Analysing Hand Gestures in Real-World Interactions.

Employing gesture coding schemes and machine learning to predict physical features of hand gestures in video footage from a crowded social setting.

Franciszek Latała[1]
Responsible Professor: Hayley Hung[1]
Supervisors: Ivan Kondyurinr[1]**, Zonghuan Li**[1]
[1]EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: Franciszek Latala
Final project course: CSE3000 Research Project
Thesis committee: Hayley Hung, Ivan Kondyurinr, Mark Neerincx

Abstract

Researching hand gestures in real-world social interactions requires very careful analysis. While gesture coding schemes were created with that purpose in mind, they are not widely utilised in research. Moreover, studies on gesture classification rarely focus on the physical nature of movements involved in gesturing, despite the fact that being able to quantify the motion could reveal useful patterns and correlations. To address those points, this research proposes the following approach: using machine learning models to automatically classify physical features of hand gestures, according to a coding scheme. Two such classifiers were created, for the left and right hand respectively. Overall, the results are quite promising - despite a small and imbalanced training set and complex features both models achieved an accuracy of roughly 60%. Moreover, the results indicate that by avoiding some of the simplifications that this research makes, and by using more balanced training data, the accuracy could be significantly increased. This is concrete evidence that machine learning models can indeed be used to classify the physical aspects of hand gestures, as defined by a coding scheme, in social interactions in the wild.

1 Introduction

McNeil [1] demonstrates that hand gestures can carry semantic meaning that could help us understand what is being communicated without knowing exactly what was said. Furthermore, gestures create an integrated system with speech and are integral to the process of communication - they provide additional depth, structure interactions, and can also have meaning of their own [2]. Moreover, as Goldin-Meadow states "Our hands are with us at all times and thus provide researchers and learners with an ever-present tool for understanding how we talk and think" [3]. These are only few of the intriguing properties of hand gestures. Nevertheless, it's clear that better understanding gestures could reveal a lot about social interactions and communication. However, as Abner [4] concludes, researching hand gestures requires very careful analysis of their rich and multi-faceted structure.

Gesture coding schemes are tools developed for that exact purpose. They are theoretical frameworks created to accurately annotate, or "code", hand gestures. This facilitates precise analysis, which, as already outlined, is crucial for understanding gestures and their role. Given the recent advances in machine learning, using a model to predict some features of hand gestures, as defined by a coding scheme, could be a promising approach with potential to accelerate gesture research.

There already exists a lot of scientific work focusing on gesture classification in video. One popular example of such research is Li et al.'s study "Hand Gesture Recognition Based on Convolution Neural Network" [5].

However, little research in this area uses gesture coding schemes, thus making it unclear how well would the currently available models perform when trained on data annotated according to such precise schemes. The reason that coding schemes are not widely adopted in gesture classification could be that most of these studies focus on classifying certain iconic gestures or interpreting their function, which doesn't necessarily require such highly precise coding.

Moreover, relatively little research focuses on the physical motion of hand gestures, possibly because it is more tempting to directly focus on other properties, such as their meaning and function. However, the meaning, or any other aspect of gestures, follows directly from the specific arrangement of hand movements. Thus having a better understanding of the motion itself could be quite revealing, since many relationships between the physical motion and other aspects of gestures may still remain undiscovered.

Finally, much of the gesture research uses video footage obtained in a lab setting or captured during lectures and speeches, which is mostly recorded from a front-facing perspective. Stergiou [6] notes that little research adapts a more social and contextual perspective in automated analysis of interactions between humans.

With the intention to address the points outlined above, the following **main research question** was formulated - Would it be possible to train a machine learning model to classify complex physical features of hand gestures, using video footage captured in a crowded social setting and annotated according to a gesture coding scheme? The following **sub-questions** had to be answered as well - What physical features of hand gestures to predict? What coding scheme would best fit that task? How to annotate the data? What kind of machine learning approach could be followed?

To be explicit, the novelty of this research lies in automatically classifying physical features of hand gestures, according to a coding scheme, and using footage from a densely crowded social setting for that task.

2 Background

To be able to fully understand the findings of this research, it is imperative to firstly understand the most basic concepts and the possible approaches. This section discusses those topics.

2.1 Hand Gestures and Dimensionality

Hand gestures are a fundamental form of nonverbal communication. They can convey various types of information, such as emotion, intentions, and instructions. McNeill [1] defines hand gestures as hand and arm movements that are meaningfully related to speech, while Kendon [2] extends that definition by describing gestures as intentional movements that can convey information alone or complement spoken language, adding depth and nuance to communication.

McNeill proposes a general classification of gestures consisting of four types: iconic, beat, deictic, and metaphoric gestures [1]. Iconic gestures visually represent some object or action - an example could be moving hand in a manner resembling a bird flapping it's wings. Beat gestures are rhythmic hand movements that accompany speech and can emphasize certain points. Deictic gestures are pointing gestures used to indicate objects, directions, or locations. Metaphoric gestures symbolize abstract concepts, for example size of some object [1]. This categorisation is often adopted in gesture research, and sometimes it is extended with emblematic gestures - gestures that have a specific meaning for some cultural group, for example the thumbs-up gesture.

However, such classification of gestures can sometimes be ambiguous - some gestures may belong to multiple categories making it hard to definitively classify them. An example of that could be a pointing gesture, which in some contexts could belong to both the deictic and beat categories. Furthermore, the classification proposed by McNeil is quite context dependent - in many cases it is required to know what was exactly said to be able assign a gesture to a category. This could be an issue, especially in the light of recently emerging social interaction research adapting privacy-sensitive approaches [7].

These issues can be addressed by adapting the concept of gesture dimensionality. Kendon first introduced it in his book "Gesture: Visible Action as Utterance" [2]. It facilitates more detailed analysis of the multifaceted roles that gestures play in communication, as well as highlighting that gestures are structured in nature. In short, gestures can be analyzed across various independent but interrelated dimensions. These dimensions focus on specific aspects of gestures, for instance their spatial characteristics (e.g., handshape, movement direction), temporal aspects (e.g., phases and rhythm), meaning (semantic content of gestures), and functional roles (e.g., emphasizing a point, referencing something).

As outlined in section 1, this research focuses on the physical aspects of hand gestures. As already stated, comparatively little research focuses on physical features of gestures, despite the fact that other aspects of gestures, such as meaning, directly follow from the movement itself. For instance a gesture mimicking the act of playing tennis only conveys that specific meaning because the arm motion resembles the actual movements one makes when playing tennis. This is quite a simple observation, but possibly there exist more intricate relationships between specific patterns of hand movements and other dimensions of gestures, for example a higher frequency of certain movement patterns could be related to certain emotions [8]. Thus being able to accurately classify and quantify physical features of hand gestures could be a good starting point for other studies focusing on other dimesnions and aspects of gestures in social interactions.

2.2 Gesture Coding Schemes

Since this study uses a gesture coding scheme for data annotation purposes, it is important to understand what precisely gesture coding schemes are and what benefits using them provides.

In research coding schemes are theoretical instruments used to categorise and assign appropriate codes to data. Their goal is to organise data in a systematic way, thus facilitating analysis and interpretation. Moreover, using coding offers deep insights into the data, that could not be observed otherwise, while also increasing transparency in research [9].

Similarly, gesture coding schemes are theoretical frameworks used to accurately describe (or "code") gestures, across various different categories. They usually use natural language for that purpose. Gesture coding schemes usually allow annotation across different categories, oftentimes directly relating to different dimensions of hand gestures. Alongside all the mentioned benefits that come with using any coding scheme, using gesture coding schemes could offer additional, more practical benefits - they could be used to generate gesture embeddings based on video footage, or they could even facilitate gesture reproduction by robots [10].

