REPUBLIC OF TURKEY

YILDIZ TECHNICAL UNIVERSITY

DEPARTMENT OF COMPUTER ENGINEERING

ACTIVITY RETRIEVAL FROM VIDEOS

19011040 — Engin MEMİŞ

20011040 — Elif Sena YILMAZ

SENIOR PROJECT

Advisor

Prof. Dr. Mine Elif KARSLIGİL

June, 2024

ACKNOWLEDGEMENTS

We would like to thank our esteemed advisor Prof. Dr. Mine Elif KARSLIGIL, who spared her experience and knowledge from us throughout our project, dedicated her valuable time to us when we needed it, guided us throughout the project and thus enabled us to successfully complete our project.

Engin MEMİŞ
Elif Sena YILMAZ

TABLE OF CONTENTS

LIST OF SYMBOLS

%	Percentage Sign
*	Asterisk
∼	Tilde

LIST OF ABBREVIATIONS

ViViT	Video Vision Transformer
CNN	Convolutional Neural Network
CIFAR	Canadian Institute for Advanced Research
VTAB	Visual Task Adaptation Benchmark
ViT	Vision Transformer
TL	Turkish Lira
CPU	Central Process Unit
GB	Gigabyte
GPU	Graphics Processing Unit
GHz	Gigahertz
RAM	Random Access Memory
LSTM	Long Short-Term Memory
RNN	Recurrent Neural Networks
3D	3 Dimensional
RGB	Red, Green and Blue

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

ACTIVITY RETRIEVAL FROM VIDEOS

Engin MEMİŞ
Elif Sena YILMAZ

Department of Computer Engineering
Senior Project

Advisor: Prof. Dr. Mine Elif KARSLIGİL

With the developing technology, the size of the video data obtained is gradually increasing. This increase in video data enables the use of this data in areas such as machine learning. This video data is used in real life security cameras, automatic tagging of videos on platforms such as Youtube. However, since the video data used in these areas can be in a wide variety of forms in terms of content, it is a very difficult problem to create structures that can process this data. In this project, the Kinetics400 dataset, which is a very complex dataset, was used to simulate these difficult problems in real life. In order to process this dataset and compare their success, 3D Convolutional Neural Networks (3DCNN) and Video Vision Transformers (ViViT) are used.

The 3DCNN model is more traditional than the ViViT model and uses convolution layers to extract information from videos. The newer ViViT model is a deep learning model that incorporates "attention" mechanisms. The two models described in this paper are compared using the previously mentioned Kinetics400 dataset to show which model is more successful in this task.

Keywords: video data, 3D Convolutional Neural Networks, Video Vision Transformers, Kinetics400, attention

ÖZET

VİDEOLARDAN AKTİVİTE ÇIKARIMI

Engin MEMİŞ
Elif Sena YILMAZ

Bilgisayar Mühendisliği Bölümü
Bitirme Projesi

Danışman: Prof. Dr. Mine Elif KARSLIGİL

Gelişen teknoloji ile birlikte elde edilen video verilerinin boyutu giderek artmaktadır. Video verilerinin bu artışı makine öğrenmesi gibi alanlarda bu verilerin kullanılabilmesine olanak sağlamaktadır. Bu video verileri gerçek hayatta güvenlik kameralarında, Youtube gibi platformlarda videoların otomatik olarak etiketlenmesinde vb. alanlarda kullanılmaktadır. Ancak bu alanlarda kullanılan video verileri içerik olarak çok çeşitli şekillerde olabileceğinden dolayı bu verileri işleyebilecek yapıların oluşturulması oldukça zor bir problemdir. Yapılan bu projede de gerçek hayattaki bu zor problemlerin simüle edilebilmesi için oldukça karmaşık bir veriseti olan Kinetics400 veriseti kullanılmıştır. Bu verisetini işleyip başarılarını karşılaştırabilmek için ise 3 Boyutlu Evrişimsel Sinir Ağları (3DCNN) ve Video Vision Transformers (ViViT) kullanılmaktadır.

3DCNN modeli ViViT modeline göre daha geleneksel bir yöntem olup videolardan bilgileri çıkarmak için konvolüsyon katmanlarını kullanır. Daha yeni bir teknoloji olan ViViT modelleri ise içerisinde "dikkat" mekanizmalarını bulunduran bir derin öğrenme modelidir. Bu çalışmada anlatılan iki model yukarıda bahsedilen Kinetics400 veriseti kullanılarak karşılaştırılıp hangi modelin bu görevde daha başarılı olduğu gösterilmektedir.

Anahtar Kelimeler: video veri, 3 Boyutlu Evrişimsel Sinir Ağları, Video Vision Transformers, Kinetics400, dikkat

1 Introduction

In recent years, with the rise of video content on various digital platforms such as social media, it has been revealed that video is a very important data for information sharing. With this rise, the importance of systems that can extract information from videos has been emphasized once again.

As technology advances, new methods and models are emerging every day in the field of artificial intelligence. These new technologies in computer vision and deep learning, combined with the need to extract data from videos, have paved the way for the development of activity recognition. Systems developed in this field can analyze videos to determine what kind of action or activity people or objects are in. The development of this field is sure to help many industries.

Video processing plays a crucial role in activity recognition. This process has important steps such as preprocessing, feature extraction and analysis. In preprocessing, the video is divided into frames and these frames go through various processes such as normalization, noise reduction, resizing. Feature extraction is the extraction of spatial and temporal information such as textures, vertices and edges from these frames. Analysis is the final process of passing this information through an algorithm to predict the activity. Various machine learning and deep learning models are being developed in this field.

Cameras are ubiquitous in our lives and with them the proliferation of video data cannot be avoided. With the development of this technology, many things can be done automatically in most industries. It can help with many things such as tracking the movements of patients in hospitals, detecting suspicious behavior from security cameras, evaluating the performance of players in different sports in the entertainment industry. The motivation behind the development of this technology is the assistance that activity recognition systems offer to people. This capability makes it easy to automatically do some operations but also helps with awareness and resource utilization in certain situations.

In the past, traditional three-dimensional convolutional neural networks (3DCNN) have been used and proven to extract activity from videos. Transformers, which have gained a foothold in natural language processing, have also stepped into image processing. This is a new approach compared to traditional CNNs and it has started to show that it can compete with them. In this paper, we will use these two different approaches and show which one is preferable and in what ways.

2
Preliminary

In this section, various articles relevant to our study are reviewed and presented.

2.1 ViViT: A Video Vision Transformer

This paper presents a transformer-based model specifically developed for use in video classification. In order to overcome the problems caused by the long videos in the used datasets, the models are modified to separate the spatial and temporal dimensions of the given video frames. In order for these 'Transformers' based models to be successful, very large datasets are usually required. In small datasets, they cannot increase their success very quickly. In order to solve the problem in these small datasets, pre-trained models are used. In this paper, datasets such as Moments in Time and Epic Kitchens and Kinetics 400 are used to test these models [1].

