by TCDN method, a majority of the supervised classifications yielded good enough accuracy. The reason for such state-of-the-art performance still cannot be found. Moreover, the multiplication of the nonlinear layers in the TCDN network caused the decision process of this method to lack transparency. Considering the interpretability of the linear SVM model, we decided to use the LRP method (Bach et al., 2015) to explain the hybrid model composed of TCDN and SVM.

As depicted in Figure 3.3, the forward transfer process of the convolution neuron network (CNN) sends the message from the node of one layer to the node of the next layer as follows:

$$z_{ij} = x_i w_{ij} \tag{19}$$

$$z_j = \sum_i z_{ij} + b_j \tag{20}$$

$$x_j = g(z_j) \tag{21}$$

The $x_i$ is the $i$-th element of the hidden layer $l$, weight $w_{ij}$links layer $l$ with the next layer $l+1$, and the variable $z_{ij}$ represents the forward message passed between the input neuron $(i)$ and the output neuron $(j)$. These forward messages were aggregated and combined after bias $(b_j)$ was added. Then, it was input into the nonlinear activation function $(g)$ to obtain the output $(x_j)$. The commonly used activation function is relu $g(z_j)$ = max(0, $z_j$).

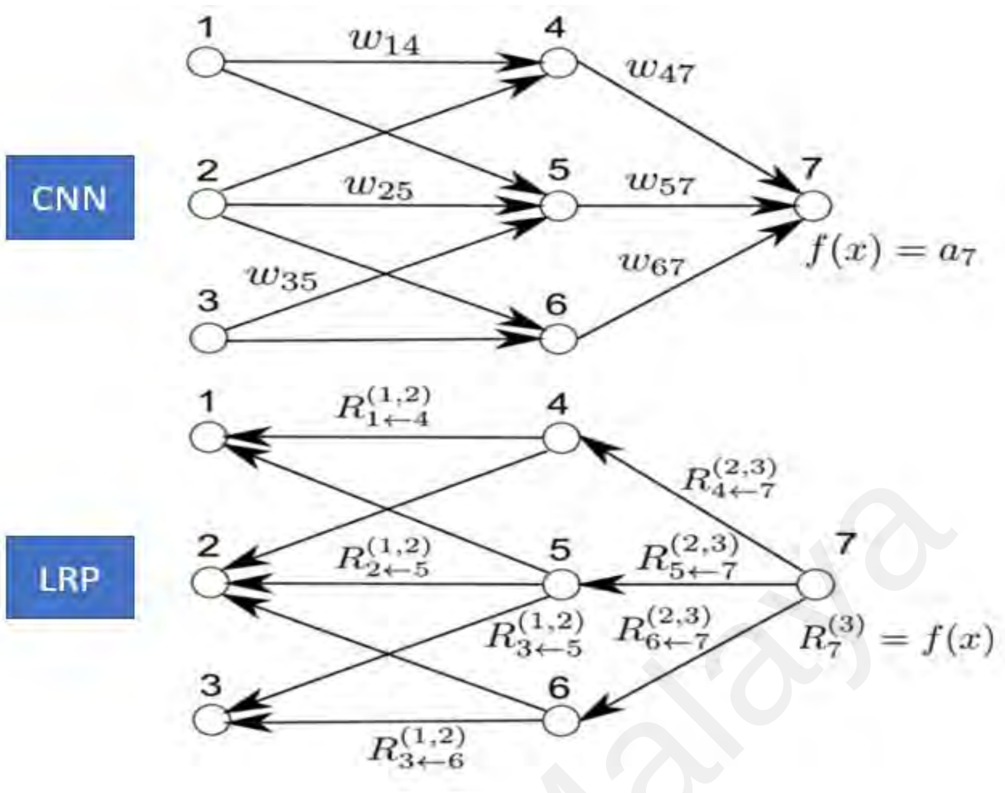

CNN
1
2
3
4
5
6
7
$w_{14}$
$w_{25}$
$w_{35}$
$w_{47}$
$w_{57}$
$w_{67}$
$f(x) = a_7$
LRP
$R_{1\leftarrow 4}^{(1,2)}$
$R_{4\leftarrow 7}^{(2,3)}$
$R_{2\leftarrow 5}^{(1,2)}$
$R_{5\leftarrow 7}^{(2,3)}$
$R_{3\leftarrow 5}^{(1,2)}$
$R_{6\leftarrow 7}^{(2,3)}$
$R_7^{(3)} = f(x)$
$R_{3\leftarrow 6}^{(1,2)}$

**Figure 3.3: TOP: A neural network-shaped classifier(such as CNN)$w_{ij}$ are weight and $a_i$ is the activation of neuron $i$. Bottom: The neural network-shaped classifier during layer-wise relevance computation time. $R_i^{(l)}$ is the relevance of neural $i$. $R_{i\leftarrow j}^{l,l+1}$ are messages which need to be computed to ensure the relevance conservation principle (Rajagopalan et al., 2013)**

Unlike forward propagation, LRP moves in the opposite direction of the layer to resolve the output of the classifier into a relevance message $R$. $R^{l,l+1}$ was set as relevance message which was sent from layer $l$+1 to layer $l$.

In the backpropagation training process of the traditional convolution neuron network, the gradient is calculated and is used to update the weight. However, the idea of backpropagation in LRP here is to explain the trained model. The weight is fixed in this method and the backpropagation is based on these weights to calculate the relevance value.

A set of constraints need to be kept to ensure the relevance conservation principle of LRP is upheld during layer-wise relevance computation time:

$$R_j^{l+1} = \sum_i R_{i\leftarrow j}^{l,l+1} \tag{22}$$

$$R_j^{l} = \sum_j R_{i\leftarrow j}^{l,l+1} \tag{23}$$

$$f(\mathrm{x}) = \cdots = \sum_{j\in(l+1)} R_j^{l+1} = \sum_{j\in l} R_j^{l} = \cdots = \sum_{d=1}^{\dim(x)} R_d^1 \tag{24}$$

As for tasks involving image classification, the overall idea of LRP was to comprehend the impacts of each pixel from the input image on the final prediction by the classifier.