While numerous coding schemes have been proposed, neither of them is established as a standard, so it wasn't immediately clear which one should be used in this research. The considered gesture coding schemes were M3D [11], NEUROGES [12], MUMIN [13] and SmartKom [14]. Table 1 presents a comparison between those schemes.

Criteria	**M3D**	**NEUROGES**	**SmartKom**	**MUMIN**
Dimensions of Analysis	Form (physical nature of the movement), Prosodic (prosodic aspects of the moevment), Meaning (semantic / pragmatic meaning)	Kinetic (kinetic features of movment), Bimanual Relation (relation between two hands), Functional (function of gestures)	No strict dimensions defined, very functional in nature	No strict dimensions defined, does however represent the handedness, shape of gesture, as well as function
Integration with Speech	Focus on temporal association between speech and gestures, as well as the relation in meaning of gestures and speech	Moderate integration with speech, might be necessary to relate to speech to annotate certain functions	Integration with speech in form of Supporting Gestures, that can further reinforce speech.	Looks at the relation between speech and gestures in turn management.
Ease of Use	Moderate (requires training), however it comes with a very good training program	Moderate (requires training), offers seminars and a training package	User-friendly (Quite small in comparison to other schemes, thus easier to use)	Moderate (requires training)
Primary Use Case	Dimensionalised annotation of gestures, "Independent and comprehensive assessment of co-speech gestures" [11]	"The NEUROpsychological GESture coding system is a tool for empirical gesture research that combines a kinetic with a functional analysis of gestural behavior"[12]	Human-computer interaction, tailored for a specific software, namely the SmartKom Project [15]	"Annotation of feedback, turn-managing and sequencing functions of multimodal expressions" [13]

Table 1: Comparison of gesture coding schemes

The SmartKom [14] coding scheme was derived to support a specific human computer interaction undertaking, the SmartKom Project [15]. It is a concept of a multi-modal computer interface that facilitates easy interaction via speech and gestures [15]. This means that SmartKom's coding approach was designed to facilitate seamless interaction with computers, rather than with gesture research in mind.

Similarily, MUMIN [13] was designed for a very specific purpose, that is turn-management analysis. It features codes for emotions, attitude, facial displays as well as hand gestures. However, the coding concept was very clearly designed with turn-management in mind, thus making it not very well suited for more general interaction analysis.

Ultimately, the MUMIN and SmartKom coding schemes were excluded from consideration. Not only were they not created with interaction analysis in mind, but also neither of these two explicitly differentiates between distinct dimensions of hand gestures (look at Table 1). As this research specifically focuses on the physical aspects of hand movements during gesturing, it was crucial for the coding scheme of choice to incorporate the concept of a physical dimension, since that facilitates annotating the physical features of gestures in isolation.

The NEUROGES [12] coding scheme does incorporate a concept of dimensions. It names the following dimensions (named as modules) of hand gestures: the Kinetic module focusing on the kinetic features of hand movement (trajectory and dynamics), Bimanual Relation module coding the relation between two hands (in-touch, separated, functional relation) and Functional module for coding the function of gestures (emphasis, emotion, conention, etc.). It was created with gesture research in mind and thus aligns with the purpose of this research.

The M3D [11] coding scheme also names distinct dimensions of hand gestures. It names the Form dimension describing the physical aspects of the movement (hand shape and orientation, trajectory, etc.), the Prosodic dimension focusing on the temporal aspects of gestures (phasing, rhytmic properties, etc.), and the Meaning dimension coding the semantic and pragmatic meaning of gestures (referntiality, mataphoricity, deixis, etc.). M3D is primarily intended for assesment of co-speech gestures, which is very relevant for interaction and communication research. Moreover, it was designed with adaptability in mind, which makes it more flexible than the other options.

Both NEUROGES and M3D seem to be quite well suited for use in social interaction research. Nevertheless, M3D appears to be the most extensive and comprehensive of the schemes, while also offering additional benefits, such as a very recent labelling tutorial with clear instruction videos. The final selection of the scheme is discussed in subsection 3.1.

2.3 Gesture Recognition in Video Footage

As classifying features of hand gestures falls within the category of action recognition, it was important to survey the available approaches and decide which methods could be applied to the task at hand.

The focus was specifically on the techniques suitable for video footage. This study focuses on the video modality for two main reasons. First of all, video data is widely adopted in social sciences and related research [16]. Secondly, directly using video footage in interaction research can be the quicker approach, since it requires less preparation and pre-processing in comparison to other modalities, such as skeleton data, which needs to be additionally extracted.

A popular and proven approach for video-based action recognition is to utilise two-stream networks [17]. This approach requires separately modelling spatial and temporal features, to later fuse them. However, a lot of the models adapting this method use a shallow architecture, which could be an issue when dealing with data as complex as hand gestures.

Another prevalent technique is to utilise Convolutional Neural Networks with 3D convolution filters [17]. In this approach a 3D tensor data structure with two spatial dimensions and one temporal dimension is used to store video footage, and subsequently a 3D convolution filter is applied to such tensor. This allows for effective modelling of the temporal relationship between frames. It is especially important when classifying gestures, which are inherently spatiotemporal phenomena.

A recently emerging approach is to utilise Transformer models [18] [19] [20]. Due to their attention mechanism, transformer models are capable of handling complex dependencies within data, while also being quite flexible and highly parallelisable during training [21]. Moreover, video Transformers achieve state-of-the-art results on various action recognition datasets [20], while also demonstrating good result on smaller datasets [18] [19]. Finally, these models are also effective at modelling spatiotemporal relationships.

All in all, any of the proposed approaches could theoretically work when classifying physical features of hand gestures based on video footage. However, given the high complexity of hand gestures and the additional complexity and noise related to the video data being recorded in a crowded social-setting, using 3D CNNs or Transformer models could be superior when compared to the two-stream network approach. The final machine learning approach is described in detail in subsection 3.2.

3 Approach

As the main research question states, the goal of the research was to verify whether it would be possible to train a machine learning model to classify complex physical features of hand gestures, using video footage captured in a crowded social setting and annotated according to a gesture coding scheme. This section introduces and motivates the general approach that was followed to achieve that goal.

The following approach was derived. Firstly, the gesture coding scheme to be used for annotating the data had to be selected. This also required choosing the exact aspect of the physical dimension of hand gestures to be labeled. Then the video footage had to be annotated, using a video labelling tool of choice. The data had to be pre-processed and finally a machine learning model had to be chosen, fine-tuned, and the relevant results had to be extracted.

The dataset, data annotation process, data prepossessing, as well as model training are described in detail in section 4. The focus of the remainder of this particular section is to explain and motivate the choices leading up to the experiment itself.

3.1 Gesture Annotation Approach

After thorough consideration, the M3D coding scheme was selected to be used for gesture annotations. As mentioned in subsection 2.2, after initially rejecting SmartKom [14] and MUMIN [13], M3D [11] and NUROGES [12] remained in consideration. There were a couple of reasons why M3D was chosen in the end. Firstly, M3D is more detailed than NEUROGES. Furthermore, the form dimension of M3D is structured more clearly in comparison to the kinetic module of NEUROGES, which was of high relevance given this study's focus on the physical aspect of gestures - the less ambiguity in annotations the better. The tutorial that M3D offers was also a factor, since it is very detailed and easy to follow. For a more detailed comparison between all the considered gesture coding schemes, please revisit the subsection 2.2.

As already outlined in subsection 2.2, M3D differntaites between the form, prosodic and meaning dimensions. The dimension most relevant to this research however, is the form dimension. It's focus is on accurate coding of the physical movements involved in gesturing.

M3D [11] distinguishes between the following four categories, or tiers, within the Form dimension: the hand shape and the palm orientation tiers focusing primarily on the hand itself, as well as the trajectory direction and trajectory shape tiers, which focus on the movement of the arm and hand in space.

