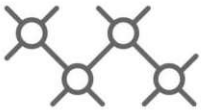


Institut für Computertechnik
Institute of Computer Technology

A MASTER THESIS ON

Multimodal Transformer Models for Human Action Classification

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

Diplom-Ingenieur

(Equivalent to Master of Science)

in

Automation and Robotic Systems (UE 066 515)

by

Zoltán Varga

11823287

Supervisor:

Univ.-Prof. Dr.-Ing. Dongheui Lee

Co-supervisors:

Projektass. Esteve Valls Mascaro

Univ.Ass. Daniel Jan Sliwowski

Vienna, Austria

September 2024

Abstract

The majority of research in deep learning focuses on processing a single modality, such as image, audio, text, or proprioception data. However, humans benefit from leveraging information from diverse senses on a daily basis for richer information acquisition. In intelligent systems, no single sensor can fully capture all aspects of the surroundings, making it necessary to integrate different types. Fusing sensor modalities allows for achieving a deeper understanding of the environment. Inspired by this, we design a transformer-based multimodal model for human action recognition and thoroughly evaluate its performance and robustness. In particular, this thesis focuses on examining the benefit of using multimodal data for human action recognition in a kitchen scenario. We conduct an ablation study of the model's understanding through a systematic quantitative evaluation and investigate the influence of uniting modalities on performance. Furthermore, we analyse the role of each modality by comparing the attention scores of our transformer architecture and explore different fusion methods to assess how various modalities are best combined. Our study shows that multimodal transformers perform better than their modality-specific equivalents. The highest boost of accuracy compared to our vision-only baseline is $+10.1\%$, achieved by late fusion trained with stochastical masking. The implemented multimodal approach of internal sensors outperforms a previous state-of-the-art model in action recognition by 32.8%. The implemented deep learning model benefits from combining vision, force, muscle activity, body pose, and sound. Lastly, we further explore the dependency between sensors by training a decoder to infer (generate) a missing modality. This decoder reconstructs tactile force with a mean error of $\pm 9.9\%$. In conclusion, our findings suggest leveraging various modalities to improve performance and robustness significantly, which can be further increased by stochastical masking.

Kurzfassung

Der Großteil der Forschung im Bereich Deep Learning konzentriert sich auf die Verarbeitung einer einzelnen Modalität, wie z.B. Bild, Ton, Text oder Bewegung. Im Gegensatz dazu nutzen Menschen tagtäglich verschiedene Sinne, um reichhaltige Informationen über die Umgebung aufzunehmen. In intelligenten Systemen kann kein einzelner Sensor alle Aspekte der Umgebung vollständig erfassen, weshalb die Integration verschiedener Sensortypen notwendig ist. Die Fusion von Sensormodalitäten ermöglicht ein tieferes Verständnis der Umgebung. Davon inspiriert, haben wir ein multimodales Modell auf Basis von Transformern zur Erkennung menschlicher Aktionen entwickelt und dessen Leistung und Robustheit umfassend evaluiert. Insbesondere konzentriert sich diese Arbeit auf die Untersuchung der Vorteile multimodaler Daten für Handlungserkennung in einer Küchenszene. Wir untersuchen das Verständnis des Modells und den Einfluss der Zusammenführung von Modalitäten durch eine systematische quantitative Analyse. Darüber hinaus ermitteln wir den Einfluss einzelnen Modalitäten anhand des Aufmerksamkeitsmechanismus unserer Transformer-Architektur und untersuchen verschiedene Fusionsmethoden, um herauszufinden, wie verschiedene Modalitäten am besten kombiniert werden können. Unsere Studie zeigt, dass multimodale Transformer besser abschneiden als ihre modalitätsspezifischen Äquivalenten. Der höchste Genauigkeitssteigerung im Vergleich zu der reinen Video-Baseline beträgt $+10,1\ \%$, was durch späte Fusion und stochastisch maskierte Training ermöglicht wird. Der implementierte multimodale Ansatz für interne Sensoren übertrifft ein vorheriges Modell, das dem Stand der Technik in Handlungserkennung entspricht, um $32,8\ \%$. Unser Deep-Learning-Modell profitiert von der Kombination von Video, Kraft, Muskelaktivität, Pose und Ton. Abschließend untersuchen wir den Zusammenhang zwischen den Sensoren, indem wir einen Decoder trainieren, eine fehlende Modalität zu generieren. Dieser Decoder rekonstruiert Griffkräfte mit einem mittleren Fehler von $\pm 9,9\ \%$. Zusammenfassend legen unsere Ergebnisse nahe, dass die Nutzung verschiedener Modalitäten die Leistung und Robustheit signifikant verbessert, was durch die Verwendung von stochastischen Masken weiter gesteigert werden kann.

Kivonat

A mély gépi tanulásban végzett kutatás döntő többsége egyetlen modalitás (érzékelő) feldolgozásával foglalkozik, mint például a kép, hang, szöveg vagy mozgás. Az emberek ezzel szemben egyszerre több érzékre támaszkodnak mindennapjaikban az alapos információszerzés végett. Az intelligens rendszerekben nincs olyan szenzor, amely egymagában képes lenne a külvilág minden részletét érzékelni, így azok összekapcsolására van szükség. A különböző érzékelők fúziója elősegíti a környezet részletgazdag megértését. Ebből ötletet merítve, egy transzformer alapú multimodális modellt tervezünk emberi tevékenységek felismerésére, aminek részleteiben elemezzük a teljesítményét és hibatűrését. Ezen szakdolgozat a multimodális adatok felhasználásának előnyeit vizsgálja ember által végzett konyhai feladatok terén. A módszer képességeit és a modalitások összefűzésének eredményességét rendszerezett kvantitatív kutatásnak vetjük alá. Ezenfelül megfigyeljük a különböző szenzorok szerepét a transzformer hálózaton belüli figyelem mértékének összehasonlításával, illetve különböző fúziós modelleket tárunk fel a modalitások hatékony összefűzésének felderítéséért. Tanulmányunk megmutatja, hogy a többérzékelős transzformerek felülmúlják az egyetlen érzékelővel ellátottakat. A legnagyobb pontosságban tett előrelépés a videóalapú felismeréshez képest $+10.1\%$, amit a véletlenszerűsített maszkolással betanított kései fúziós technika ér el. A belső szenzorokon alapuló modell 32.8%-kal túlteljesít egy, a közelmúltban korszerű tevékenység-felismerő neurális hálózatot. A megvalósított mély gépi tanulási módszer előnyt kovácsol a videó, az erőmérés, az izomműködés, a póz és a hang összefűzéséből. Végezetül egy hiányzó modalitás kikövetkeztetésére tanítjuk be dekóderünket, tovább kutatva a szenzorok összefüggését. Ezen dekóder $\pm 9.9\%$-os átlagos hibával számolja ki a hiányzó erőméréseket. Összefoglalva, a modalitások összekapcsolása a teljesítmény és hibakezelés jelentős előrehaladásához vezet, mely véletlenszerűsített maszkolással tovább növelhető.