2.2 Video Transformers: A Survey

This paper presents research on how Transformer-based models can handle very long data in video datasets. While the work up to this paper has generally attempted to show improvements in Transformer-based architectures, none of it has examined the details of processing long videos. In this paper, we focus on video classification using datasets such as Kinetics400 to demonstrate its ability to process long videos. While doing this research, it is investigated how long videos can be processed more efficiently, how long video data can be shortened without losing its main skeleton and how it can be processed successfully within the model. As a result of this research, even if lower computational costs are used in order to process long videos more efficiently, the success of 3DCNNs, which is the traditional method used so far in video classification, is shown [2].

2.3 An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale

Although the ‘Transformers’ architectures used so far have found a place for themselves in the field of natural language processing, their use in image processing is still limited. In the field of image processing, the Attention mechanism in the Transformers architecture is either used with convolution layers in CNN models, which are frequently used in the field of image processing, or some structures are removed and these Attention mechanisms are added. In this study, it is argued that this dependence on CNNs in image processing is unnecessary and it is tried to show that the use of Transformers structures directly on image data is more successful. By first training this model with large data sets and then transferring it to smaller structures, it is shown that Transformers architectures require less resources and give more successful results in the field of image processing [3].

2.4 Human Activity Classification Using the 3DCNN Architecture

Nowadays, unlike image datasets, the interest in video datasets is increasing due to the increasing size of video datasets. This increase in interest also leads to an increase in studies for processing video data. In this paper, 3DCNN architecture is proposed instead of common architectures such as Convolutional Long Short-Term Memory (ConvLSTM) developed to identify activities in video data. In order to test this proposed architecture, datasets such as UCF101, UCF Youtube Action, UCF50 were used. As a result of these experiments, 3DCNN architecture was found to be more successful than previously used neural networks [4].

2.5 A Review of Video Action Recognition Based on 3D Convolution

Unlike the above articles, this study is not intended to solve any problem, but to show what solutions previous studies have provided and what problems they have encountered through the problems identified so far for the researches to be carried out by new researchers in the field of video action recognition. After showing these problems, it is presented which popular datasets and methods are used in the studies and analyses of these methods and datasets [5].

3
Feasibility

This chapter describes the feasibility study in 4 different sections: technical feasibility, economic feasibility, legal feasibility and labor and time feasibility.

3.1 Technical Feasibility

Technical feasibility is described under two subheadings: software feasibility and hardware feasibility.

3.1.1 Software Feasibility

This section describes the development environment and programming languages used in the project.

3.1.1.1 Development Environment

Google Colab:
Colaboratory ("Colab" for short) is a product offered by Google Research [6]. Colab was chosen because it offers free or affordable GPU access. Colab's connection to Drive also facilitates access to the dataset.

3.1.1.2 Programming Language

In this project, it was preferred to use the Python language. This language is frequently used in artificial intelligence and image processing with many free libraries to use.

3.1.2 Hardware Feasibility

This section provides minimum and recommended system requirements. Since the dataset consists of videos, a large disk space is required.

Table 3.1 Minimum System Requirements

RAM	8 GB
Required Disk Space	20 GB
CPU	2.4 GHz
GPU	4 GB

Table 3.2 Recommended System Requirements

RAM	32 GB
Required Disk Space	80 GB
CPU	2.6 GHz
GPU	16 GB

3.2 Economic Feasibility

In this section, the price of the computer needed for the project and the salaries of the staff developing the project are given.

Table 3.3 Required Hardware Cost

Product	Price
1x Recommended Computer	18000 TL
1x Recommended GPU	40000 TL

Table 3.4 Personnel Expense Table

Personnel	Person	Day	Daily Salary	Total
System Analyst	2	5	950 TL	9500 TL
Software Developer	2	27	1100 TL	59400 TL
Software Tester	1	5	750 TL	3750 TL
			Total Expense	72650 TL

3.3 Legal Feasibility

Since the data sets used during the project are publicly available, no patent and trademark rights have been violated. Apart from that, open source and free products are used. When the necessary laws are examined, there are no legal obstacles to the project. Therefore, the project is legal.

3.4 Labor and Time Feasibility

In this section, the task and time distribution of the project is shown in Figure 3.1.

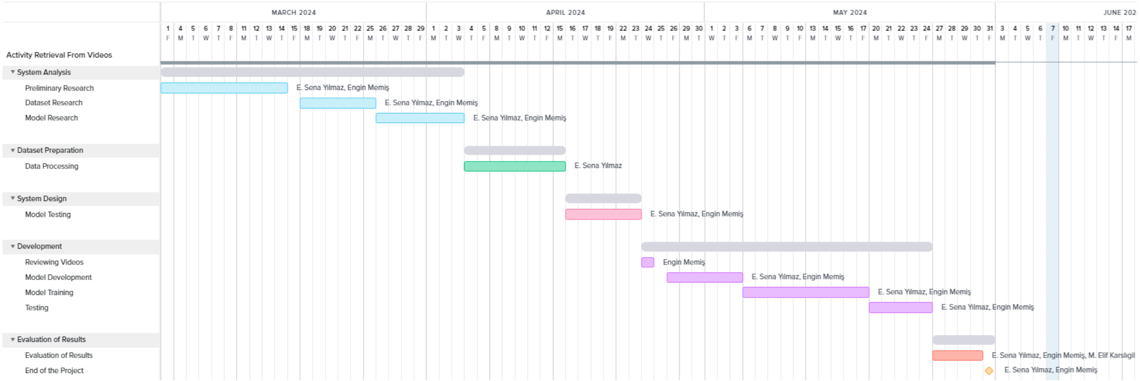


Figure 3.1 Gantt Chart

4
System Analysis

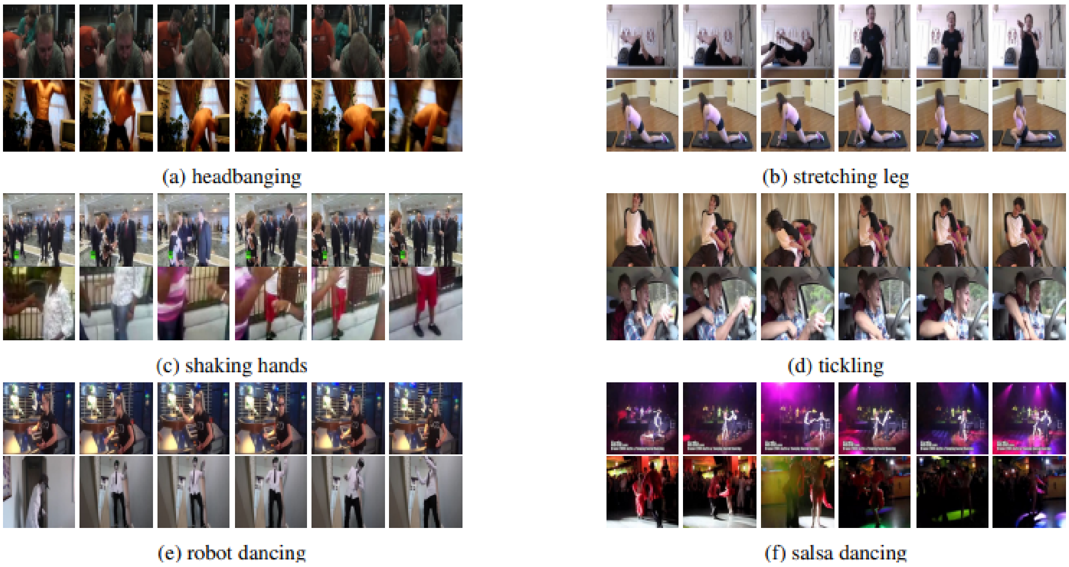
4.1 Project Objective

The aim of this study is to investigate how the models and architectures used for activity recognition from videos, which are frequently studied today, work, and to compare these models by testing them with commonly used datasets. Among the models investigated, one traditional convolutional neural network (CNN) which is already widely used in the industry and the more innovative Video Vision Transformer (ViViT) models will be compared.