This research strictly focuses on the trajectory direction tier of the form dimension. It's meant for precise coding of the direction in which the hand is moving when gesturing. There are several reasons behind that decision. Firstly, there exists studies suggesting that upwards arm movements could express positive emotions while downwards movements could relate to negative emotions [8], so being able to automatically classify such motions and calculate their frequency could help in estimating the emotional valence of interactions. This is just an example, but it demonstrates that being able to automatically classify and quantify the exact directions of hands when gesturing could be quite helpful in social interaction analysis. Secondly, it could be possible to infer the shape of the movement based on subsequent movement directions. This means that similar studies focusing on the trajectory shape tier could potentially build directly on top of the findings of this study.

According to M3D, the trajectory direction of hand gestures should be annotated in the following way; first, it proposes annotating the G-unit, which it defines as a segment of continuous motion that represents a complete gesture [11]. Afterwards, the main manual articulator should be annotated, which stands for the main gesturing hand, and finally the trajectory direction should be annotated for both the manual articulators. There are 16 distinct annotations for trajectory direction - up (U), down (D), left (L), right (R), towards the speaker (S), forward and away from the body (F), rotation (Ro), no movement (NM), as well as a total of 8 diagonal movement codes (diag-UR, diag-UL, diag-DR, diag-DL, diag-US, diag-UF, diag-DS, diag-DF).

3.1.1 Annotation Framework Selection

Annotating the trajectory direction of hand gestures in video footage required using a data annotation framework. Choice of the tool could have had direct impact on the labelling process, as the quality of annotations and the amount of labeled data can directly influence classifier accuracy [22]. The annotations had to be very consistent, while also the labelling process itself couldn't be overly time consuming; due to time constraints only a 2 week period was allocated for that part. Two tools were initially considered, that being Covfee [23] and ELAN [24].

Covfee is a continuous-time annotation framework, meant for labelling video and audio data involving social interactions. It aims to speed up the process by allowing for continuous video annotation, without the need for pausing [23]. When this study was taking place Covfee only supported binary annotations, which would require annotating each of the trajectory directions in separation. This could reduce the annotation speed, thus ultimately countering the speed benefit of Covfee.

The other considered framework was ELAN [24]. It was created with annotation and transcription of video and audio data in mind. It allows for precise multi-layer annotations thus making it possible to annotate both hands in parallel. Furthermore, contrary to the binary approach of Covfee, it facilitates annotating multiple labels at the same time, which could be very useful, given the multitude of codes offered by M3D [11]. ELAN doesn't support continuous data annotation and it requires manually choosing the start and end frames for each annotation, but it makes very precise temporal alignment possible, which is especially important for annotating hand gestures, since they can be very complex and fast.

In the end, after consulting the matter with the developers of the Covfee framework, it became clear than ELAN would be better suited for the specific task of annotating trajectory direction of hand gestures. Not only does it allow for precise multi-layer annotations, but also the M3D coding scheme provides an annotation template that can be directly imported into the ELAN tool.

3.2 Machine Learning Approach

The final consideration before carrying out the research experiment was the selection of a machine learning approach. As outlined in subsection 2.3, two stream models, 3D CNNs and video Transformers were considered.

There were several criteria that the final approach had to fulfill. Firstly, it had to employ the concept of time-space attention, since gesturing, and more specifically the hand trajectory direction while gesturing, is inherently a spatiotemporal phenomena - to be able to classify whether a hand is moving in a certain direction, it is necessary to know where exactly in space it was across a whole time interval. This is not a new concept and some research has already demonstrated the effectiveness of time-space attention in skeleton-based action recognition [25]. Moreover, the model had to be able to generalize well on small datasets, since, as already mentioned in subsubsection 3.1.1, the time available to annotate footage was limited.

As already outline in subsection 2.3, the 2 stream approach was rejected because it usually involves shallower models. There was a risk that such models could fail to handle the complexity of hand gestures.

However, both video transformers and 3D CNNs fulfill the criteria. As outlined in subsection 2.3 they are both capable of handling complex spaciotemporal dependencies within data and both are proven in the field of action recognition. While using 3D CNNs could have also been a perfectly viable approach, because of the very good performance that Transformers achieve on small datasets [18] [19] and also because of their recent popularity and success in various areas [26], the decision to use a transformer model was made, especially that it doesn't present any immediate drawbacks.

The availability of open-source transformer models capable of video classification was quite limited. In the end the following models were considered: ViViT [18], VideMAE [19], and TimeSformer [20]. All the models performed well during test runs on the UCF 101 dataset [27] reaching accuracy of roughly 87.5 - 100%. In the end the VideoMAE model was chosen. It achieves exceptionally good performance on small datasets because of a novel approach which leverages self-supervised learning [19], while also offering the most pre-trained and fine-tuned versions using various datasets and for various numbers of epochs. VideoMAE is a Masked Autoencoder model that uses a ViT [28] backbone, which means it uses a smart approach that enables the ViT to be used for video classification. One important consideration is that it uses temporal down-sampling, meaning that it will only look at a subset of in-between frames, instead of the whole video. This should not be an issue however, since capturing the direction of the hand movements should still be possible when looking only at selected frames that are evenly spread across the whole hand motion.

4 Experiment

The research experiment was made up out of the following three parts: data annotation, data pre-processing, and model training. The general plan for the experiment was to annotate the video footage from the dataset according to the M3D gesture coding scheme, more specifically annotate the trajectory direction tier of the form dimension, and later, after prepossessing, fine-tune the VideMAE [19] model. The goal was to achieve accuracy that would imply that the model can actually recognise the features, so higher than majority-class and random classifiers would achieve.

4.1 ConfLab Dataset and the Real-Life Aspect

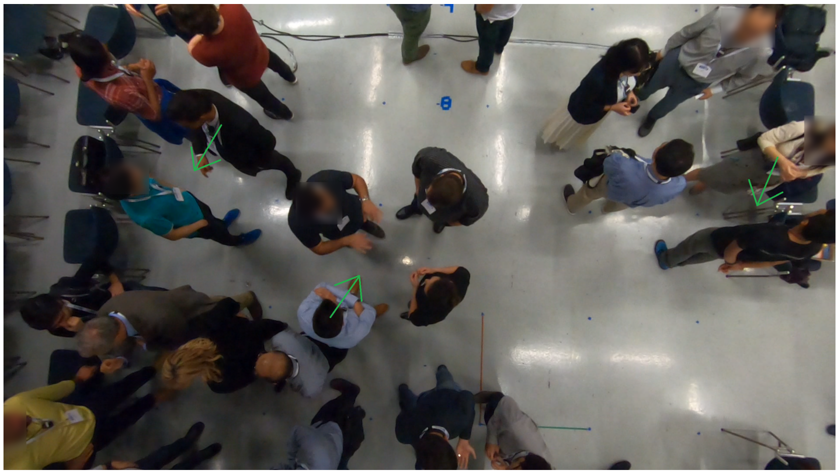


Figure 1: A frame from the sample video offered by the ConfLab [29] dataset, with manually added arrows highlighting varied positioning of the participants.

The ConfLab [29] dataset was used. It is very well aligned with the goals of this research, as it was created with the intention to be used for analysis of free-standing social interactions in a real-world setting and it captures natural and unscripted human behaviour. It contains data of different modalities, that being video footage, as well as acceleration data and low frequency audio captured via a wearable sensor. Since the focus of this research is on the video modality, it is imperative to understand the captured video. Eight overhead cameras were used to record 1920x1080 video at 60 frames per second. The videos involve 48 participants from diverse professional backgrounds talking in groups and naturally using hand gestures in the process. The participants form groups of various sizes, which corresponds well to real-life scenarios. This means that in the larger groups with more people participating in the discussion there will be more noise in form of other people gesturing, which also corresponds to reality. Finally, as demonstrated by Figure 1, the aerial view provides a mostly unobstructed view of all participants' hands while maintaining privacy by avoiding direct capture of the faces, which is an additional benefit.