Preface

Parts of this thesis were submitted to the 12th International Conference on Robot Intelligence Technology and Applications (RiTA 2024) as

Z. Varga, E. V. Mascaro, D. J. Sliwowski, and D. Lee, *"Multimodal Transformer Models for Human Action Classification"*.

Erklärung

Hiermit erkläre ich, dass die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt wurde. Die aus anderen Quellen oder indirekt übernommenen Daten und Konzepte sind unter Angabe der Quelle gekennzeichnet.

Die Arbeit wurde bisher weder im In- noch im Ausland in gleicher oder in ähnlicher Form in anderen Prüfungsverfahren vorgelegt.

Copyright Statement

I, Zoltán Varga, hereby declare that this thesis is my own original work and, to the best of my knowledge and belief, it does not:

- Breach copyright or other intellectual property rights of a third party.
- Contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
- Contain material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a university or other institution of higher learning.
- Contain substantial portions of third party copyright material, including but not limited to charts, diagrams, graphs, photographs or maps, or in instances where it does, I have obtained permission to use such material and allow it to be made accessible worldwide via the Internet.

Signature: _____

Vienna, Austria, September 2024 Zoltán Varga

Contents

Abstract iii

Abstract (German) iv

Abstract (Hungarian) v

Preface vii

1 Introduction 1

1.1 Arising need for combining modalities . . . 1

1.2 Problem definition . . . 2

2 Related Work 5

2.1 Human action recognition . . . 5

2.2 Action recognition with transformers . . . 6

2.3 The standard transformer . . . 6

2.4 Image and video transformers . . . 8

2.5 Masked Autoencoder . . . 9

2.6 Multimodal transformers . . . 10

2.7 Cross-modal inference . . . 11

3 Methodology 13

3.1 Dataset . . . 13

3.2 Human action recognition . . . 19

3.3 Model architecture . . . 19

3.4 Training the model . . . 25

4 Unimodal Action Recognition 29

4.1 Internal sensors . . . 29

4.2 External sensors 31
4.3 Discussion 33

5 Multimodal Action Recognition 35
5.1 Ablation study 35
5.2 The combination of tactile force and body skeleton pose 36
5.3 Multimodal model of internal sensors 38
5.4 The combination of tactile force and vision 39
5.5 Multimodal model using all five modalities 40
5.6 Masking the encoder 41
5.7 Modality inference 46
5.8 Discussion 46

6 Conclusion 49
6.1 Approach and results 49
6.2 Limitations 50
6.3 Further research 51

Bibliography 52

A Masked Autoencoder 57
A.1 Model architecture 57
A.2 Experiments 57
A.3 Discussion 58

List of Tables

3.1 Comprehensive list of human actions in ActionSense [1]. 20

3.2 Implementation details 27

4.1 Validation accuracies for subject S00 using different sampling frequencies for internal sensor data. In the last row, Δ denotes the difference between the two options. 31

4.2 Comparison of training the model on body joint orientation angles only or body and finger joint orientation angles. 32

5.1 Multimodal advantage of models from Section 5.2 to 5.5. All values in %. The model that achieves the highest accuracy is highlighted in bold, and the second highest is underlined. 44

1 Summary Tab for the classification metrics of Chapters 4 and 5. All values in %. The different cameras and microphones are noted in accordance with Sections 3.1.4 and 3.1.5. 56

A.1 Cross-validation accuracy and multimodal increase of masked autoencoders for a downstream classification task. All values in %. 58

List of Figures

2.1 The standard transformer from [2]. 8

3.1 Sample snippets of *Slicing cucumbers* as viewed from the muscle activity sensor and tactile force gloves. Source: [1] 15

3.2 Example of a body pose skeleton for *Spreading almond butter on a bread slice* 16

3.3 Three spectrogram snippets of performing the task of *Slicing potatoes*. These examples were recorded with the sink microphone. Light colours represent high intensity in frequency. 17

3.4 Placement of the environmental cameras in the recording space. M1-2 denote the microphones, and C1-5 the cameras. Records of the depth camera were not published. Source: [1] 17

3.5 Sample video frames of *Setting table*. Top: egocentric camera. Bottom, left to right: environmental camera No. 2 and 4. 18

3.6 Cropping and splitting a uniformly under-sampled video into patches where $x_{i,j}$ denotes the j^{th} patch of the segment's i^{th} frame. The displayed action is *Cleaning plate with a sponge*. 22

3.7 Overview of the multimodal transformer model using early fusion. Modules with trainable parameters are highlighted in blue. 24

3.8 Overview of the multimodal transformer model using late fusion. Modules with trainable parameters are highlighted with a light blue background. 24

3.9 Extending the model with a decoder to infer a selected modality. Modules with trainable parameters are highlighted with a light blue background. 26

4.1 Comparison to the baseline (LSTM) based on cross-validation accuracy. The results of the detailed classification task are displayed on the left and the general one on the right. 30

4.2 Example snippets of confusion matrices. Force modality (left column) can mostly recognise *Peeling* and *Slicing* tasks while it completely fails at *Fetching items*. The situation is reversed for the body skeleton modality (right column). 30
4.3 Unimodal classification accuracy of the external sensors. The dotted line represents the average of *internal* sensors. 32

5.1 Multimodal force and body skeleton models using different fusion methods compared to their unimodal counterparts and the baseline. 36
5.2 Confusion matrices for the general classification task comparing the models based on force and body skeleton. 37
5.3 Multimodal internal sensor models using different fusion methods compared to their unimodal counterparts and the baseline. 38
5.4 Multimodal models fusing tactile force and vision compared to the unimodal counterparts. 39
5.5 Fusion of all modalities. 40
5.6 Attention score of actions expressed as a percentage. The three highest values of each column are marked in bold. 42
5.7 Comparing the effect of different attention masks during training. R and V denote a Randomly chosen modality and the Video modality, respectively. 43
5.8 Attention score of each modality and the class token in late fusion. A vision mask changes the distribution of attention and forces the model to focus more on non-visual inputs. 44
5.9 Performance and robustness achieved by different masks in the training phase. R denotes a randomly selected modality. 45
5.10 Inferring tactile force $F(t)$ from other modalities (body skeleton $B(t)$, muscle activity $E(t)$, audio $A_{\text{Table}}(\tau)$ and video $V_{\text{Table}}(t)$). 46

A.1 Partially masked autoencoder. Modules with trainable parameters are highlighted with a light blue background. After pre-training finishes, the decoder is removed and the class token is fed through a linear layer, providing classification labels for the human actions. 59

Acronyms

CNN Convolutional Neural Network. 10

EMG Electromyography sensor. 14, 21, 38

FFT Fast Fourier Transform. 16

IMU Inertial Measurement Unit. 10, 11

LLM Large Language Model. 2

LSTM Long Short-Term Memory. 6, 14, 32, 33, 35, 39, 47, 50

MLP Multilayer Perceptron. 10

MML Multimodal Learning. 10

RNN Recurrent Neural Network. 6

RvNN Recursive Neural Network. 5, 6

ViT Vision Transformer. 6, 8, 21, 22

ViViT Video Vision Transformer. 8

Chapter 1

Introduction

As humans, we use multiple senses to interact with our environment. For instance, relying solely on our vision during a cooking task might become difficult. We must merge information from movement, sound, sense of touch, and vision to better understand and accomplish the task at hand. Despite the rapid growth of deep learning in the research of task understanding, most works focus on analyzing the environment and actions purely based on visual perception [3, 4]. They operate without considering different sensors available for humans, such as the forces involved when executing a task. Therefore, these works are limited to visible knowledge, which fails to translate some of the embodied intelligence into robots. For this thesis, we propose a Transformer model [2] to merge multiple modalities such as pose, muscle activity, tactile information, audio, and visual data to classify various human actions in a kitchen scenario. Transformer-based architectures have shown great performance for multimodal learning tasks due to the attention mechanism they are equipped with.