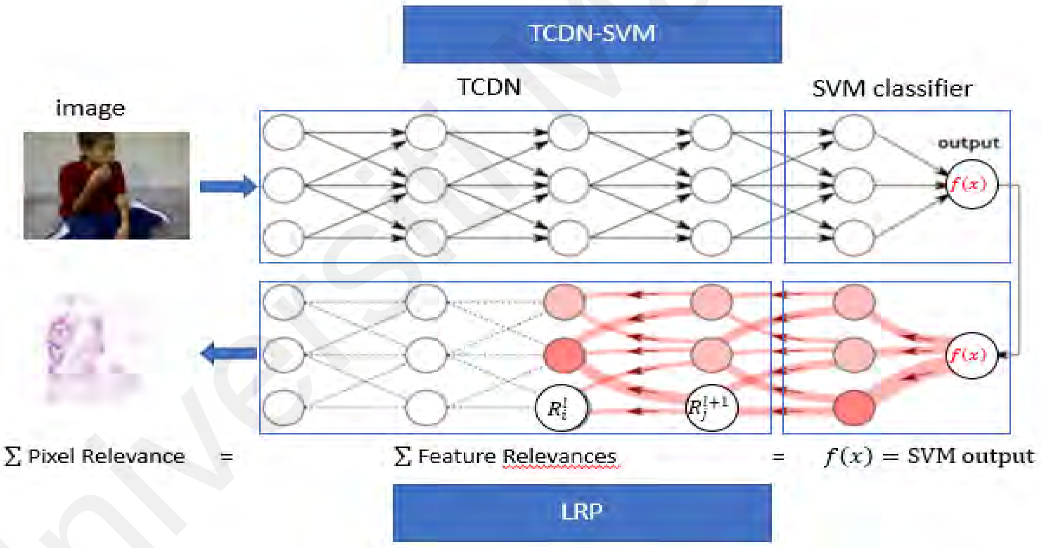


**Figure 3.4: Visualization of the Layer-wise Relevance Propagation (LRP) decomposition process. In the classification step, the image is converted to a feature vector representation by TCDN and an SVM classifier is used to get a category. The LRP method decomposes the SVM output $f(x)$ into the sum of feature and pixel relevance score. The final relevances (heatmap) visualize the contributions of single pixels to the prediction.**

In this study, three two-class SVM classifiers are equipped to complete the multi-classification task using one vs all strategy. In this strategy, for each binary learner, one of the three autistic behaviours was set as positive and all the remaining classes as negative. Then, the SVM is obtained for multi-tasking purposes. When the behavioural videos of people with autism were analysed, TCDN was used to extract the features of the frame. This input was then passed to the SVM classifier. Due to the similar structure between linear SVM and full connect layers in the AlexNet, the LRP method was used to explain TCDN-SVM, a model that is a mixture of TCDN and SVM.

This study utilised the fully connected linear LRP layer to perform the decomposition process of linear SVM because the structure of linear SVM is similar to that of the fully connected linear layer.

$$z_{ij} = x_i w_{ij} \qquad \mathrm{z_j} = \sum_{\mathrm{i}} \mathrm{z_{ij}} + \mathrm{b_j} \tag{25}$$

$$\mathrm{R}_{\mathrm{i \leftarrow j}}^{\mathrm{l,l+1}} = \frac{z_{ij}}{z_j} \mathrm{R}_{\mathrm{j}}^{\mathrm{l+1}} \tag{26}$$

consider if the $z_j$ is very small the relevant message $R_{i \leftarrow j}^{(l,l+1)}$ may become unbound. $\varepsilon$-decomposition formula was chosen, which introduces a sign-dependant numerical stabilizer ε in the formula.

$$\mathrm{R}_{\mathrm{i \leftarrow j}}^{\mathrm{l,l+1}} = \frac{z_{ij}}{\mathrm{z_j} + \varepsilon \cdot sign(\mathrm{z_j})} \mathrm{R}_{\mathrm{j}}^{\mathrm{l+1}} \tag{27}$$

In an ordinary image classification task, LRP usually uses the output of the softmax layer or fully connected layer in the artificial neural network as the initial input of LRP backpropagation. LRP can divide the relevance scores into positive and negative values in each layer using this step. When the LRP is propagated back into the image input layer, the relevance score at a pixel of the image becomes positive indicating that the pixel helps the model to classify the image into the correct category, the colour is set to red in the heatmap. Conversely, if the correlation is negative, the pixel prevents the model from classifying the image into the correct category (the colour is set to blue). Hence, we can determine which area in the picture is important for the classification task. In our model, the output of SVM represents the distance between the sample and the hyperplane. This distance is used as the initial input of the LRP algorithm to determine the areas in each frame of the autistic patient's video that affects the classification results (distance).

# CHAPTER 4: EVALUATION AND RESULTS

During the start of the study, the dataset was subjected to data pre-processing. Subsequently, the evaluation method was introduced. In this study, the structure of the methods will require the results to be splitted into three parts: In the first part, the K-means is used to classify the features and to evaluate the effectiveness of the features obtained using unsupervised learning method. The results were then compared with the baseline method (random classification method) by condition entropy. In the second part, the accuracy and confusion matrix was used to evaluate the performance of the different supervised methods. Finally, the LRP output is analysed to assess the classification model.

## 4.1 Data

In order to use real-time detection of the children's behaviours and provide early warning for parents in an uncontrolled environment, the SSBD was selected as our training test dataset (Arthur & Vassilvitskii, 2007). The SSBD dataset contains 75 self-stimulatory behaviour videos of autistic children, which were classified into three categories. However, since these videos were obtained from public domains such as Youtube, seven of the videos could not be downloaded due to copyright issues. We managed to randomly select 20 videos from each of the three classes to obtain a new dataset that contained 60 self-stimulatory behaviour videos.

Figure 4.1 illustrates some snapshots of the three types of actions. The faces of the patients were masked with mosaics to protect their identities.

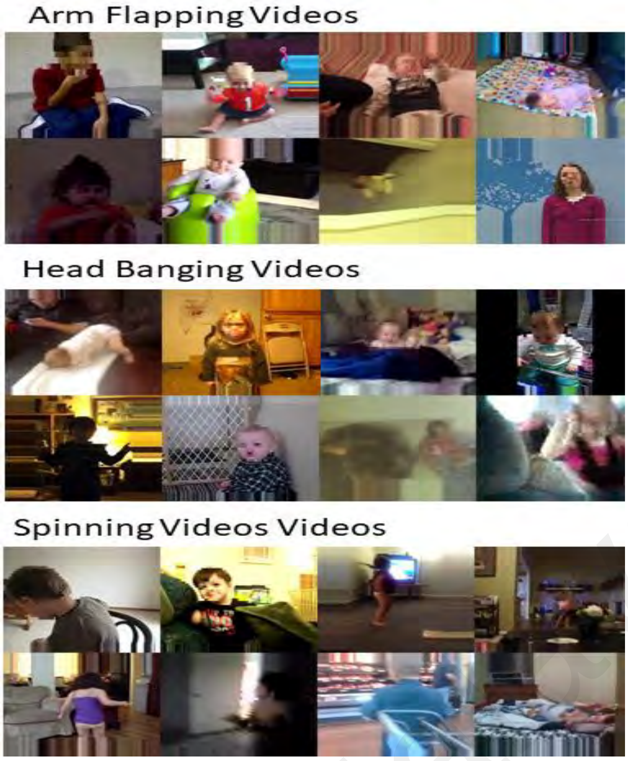


**Figure 4.1: Introduction of SSBD dataset (Arthur & Vassilvitskii, 2007)**

## 4.2 Classification using an unsupervised method to evaluate features

In order to ensure that our method becomes fully unsupervised, different K-means methods were used to classify the video data.