4.2 Requirement Analysis

OpenCV, NumPy, PyTorch, Transformers and a few simple libraries were used in the development of the code written in the project. Anaconda Navigator, Visual Studio Code and Google Colab were chosen as development environments.

In this project, the Kinetics400 dataset was selected to be used for testing the models. Kinetics400 is considered to be one of the benchmark datasets in the action recognition problem. This dataset contains 400 different classes of human activity taken from YouTube videos and each class contains approximately 800 videos. In this dataset, there are activities such as daily activities or sports. Since this dataset consists of videos taken by people themselves and no model training was considered, it was thought that it would better simulate real life in training as it contains various camera angles, shaky images, videos that are either too long or too short.



(a) headbanging

(b) stretching leg

(c) shaking hands

(d) tickling

(e) robot dancing

(f) salsa dancing

Figure 4.1 Example classes from the Kinetics dataset [7]

4.3 Use Case Scenario

In this section, the use case of the project is shown in Figure 8.5.

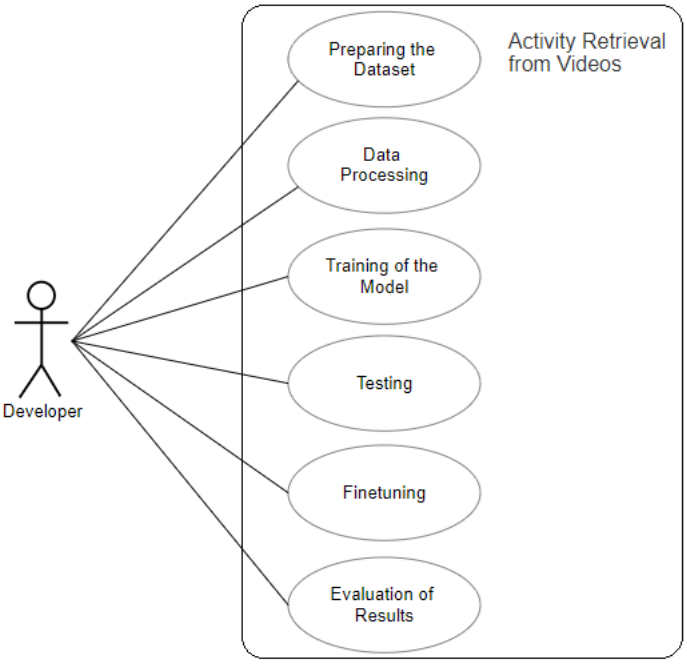


Figure 4.2 Use Case Diagram

5
System Design

This section describes the software design of the project.

5.1 Software Design

5.1.1 Data Preprocessing

In the models used, videos cannot be trained as they are. Therefore, the videos in the databases must go through certain pre-processing steps before they are given to the models for training. Firstly, since the data in the videos can be very long and all of them can be of different sizes, frames are selected from the videos according to the total number of frames selected in accordance with the model architecture. Then, these frames are subjected to certain normalisation processes and the pixel values of the frames in the video are finally brought to the range of zero to one. Thanks to these pre-processing steps, the videos in the databases are made trainable in the models used.

5.1.2 Deep Learning Architectures for Video Classification

With the development of technology in video classification, a wide variety of architectural models have been used. Firstly, Convolution Neural Networks (CNN) working on images were used in video classification problem. However, CNN models were not very successful in videos due to feature extraction only on a single image. Subsequently, Long Short Memory (LSTM) and Recurrent Neural Networks (RNN) have been used for video classification. The reason for using these models is their ability to perform feature extraction on sequential images. In this way, these models have become more successful in video classification than CNNs.

Today, although the models mentioned above are still used, Transformers models, which are more innovative methods, which include the 'Attention' mechanism, give very successful results in video classification.

5.1.2.1 Convolutional Neural Networks (CNNs)

CNN models, which have been used since the beginning of studies in the field of image processing, have been tested on video data before the development of new technologies due to the increase in video data. However, although 2DCNNs developed on 2D images were successful in this field, they could not be successful in this problem due to the fact that videos are sequential images.

3DCNN models developed specifically for the classification of videos with sequential images have become more successful in this problem because they can extract more detailed information that 2DCNN models cannot extract [8].

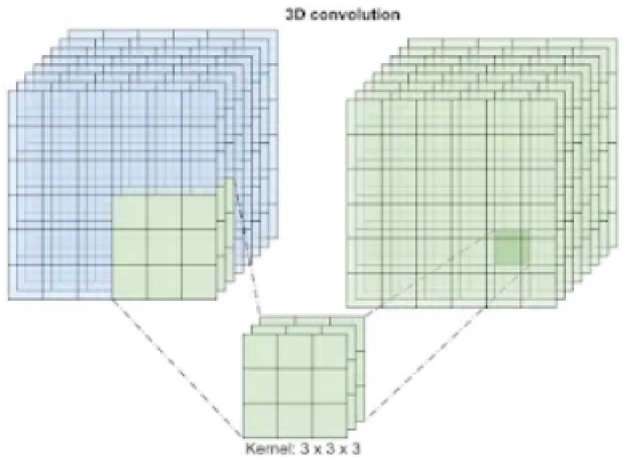


Figure 5.1 3D CNN Structure [8]

The architecture of the 3DCNN model, which is developed to be used for action extraction from videos, which is the subject of the project, and has 3 convolution layers, is as follows.

The preprocessed video data in RGB colour space are passed through 32 3x3x3 filters in the first convolution layer and features are extracted. These extracted features pass through the pooling layer of 1x2x2 size.

The data from the first convolution layer passes through 64 3x3x3 filters in the 2nd convolution layer and more detailed information is extracted compared to the first convolution layer. Again, this data passes through a 2x2x2 pooling layer.

The data coming to the last convolution layer passes through 128 3x3x3 filters and then through a 2x2x2 pooling layer in order to extract more detailed features that the previous convolution layers could not extract.

The data from the convolution layers are flattened and passed through an artificial neural network with 128 and 512 neurons. In the last layer of this neural network, the data is reduced to the number of classes in the used dataset, resulting in the predicted class of the video.