Approaching the topic from a biological point of view, humans have a total of five senses: sight, hearing, touch, taste, and smell. Some of them get prioritized depending on the situation and the objectives people use them for, while others get occasionally ignored. For various tasks in our daily lives, it is crucial to combine a handful of them and know which to focus on. The equivalent of senses in robotics is modalities, which are provided by various sensors. Although the available sensors for robotic systems often differ from the senses of humans, they fulfil the same role in understanding the environment and interacting with it.

1.1 Arising need for combining modalities

Most work in deep learning focuses on vision or text only. Visual data can be interpreted quickly and efficiently by humans, and it is also easy to acquire as cameras have become affordable and easy to use for the public. It can condense a vast quantity of information in a universal way; hence, a diverse range

of methodologies can work with it. Although it may be the most comprehending modality, vision alone is insufficient for many robotic applications and other autonomous systems. Since robots interact with their environment dynamically, it is necessary to obtain visually non-observable information besides that, for example, force and audio. With a force or tactile sensor, it can be easier to learn and precisely control the interaction forces with the environment. The incorporation of multimodal approaches may pave the way for novel research opportunities.

Training deep learning models requires a high amount of data. The internet provides an immense amount of audiovisual records, making large-scale training achievable without data collection. On the contrary, only a handful of multimodal datasets that contain modalities other than audio, vision, or text are available to the public. The majority of robotic applications employ sensors apart from these three modalities, raising the need to record new data and design models for this cause.

For instance, through vision only, models often fail to distinguish between tasks such as grasping and squeezing, as the main difference lies in the applied pressure. Certain actions cannot be recognized without proprioception, the sense of force, and body position. Proprioception plays a vital role in the correct execution of everyday tasks as well. Opening a jar or popping a bottle requires adequate force, which cannot be determined only through visual means. To this end, we consider that leveraging multimodal data that extends from vision and text data would improve the grounding of models such as Large Language Models (LLMs) with the real world.

1.2 Problem definition

We consider the problem of classifying the action a human is performing in a kitchen scenario based on multiple sensory data. In our work, we propose to study the feasibility and performance of recognizing human actions based on one or more modalities and observe if there is a benefit in using multiple data streams to better capture and translate the embodied information for robots.

In the next step, the model gets extended with a decoder that transforms the encoded data into another modality (e.g., video data into force) that can regress a modality from other ones. This could be used for humanoid robots, allowing them to learn from demonstrations and mimic human movements.

Research question

Does combining modalities improve the transformer model's understanding of human actions?

1. Does the classification improve compared to that of single modalities?
2. Can a multimodal transformer model distinguish classes that a unimodal one cannot?
3. Is it possible to extend the model with a decoder to infer a modality that is not part of the input data?

Chapter 2

Related Work

In this chapter, we review related studies in the field of machine learning. First, we show examples of human action recognition tasks and discuss which modalities were utilized. Then, the attention mechanism of transformers and the encoder-decoder structure are discussed in detail. Finally, we examine some relevant multimodal transformer models and compare them to our approach.

2.1 Human action recognition

Recognizing human actions is a crucial part of human-robot interactions. From a robotic point of view, it can be used in autonomous navigation systems [5], surveillance systems [6], and many other applications. Most works focus on RGB or grayscale videos, as humans mainly rely on perception when observing the actions of others. Visual modalities, especially RGB videos, have been shown to be very effective for this purpose in deep learning [3]. Besides vision, many other modalities can be utilized for action recognition, such as skeleton pose [7] and audio [8].

To understand human actions, it may be insufficient to observe the momentary state of the human in the environment. One needs to comprehend the evolution of the actions in time as well, which allows one to gain knowledge on temporally progressive tasks. For instance, by observing a human grabbing the fridge handle, one can not determine if the action is to close or to open the fridge. Understanding this sequence of motion primitives allows for excelling in the corresponding action recognition. Human motion and interactions with the environment must be leveraged to understand the action at hand.

Previous works for understanding actions proposed to model the sequential dimension of the data by updating a representation that encapsulates the prior information. A Recursive Neural Network (RvNN) [9] applies the same set of weights recursively on variable-sized inputs, to sum up the information more compactly. This calculation is repeated until a single representation is achieved. A widely known issue with RvNNs is the phenomenon of forgetting. At the end of a more extended sequence,

the information processed initially has little effect on the final result. To this end, Recurrent Neural Networks (RNNs) were developed to gain more control over forgetting. The default choice for a RNN, the Long Short-Term Memory (LSTM) [10], leverages the importance of the information stored in its memory state in each recurrence. This results in a more successful way of merging sequence entries into a common representation. Although an RvNN or an LSTM can recognize temporal progress [1], they often perform poorly for complex patterns.

2.2 Action recognition with transformers

There are numerous other methods capable of action recognition based on sequential data. The reason behind choosing the transformer [2] for this specific problem is that it has shown outstanding performance for sequence modelling in different modalities like natural language [11], video prediction [12], and image classification [13,14]. Although Vision Transformers (ViTs) [14] are bound to a single modality and are therefore unimodal, the way they process visual data and utilise spatio-temporal relations can also be directly applied in a multimodal approach. Transformer-based models aim to solve the limitations of prior approaches by using the attention mechanism to propagate information over time and learn the relationship between the short and long data sequences.

2.3 The standard transformer

The transformer is a sequence transduction model first described in [2]. Sequence transduction is a machine-learning problem where an input sequence is transformed into an output sequence that may or may not have the same length as the input. Early transformers were implemented for language modelling and machine translation with an autoregressive encoder-decoder structure. Later, they were extended for classification [13] and segmentation [14] as well, suggesting that the model can solve problems other than sequence transduction.