4.2 Data Annotation Process

This first part of the experiment was instrumental to achieving the desired results. Not only enough data had to be annotated within the 2 week period, but the annotations also had to be consistent.

The trajectory direction subcategory of the form dimension of the M3D [11] coding scheme comes with a template for hand gesture annotations with the ELAN annotation framework [24]. Initially the plan was to follow that template exactly, but after the annotation process began it became obvious that for practical reasons some adjustments had to be made.

As outlined in subsection 3.1, M3D [11] proposes first annotating the G-unit, which it defines as a segment of continuous motion that represents a complete gesture. Afterwards, the main manual articulator should be annotated (the main gesturing hand), and finally the trajectory direction should be annotated for both manual articulators. However, such approach required oscillating between annotating the left and right hand in each of the manual articulator tiers. In practice this was too slow and resulted in confusion and mistakes while annotating. The decision was made to skip the annotation of the main articulating hand, and to just separately annotate the left and right hand. Knowing the main articulating hand could be important for inferring the semantic meaning of gestures later on, but this simplification was made under an assumption that it should be possible to train another model to predict that based on the trajectory direction of both hands, especially in scenarios when only one hand is articulating, since M3D includes a no movement (NM) label.

Secondly, M3D [11] proposes a total of 16 labels within the trajectory direction tier - up (U), down (D), left (L), right (R), towards the speaker (S), forward and away from the body (F), rotation (Ro), no movement (NM), as well as a total of 8 diagonal movement codes (diag-UR, diag-UL, diag-DR, diag-DL, diag-US, diag-UF, diag-DS, diag-DF). This leads to very little ambiguity, which is desired, but in practice having this many labels made annotating too slow. Furthermore, the diagonal movement annotations were especially time consuming, since it oftentimes was hard to tell what kind of diagonal motion was observed. Video in ConfLab [29] is recorded from a top-down perspective, making it hard to tell if the hand is moving slightly upwards or downwards, and those two movement directions are a component of each of the diagonal annotations. Thus the decision to disregard the diagonal labels was made. This introduced some ambiguity, but to counter that always the primary trajectory direction of a hand movement was annotated. For instance, in a scenario where a hand was moving slightly downwards but mostly to the left, the left (L) annotation would be selected.

These adjustments slightly simplify the problem, however given the 10 week time-frame to complete the whole research, these trade-offs had to be made. Nevertheless, the principle of using a coding scheme for gesture annotations still holds, especially given that M3D [11] is quite extensive in comparison to other coding schemes to begin with.

After making the described adjustments, the annotation process went as follows. Firstly, a set of 8 videos, each 2 minutes in length, was selected from the ConfLab [29] dataset. The videos came from various different cameras (cameras 2, 4, 6, 8 [29]) and included different people.

From each of the selected videos a subset of visible participants was chosen. In total annotations for 12 participants were prepared, over the whole duration of the 2 minute clips. It is important to note that the selected participants were facing different directions and were located in various areas of the frame. As depicted by Figure 1, this meant that some people were more easily visible than others and they could be seen from varying camera angles. This was a deliberate choice so that the data for the classifier was more varied.

For each of the selected participants the gesture units (G-Units) were annotated, exactly as the M3D [11] labelling guide instructs. This was done by selecting a time-interval corresponding to a respective G-Unit on the timeline within the ELAN [24] tool. Afterwards the trajectory direction of the hand movement was annotated for each of the hands separately. This again required selecting a time-interval and then choosing a corresponding label that best described the hand movement trajectory direction in that interval.

An example annotation can be seen in Figure 2. It depicts the right (R) annotation for the right hand, and no movement annotation for the left hand (NM). On the cropped frames it can be observed how both hands indeed follow these trajectory directions. Figure 2 also shows how precise the annotations were - the right (R) annotation spans less than one second.

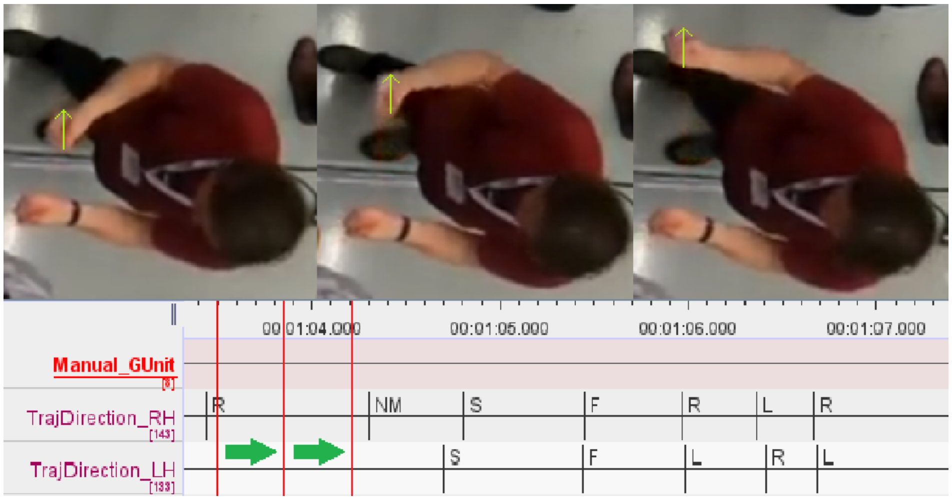
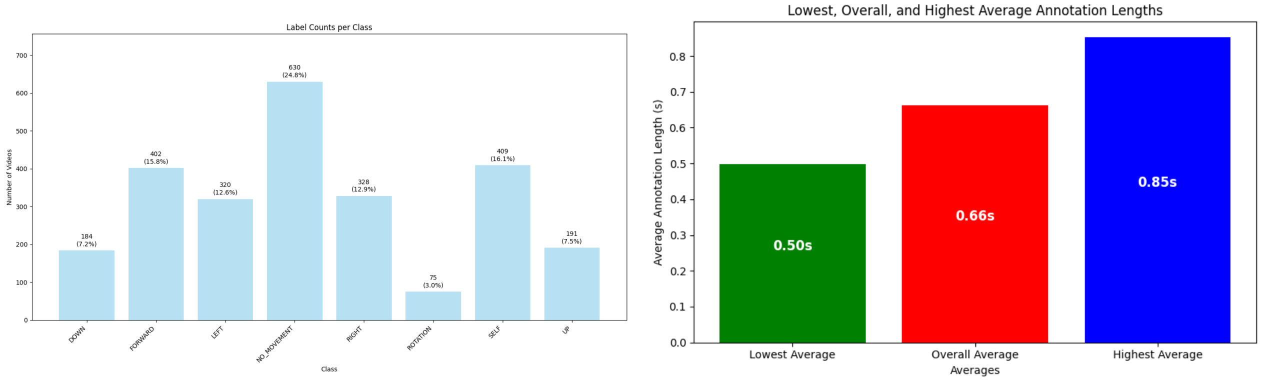


Figure 2: An example annotation in ELAN [24] for a short segment of a video.



(a) Distribution of annotatted classes.

(b) Average per-participant annotation lengths.

Figure 3: Details of the annotations, irrespective of the hand.

Some interesting observations were made when annotating the videos. Firstly, some trajectory directions were significantly more prevalent than others, for instance rotation (Ro) was very rarely observed as the primary trajectory direction. This imbalance, irrespective of the hand, is depicted by Figure 3a.

Moreover, the gesturing styles varied significantly between participants. For example some participants tended to make hand movements that were much longer and more pronounced in comparison to others, who would move their hands quickly but in very short strokes. This is reflected in the varying average annotation length across the participants, as shown by Figure 3b. This observation aligns with the idea that gesturing styles can vary from person to person [30] and it further complicated the annotation process.