After classification with different K-means. the random classification method was chosen as the baseline method as it provided a reasonable reference standard. In this way, the classification performance of the K-means classifier can be evaluated by comparing it with the baseline. The performance of any classification method should be better than the random classification method. Referring to the paper published by (Redondo-Cabrera & Lopez-Sastre, 2019), the random classification method was used as a baseline for human motion recognition. Hence, the random classification method was employed to evaluate the efficiency of the unsupervised method on the autistic action dataset.

**Table 4.1: Condition entropy (CE) of K-means**

| **Method**<br>margin | CE<br>0.5 | CE<br>1 | CE<br>1.5 | CE<br>2 |
|---|---|---|---|---|
| **random(baseline)** | **1.56** | **1.56** | **1.56** | **1.56** |
| K-means uniform | 1.12<br>($\sigma$ 0.17) | 1.11<br>($\sigma$ 0.20) | 1.06<br>($\sigma$ 0.21) | 1.11<br>($\sigma$ 0.23) |
| K-means sample | 1.16<br>($\sigma$ 0.14) | 1.17<br>($\sigma$ 0.16) | 1.15<br>($\sigma$ 0.17) | 1.17<br>($\sigma$ 0.18) |
| K-means plus | 1.13<br>($\sigma$ 0.17) | 1.13<br>($\sigma$ 0.18) | 1.09<br>($\sigma$ 0.19) | 1.15<br>($\sigma$ 0.18) |

On the other hand, the parameters in the model were analyzed to assess the best performance. In this model, three important parameters may influence the performance of our model, namely (a) the margin $\alpha$, (b) the temporal window to consider contiguous frames ($w$) and the non-neighbour frame index ($n$). However, considering that different videos may have different frame rates in the wild environment, the temporal window might not be suitable. Therefore, it is very important to evaluate this parameter in a sufficiently large dataset. In this article, we have referred to the settings of the published article (Rajagopalan & Goecke, 2014) ($w$=1, $n$=20) because they have utilized a similar method for human motion recognition, and verified the temporal window on a larger data set (UCF101). However, considering that the movements of autistic patients are very different from that of normal people, we decided to analyse the optimal value of the margin parameter on our dataset.

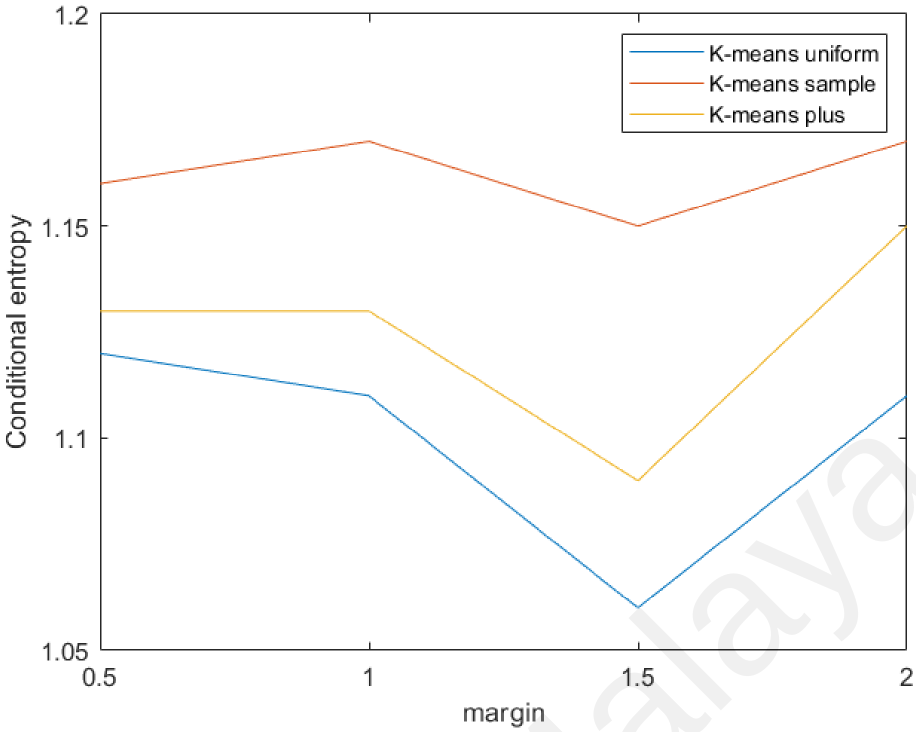


**Figure 4.2: The influence of the margin**

Table 4.1 lists the conditional entropy(CE) and standard deviation($\sigma$) obtained from various classification methods as the TCDN algorithm adopted different margin parameters. According to Table 4.1, the conditional entropy of random classification was 1.56, very close to the maximum conditional entropy $\log_2(3)$=1.58. If the conditional entropy of a classifier results in a maximum conditional entropy, this classifier can be considered completely futile (such as a random classifier). Therefore, we can prove our baseline method (random classify) is absolutely random. However, each of our K-means methods with different K-means initialization methods was better than the baseline. In order to intuitively illustrate the impact of different margin parameters and the K-means algorithm on the classification effects, Figure 4.2 was plotted. For the self-stimulatory behaviour classification task, using margin 1.5 and K-means uniform method, our proposed model classified the different autistic behaviours very well.

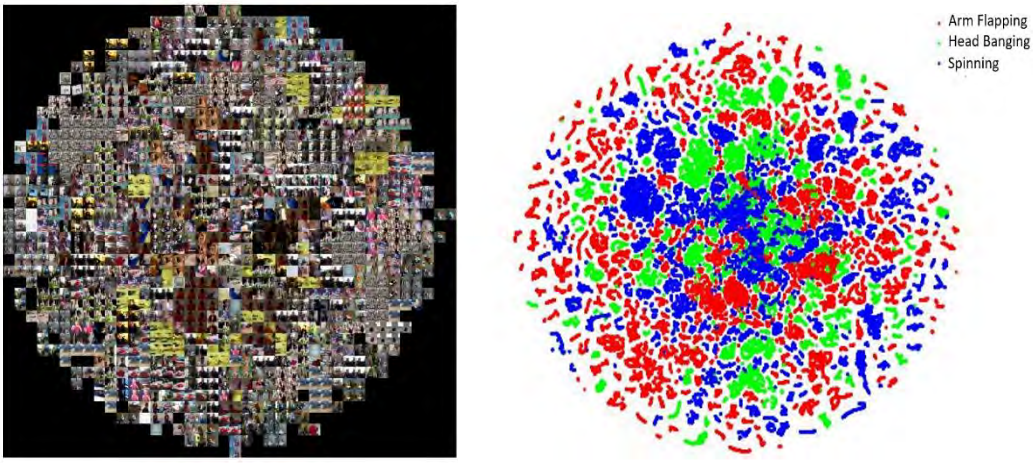


**Figure 4.3: Barnes-Hut t-SNE 2-dimensional embedding with our 1024-dimension fc7-features TOP: Drawn with frames BOTTOM: Drawn with scatter**

In general, when our unsupervised temporal coherency deep network method and K-means uniform method were combined, the remaining uncertainty on the real autism behaviour categories was reduced from a random classification of $2^{1.56}$= 2.95 to $2^{1.06}$= 2.08 (our method). This result indicates that the proposed unsupervised method is useful for the autistic data as the features extracted using this method are very effective.