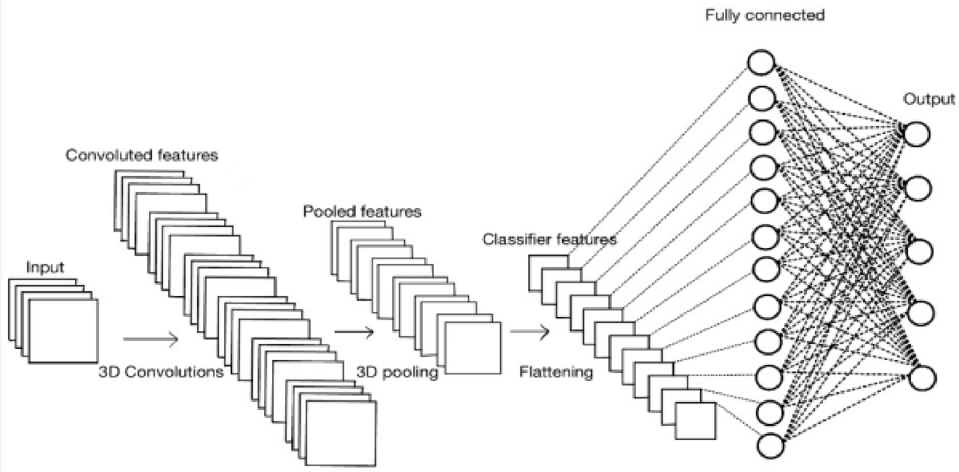


Figure 5.2 3D CNN Structure [9]

5.1.2.2 Video Vision Transformers (ViViT)

Unlike traditional models such as CNN, RNN and LSTM, which have been used since the early days of image processing, ViViT, which is a more innovative structure with the developing technology, is a Transformers with an ‘Attention’ mechanism.

ViViT models work on those structures by dividing them into smaller structures while processing videos. In this way, it can reach much more detailed features. While performing operations on the smaller structures it divides, it decodes the relationships within the videos by using the Attention mechanisms inside. At the end of the structure, there is a flatten layer and a softmax layer. These layers are used to determine which class the video belongs to from the features of the videos [10].

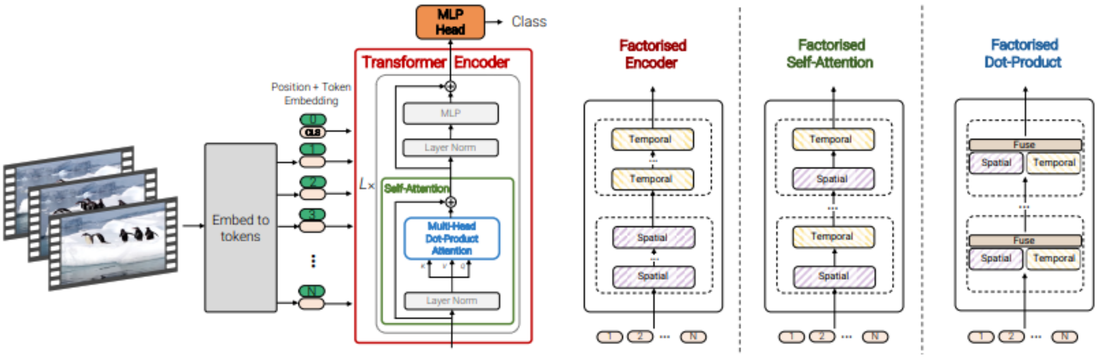


Figure 5.3 ViViT Architecture [1]

The literature review showed that the parameters of the most commonly used ViViT models are as shown in the Table 5.1. The models with an asterisk (*) next to them in the table are the parameters that Google uses for the ViViT models.

Table 5.1 Previously Used ViViT Model Parameters [11]

Model Name	Hidden Size	MLP Dimension	Number of Attention Heads	Number of Encoder Layers	Tubelet Spatial Size
Tiny	192	768	3	12	16
Small	384	1536	6	12	16
Base*	768	3072	12	12	16
Large*	1024	4096	16	24	16
Huge	1280	5120	16	32	14

6 Application

As will be seen in the following sections, we conducted our experiments using the 3DCNN and ViViT models. In this section, we will give the parameters of these two models, describe how the dataset was used and how we preprocessed the data to feed it to the models. Also there is the results of the first test with these features. The models whose results are presented in this section are the ViViT's Tiny and Small models described in the previous section.

6.1 Dataset

Detailed information about the Kinetics400 dataset was given in the requirements analysis section. Here we will talk about how we adapted this dataset to our own experiments.

In our circumstances it was not possible to process the entire dataset, so in our initial tests we decided to reduce the number of classes from 400 to 100. We also reduced the number of videos in each class and decided to have 100 videos in the train set and 30 videos in the validation and test set for each class.

After doing some tests with this dataset this approach did not yield good results and we decided to change the way we split the dataset again. We chose 20 classes instead of 100, and from each of these classes we selected about 400 videos for training. There are approximately 100 videos in the test set and about 50 videos in the validation set for each class in this new data set.

6.1.1 Preparing the Data

The videos were pre-processed before being given to the model. First, starting from the middle of the video, frames were removed from the video using a stride value of two, resulting in 64 frames from each video. These frames were then resized to 224x224. The normalization process is carried out to rescale pixel values within the range of 0 to 1 before importing these frames into the model. As a result of these processes, these frames are ready to be given to the model.

6.2 Training Parameters

Some of the key parameters used to train these models are listed below.

- Batch Size: 4
- Number of Epochs: 5/10/15
- Learning Rate: 0.00001
- Optimizer: Adam

6.3 Loss Function

Cross Entropy, which is widely used in this field, was used as the loss function in training for both models.

6.4 Model Testing

To expedite the evaluation of parameter suitability, only 5% and 10% subsets of the original Kinetics dataset were utilized to test the training outcomes for efficiency. How we created these subsets was explained earlier.

Table 6.1 Parameter Testing Results

Model	Dataset	Accuracy
Tiny	Kinetics 400 (5%)	13.4%
Tiny	Kinetics 400 (5%)	13.6%
Small	Kinetics 400 (10%)	14.8%
Small	Kinetics 400 (10%)	15.2%

Due to the low success results, we changed our perspective on this problem and reduced the number of classes from 100 to 20 and increased the number of videos in the following tests. While searching for another suitable ViViT model to use, we performed our first experiments with the 3DCNN model.

7
Experimental Results

In this section we will present the results of the tests we conducted during the development of our project and explain the reasons for some of the decisions that led to the final model. Finally, the results of the final version of both models will be presented and analyzed.

7.1 Tests For Regularization and Video Amount Using 10 Classes

All the experiments in this section were performed on the 3DCNN model and tested on ViViT with the finalized features. Below are the measurements we made using only 10 classes of the dataset to answer questions such as whether there should be regularization in the final model or how many videos should be used. Below this table there is another table that shows the training time for different numbers of videos.

Table 7.1 Test Results For 10 Classes

	No Regularization		**With Regularization**	
	5 Epoch	**10 Epoch**	**5 Epoch**	**10 Epoch**
100 Videos	59.55%	60.91%	61.40%	59.43%
200 Videos	64.48%	67.57%	63.00%	71.52%
All Videos (∼370)	68.92%	69.22%	69.91%	71.76%

Table 7.2 Model Training Times for 10 Classes

	10 Epoch
100 Videos	95 min
200 Videos	135 min
All Videos (∼370)	210 min

Examining these results, it was decided that the measurements should continue with the model with regularization. Since it is thought that the number of videos in each class may increase the success rate when the number of classes is increased, the number of videos to be used was not decided and it was thought that the experiments should continue. The confusion matrix of the most successful model is shown below.