The core idea behind the transformer lies in the attention mechanism. The dot-product attention, described in [2], is defined according to Equation (2.2) where $Q \in \mathbb{R}^{t \times d_k}$ is the query matrix, $K \in \mathbb{R}^{t \times d_k}$ the key matrix and $V \in \mathbb{R}^{t \times d_v}$ the value matrix. The softmax function is applied row-wise, normalising the attention values to $[0, 1]$. The sizes d_k and d_v are independent hyper-parameters, and t is the number of tokens in the sequence. The keys, queries, and values are obtained by projecting input tokens $T \in \mathbb{R}^{t \times d}$ via dense layer (Equation (2.1)).

$$\begin{aligned} Q &= TW_Q \\ K &= TW_K \\ V &= TW_V \end{aligned} \tag{2.1}$$

$$\begin{aligned} \text{attention}(Q, K) &= \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right) \\ \text{head}(Q, K, V) &= \text{attention}(Q, K)V \end{aligned} \tag{2.2}$$

Instead of calculating Equation (2.2) once for each sequence, repeating it h times using different weights is beneficial. This technique splits the attention calculation, allowing the model to focus on more details. The sub-calculations are referred to as heads. Using multi-head attention, each head $i \in 1 \dots h$ has a unique linear projection layer with learnable weights for the inputs Q_i, K_i, and V_i. The resulting attention scores get concatenated and fed through an additional linear layer $L : \mathbb{R}^{d_l} \to \mathbb{R}^d$. The concatenation requires the dimensions to be consistent, meaning $d_l = hd_v$. For a fixed d_l, increasing the number of heads does not cause additional computational burden.

The standard transformer [2] has encoder-decoder architecture (Figure 2.1) and performs sequence-to-sequence modelling. On the other hand, only the encoder is needed for action recognition. The encoder has l sequential layers, each consisting of two blocks: a multi-head attention and a feed-forward block. Both of them have a subsequent residual connection and normalisation layers. The attention block of the encoder takes the same input for calculating the queries, keys, and values, making it a self-attention block. It focuses on the relations between tokens of the same input source.

The attention mechanism on its own does not take the order of tokens into consideration. One can easily see that by applying a permutation matrix π to the sequence of projected tokens $Q' = \pi Q$, $K' = \pi K$, and $V' = \pi V$, the same permutation appears in the output sequence. Considering Theorems 2.1 and 2.2, the corresponding output of the permuted tokens is derived in Equation (2.3).

Theorem 2.1. The softmax function is permutation invariant. $\text{softmax}(\pi X) = \pi\text{softmax}(X)$ as well as $\text{softmax}(X\pi) = \text{softmax}(X)\pi$.

Theorem 2.2. Permutation matrices are orthogonal. $\pi^{-1} = \pi^\top$

$$\text{head}(Q', K', V') = \text{softmax}\left(\frac{\pi QK^\top\pi^\top}{\sqrt{d_k}}\right)\pi V = \pi\text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)\pi^\top\pi V = \pi\text{head}(Q, K, V) \tag{2.3}$$

Although the permutation invariance of attention can be exploited for alternative models, e.g. the Set Transformer [15], it raises the need for adding spatial or temporal awareness when solving problems

in sequence transduction. Vaswani et al. [2] altered the encoder and decoder inputs using positional encoding to address this issue. The positional encoding is added to the input tokens, changing each of them differently. There are diverse methods to this end, some being learnable and others fixed. Many implementations use sinusoidal functions of different frequencies, resulting in unique encodings at each position.

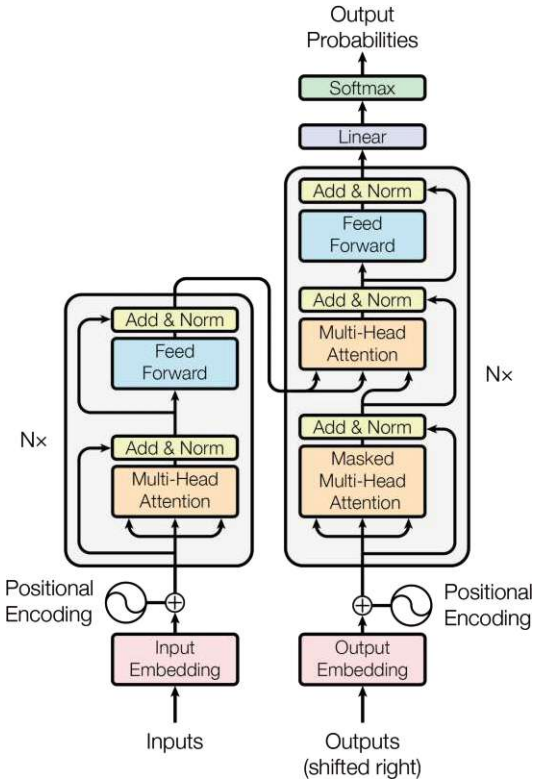


Figure 2.1: The standard transformer from [2].

2.4 Image and video transformers

ViT [12,14] is a transformer model specially developed for working with visual data. The process begins with splitting video frames into non-overlapping subsets called patches. This procedure reduces the size of a single item, simplifying the encoder and enabling it to learn feature recognition more efficiently. After that, the patches are flattened and prepended by a class token with the same dimension as the embedding of a patch. Spatial information is added to the sequence using positional embedding, the same way as it was introduced in the original Transformer [2]. This creates a spatial sequence of tokens that the transformer can directly work with. The class token of the last layer is trained to represent the entire image in one token.

In Video Vision Transformers (ViViTs), the data has temporal information, which in turn requires more complex modelling. Numerous methods mentioned in [12] utilise the embedded spatiotemporal

relations in videos. Each method aims to convert a given snippet of video data into a single token that compresses relevant information. This requires the attention mechanism to match information along the time axis with elements from the spatial axis. Due to the computationally expensive calculations of this interaction, reducing the number of paired tokens by separating (factorising) the temporal and spatial dimensions from each other is beneficial. This factorisation can happen at one of three encoding stages: at the encoder level, in the self-attention blocks, or when computing attention scores. Arnab et al. [12] have tested the computationally expensive spatiotemporal attention and three factorised models on the Kinetics-400 [16] dataset and found the factorised encoder to achieve the highest accuracy in classification while also being the quickest in the calculation.

To further reduce the high number of tokens, a so-called tubelet embedding can be implemented. This embedding uses a 3D convolution to extract information from sequential patches in time. It sums up several patches of fixed position and variable time into one patch. A more straightforward method for reducing the number of tokens is sampling the frames at a fixed interval. In our implementation, we choose the latter one.

2.5 Masked Autoencoder

Transformer encoders can be extended into an autoencoder by adding a decoder and training the model to reconstruct missing parts of the original input. Such autoencoders yield high-level feature extraction, which makes them excel in downstream classification tasks [17] as well. A vision-based Masked Autoencoder [17] has been found to require fewer epochs for training while outperforming similar methods. Its computational efficiency lies in using only a small subset of inputs at once during pre-training and having a relatively small decoder.

The Masked Autoencoder [17] is trained in two phases. During *pre-training*, a random subset containing patches of the original image is masked (i.e. removed). The remaining patches with added positional embeddings are encoded via a set of transformer blocks. Then, the encoded tokens are appended by mask tokens to represent missing data. Each mask token is a shared, learnable parameter vector that receives the same positional embedding as actual data. The decoder then reconstructs the missing tokens using tokens from the encoder. Finally, the last layer is a linear projection layer connecting the token space with the image space. After pre-training, the decoder is removed. The encoder is slightly modified to perform a classification task in the second training phase, called *fine-tuning*. Experiments show that the encoder can be frozen or trainable, the latter having improved recognition but requiring more time for training [17]. Its output (the class token or the average of all tokens) is probed with a new linear layer, fulfilling the recognition task.