It is important too note that the amount of annotated footage was limited by the time constraints. On average, after including the setup time and accounting for the initial learning curve, it took roughly 2 hours to annotate one minute of footage per participant.

4.3 Data Pre-Processing

Before the data could be used to fine-tune a classifier, it had to be pre-processed. Initially the plan was to split up the annotated videos into segments, where each segment would be assigned both the label for the right and left hand. This way a multi-class classifier, capable of labelling the trajectory direction for both hands, could be trained. However, the movements of the left and right hand aren't necessarily synchronised. To align the annotations between both hands, many video segments capturing a single hand stroke corresponding to a specific direction annotation would need to be broken down into even shorter segments, only corresponding to a part of the annotated hand movement. This could result in information loss in the data and could thus affect the accuracy, so in the end a more straightforward alternative was chosen - training two separate classifiers, one for the left and one for the right hand respectively.

To facilitate that, the data was pre-processed. Firstly, each of the videos was cropped and centered around the annotated person, but reasonable margins were included and the neighbouring participants were still visible in the frame. This was done to reduce the noise, in form of all the other participants in neighbouring groups. This was done manually, however algorithms capable of doing so automatically already exist [31]. Secondly, each of the annotated videos was broken down into segments corresponding to the annotated trajectory direction, separately for each hand. For instance, if a short segment of a video had the up (U) label for the right hand assigned, it would be copied and saved as a separate video, with the right hand and up (U) labels assigned to it. Finally, the resulting clips were shuffled and divided into the training set (75%), the validation set (15%), as well as the test set (15%). This was achieved by iterating through all the annotation classes and randomly assigning clips from each class to a respective set, with the probabilities being 75% for training, 15% for validation and 15% for testing. Afterwards, data augmentation techniques were applied, this being random shift, random flip and random rotation, as well as resizing to meet the VideoMAE's [19] resolution requirements.

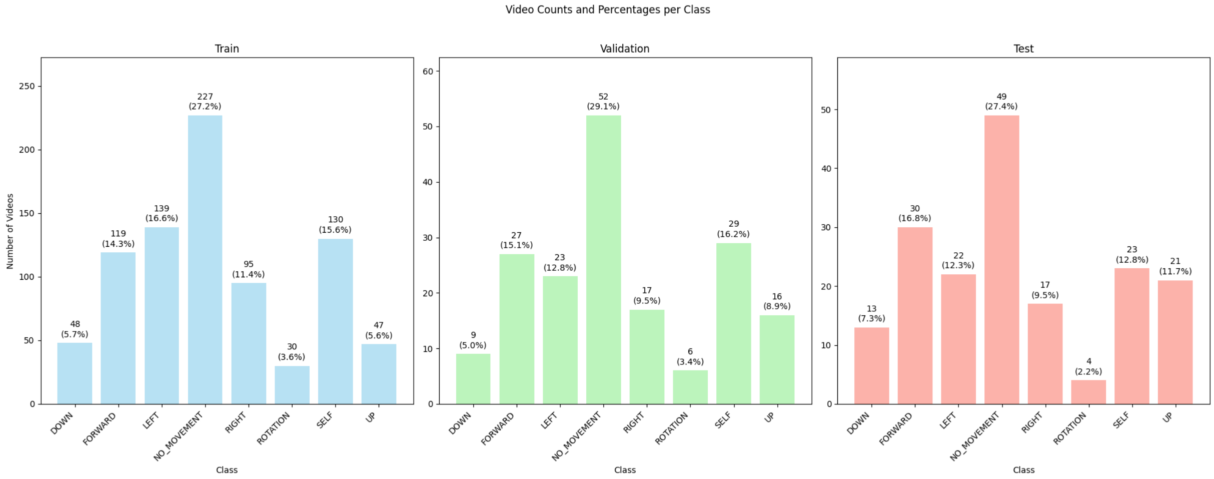


Figure 4: Left hand label distribution in train, validation, and test sets.

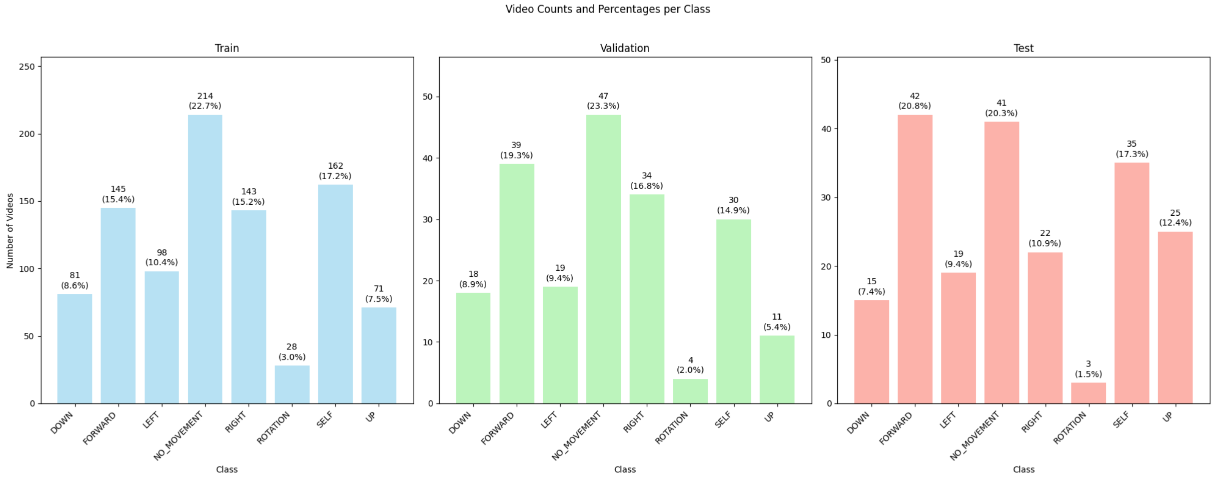


Figure 5: Right hand label distribution in train, validation, and test sets.

In the end a total of 1346 video segments were prepared for the right hand and 1193 segments for the left hand. The prepared datasets were quite imbalanced, which was not unexpected, given the limited number of annotated videos and accounting for the disproportion in trajectory direction prevalence that was already observed when annotating the data (subsection 4.2). The final data set distribution is depicted by Figure 4 for the left hand, and Figure 5 for the right hand respectively. It is also important to note that the distribution of the data in validation and test sets was roughly similar to the training set's distribution.

4.4 Model Training

The final step of the experiment was training the model. As explained in subsection 3.2, the VideoMAE [19] model was selected for the task. There were several versions of the model available, but in the end the version pre-trained and later fine-tuned on the Kinetics400 dataset [32] was selected (Huggingface: MCG-NJU/videomae-base-finetuned-kinetics). This decision was made because after early test runs this version appeared to consistently converge on the data. This was not surprising, since Kinetics [32] is a large human action recognition dataset and it contains videos somewhat similar to the ones in the ConfLab [29] dataset, although usually they are filmed from a different perspective.