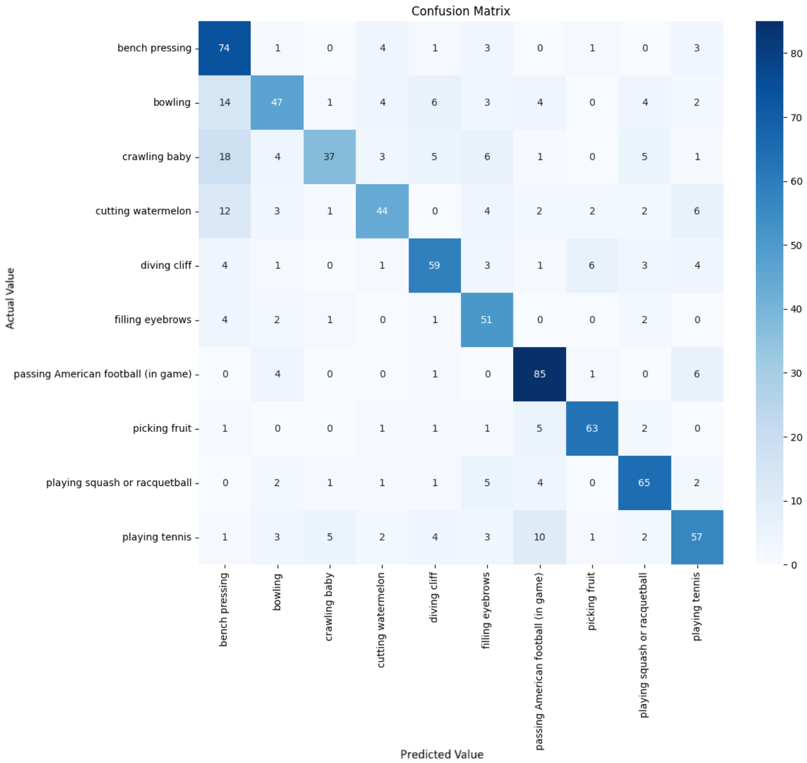


Figure 7.1 3DCNN 10 Classes All Videos 10 Epoch With Reg. Confusion Matrix

When the confusion matrix is analyzed, it is seen that the most correctly predicted class is “passing American football (in game)”, while “crawling baby” is the most confused class and is mostly confused with “bench pressing”.

7.2 Tests Using 20 Classes

In order to observe how the success rate will change when the number of classes is increased to 20 and how the number of videos in each class will contribute to this success, tests were conducted. The results of these tests and how long each training lasted are given in the tables below.

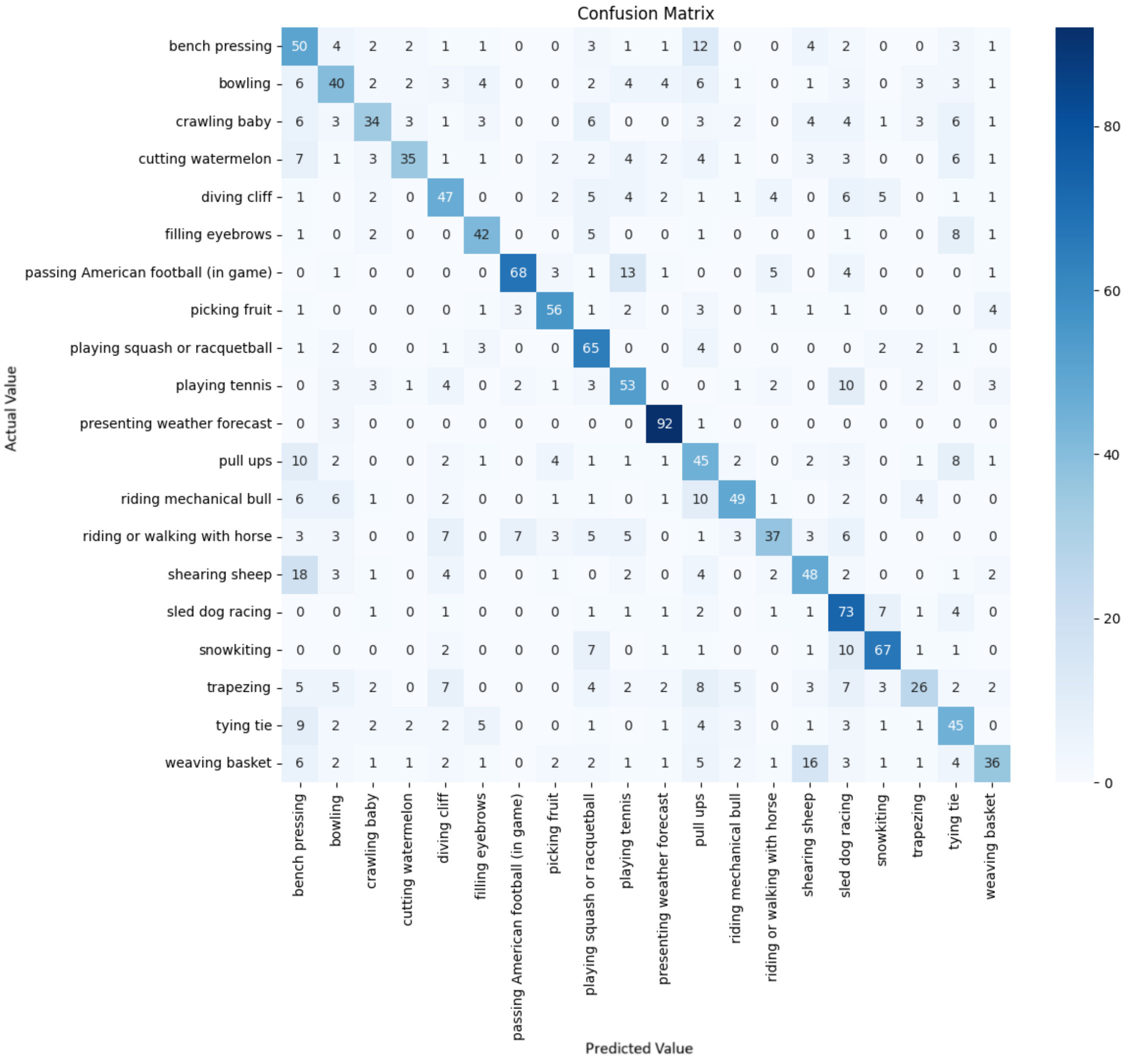
Table 7.3 Test Results For 20 Classes

	5 Epoch	**10 Epoch**
100 Videos	47.09%	49.70%
200 Videos	51.78%	52.42%
All Videos (∼370)	58.72%	59.89%

Table 7.4 Model Training Times for 20 Classes

	10 Epoch
100 Videos	156 min
200 Videos	257 min
All Videos (∼370)	569 min

Based on the results of these tests, it was decided that the final model should be trained with all the videos, since the success rate increased as the number of videos used increased. Also, since there was no significant drop in success despite doubling the number of classes, it was decided to keep the number of classes at 20 for diversity. The confusion matrix of the 3DCNN model with all videos is given below.