Our approach for a masked autoencoder differs from that of [17] since reconstructing the original input would not be possible for some modalities.

2.6 Multimodal transformers

Recent studies have found different means of abstraction for multimodal problems. For instance, ImageBind [18] provides a shared embedding space across six different modalities, and as a result, it can perform cross-modal retrieval and zero-shot classification. It considers images, videos, text, audio records, Inertial Measurement Unit (IMU) data, temperature, and depth maps. The ImageBind model was trained on modality pairs consisting of a visual modality – image or video – and a second arbitrary modality. The resulting network can recognize the connection between emerging pairs by binding them through visual modality. For example, training the model on image-text and image-audio pairs enables it to perform zero-shot audio classification using text prompts, even though it was never explicitly trained on text and audio pairs. This ability is called emergent zero-shot classification.

Zhang et al. [19] improved upon existing works by proposing a Meta-Transformer that can handle a wide range of tasks across 12 modalities. They highlighted a *promising trend toward developing unified multimodal intelligence with a transformer backbone.* They emphasized the importance of simple networks such as Convolutional Neural Networks (CNNs) and Multilayer Perceptrons (MLPs) in data tokenization. Contrary to our approach, they froze the parameters of the multimodal module and trained the tokenizers only. The primary limitation of their model was the lack of temporal awareness. This means that the model cannot process a temporal sequence of tokens (e.g., a video) in its entirety, only individual frames. This makes the model ineligible for recognizing specific actions where temporal progress is crucial. Besides, embedding tokens for 12 modalities in parallel has caused a significant computational burden, making it difficult to scale the model. Our approach factorizes the encoder, which reduces computation and introduces an efficient form of spatio-temporal attention.

Multimodal neural networks differ in architecture regarding the exact way features of different modalities interact with each other. There is no consensus regarding the correct way of modality fusion, as a whole scale of various approaches is applicable [20, 21]. Xu et al. [20] indicated that *existing multimodal transformer models are superior only for specific Multimodal Learning (MML) tasks, as they are designed specifically for only a subset of specific tasks*, highlighting that the most significant challenge in universal modality learning is to find a level of generalisation that works for various inputs while retaining modality-specific modules.

2.7 Cross-modal inference

Observing the same action from different modalities carries a redundancy of information. This can be useful for inferring missing data using other modalities, i.e., sensors that do not directly measure the data of interest. In most prior works, IMU data is inferred from vision. This can be achieved by projecting the shared information of distinct modalities onto a shared manifold [22]. The manifold allows the model in [22] to understand the connection between modalities such as tactile force and video, achieving bidirectional interference. Similarly, humanoid pose and force can be inferred as well [23]. Our work differs from these models as it has multiple input modalities inferring a single output.

Cross-modal inference can appear in various robotic applications, including teleoperation [24], and it has been shown to improve object recognition as well [25]. Since gathering information from multiple input modalities enhances the capability of object recognition [25], we pursue the same technique for an action recognition task. Some prior works use zero-shot classification [18, 25] to show the benefit of multimodal models. As the dataset our model is trained on is relatively small, we do not aim for this. Instead, the accuracies are compared to that of a vision-based model to quantitatively measure the capabilities of multimodal models.

Chapter 3

Methodology

This section provides insight into the architecture of the models and the utilized dataset. First, we explain the data structure and visualize the different modalities. A precise data structure description is necessary to define the desired input shape and the conducted preprocessing. Then, we introduce the action recognition objective and discuss the identified tasks. We proceed with outlining the different architectures and give a detailed, reproducible description of the modules. This involves a comparison of fusion methods for classification, as well as an explanation of modality inference with a decoder. Finally, we discuss the training procedure and provide details about implementing our ablation study.

3.1 Dataset

We consider the ActionSense dataset [1] as it contains various sensory data like body and finger tracking, muscle activity, tactile, audio, and video data from different perspectives in a kitchen scenario where a person performs simple tasks. The reason behind choosing this particular dataset is that food preparation and corresponding household chores are some of the most desired topics in the field of automation and robotics. The modalities of [1] enable the transformer to capture the same tasks via diverse sensors and recording devices. We expect this to help the model find relations between modalities and thus improve its understanding of human actions.

DelPreto et al. [1] used carefully selected preprocessing mechanisms to create feature vectors from internal sensor data. This involves filtering out noise and resampling the data with a uniform sampling frequency f_s for all modalities except audio (See Section 3.1.4 to understand why audio data is treated differently). We follow the same procedure during the experiments to ensure a feasible baseline comparison.

At the time of writing, ActionSense contains a total of 5 subjects, most of which perform 20 different actions. Without calibration, the unsegmented recording for a single subject lasts around one hour. The

dataset is imbalanced regarding the length of segments in each class. While extracting feature vectors from raw data, over- and undersampling are applied to balance classes. The sampler imports a fixed number of segments n_{seg} of length T_{seg} for each class. The resulting dataset has an equal number of instances for each class and each subject. This way, the classification accuracy is more reliable as a metric for recognizing actions since all actions are prioritized equally.

The five modalities can be split into two categories. The human pose sensor, the muscle activity sensor, and the tactile force sensor are internal sensors collecting data about the subject's state. The other group collects information about external observations: audio and video data characterize actions in the environment and the state of surrounding objects.

In the publication [1] about the previously mentioned dataset, the authors have trained an LSTM on some of the acquired data (force, Electromyography sensor (EMG), and body skeleton). Their approach serves as a baseline for our metrics. Since there was a significant error in their validation, the baseline results of the LSTM are recalculated here and do not match the results reported in [1].

3.1.1 Tactile force sensor data

The ActionSense dataset [1] introduces a custom tactile sensor. This sensor measures the force applied to the hands of the subject. The features resulting from a single measurement are stored in a 32×32 matrix for both hands. Following [1], the matrices are aggregated to reduce their high dimensionality. The matrix gets split into 8×8 sub-matrices for the aggregation. Each sub-matrix is represented by the mean of its elements, resulting in a scalar. By stacking the aggregated scalars and concatenating left and right sides, we create feature vectors of length 32, formalized in Definition 3.1. They are normalized to $[-1, 1]$ to enhance interpretability for our network.

$$F(t) \subset \left\{ t \to \mathbf{x} \in \mathbb{R}^{32} \mid \mathbf{x} = (x_i),\ x_i \in [-1, 1] \right\} \tag{3.1}$$

3.1.2 EMG data

The dataset uses [1] Myo armbands to record muscle activity. Each armband consists of eight EMG sensors and is placed on each arm of the subject. EMG sensors characterize muscle activity as voltage. Similarly to other modalities, we follow the practices described in [1] to create feature vectors from the voltage. First, the absolute value is taken to represent muscle activity. Then, noise is filtered out with a cutoff frequency of $f_c = f_s/2$ using a 5^{th} order Butterworth filter. The filtered data is resampled with f_s to stay consistent with the sampling theorem [26]. Similarly, as with the force sensor, the feature vectors (Definition 3.2) are normalized and concatenated. Some examples of muscle activity along with the force sensor are visualized in Figure 3.1. Note that Figure 3.1 shows the overall activation since

plotting all vector elements separately would be unintelligible to humans. The overall activation is calculated by taking the vector norm of the features and normalizing it to $[0, 1]$ along the time axis.