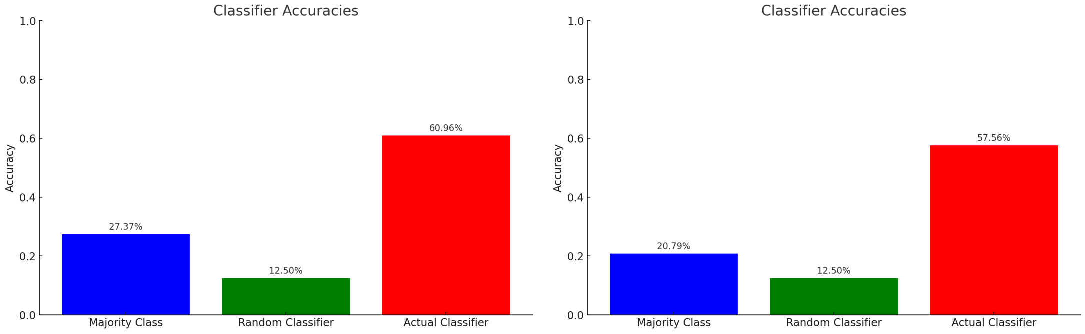
Two separate models were fine-tuned - one for the left hand [33] and one for the right hand [34]. The pre-processed video segments (look at subsection 4.3) were normalised and used as input. The default mean squared error loss function was used when fine-tuning the final model. Some issues with over-fitting were also encountered but they were resolved after adjusting the learning rate. Table 2 presents the hyper-parameters that were used during training of both the left-hand and right-hand models:

Parameter	**Value**
Number of epochs	7
Learning rate	5×10^{-5}
Training / Evaluation batch size	8
Seed	42
Adam optimizer	betas = (0.9, 0.999), epsilon = 1×10^{-8}
Total training steps	728

Table 2: Training hyper-parameters

5 Results

In this section in-depth results of the experiment are presented with the help of figures and the findings most relevant in the context of this research are further highlighted.



(a) Classifier accuracies for the left hand.

(b) Classifier accuracies for the right hand.

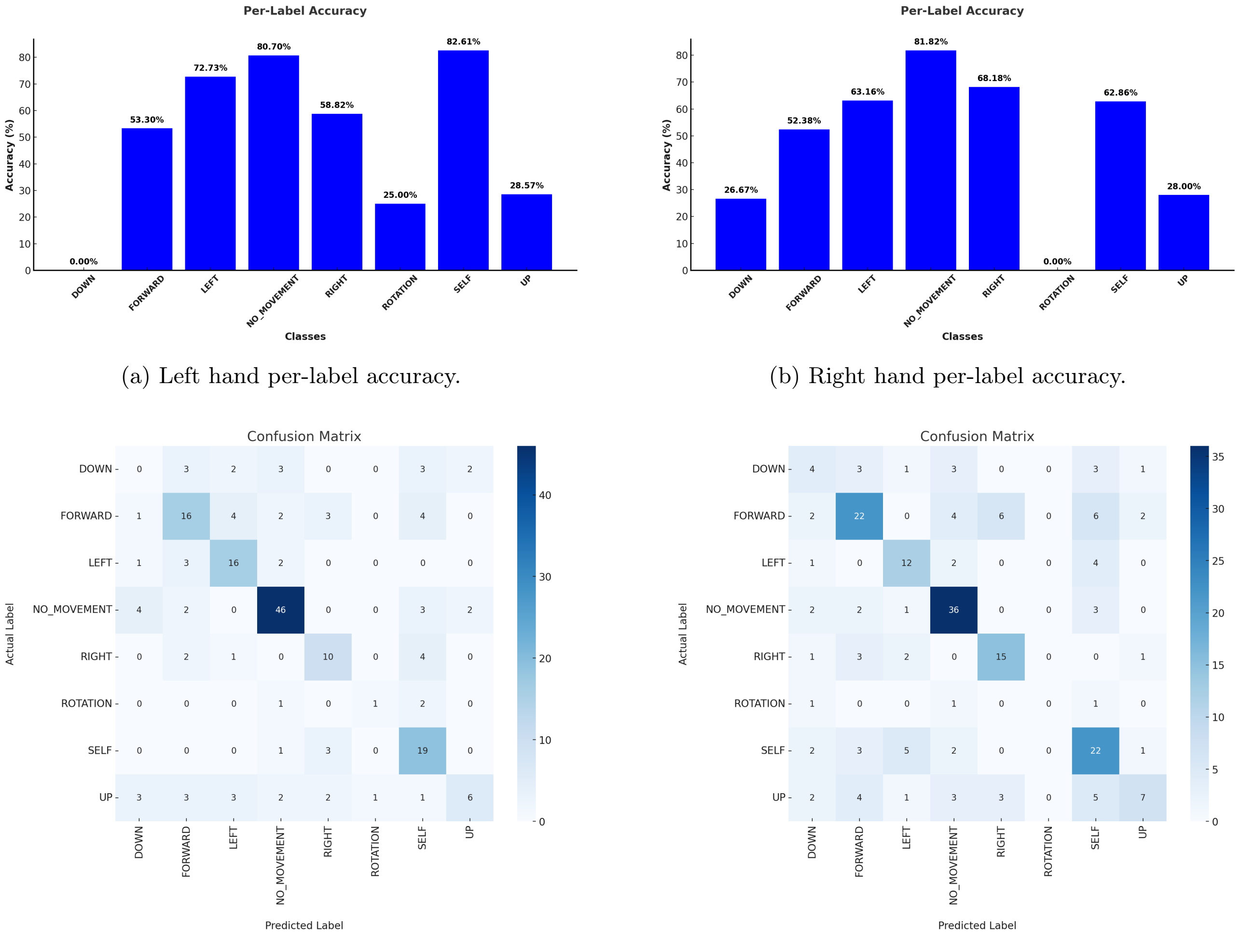
Figure 6: Comparison of classifier performance.

Metric	**Left-Hand Classifier**	**Right-Hand Classifier**
Validation Set Accuracy	0.6330	0.6019
Test Set Accuracy	0.6096	0.5756
Test Set Precision	0.6004	0.5670
Test Set Recall	0.6096	0.5756

Table 3: Classifier evaluation metrics

Firstly, as depicted by Figure 6, both the left-hand classifier and the right-hand classifier achieved results significantly (p « .000000001 for all the statistics, paired t-test was used) better than a majority-class classifier and a random classifier (on average, given 8 classes).

Looking at Table 3, a slight difference in left hand and right hand classification performance can be observed. The accuracy of the left-hand classifier was higher both on the validation and test set. Similarly, fewer false positives (precision measure) and more true positives (recall measure) were observed. Nevertheless, the observed difference is statistically insignificant ($p > .48$). Averaging the differences across all these metrics shows that the left hand classifier performed roughly 5.6% better than the right hand classifier.



(a) Left hand per-label accuracy.

(b) Right hand per-label accuracy.

(c) Left hand confusion matrix.

(d) Right hand confusion matrix.

Figure 7: Plots presenting detailed training results.

Significant differences between per-class classification accuracies can be observed. Figure 7a shows the per-label accuracy of the left hand classifier and it clearly achieved the worst results for the ’down’ (0%), ’rotation’ (25%), and ’up’ (28.6%) labels. These classes were also the least prevalent in the left-hand training dataset, as shown by Figure 4. Similarly, as shown in Figure 7b, the right-hand classifier achieved lowest accuracy for the same labels: ’rotation’ (0%), ’down’ (26.7%), and ’up’ (28%). Again, these classes also were underrepresented in the right-hand training set, as it is shown in Figure 5. Contrastingly, both classifiers achieved very high accuracy (over 80%) for the no-movement label, which was the most prevalent in both training sets. Finally, for the remaining label both classifiers achieve an accuracy of over 50%, while reaching as high as 80% for some classes and oscillating around the 60-70% mark for most.

Looking at confusion matrices, some insightful observations can be made. As shown by Figure 7c and Figure 7d, for both classifiers there were very few misclassifications between the ’right’ and ’left’ classes (1 for the left-hand classifier and 2 for the right-hand one), despite the fact that in the video data the participants were facing various directions (Figure 1).

On the contrary, many confusions can be observed for the ’up’ and ’down’ classes. As shown by Figure 7c for the left hand classifier the ’down’ label wasn’t predicted correctly even once, while the reaming labels were assigned to ’down’ motion 13 times. A similar observation can be made for the ’up’ label, which the classifier managed to predict correctly only 6 times, while the other labels were assigned to it 15 times. Very similar performance for the ’down’ and ’up’ labels can be also observed in the right-hand classifier (Figure 7d), where correct-incorrect prediction ratio for the ’up’ motion equated 7-18, and for the ’down’ label the correct-incorrect ratio was 4-11. For both classifiers no primary confusion pattern can be observed for the ’up’ and ’down’ misclassifications, but rather all the remaining labels were roughly evenly predicted instead.