	bench pressing	bowling	crawling baby	cutting watermelon	diving cliff	filling eyebrows	passing American football (in game)	picking fruit	playing squash or racquetball	playing tennis	presenting weather forecast	pull ups	riding mechanical bull	riding or walking with horse	shearing sheep	sled dog racing	snowkiting	trapezing	tying tie	weaving basket
bench pressing	50	4	2	2	1	1	0	0	3	1	1	12	0	0	4	2	0	0	3	1
bowling	6	40	2	2	3	4	0	0	2	4	4	6	1	0	1	3	0	3	3	1
crawling baby	6	3	34	3	1	3	0	0	6	0	0	3	2	0	4	4	1	3	6	1
cutting watermelon	7	1	3	35	1	1	0	2	2	4	2	4	1	0	3	3	0	0	6	1
diving cliff	1	0	2	0	47	0	0	2	5	4	2	1	1	4	0	6	5	0	1	1
filling eyebrows	1	0	2	0	0	42	0	0	5	0	0	1	0	0	0	1	0	0	8	1
passing American football (in game)	0	1	0	0	0	0	68	3	1	13	1	0	0	5	0	4	0	0	0	1
picking fruit	1	0	0	0	0	1	3	56	1	2	0	3	0	1	1	1	0	0	0	4
playing squash or racquetball	1	2	0	0	1	3	0	0	65	0	0	4	0	0	0	0	2	2	1	0
playing tennis	0	3	3	1	4	0	2	1	3	53	0	0	1	2	0	10	0	2	0	3
presenting weather forecast	0	3	0	0	0	0	0	0	0	0	92	1	0	0	0	0	0	0	0	0
pull ups	10	2	0	0	2	1	0	4	1	1	1	45	2	0	2	3	0	1	8	1
riding mechanical bull	6	6	1	0	2	0	0	1	1	0	1	10	49	1	0	2	0	4	0	0
riding or walking with horse	3	3	0	0	7	0	7	3	5	5	0	1	3	37	3	6	0	0	0	0
shearing sheep	18	3	1	0	4	0	0	1	0	2	0	4	0	2	48	2	0	0	1	2
sled dog racing	0	0	1	0	1	0	0	0	1	1	1	2	0	1	1	73	7	1	4	0
snowkiting	0	0	0	0	2	0	0	0	7	0	1	1	0	0	1	10	67	1	1	0
trapezing	5	5	2	0	7	0	0	0	4	2	2	8	5	0	3	7	3	26	2	2
tying tie	9	2	2	2	2	5	0	0	1	0	1	4	3	0	1	3	1	1	45	0
weaving basket	6	2	1	1	2	1	0	2	2	1	1	5	2	1	16	3	1	1	4	36

Figure 7.2 3DCNN 20 Classes All Videos 10 Epoch Confusion Matrix

The matrix shows that the most successful classes are “presenting weather forecast”, “sled dog racing” and “snowkiting”. Although the overall success of most of the classes decreased slightly with the increasing number of classes, it was still mostly predicted correctly, and one of the most confused classes was “shearing sheep”.

7.3 Test with the ViViT Model

Finally, we tested ViViT with the features we decided on after all the tests, using 20 classes and all videos. For ViViT, we used the parameters of the Base model that Google uses in its own work.

The success rate of the test was 54.98%. The training took 309 minutes. The confusion matrix for this model is shown below.

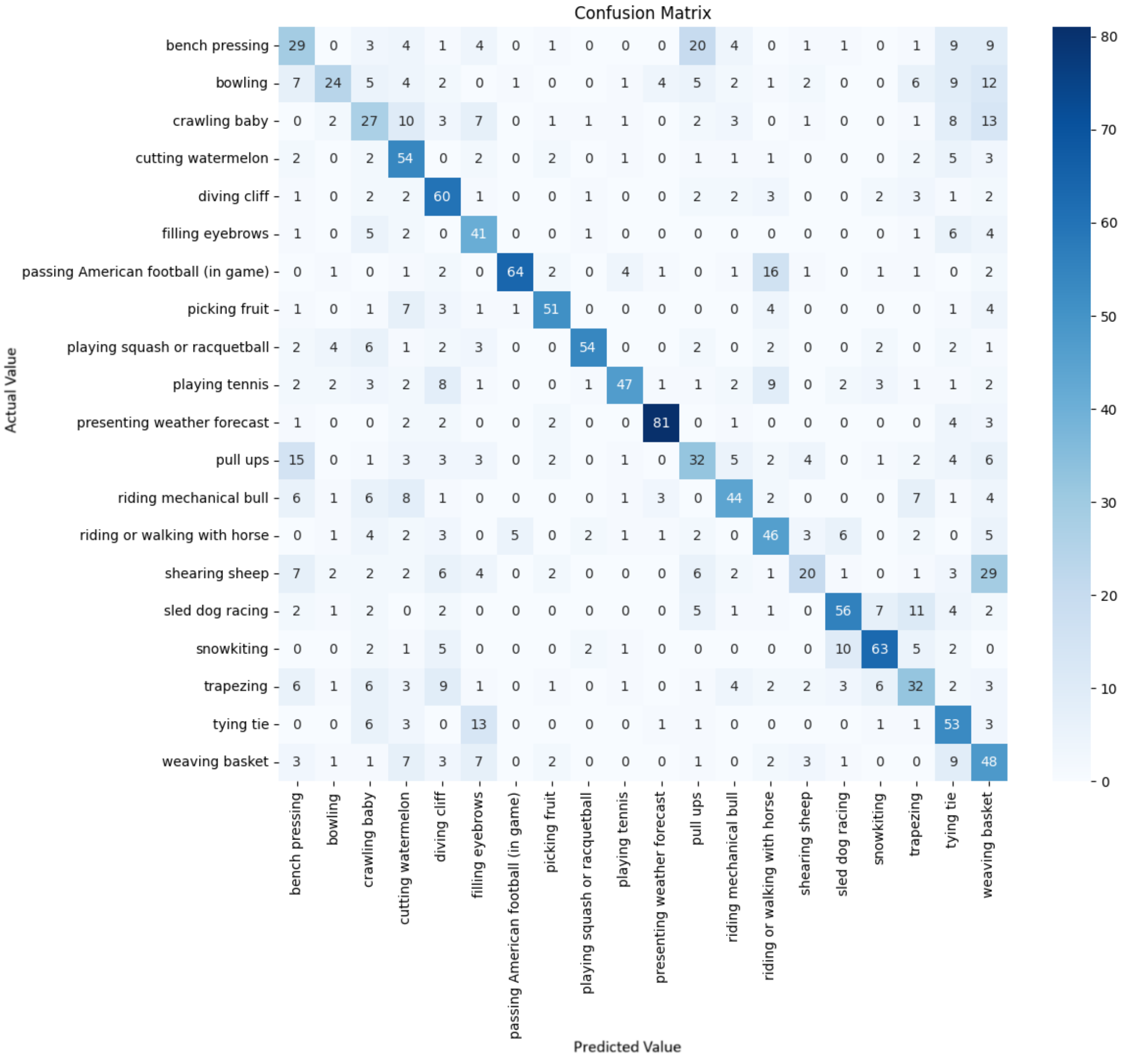


Figure 7.3 ViViT 20 Classes All Videos 10 Epoch Confusion Matrix

8
Performance Analysis

In this section, we will compare and comment on the performance of the final 3DCNN and ViViT models we trained in various aspects.