Figure 4.3 depicts a 2-dimensional embedding using the Barnes-Hut t-SNE method. This method reduced the 1024- dimensional features data in this study to two dimensions by arranging pictures with similar features closely. This figure has shown the utilization of two different ways to show the clustering results of all self-stimulatory behaviours.

In this section, all our K-means methods used the five-fold cross-validation. In order to test the validity of our features more rigorously, we then repeated our experiment 20 times. The average of these experimental results was accepted as our final result.

### 4.2.1 Implementation details

In the training process of TCDN, the mini-batch Stochastic gradient descent (SGD) method was used to train our unsupervised TCDN due to the lack of training resources and to maintain the stability of the training process. In the network, the convolutional layer of AlexNet was used as the basic structure before adding two fully connected layers on the pool of 5-layer outputs. Hence, we obtained 1024-dimensional features to calculate the loss function. During the training process, we set the batch size to 40 tuples of frames. As for the parameters of the network, the start learning rate was set at 0.001, while the temporal window was set at 20.

### 4.3 Experimental Setup and Results of Supervised Method

In this section, the dataset from section 4.1 is used. Initially, all the videos in the dataset are used to train the TCDN network. Then, the said network is used to extract the features of frames in all the videos. In total, 136613 features were gathered for all the frames. A 5-fold cross-validation method was used to evaluate the performance of our supervised methods. In this experiment, 11 supervised methods were used.

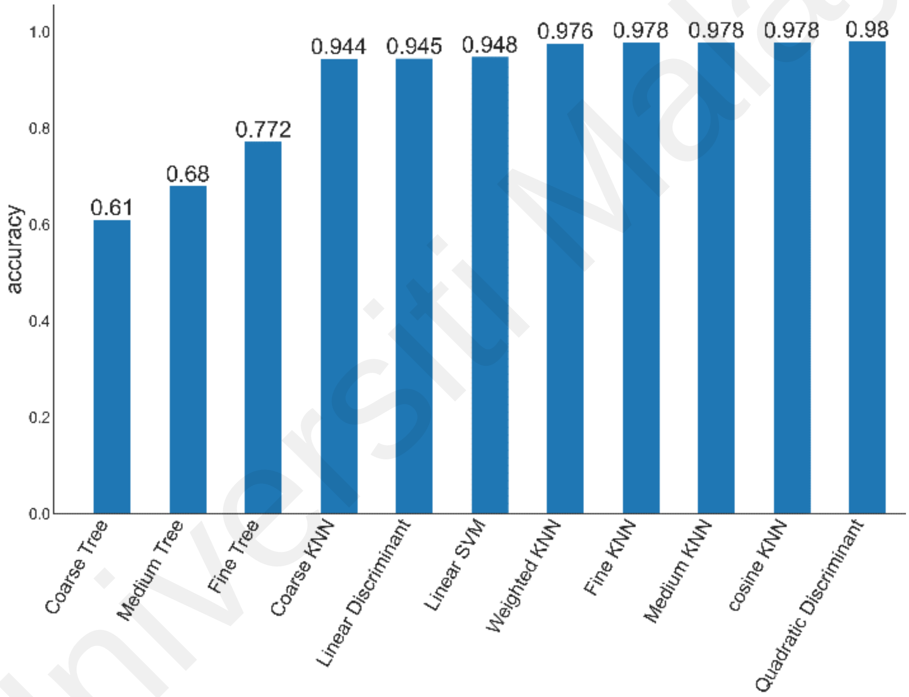
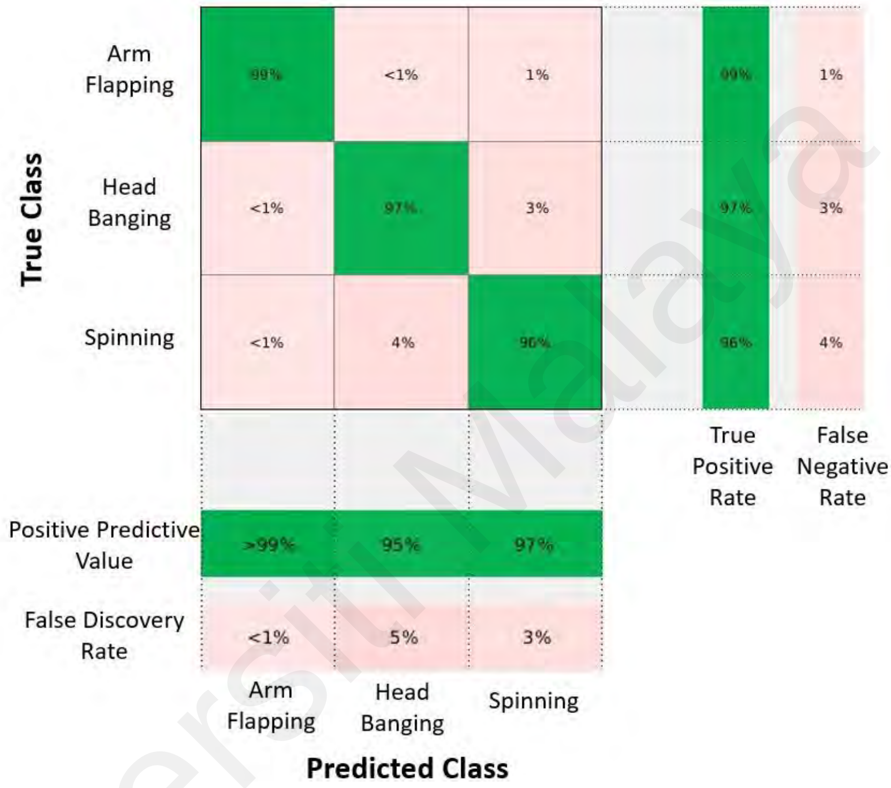