Before the training of the model there were some concerns that the left-hand model could quite easily confuse the ’forward’ class with the ’left’ class and the ’self’ class with the ’right’ class. As depicted by Figure 7c, the left hand model only confused ’forward’ and ’left’ 7 times and the label ’self’ with ’right’ also 7 times, which means that these misclassifications did occur, but they were not much more prevalent than other confusion patterns. Similarly, the anticipation was that the right-hand model could confuse the ’right’ and ’forward’, as well as the ’self’ and ’left’ classes. Figure 7d demonstrates that the ’right-forward’ confusions occurred 9 times and the ’left-self’ confusions occurred also 9 times, which was not as severe as anticipated. The overall accuracy for all these labels falls into the 50-80% range.

6 Discussion

Overall, the results are quite promising. Both the left hand and right hand models achieved an overall accuracy of around 60% on both the validation and test sets, thus significantly exceeding the performance of a majority class classifier as well as surpassing the average performance of a random classifier. This indicates that it is possible to use using machine learning models for automatic prediction of physical features of hand gestures in social interactions, as coded by a coding scheme.

As highlighted in section 5 the left hand classifier performed a bit better than the right hand classifier, however the difference is not statistically significant and could be attributed to randomness in data or slightly better quality of the left-hand annotations.

The results also demonstrate the direct impact of the imbalanced dataset. For both the classifiers, the lowest accuracy was achieved for the ’up’, ’down’, and ’rotation’ classes, which directly corresponds to those classes being underrepresented in both the datasets. To counter the imbalance in the training datasets several attempts to use a weighted cross-entropy loss function were made, instead of using mean-squared error. It did work to some degree, but the overall accuracy was negatively affected, and getting the label weights right required continuous fine-tuning of the model, which wasn’t possible due to limited access to the compute cluster. Nevertheless, it could be a useful technique to utilize in future research.

On the contrary, both models managed to predict the ’left’ and ’right’ labels accurately and these were virtually never confused with each another. Given that the participants captured in the dataset video footage were facing different directions, this implies that the models managed to learn that feature quite well and thus distinguish between participants’ left and right sides no matter the direction they were facing.

However, high ambiguity within the data annotated with the ’down’ and ’up’ labels was observed, as all of the other labels were miss-assigned to those movements with similar frequency. This strongly aligns with the decision to ignore the annotations for movements with a diagonal trajectory (subsection 4.2), since those discarded annotations are specifically meant for more detailed coding of movements that have a partially upwards or downwards direction. Using the complete trajectory direction category of the M3D scheme when annotating would have introduced more granularity in the annotations for those movements, which in turn could have resulted in less ambiguity and higher overall accuracy.

On a positive note, some of the ambiguity that I anticipated to be problematic wasn’t as prevalent as expected. The left-hand classifier produced fewer ’forward-left’ and ’self-right’ miss-classifications than predicted, while the right hand classifier demonstrated fewer ’forward-right’ and ’self-left’ miss-classifications than I thought it would. This indicates that the quality of annotations for those trajectory directions was sufficient and that in the majority of cases the model managed to distinguish between those somewhat similar movements.

All in all, the results are quite promising and they can also be explained, given the decisions made in the experiment and some other factors. There are multiple signs indicating that having a more balanced dataset, as well as using the complete trajectory direction annotation scheme, could further improve the overall performance.

7 Limitations

It is important to acknowledge that there were several limitations that certainly influenced the process and the outcome of this research.

First of all, the research was bound by time. Since this research is a bachelor's degree thesis, a period of 9 weeks (effectively 8 weeks) was allocated for its completion. Using a simplified version of trajectory direction annotation scheme was a direct result of that limitation, since it was necessary to speed up the data annotation process. As outlined in section 6, this very likely had a direct impact on the results. Moreover, a more thorough evaluation could have been carried out given more time.

Access to computational resources was also limited. Model training and evaluation had to be performed on the DelftBlue [35] cluster, since the ConfLab dataset is strictly bound by GDPR and therefore no commercial platform could be used. A rather small node group was assigned to all the participants of the Research Project course and thus it was difficult to get a compute node with enough resources allocated. This had a direct impact on the findings, since it halted further experiments with weighted cross-entropy loss function, while also preventing a more thorough evaluation.

In summary, there is a high likelihood that having more time allocated as well as having better access to compute resources could have resulted in even more promising results.

8 Future Work

The work completed throughout this research could be further extended, while also providing various interesting insights that could be used as a basis for future work.

The first step would be to carry out a more extensive study concerning the same topic. Based on the remarks made in section 6, better performance could be achieved by preparing a more balanced dataset and using some balancing techniques, as well as by following the trajectory direction annotations exactly as proposed by M3D.

Secondly, after a more accurate classifier for the trajectory direction is derived, the possible applications for it could be explored. One interesting venue could be automatic creation of embeddings that capture the trajectory direction of hand gestures based on video footage. As explained in subsection 2.1, this could be useful for quantifying the hand motion, which in turn could help classify related features, such as the main articulating hand, the emotional valence of interactions, or even the semantic meaning of hand gestures.

9 Conclusion

This research has managed to successfully demonstrate the feasibility of using machine learning models to classify complex physical features of hand gestures in video footage captured in crowded social setting. By employing the M3D [11] gesture coding scheme and using a transformer-based model, two classifiers capable of that task were derived. Both the left-hand and right-hand classifiers achieved accuracies that significantly surpass the baselines (majority class and random approaches), thus indicating that the subtle variations in the trajectory direction of hand movements can be effectively captured and distinguished.

Moreover, this study presents a complete approach that could be applied to other research with a similar focus. Gesture annotation according to the M3D coding scheme in combination with the ELAN tool was demonstrated, alongside the complete pre-processing and model training approaches.

Despite the promising results, some issues were identified, spanning from an unbalanced dataset and the slight simplifications that were made during the annotation process. As section 6 demonstrates, these factors likely constrained the overall performance of the derived models. Future research should focus on addressing these limitations by mitigating the dataset imbalance and by more strictly adhering to the annotations that the M3D coding scheme proposes. Further improvements in classifier accuracy could facilitate the application of these models in various areas of social interaction analysis, such as emotion or meaning prediction.

In summary, this research validates the potential of machine learning for detailed gesture analysis in real-life social settings and provides a robust foundation for future explorations and applications in the field of gesture analysis in social interactions.

10 Responsible Research

Every researcher should place great focus on ethical implications of their work, as well as reproducibility and transparency. This section reflects on these aspects of this research.

10.1 Ethical Implications

While there is a lot of potential for applying hand gesture research in a positive and generally beneficial ways, it also crucial to acknowledge that there are some ethical concerns related to it.

Achieving high accuracy in understanding the semantics of hand gestures could facilitate development of new surveillance technologies infringing on peoples' privacy. Nowadays this is especially concerning, since in the past there have been numerous cases of both corporations and governments misusing technology for that exact purpose.

Another concern is related to algorithmic bias. The used training set was quite small and while the ConfLab [29] dataset includes participants of various ethnicities and genders, not all the groups were equally represented in the processed data used in the experiment, as those were not the primary driving factors during selection of the participants for whom the trajectory direction was annotated.

10.2 Reproducibility

I believe that the results achieved in this research are reproducible. Firstly, the section 4 outlines which dataset was used, as well as explaining the selection of data to be annotated. Moreover, data annotation and pre-processing are explained in great detail, and finally a precise description of the training process is given. This includes providing the hyper-parameters used when training the model, as well as the exact open-source version of the base model. Furthermore, the code used for preparing the datasets as well as the notebook used for model training are available in a public GitHub repository (`https://github.com/flatala/hand-gesture-classification-rp-2024`). Given all these details, setting up the machine learning pipeline should be possible for anyone with a computer science background.