8.1 Accuracy Comparison

The following table shows the accuracy of the two models over 10 epochs.

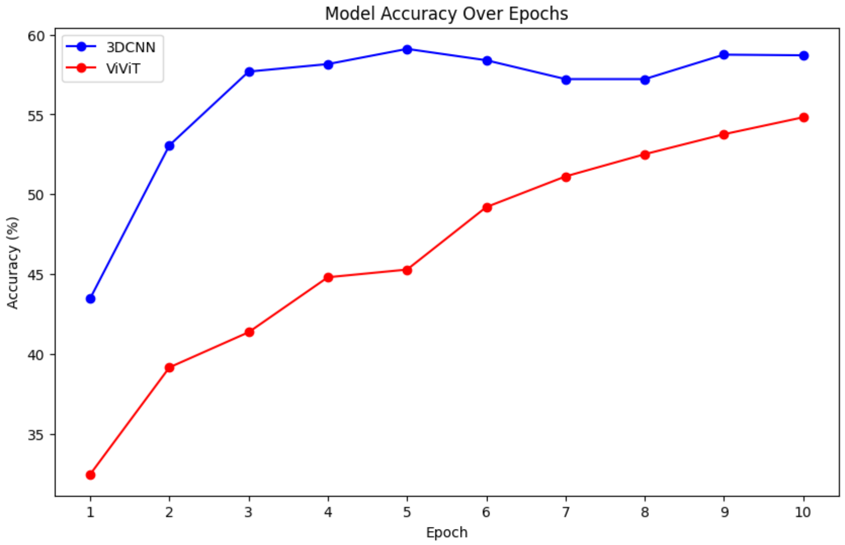


Figure 8.1 Accuracy Comparison Between ViViT and 3DCNN

When the graph showing the success rates according to the number of epochs given above is analysed, it is seen that although the success of the ViViT model starts at a lower level than the success of the 3DCNN model, it is trained with a steady increase for 10 epochs. On the other hand, when we look at the 3DCNN model, although its success starts at a higher level than the success of the ViViT model, it is seen that its success is very close to its best success at the end of the 4th epoch and that it is in a state of continuous increase and decrease between certain success rates between the 4th and 10th epochs.

In the table below, the values for precision, recall and F1 score were extracted for both models and presented side by side.

Table 8.1 Classification Report for 3DCNN and ViViT

	precision		**recall**		**f1-score**	
	3DCNN	**ViViT**	**3DCNN**	**ViViT**	**3DCNN**	**ViViT**
bench pressing	0.38	0.34	0.57	0.33	0.46	0.34
bowling	0.5	0.6	0.47	0.28	0.48	0.38
crawling baby	0.61	0.32	0.42	0.34	0.5	0.33
cutting watermelon	0.76	0.46	0.46	0.71	0.57	0.56
diving cliff	0.54	0.52	0.57	0.73	0.56	0.61
filling eyebrows	0.68	0.47	0.69	0.67	0.68	0.55
passing American football	0.85	0.9	0.7	0.66	0.77	0.76
picking fruit	0.75	0.77	0.76	0.69	0.75	0.73
playing squash or racquetball	0.57	0.87	0.8	0.67	0.66	0.76
playing tennis	0.57	0.8	0.6	0.53	0.59	0.64
presenting weather forecast	0.84	0.88	0.96	0.84	0.89	0.86
pull ups	0.39	0.4	0.54	0.38	0.45	0.39
riding mechanical bull	0.7	0.61	0.58	0.52	0.64	0.56
riding or walking with horse	0.69	0.5	0.45	0.55	0.54	0.53
shearing sheep	0.55	0.54	0.55	0.23	0.55	0.32
sled dog racing	0.51	0.7	0.78	0.6	0.62	0.64
snowkiting	0.77	0.73	0.74	0.69	0.75	0.71
trapezing	0.58	0.42	0.31	0.39	0.41	0.4
tying tie	0.48	0.43	0.55	0.65	0.51	0.51
weaving basket	0.65	0.31	0.41	0.55	0.5	0.4

The tested ViViT model has higher precision values than the 3DCNN model in the activity classes “playing squash or racquetball”, “presenting weather forecast” and “playing tennis”. On the other hand, 3DCNN model has a higher precision value in “crawling baby” and “cutting watermelon” activity classes.

Looking at the recall scores of the 3DCNN model, it is seen that it is much more successful than the ViViT model in the ‘presenting a weather forecast’ and ‘sled dog racing’ classes. Although ViViT model is less successful than 3DCNN model in other classes, it is still competitive.

While the ViViT model performs better in the 4 classes marked in the table above, the 3DCNN model performs better in all other classes.

The 3DCNN model generally performs better on all precision, recall and F1 scores, indicating that it has an overall superior classification performance compared to the ViViT model.

8.2 Comparison of Efficiency and Resource Utilization

The number of trainable parameters, GPU usage and training times for 3DCNN and ViViT models are as shown in the table below.

Table 8.2 Training and Resource Utilization for ViViT and 3DCNN

	ViViT	**3DCNN**
Trainable Parameters	52,585,748	822,373,652
GPU Usage	4 GB	21.8 GB
Training Time	309 min	569 min

When the graph above, which shows the time and resource utilisation of both models, is examined, it is seen that the 3DCNN model is a much more costly model than the ViViT model in terms of trainable parameters and GPU memory usage. This difference between the 2 models has also created a significant difference in training times.

8.3 Sample Outputs of the Models

This section shows the outputs of the models for various classes and situations.

8.3.1 Example of Successful Prediction by Both Models

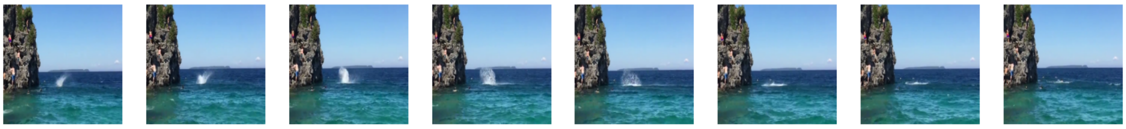


Figure 8.2 Diving Cliff Sample Video Frames

Table 8.3 Model Predictions for Diving Cliff

3DCNN	
diving cliff	99.93%
playing tennis	0.06%
pull ups	∼0.00%
trapezing	∼0.00%
passing American football	∼0.00%

ViViT	
diving cliff	99.16%
playing tennis	0.4%
presenting weather forecast	0.2%
trapezing	0.07%
pull ups	0.03%

8.3.2 Example of Successful Prediction by Only 3DCNN

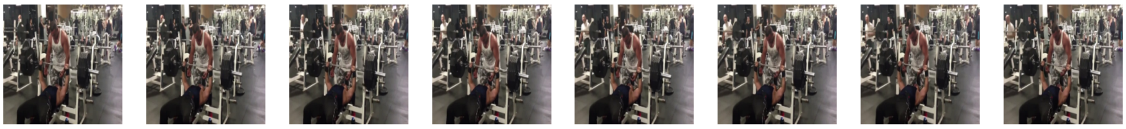


Figure 8.3 Bench Pressing Sample Video Frames

Table 8.4 Model Predictions for Bench Pressing

3DCNN	
bench pressing	99.77%
pull ups	0.13%
weaving basket	0.05%
shearing sheep	0.02%
riding mechanical bull	0.01%