**Figure 4.4: Comparison of classification accuracy at the frame level**

Figure 4.4 indicates the accuracy of each supervised method. Based on the observation, the Quadratic Discriminant method demonstrated the best accuracy at the frame level, up to 0.98. To comprehensively evaluate the performance of the Quadratic Discriminant method, the Confusion matrix of our supervised methods were measured. Figure 4.5 represents the



| True Class | Arm Flapping | Head Banging | Spinning | True Positive Rate | False Negative Rate |
|---|---|---|---|---|---|
| Arm Flapping | 99% | <1% | 1% | 99% | 1% |
| Head Banging | <1% | 97% | 3% | 97% | 3% |
| Spinning | <1% | 4% | 96% | 96% | 4% |
| Positive Predictive Value | >99% | 95% | 97% | | |
| False Discovery Rate | <1% | 5% | 3% | | |

Predicted Class

**Figure 4.5: Confusion matrix of Quadratic Discriminant**

Confusion matrix of the Quadratic Discriminant. Since our classification method is to classify by frame, the proposed method detected the movements of children with autism in real-time and they are diagnosed in real-time. However, in order to compare the results

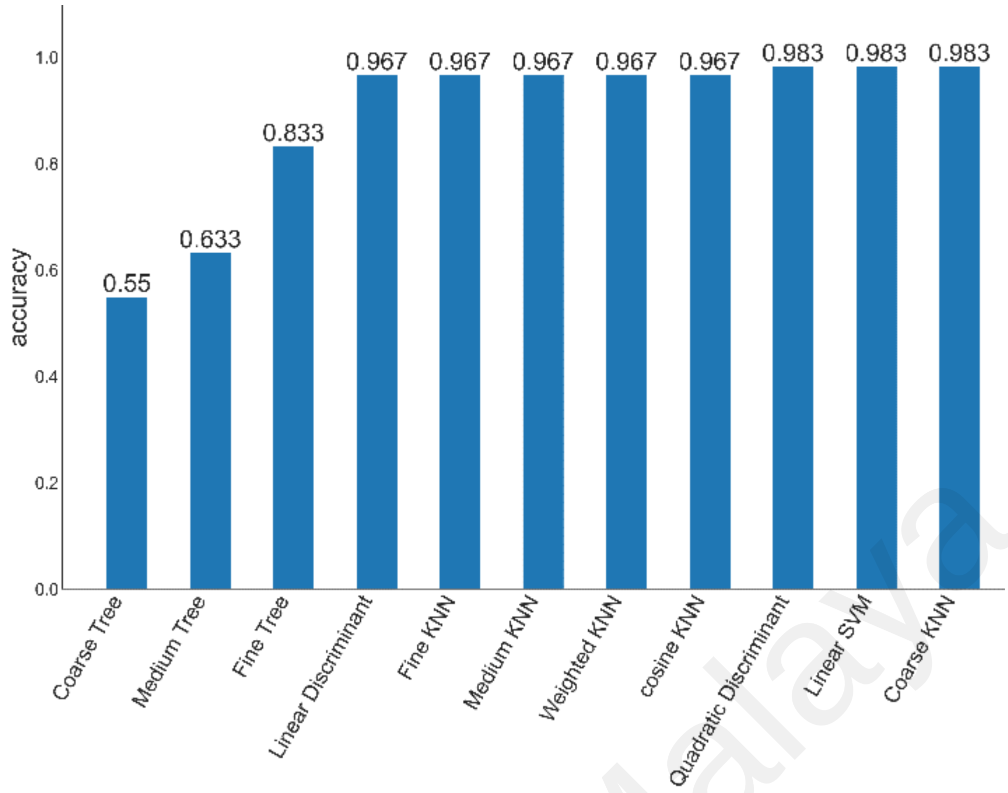


**Figure 4.6: Accuracy at the video level**

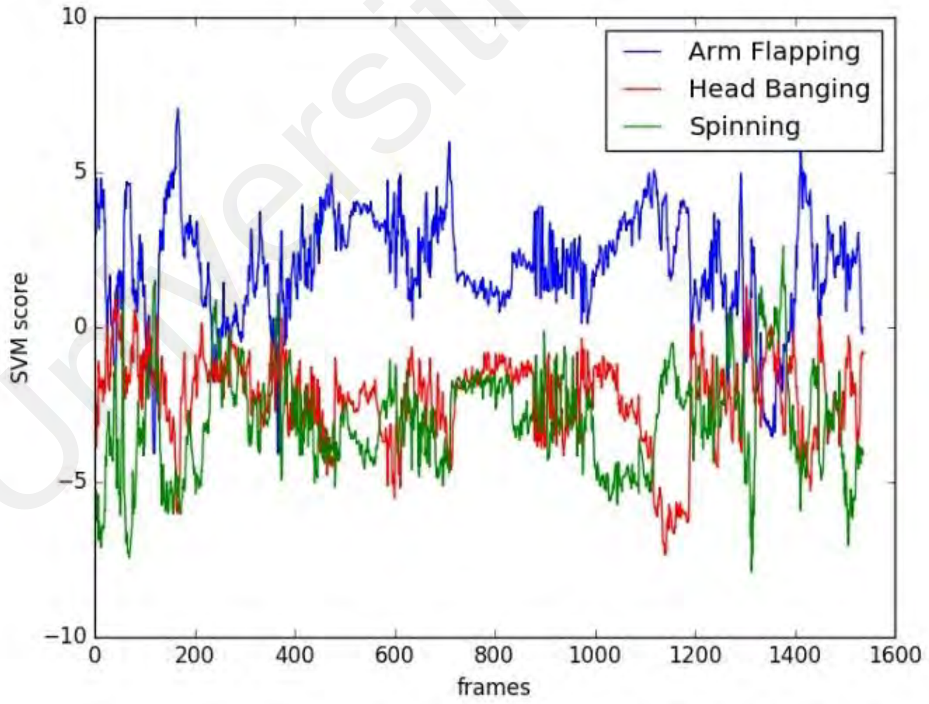


**Figure 4.7: The SVM classifier output(score) in the TCDN-SVM model. This figure shows the score of each frame in an Arm Flapping video.**

with previous studies, the accuracy of the video level was also calculated. In this study, the average classification score of all frames from a video was calculated (Figure 4.6), whereby the class with the largest average score was used as the classification of this video. The state-of-the-art accuracy for the Quadratic Discriminant, Linear SVM and Coarse k-NN methods were estimated at 98.3%. Figure 4.7 shows the TCDN-SVM classification result of one Arm Flapping video.