10.3 Data Security

The ConfLab [29] dataset is strictly bound by GDPR. All the necessary permissions to access and use the dataset for the purpose of this study were arranged beforehand. Furthermore, to prevent any violations, all of the data was securely stored locally and the DelftBlue [35] cluster was used to train and evaluate the models, instead of commercial clusters. Finally, all of the vulnerable data was removed from the cluster's storage and local storage after the research was concluded.

10.4 Utilisation of LLMs

Large Language models were used as an assisting resource during this research. More specifically, OpenAi's GPT-4o [36] model was used. There were several use-cases for the model:

- Reformatting references into the correct BibTeX format.
- Formatting LaTeX code for the figures and tables.
- Finding research papers related to a specified topic.
- Rephrasing small sections of the report (mostly conclusion).
- Asking general questions related to the filed of research when looking into certain topics.
- Assistance with coding, troubleshooting issues with code and coding environments.

All of the outputs provided by the AI models very verified for correctness and critically assessed. For instance the BibTeX citations that the model provided were all manually checked at the end and corrected if necessary. Example LLM prompts can be found in Appendix A.

References

[1] D. McNeil. *Hand and mind: What gestures reveal about thought.* University of Chickago Press, 1992.

[2] Adam Kendon. *Gesture: Visible action as utterance.* Cambridge University Press, Cambridge, UK, 2004.

[3] Susan Goldin-Meadow and Martha W. Alibali. Gesture's role in speaking, learning, and creating language. *Annual Review of Psychology*, 64:257–283, 2013. Epub 2012 Jul 25.

[4] Natasha Abner, Kensy Cooperrider, and Susan Goldin-Meadow. Gesture for linguists: A handy primer. *Language and Linguistics Compass*, 9(11):437–451, November 2015.

[5] Li et al. Hand gesture recognition based on convolution neural network. *Cluster Computing - the Journal of Networks, Software Tools and Applications*, 2019.

[6] Alexandros Stergiou and Ronald Poppe. Analyzing human–human interactions: A survey. *Computer Vision and Image Understanding*, 188:102799, November 2019.

[7] Jose Vargas Quiros and Hayley Hung. Cnns and fisher vectors for no-audio multimodal speech detection. In *MediaEval*, 2019.

[8] George Lakoff and Mark Johnson. *Philosophy in the Flesh: The Embodied Mind and Its Challenge to Western Thought.* Basic Books, 1999.

[9] Mai Linneberg and Steffen Korsgaard. Coding qualitative data: a synthesis guiding the novice. *Qualitative Research Journal*, 05 2019.

[10] Kim et al. Gesture encoding and reproduction for human-robot interaction in text-to-gesture systems. *Industrial Robot: An International Journal*, 2012.

[11] Patrick Louis Rohrer, Ulya Tütüncübasi, Ingrid Vilà-Giménez, Júlia Florit-Pons, Núria Esteve Gibert, PeiLin Ren, Stefanie Shattuck-Hufnagel, and Pilar Prieto. The multimodal multidimensional (m3d) labeling system. *Open Science Framework*, 2023.

[12] Hedda Lausberg and Han Sloetjes. Coding gestural behavior with the neuroges-elan system. *Behavior Research Methods*, 41(3):841–849, 2009.

[13] Jens Allwood, Loredana Cerrato, Laila Dybkjaer, Kristiina Jokinen, Costanza Navarrettan, and Patrizia Paggio. The mumin multimodal coding scheme. 2005.

[14] Silke Steininger, Bernd Lindemann, and Thorsten Paetzold. Labeling of gestures in smartkom - the coding system. In *Gesture and Sign Language in Human-Computer Interaction*, pages 215–227, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[15] Wolfgang Wahlster, Norbert Reithinger, and Anselm Blocher. Smartkom: multimodal communication with a life- like character. pages 1547–1550, 09 2001.

[16] S. Fazeli, J. Sabetti, and M. Ferrari. Performing qualitative content analysis of video data in social sciences and medicine: The visual-verbal video analysis method. *International Journal of Qualitative Methods*, 22, 2023.

[17] Yi Zhu, Xinyu Li, Chunhui Liu, Mohammadreza Zolfaghari, Yuanjun Xiong, Chongruo Wu, Zhi Zhang, Joseph Tighe, R. Manmatha, and Mu Li. A comprehensive study of deep video action recognition, 2020.

[18] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6836–6846, October 2021.

[19] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training, 2022.

[20] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)*, 2021.

[21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[22] Christopher A Ramezan, Timothy A Warner, Aaron E Maxwell, and Bradley S Price. Effects of training set size on supervised machine-learning land-cover classification of large-area high-resolution remotely sensed data. *Remote Sensing*, 13(3):368, 2021.

[23] Jose Vargas Quiros, Stephanie Tan, Chirag Raman, Laura Cabrera-Quiros, and Hayley Hung. Covfee: an extensible web framework for continuous-time annotation of human behavior. In *Understanding Social Behavior in Dyadic and Small Group Interactions*, volume 173 of *Proceedings of Machine Learning Research*, pages 265–293. PMLR, 16 Oct 2022.

[24] Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. ELAN: a professional framework for multimodality research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, Genoa, Italy, May 2006. European Language Resources Association (ELRA).

[25] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[26] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. A survey of transformers. *AI Open*, 3:111–132, 2022.

[27] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[28] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[29] Chirag Raman, Jose Vargas Quiros, Stephanie Tan, Ashraful Islam, Ekin Gedik, and Hayley Hung. Conflab: A data collection concept, dataset, and benchmark for machine analysis of free-standing social interactions in the wild, 2022.

[30] Özer, Derya and Göksun, Tilbe. Gesture use and processing: A review on individual differences in cognitive resources. *Frontiers in Psychology*, 11:573555, Nov 2020.

[31] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, March 2020.

[32] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset, 2017.

[33] Franciszek Latała. Left-hand trajectory direction classifier. `https://huggingface.co/flatala-research/VideoMAE-conflab-traj-direction-LH`, 2024.

[34] Franciszek Latała. Right-hand trajectory direction classifier. `https://huggingface.co/flatala-research/VideoMAE-conflab-traj-direction-RH`, 2024.

[35] Delft High Performance Computing Centre (DHPC). DelftBlue Supercomputer (Phase 2). `https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase2`, 2024.

[36] OpenAI. Chatgpt-4: Generative pre-trained transformer, 2023.

A Example LLM Prompts

Below are some of the LLM prompts used throughout this research. Keep in mind that the original prompts contained spelling mistakes and to increase readability such mistakes were corrected in the quoted phrases.

> “Please find me research papers focusing on practical applications of gesture coding schemes.”
>
> “I will send you an excel file containing a table comparing different coding schemes. Please covert this table into LaTeX code.”
>
> “A. Kendon. Gesture: Visible action as utterance. 2004. Give me a complete citation in BibTeX.”
>
> “What are the most popular methods to mitigate an imbalanced dataset?”
>
> “What could be causing the following error:

```
ERROR: Ignored the following yanked versions: 0.1.6, 0.1.7,
0.1.8, 0.1.9, 0.2.0, 0.2.1, 0.2.2, 0.2.2.post2, 0.2.2.post3
ERROR: Could not find a version that satisfies the requirement
torchvision==0.13.1 (from versions: 0.17.0, 0.17.1, 0.17.2, 0.18.0)
ERROR: No matching distribution found for torchvision==0.13.
```

> ”
>
> “How to remove non empty directories os.removedirs?”
>
> ”I have the following folder structure: in the root folder there are multiple folders, each named as the video label it corresponds to. In each of the subfolders there is videos. Please write me a python script that saves a chart plotting the video counts per class.”