$$E(t) \subset \{t \to \mathbf{x} \in \mathbb{R}^{16} \mid \mathbf{x} = (x_i),\ x_i \in [-1, 1]\} \tag{3.2}$$

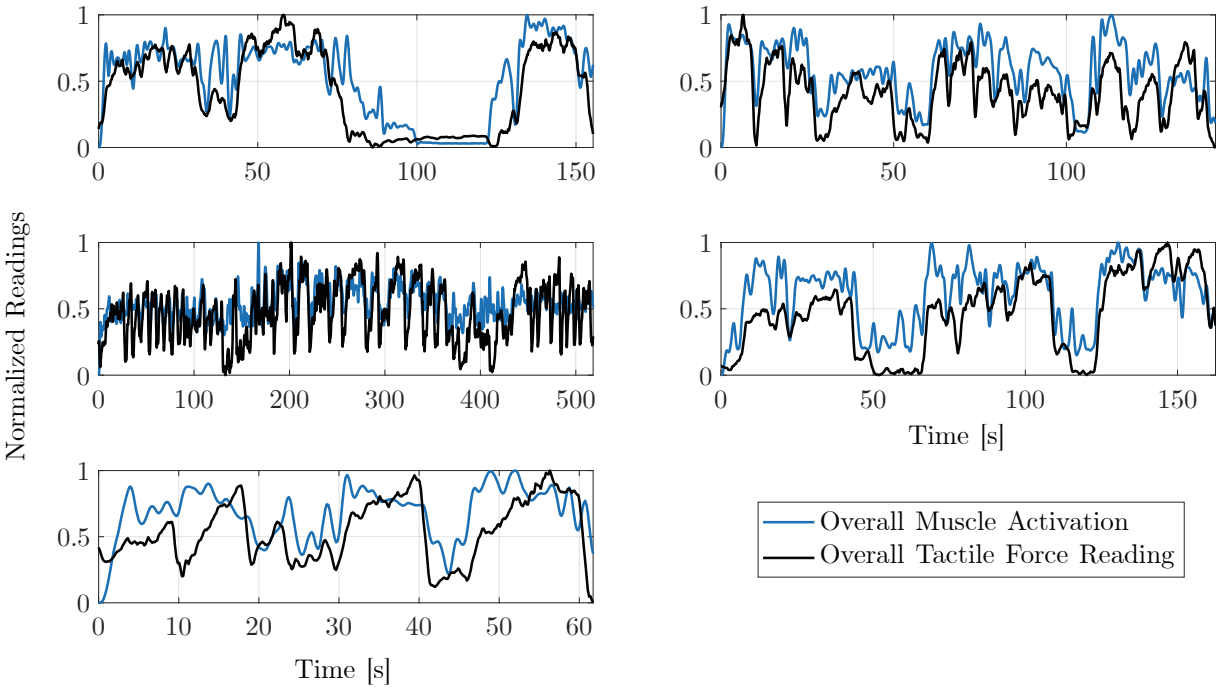


Figure 3.1: Sample snippets of *Slicing cucumbers* as viewed from the muscle activity sensor and tactile force gloves. Source: [1]

3.1.3 Body tracking sensor data

The Xsens body tracking sensor records the orientation of 22 body joints. Each orientation is represented by a rotation angle around the x, y, and z axes in degrees from $-180°$ to $180°$. The measurements are resampled with f_s and normalized to lay between -1 and 1. The joints are visualized in Figure 3.2 to gain a better understanding. Considering that the exact anatomy (relative location of the segments and their length) has to be defined to create a skeleton image from joint angles, the data used for visualization is not equivalent to the actual features. The anatomy is only added to the figure.

In addition to the body joint orientations $B(t)$ (Definition 3.3a), the dataset also contains records of finger joints (Definition 3.3b). This part of the data has the same angle format as the body tracking sensor but was recorded with a different device. The orientation of 19 joints per hand was recorded with a Manus finger tracking tool.

$$B(t) \subset \{t \to \mathbf{x} \in \mathbb{R}^{66} \mid \mathbf{x} = (x_i),\ x_i \in [-1, 1]\} \tag{3.3a}$$

$$B_F(t) \subset \{t \to \mathbf{x} \in \mathbb{R}^{180} \mid \mathbf{x} = (x_i),\ x_i \in [-1, 1]\} \tag{3.3b}$$

Figure 3.2: Example of a body pose skeleton for *Spreading almond butter on a bread slice*

3.1.4 Audio data

The dataset has two mounted environmental microphones, shown in Figure 3.4; one is located at the sink (M2) and the other over the table (M1). We include both of them in our ablation study.

As raw audio data has a significantly higher sampling rate ($f_{s,\text{Audio}} = 16\,\text{kHz}$), we first transform it to the frequency domain using mel-scaled filter banks [27]. The resulting time-frequency distribution represents the information more compactly by converting the waveform from the time domain to a two-dimensional feature space. To this end, n_f bandpass filters are applied to the waveform independently, resulting in n_f sub-banded signals. These signals get divided up using Hamming windows [28] at fixed time interval t_H. A Fast Fourier Transform (FFT) is applied to the segments, resulting in a spectrum for each frequency bin. Finally, the spectra are converted to intensities by summing over their absolute values. The resulting spectrograms have two dimensions, one representing the time and one the frequency axis. Hyper-parameters of the data loader are set following ImageBind [18]. Hence, the imported segments are divided into sub-segments of length T_{IB}, resulting in fewer data points per segment than other modalities. The spectrograms are normalised to zero mean $\overline{\mathbf{x}} = 0$ and unit variance $\sigma_{\mathbf{x}} = 1$ (see Definition 3.4). Some examples of a spectrogram are displayed in Figure 3.3.

$$A_{\{\text{Sink, Tab}\}}(\tau) \subset \{\tau \to \mathbf{x} \in \mathbb{R}^{128} \mid \overline{\mathbf{x}} = 0,\ \sigma_{\mathbf{x}} = 1\}, \quad \text{where } \tau = \lfloor t/10\,\text{ms} \rfloor \tag{3.4}$$

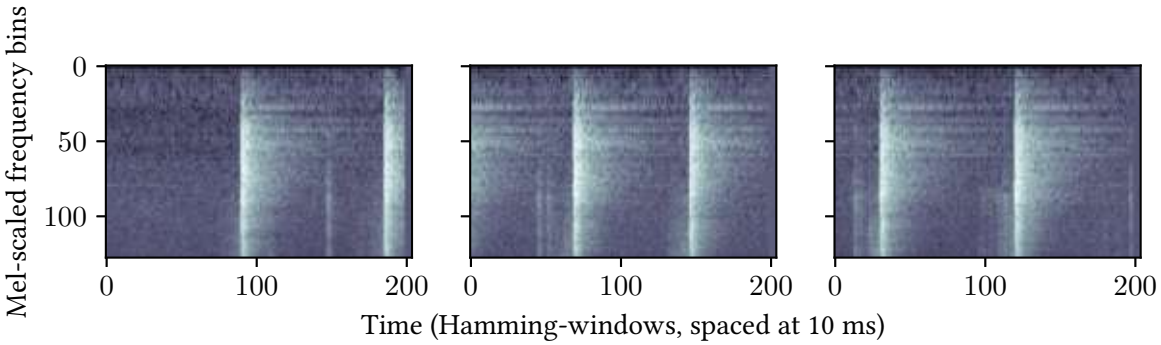


Figure 3.3: Three spectrogram snippets of performing the task of *Slicing potatoes*. These examples were recorded with the sink microphone. Light colours represent high intensity in frequency.