**Table 4.2: Comparison With The Recent State-Of-The-Art Result**

| **Method** | **accuracy** |
|---|---|
| **Poselet bounding box selection+ Histogram of Dominant Motions (HDM)+ discriminatory model (Rajagopalan & Goecke, 2014)** | **73.6%** |
| **TCDN and Coarse KNN** | **98.3%** |
| **TCDN and Linear SVM** | **98.3%** |
| **TCDN and Quadratic Discriminant** | **98.3%** |
| **TCDN and Cosine KNN** | **96.7%** |
| **TCDN and Weighted KNN** | **96.7%** |
| **TCDN and Medium KNN** | **96.7%** |
| **TCDN and Fine KNN** | **96.7%** |
| **TCDN and Linear Discriminant** | **96.7%** |
| **TCDN and Fine Tree** | **83.3%** |
| **TCDN and Medium Tree** | **63.3%** |
| **TCDN and Coarse Tree** | **55%** |

In the Matlab classification learner application, the three-class SVM classification task is decomposed into three binary classification tasks. The classifier performs classification by calculating the distance between the features extracted by the TCDN algorithm and the three hyperplanes, and the classification category of the hyperplane with the largest distance is used as the final category of the feature.

Table 4.2 shows the comparison between our results and the state-of-the-art results published by (Rajagopalan & Goecke, 2014). In the article published by (Rajagopalan & Goecke, 2014), in order to track the child's body motion, they have utilized the nearest neighbour algorithm to select the postlet bounding box. In these detected body regions, a Histogram of Dominant motions (HDM) descriptors is computed and are being used to train a discriminatory model. On the other hand, we have instead, used slow-changing discriminative self-stimulatory behaviours features to train supervised models (such as SVM, k-NN, Discriminant), and to classify self-stimulatory behaviours. Compared with the recent state-of-the-art result, we have obtained an improvement of 24.7%.

## 4.4 Explaining TCDN-SVM Using LRP Method

In order to understand the internal mechanism of the TCDN-SVM model, we visualize the output result ($\boldsymbol{R_1}$) of the LRP algorithm and used the heatmap to represent it.

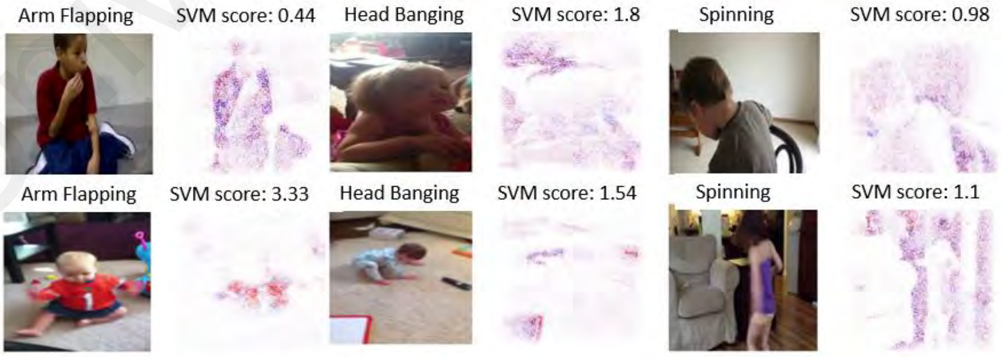


**Figure 4.8: Heatmap of the LRP output relevance($\boldsymbol{R_1}$)**

Figure 4.8 suggested that the basis of the model may be related to the magnitude of the action. When the model classifies arm-flapping behaviour, the model mainly focused on the effects of the arm on other parts of the body (such as occlusion). The headbanging behaviour also has the upper body of the patient moving along with the head, then the model begins to give some attention to the environment around the body. When the model begins to recognise the spinning behaviour, as the patient mainly rotates the whole body, the model recognises the human body influence on the surrounding environment (such as occlusion). Therefore, the heat map revealed that the model focused mainly on the environment near the body.

# CHAPTER 5: DISCUSSION

In this research, an unsupervised feature learning method, that is TCDN, was introduced to extract features from unlabelled videos. This method uses the local temporal coherence between contiguous frames as free supervision to obtain the ability to learn from unlabelled videos. As shown in Figure 4.2, it was found that different margins had different impacts on the clustering results. Thus, in order to separate the representation of different videos, a proper global discrimination margin is necessary.

Other than that, Table 4.1 has revealed the effectiveness of a completely unsupervised method that combines TCDN and k-means, after comparing the results of different k-means classifiers with the random classification method (baseline). This has provided users with the ability to take advantage of many unlabelled autistic self-stimulatory behaviours videos.

Compared with the state-of-the-art result (73.6%) using HDM descriptor features and discriminant classifier, our slow-changing discriminative self-stimulatory behaviours features and discriminant was able to achieve a higher accuracy (98.3%). This means that the features extracted by TCDN can improve the accuracy of 24.7% in autistic self-stimulatory behaviour classification tasks. The huge improvement of classification performance strongly proves the effectiveness and superiority of the TCDN method.

Considering the future application in the medical field, it is necessary to understand the internal mechanism of the model. The analysis of the LRP output (Heatmap) indicated that our model classified the self-stimulatory behaviours by analysing the interaction between autistic patients and their surrounding environment. This mechanism has laid a solid foundation for an accurate classification of the model. The success and reasonable explanation of our model directly indicate the effectiveness of our model.

In this thesis, the unsupervised feature extraction method has been applied on the autism self-stimulatory behavior recognition task. The advantage of the unsupervised method is that the model can match the unlabelled dataset better, and can be used as a prior input of the supervised method. The combined model of supervised and unsupervised method (TCDN-SVM) had been proven to obtain better accuracy for the autism self-stimulatory behavior recognition task. This new structure and approach with significant performance will greatly contribute to the autism self-stimulatory behavior recognition task.

# CHAPTER 6: CONCLUSION

In this study, an unsupervised feature learning method was used to extract the slow-changing discriminative self-stimulatory behaviours features from unlabelled videos.

Comparison of the conditional entropy results of the k-means classifier and the random classifier shows the efficiency of completely unsupervised classification using TCDN and k-means.

As compared to the recent state-of-the-art result (73.6%), our method was able to achieve a higher accuracy (98.3%). Considering that the same classifier was used in both studies, it is proven that TCDN feature extraction method was more efficient.

In conclusion, based on the results obtained from the experiment, it is possible to identify effective representations from unlabelled self-stimulatory behaviour video data， subsequently using them as a prior input for supervised learning methods. This entire structure of training was able to perform well in the autism spectrum self-stimulatory behaviors classification task.

# CHAPTER 7: FUTURE WORKS

Although our proposed methods were considered successful, there are still some limitations to look into. The dataset used in this study was small, hence it could not be generalized to fit into a nationwide data. Thus, the collection of more self-stimulatory behaviours videos of autistic patients would definitely help in expanding the dataset for better model training. The deep learning neural network contains a great amount of parameters and it poses substantial challenges to researchers who are trying to accurately explain the structure. Future works in this area will require us to continue on this journey of explaining the neural networks for better understanding and development.