ViViT	
pull ups	64.79%
bench pressing	29.64%
trapezing	2.28%
weaving basket	1.77%
shearing sheep	0.85%

8.3.3 Example of Successful Prediction by Only ViViT

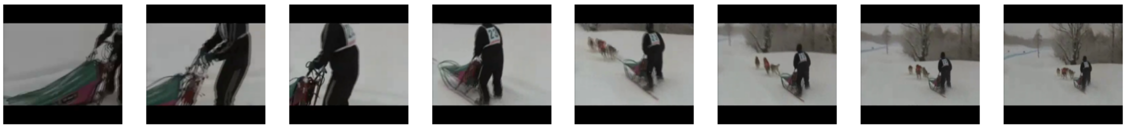


Figure 8.4 Sled Dog Racing Sample Video Frames

Table 8.5 Model Predictions for Sled Dog Racing

3DCNN	
playing squash or racquetball	87.04%
trapezing	6.38%
riding mechanical bull	1.74%
playing tennis	0.99%
passing American football	0.77%

ViViT	
sled dog racing	82.5%
riding or walking with horse	4.27%
trapezing	3.9%
snowkiting	3.19%
crawling baby	1.85%

8.3.4 Example of Failure Prediction by Both Models

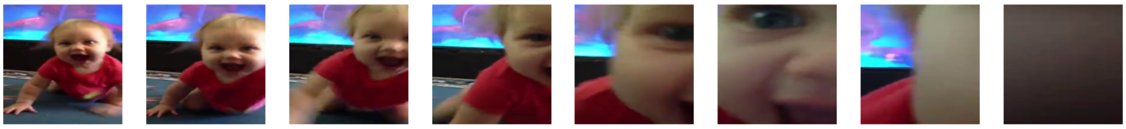


Figure 8.5 Crawling Baby Sample Video Frames

Table 8.6 Model Predictions for Crawling Baby

3DCNN	
cutting watermelon	56.20%
crawling baby	17.51%
riding mechanical bull	13.77%
bowling	2.81%
diving cliff	2.71%

ViViT	
bowling	32.05%
crawling baby	31.43%
riding mechanical bull	17.22%
weaving basket	12.36%
riding or walking with horse	2.82%

9
Conclusion

In this study, 3DCNN and ViViT models, which are widely used for the task of action extraction from video, are analysed and as a result of the tests, which model is more successful in which situations for action extraction from video is compared.

Due to limited resources, the two models mentioned above were not tested with the 400 classes in the Kinetics400 dataset, but with smaller datasets created from this dataset. As a result of testing these datasets, the best results were obtained with 20 classes and 400 videos in each class. The 3DCNN model has a success rate of 59.89% after 10 epochs on this dataset, while the ViViT model has a success rate of 54.98% after 10 epochs. Considering these success rates, it is seen that the 3DCNN model is more successful than the ViViT model in our study.

When the resources required by these 2 models during the training process are compared, it is seen that the ViViT model is more efficient than the 3DCNN model in terms of time and memory usage. It is thought that ViViT model can be chosen instead of 3DCNN model considering the resources since the difference in success rates is not too much when resources are low.

It is thought that the ViViT model may be more successful than the 3DCNN model after more training, since the 3DCNN model increases and decreases in a certain range towards the end of the training and the ViViT model increases regularly throughout the training. However, this idea could not be tested due to limited resources. Therefore, this theory can be tested in future studies to see whether it is true or false.

As a result, it was observed that the 3DCNN model was generally more successful than the ViViT model in the studies conducted in this project. However, in some classes, the ViViT model is also more successful than the 3DCNN model. It seems possible to choose between 3DCNN or ViViT models according to both limited resource use and the area where the models will be used in real life.

References

[1] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," 2021. arXiv: 2103.15691 [cs.CV].

[2] J. Selva, A. S. Johansen, S. Escalera, K. Nasrollahi, T. B. Moeslund, and A. Clapes, "Video transformers: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2023, ISSN: 1939-3539. DOI: 10.1109/tpami.2023.3243465. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2023.3243465.

[3] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021. arXiv: 2010.11929 [cs.CV].

[4] R. Vrskova, R. Hudec, P. Kamencay, and P. Sykora, "Human activity classification using the 3dcnn architecture," *Applied Sciences*, vol. 12, no. 2, 2022, ISSN: 2076-3417. DOI: 10.3390/app12020931. [Online]. Available: https://www.mdpi.com/2076-3417/12/2/931.

[5] X. Huang and Z. Cai, "A review of video action recognition based on 3d convolution," *Computers and Electrical Engineering*, vol. 108, p. 108 713, 2023, ISSN: 0045-7906. DOI: https://doi.org/10.1016/j.compeleceng.2023.108713. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790623001374.

[6] Google. "Colaboratory nedir?" (no date), [Online]. Available: https://research.google.com/colaboratory/intl/tr/faq.html (visited on 04/29/2023).

[7] W. Kay *et al.*, "The kinetics human action video dataset," 2017. arXiv: 1705.06950 [cs.CV].

[8] A. Helwan. "Video classification using cnn and transformer." (2023), [Online]. Available: https://abdulkaderhelwan.medium.com/video-classification-using-cnn-and-transformer-798b84c345bd (visited on 12/03/2023).

[9] C. Wang, "A review on 3d convolutional neural network," in *2023 IEEE 3rd International Conference on Power, Electronics and Computer Applications (ICPECA)*, 2023, pp. 1204–1208. DOI: 10.1109/ICPECA56706.2023.10075760.

[10] SternDS. "Video transformer(vit): A deep learning model for video processing." (2023), [Online]. Available: https://medium.com/@nadav6stern/video-transformer-vit-a-deep-learning-model-for-video-processing-442268c8c3b4 (visited on 12/03/2023).

[11] S. Yan *et al.* "Multiview transformers for video recognition." arXiv: 2201.04288 [cs.CV]. (2022).

Curriculum Vitae

FIRST MEMBER

Name-Surname: Engin MEMİŞ
Birthdate and Place of Birth: 30.03.2000, İstanbul
E-mail: engin.memis@std.yildiz.edu.tr
Phone: 0534 244 87 84
Practical Training: YTÜ Olasılıksal Robotik Araştırma Grubu
HAVELSAN A.Ş. Komuta Kontrol Savunma Teknolojileri

SECOND MEMBER

Name-Surname: Elif Sena YILMAZ
Birthdate and Place of Birth: 21.05.2001, Ankara
E-mail: sena.yilmaz4@std.yildiz.edu.tr
Phone: 0551 251 35 54
Practical Training: HAVELSAN A.Ş. Komuta Kontrol Savunma Teknolojileri

Project System Informations

System and Software: Windows Operating System, Python, Google Colab
Required RAM: 32GB
Required Disk: 256GB