Figure 3.4: Placement of the environmental cameras in the recording space. M1-2 denote the microphones, and C1-5 the cameras. Records of the depth camera were not published. Source: [1]

.

3.1.5 Video data

The dataset contains the recordings of one egocentric camera placed on the subject's head and five environmental cameras around the kitchen (see C1-C5 in Figure 3.4). We only consider the cameras C2, C4, C5, and the egocentric camera for our experiments. Cameras C1 and C3 are discarded as they have similar viewing angles as C2 and C4 and would not provide new information. Camera number 5 is referenced as the *Table camera* (Tab), as it records the kitchen table from above. Example images are shown in Figure 3.5. Some videos of the egocentric view have a green dot representing the human's gaze overlaid on the video footage, while some lack this information. We choose the one without gaze overlay in cases where both versions are available.

All videos are sampled down to 5 frames per second to reduce computation time and storage space. The actions considered do not have any highly dynamic movements, and we do not oversee any loss of information caused by this transformation. The feature tensors $V_{\text{Ego}}(t)$, $V_{\text{C2}}(t)$, $V_{\text{C4}}(t)$ and $V_{\text{Tab}}(t)$ are described by Definition 3.5.

$$V_{\{\text{Ego, C2, C4, Tab}\}}(t) \subset \{t \to \mathbf{x} \in \mathbb{R}^{224\times224\times3}\} \tag{3.5}$$

Figure 3.5: Sample video frames of *Setting table*. Top: egocentric camera. Bottom, left to right: environmental camera No. 2 and 4.

3.2 Human action recognition

The performed activities of [1] were carefully selected to contain simple and complex tasks with various objects. For example, *Spreading jelly* or ∼ *almond butter on bread* are simple actions for which the movement is similar but can be performed with two different objects. On the other hand, *Setting the table* and *Stacking plates on the table* are more complex tasks involving the same items, where the only difference lies in the placement of the dishes.

The dataset makes it possible to distinguish labels on two levels, resulting in 20 or 10 labels. We call them detailed and general actions, respectively. Recognising the detailed actions requires understanding the time evolution and the objects involved. In this case, *Loading the dishwasher* and *Unloading* ∼ are separate classes. By looking at a single frame of the record, it is impossible to distinguish these two. One needs to consider at least a few seconds of the recording to observe how the action advances and see whether the dishes go in the dishwasher or are being taken out. A complete set of the classification labels is listed in Table 3.1. The abbreviated code for each is noted in parentheses.

Among detailed actions, there are four cleaning-related tasks, as the labelling distinguishes two tools – sponge and towel – and two objects – plate and pan. Both have to be recognised accordingly for a correct classification, which can be difficult for some modalities. For instance, using a skeleton pose only makes it challenging to see whether the object being held is a plate or a pan. To simplify classification, a more generalised listing of human actions is defined. These labels no longer distinguish the objects; the focus shifts to the movement itself. Using general classification reduces the number of distinct labels to half. The corresponding general label can be easily found by removing the digit from the code of the detailed label.

Using the segmentation described in Section 3.1, the training split consists of 1580–1600 and the validation split of 380–400 items[1]. Cross-validation enables all 1990 segments to contribute to the final metrics, yet it is still only a fraction of the dataset size other transformer models were trained on. Therefore, we expect modalities enhanced with a backbone to overfit less and perform significantly better than those where all parameters were trainable.

3.3 Model architecture

Multimodal models have to be designed to be compatible with different modalities. The first step is to bring feature vectors of various shapes closer together by creating a uniform token format. These tokens then get fused together using a transformer encoder that considers both the time evolution and modality-specific information embedded in the tokens. The fusion can be carried out at different levels,

[1]The varying size is due to subject *S00* not performing the task of *Clearing the cutting board*.

№	Detailed action	№	General action
1.	Peeling cucumber (P1)	1.	Peeling (P)
2.	Peeling potato (P2)		
3.	Slicing cucumber (S1)	2.	Slicing (S)
4.	Slicing potato (S2)		
5.	Slicing bread (S3)		
6.	Clearing cutting board (CB)	3.	Clearing cutting board (CB)
7.	Spreading almond butter on a bread slice (B1)	4.	Spreading on bread (B)
8.	Spreading jelly on a bread slice (B2)		
9.	Opening and closing a screw-top jar (JA)	5.	Open/close jar (JA)
10.	Pouring water from a pitcher into a glass (WA)	6.	Pouring water (WA)
11.	Cleaning plate with sponge (C1)	7.	Cleaning (C)
12.	Cleaning plate with towel (C2)		
13.	Cleaning pan with sponge (C3)		
14.	Cleaning pan with towel (C4)		
15.	Getting items from refrigerator, cabinets or drawers (G1)	8.	Fetching items (G)
16.	Getting items from cabinets: 3 each large/small plates, bowls, mugs, glasses, sets of utensils (G2)		
17.	Setting table: 3 each large/small plates, bowls, mugs, glasses, sets of utensils (T1)	9.	Table tasks (T)
18.	Stacking on the table: 3 each large/small plates, bowls (T2)		
19.	Loading dishwasher: 3 each large/small plates, bowls, mugs, glasses, sets of utensils (D1)	10.	Dishwasher tasks (D)
20.	Unloading dishwasher: 3 each large/small plates, bowls, mugs, glasses, sets of utensils (D2)		

Table 3.1: Comprehensive list of human actions in ActionSense [1].

of which two are considered in this case: early and late fusion. The exact way of modality fusion is discussed in Section 3.3.4. Finally, the encoder's output is fed through a linear layer responsible for the classification objective.

While the main objective remains to assess the use of multimodal data, the model can also work with unimodal inputs. In this case, modality fusion is dropped while other modules remain unchanged, including all hyperparameters and embedding sizes. The ablation study of the unimodal model is discussed in Chapter 4, while Chapter 5 evaluates the use of multimodal data.

3.3.1 From feature vectors to tokens

The original Transformer described in [2] is implemented mainly for textual inputs. Further steps are necessary to extend it to the various modalities of [1]. Since the input tokens of the multimodal encoder must have the same embedding size, the uniformly sampled feature vectors of the sensor data get projected into a new vector that has an equal shape among all recording devices.