# REFERENCES

Alnajjar, F., Cappuccio, M., Renawi, A., Mubin, O., & Loo, C. K. (2021). Personalized robot interventions for autistic children: an automated methodology for attention assessment. *International Journal of Social Robotics*, *13*(1), 67-82.

Alnajjar, F. S., Renawi, A. M., Cappuccio, M. L., & Mubin, O. (2019). A Low-Cost Autonomous Attention Assessment System for Robot Intervention with Autistic Children. *2019 IEEE Global Engineering Education Conference (EDUCON)*, 787-792.

Arthur, D., & Vassilvitskii, S. (2007). *K-means++: The advantages of careful seeding*, Association for Computing Machinery.

Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., & Baskurt, A. (2011). Sequential deep learning for human action recognition. International workshop on human behavior understanding,

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, *10*(7), e0130140.

Berkes, P., & Wiskott, L. (2005). Slow feature analysis yields a rich repertoire of complex cell properties. *Journal of vision*, *5*(6), 9-9.

Bobick, A. F., & Davis, J. W. (2001). The recognition of human movement using temporal templates. *IEEE transactions on pattern analysis and machine intelligence*, *23*(3), 257-267.

Bodfish, J. W., Symons, F. J., Parker, D. E., & Lewis, M. H. (2000). Varieties of repetitive behavior in autism: Comparisons to mental retardation. *Journal of autism and developmental disorders*, *30*(3), 237-243.

Boureau, Y.-L., Bach, F., LeCun, Y., & Ponce, J. (2010). *Learning mid-level features for recognition*.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (2017). *Classification and regression trees*. Routledge.

Bright, T., Bittick, K., & Fleeman, B. (1981). Reduction of self-injurious behavior using sensory integrative techniques. *The American Journal of Occupational Therapy*, *35*(3), 167-172.

Bryson, S. E., Zwaigenbaum, L., McDermott, C., Rombough, V., & Brian, J. (2008). The Autism Observation Scale for Infants: scale development and reliability data. *Journal of autism and developmental disorders*, *38*(4), 731-738.

Burack, J. A., Charman, T., Yirmiya, N., & Zelazo, P. R. (2001). Development and autism: Messages from developmental psychopathology. *The development of autism: Perspectives from theory and research*, 3-15.

Caillette, F., Galata, A., & Howard, T. (2008). Real-time 3-D human body tracking using learnt models of behaviour. *Computer Vision and Image Understanding*, *109*(2), 112-125.

Cantin-Garside, K. D., Kong, Z., White, S. W., Antezana, L., Kim, S., & Nussbaum, M. A. (2020). Detecting and classifying self-injurious behavior in autism spectrum disorder using machine learning techniques. *Journal of autism and developmental disorders*, *50*(11), 4039-4052.

Chen, J.-H., Chen, H.-M., & Ho, S.-Y. (2005). Design of nearest neighbor classifiers: multi-objective approach. *International Journal of Approximate Reasoning*, *40*(1-2), 3-22.

Chen, Y. Q., Nixon, M. S., & Damper, R. I. (1995). Implementing the k-nearest neighbour rule via a neural network. Proceedings of ICNN'95-International Conference on Neural Networks,

Cover, T. (1968). Nearest neighbor pattern classification. *IEEE Trans. Information Theory*, *4*(5), 515-516.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press.

Dawood, F., & Loo, C. K. (2016). Incremental episodic segmentation and imitative learning of humanoid robot through self-exploration. *Neurocomputing*, *173*, 1471-1484.

Dawood, F., & Loo, C. K. (2018). Developmental approach for behavior learning using primitive motion skills. *International journal of neural systems*, *28*(04), 1750038.

De'ath, G., & Fabricius, K. E. (2000). Classification and regression trees: a powerful yet simple technique for ecological data analysis. *Ecology*, *81*(11), 3178-3192.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). *Long-term recurrent convolutional networks for visual recognition and description*.

Efros, A. A., Berg, A. C., Mori, G., & Malik, J. (2003). Recognizing action at a distance. Computer Vision, IEEE International Conference on,

Ekblad, L., & Pfuhl, G. (2017). Ekblad Pfuhl Autistic self-stimulatory behaviors (stims): Useless repetitive behaviors or nonverbal communication?

Estes, A., Munson, J., Rogers, S. J., Greenson, J., Winter, J., & Dawson, G. (2015). Long-term outcomes of early intervention in 6-year-old children with autism spectrum disorder. *Journal of the American Academy of Child & Adolescent Psychiatry*, *54*(7), 580-587.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, *7*(2), 179-188.

Is there an epidemic of autism?, 107 American Academy of Pediatrics 411-413 (2001).

Fong, L., Wilgosh, L., & Sobsey, D. (1993). The experience of parenting an adolescent with autism. *International Journal of Disability, Development and Education*, *40*(2), 105-113.

Fukushima, K., & Miyake, S. (1982). Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition. In *Competition and cooperation in neural nets* (pp. 267-285). Springer.

Hand, D. J., & Vinciotti, V. (2003). Choosing k for two-class nearest neighbour classifiers with unbalanced classes. *Pattern recognition letters*, *24*(9-10), 1555-1562.

Hashemi, J., Spina, T. V., Tepper, M., Esler, A., Morellas, V., Papanikolopoulos, N., & Sapiro, G. (2012). A computer vision approach for the assessment of autism-related behavioral markers. 2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL),

He, K., & Sun, J. (2015). Convolutional neural networks at constrained time cost. Proceedings of the IEEE conference on computer vision and pattern recognition,

Connectionist learning procedures, Machine learning 555-610 (Elsevier 1990).

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, *313*(5786), 504-507.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*(8), 1735-1780.

Hubel, D. H., & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, *160*(1), 106.

Hurri, J., & Hyvärinen, A. (2003). Simple-cell-like receptive fields maximize temporal coherence in natural video. *Neural computation*, *15*(3), 663-691.

Jayaraman, D., & Grauman, K. (2016). Slow and steady feature analysis: higher order temporal coherence in video. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,

Ji, S., Xu, W., Yang, M., & Yu, K. (2012). 3D convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, *35*(1), 221-231.

Kanner, L. (1943). Autistic disturbances of affective contact. *Nervous child*, *2*(3), 217-250.

Kanner, L. (1971). Follow-up study of eleven autistic children originally reported in 1943. *Journal of autism and childhood schizophrenia*, *1*(2), 119-145.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). *ImageNet Classification with Deep Convolutional Neural Networks* Red Hook, NY, USA, Curran Associates Inc.

Kuncheva, L. I. (1997). Fitness functions in editing k-NN reference set by genetic algorithms. *Pattern Recognition*, *30*(6), 1041-1049.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436-444.

Lord, C., Brugha, T. S., Charman, T., Cusack, J., Dumas, G., Frazier, T., Jones, E. J., Jones, R. M., Pickles, A., & State, M. W. (2020). Autism spectrum disorder. *Nature reviews Disease primers*, *6*(1), 1-23.

Lord, C., Risi, S., Lambrecht, L., Cook, E. H., Leventhal, B. L., DiLavore, P. C., Pickles, A., & Rutter, M. (2000). The Autism Diagnostic Observation Schedule—Generic: A standard measure of social and communication deficits associated with the spectrum of autism. *Journal of autism and developmental disorders*, *30*(3), 205-223.

Lovaas, I., Newsom, C., & Hickman, C. (1987). Self‐stimulatory behavior and perceptual reinforcement. *Journal of applied behavior analysis*, *20*(1), 45-68.

Maenner, M. J., Shaw, K. A., Bakian, A. V., Bilder, D. A., Durkin, M. S., Esler, A., Furnier, S. M., Hallas, L., Hall-Lande, J., & Hudson, A. (2021). Prevalence and characteristics of autism spectrum disorder among children aged 8 years—autism and developmental disabilities monitoring network, 11 sites, United States, 2018. *MMWR Surveillance Summaries*, *70*(11), 1.

Magnússon, P., & Sæmundsen, E. (2001). Prevalence of autism in Iceland. *Journal of autism and developmental disorders*, *31*(2), 153-163.

Mandy, W. P., & Skuse, D. H. (2008). Research review: What is the association between the social‐communication element of autism and repetitive interests, behaviours and activities? *Journal of Child Psychology and Psychiatry*, *49*(8), 795-808.

Min, C.-H. (2017). Automatic detection and labeling of self-stimulatory behavioral patterns in children with Autism Spectrum Disorder. 2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC),

Polana, R., & Nelson, R. (1994). Low level recognition of human motion (or how to get your man without finding his body parts). Proceedings of 1994 IEEE Workshop on Motion of Non-rigid and Articulated Objects,

Powers, S., Thibadeau, S., & Rose, K. (1992). Antecedent exercise and its effects on self‐stimulation. *Behavioral Interventions*, *7*(1), 15-22.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106.

Rad, N. M., Kia, S. M., Zarbo, C., van Laarhoven, T., Jurman, G., Venuti, P., Marchiori, E., & Furlanello, C. (2018). Deep learning for automatic stereotypical motor movement detection using wearable sensors in autism spectrum disorders. *Signal Processing*, *144*, 180-191.

Rajagopalan, S., Dhall, A., & Goecke, R. (2013). Self-stimulatory behaviours in the wild for autism diagnosis. Proceedings of the IEEE International Conference on Computer Vision Workshops,

Rajagopalan, S. S., & Goecke, R. (2014). Detecting self-stimulatory behaviours for autism diagnosis. 2014 IEEE International Conference on Image Processing (ICIP),

Redondo-Cabrera, C., & Lopez-Sastre, R. (2019). Unsupervised learning from videos using temporal coherency deep networks. *Computer Vision and Image Understanding*, *179*, 79-89.

Rehg, J. M. (2013). *Behavior imaging and the study of autism* Proceedings of the 15th ACM on International conference on multimodal interaction, Sydney, Australia.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533-536.

Sadouk, L., Gadi, T., & Essoufi, E. H. (2018). A novel deep learning approach for recognizing stereotypical motor movements within and across subjects on the autism spectrum disorder. *Computational intelligence and neuroscience*, *2018*.

Sargano, A. B., Wang, X., Angelov, P., & Habib, Z. (2017). Human action recognition using transfer learning with deep representations. 2017 International joint conference on neural networks (IJCNN),

Schreibman, L., & Carr, E. G. (1978). Elimination of echolalic responding to questions through the training of a generalized verbal response. *Journal of applied behavior analysis*, *11*(4), 453-463.

Seltzer, M. M., Krauss, M. W., Shattuck, P. T., Orsmond, G., Swe, A., & Lord, C. (2003). The symptoms of autism spectrum disorders in adolescence and adulthood. *Journal of autism and developmental disorders*, *33*(6), 565-581.

Shattuck, P. T., Durkin, M., Maenner, M., Newschaffer, C., Mandell, D. S., Wiggins, L., Lee, L.-C., Rice, C., Giarelli, E., & Kirby, R. (2009). Timing of identification among children with an autism spectrum disorder: findings from a population-based surveillance study. *Journal of the American Academy of Child & Adolescent Psychiatry*, *48*(5), 474-483.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Smith, A. M. (2001). *Alhacen's Theory of Visual Perception: A Critical Edition, with English Translation and Commentary, of the First Three Books of Alhacen's De Aspectibus, the Medieval Latin Version of Ibn Al-Haytham's Kitab Al-Manazir* (Vol. 1). American Philosophical Society.

Smith, E. A., & Van Houten, R. (1996). A comparison of the characteristics of self-stimulatory behaviors in “normal” children and children with developmental delays. *Research in developmental disabilities*, *17*(4), 253-268.

Soucy, P., & Mineau, G. W. (2001). A simple KNN algorithm for text categorization. Proceedings 2001 IEEE international conference on data mining,

Srivastava, S., Gupta, M. R., & Frigyik, B. A. (2007). Bayesian quadratic discriminant analysis. *Journal of Machine Learning Research*, *8*(6).

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition,

Tantam, D. (2000). Adolescence and adulthood of individuals with Asperger syndrome. *Asperger syndrome*, 367-399.

Tuytelaars, T., Lampert, C. H., Blaschko, M. B., & Buntine, W. (2010). Unsupervised object discovery: A comparison. *International journal of computer vision*, *88*(2), 284-302.

Frank Rosenblatt: Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms, Brain Theory 245-248 (Springer Berlin Heidelberg 1986).

van Hateren, J. H., & Ruderman, D. L. (1998). Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, *265*(1412), 2315-2320.

Westeyn, T., Vadas, K., Bian, X., Starner, T., & Abowd, G. D. (2005). Recognizing mimicked autistic self-stimulatory behaviors using hmms. Ninth IEEE International Symposium on Wearable Computers (ISWC'05),

Wiskott, L., & Sejnowski, T. J. (2002). Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, *14*(4), 715-770.

Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.

Yamato, J., Ohya, J., & Ishii, K. (1992). Recognizing human action in time-sequential images using hidden Markov model. CVPR,

A review of recurrent neural networks: Lstm cells and network architectures, 31 MIT Press Journals 1235-1270 (2019).

Yue-Hei Ng, J., Hausknecht, M., Vijayanarasimhan, S., Vinyals, O., Monga, R., & Toderici, G. (2015). *Beyond short snippets: Deep networks for video classification*.

Zwaigenbaum, L., Bryson, S., & Garon, N. (2013). Early identification of autism spectrum disorders. *Behavioural brain research*, *251*, 133-146.