Internal sensor features – namely body skeleton pose, tactile force, and EMG data – are projected into an embedding of size d using a linear layer. This embedding size is chosen to be compatible with the visual modality, as the transformer encoder requires all inputs to have the same dimension.

Unlike internal sensors, audio data needs special preparation. Each segment of the audio spectrogram (see Figure 3.3) gets split into patches of a given shape P_{IB}. The resulting patches get encoded using the ImageBind [18] multimodal transformer model as the backbone. Out of the seven modalities, only the audio processing module is taken. ImageBind [18] was trained on Audio Set [29], a large-scale dataset of manually annotated audio events. They utilised one of its subsets containing $4,971$ hours of audio records. They segmented data into snippets of length T_{IB}, and we follow the same practice to stay consistent with that. The encoded tokens have a different dimension than those of other modalities. Therefore, a linear projection layer $d_{\text{IB}} \to d$ is applied to the data before encoding it again in the multimodal module.

Besides the audio encoder, ImageBind [18] also has a module for visual data, but it has 632 million parameters, which would increase model overfitting more than smaller models. Instead, we use the DINOv2 [30] for the backbone of the visual modality. It is a robust feature extractor trained on the ImageNet dataset of $14,197,086$ images. Internally, the model functions as a ViT. Therefore, its structure is highly similar to the rest of our model. The backbone implementation already includes an algorithm that splits each video frame into patches, which we do not modify. The only preprocessing is cropping and resizing the frames to satisfy the default DINOv2 [30] configuration and normalising them to ImageNet's mean and standard deviation. This modality does not need a projection layer, as

the token size d of the multimodal encoder was set to have the same embedding size as the ViT. A visualisation of how patches are extracted out of video footage is shown in Figure 3.6.

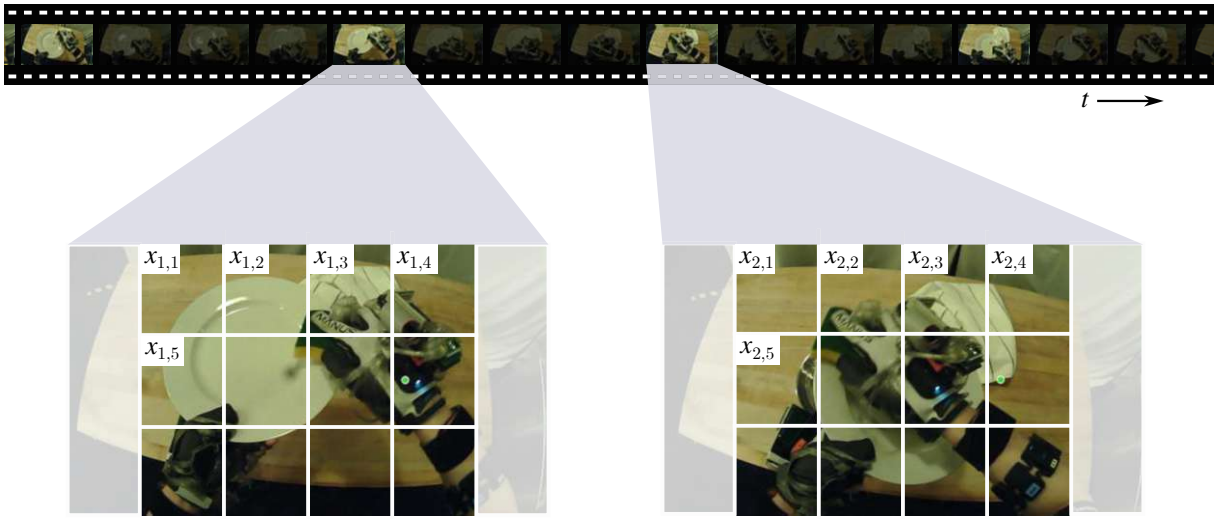


Figure 3.6: Cropping and splitting a uniformly under-sampled video into patches where $x_{i,j}$ denotes the j^{th} patch of the segment's i^{th} frame. The displayed action is *Cleaning plate with a sponge*.

We do not optimise the weights of ImageBind and DINOv2. This reduces the number of learnable parameters and makes the model less prone to overfitting. Using pre-trained backbones speeds up the training and reduces computational burden during optimisation.

3.3.2 Transformer encoder

After creating tokens from raw data, they get passed to the encoder. As mentioned in Section 2.3, the encoder requires token embedding to preserve positional information due to the permutation-invariant calculations in the attention module. Therefore, a fixed, sinusoidal embedding is applied before processing a temporal sequence of tokens. Learnable embeddings are additionally used if the sequence contains tokens of different modalities. Embedding tokens is achieved by tensor addition, which is a commutative operation. Different types can combined by applying them sequentially, regardless of execution order. Combining embeddings this way alters each token uniquely, depending on the modality and the timestamp it represents.

Before the actual encoding, the token sequence gets prepended with a class token. The class token's purpose is to summarise the information from the sequence of tokens. For the encoder, we use l layers of attention blocks equipped with h self-attention heads [2] to capture and understand the relationships between the input features. The output of each encoder is the class token; other tokens are discarded.

3.3.3 Masking the encoder

The attention scores described in Section 2.3 correspond to the relevance of a token in the sequence at hand. The attention score can be forced to 0% by setting the key-query pair to negative infinity before applying the softmax function. This process, named masking, can be helpful in various cases. It makes the model focus more on the remaining tokens and acts as a powerful form of regularisation if applied stochastically. Our implementation generates masks separately for each element of the batched input. This mask then gets passed to the attention blocks of each head and each layer in the encoder. The masks are added to the query-key product before applying the softmax function.

3.3.4 Modality fusion in the encoder

The art of combining different modalities is a long-standing challenge in deep learning communities. Here, we consider two different procedures for fusing tokens of separate modalities in the same attention block. On one hand, in *early fusion*, the temporal sequence of each modality gets stacked together before encoding. This happens immediately after creating tokens from features and equipping them with modality and time embedding. This fusion method has only one class token, summarising information along the time axis and among modalities. After adding the class token, the sequence consists of $m \cdot f_s \cdot T_{\text{seg}} + 1$ tokens where m denotes the number of modalities. A block diagram of this fusion method can be seen in Figure 3.7.

On the other hand, in *late fusion*, each modality has a separate encoder. The first block of encoders processes information along the time axis, focusing on one modality at a time. Each of the m encoders works with $f_s \cdot T_{\text{seg}}$ tokens and an additional class token that is not shared among modalities. After the first block of encoders, only the class tokens are passed to the final block. They are stacked into a sequence where modality embeddings and a new class token are added. The second block consists of a single encoder. An overview of the multimodal transformer model in this configuration is displayed in Figure 3.8.

3.3.5 Classification layer

The encoder alone does not provide classification labels. To this end, the encoded class token gets fed through a final linear layer followed by a softmax activation function that provides a confidence score for each class. Each prediction sums up to 1, and the score with the highest value is the predicted class. One item results in a tensor of size N_C, where N_C denotes the number of classes.

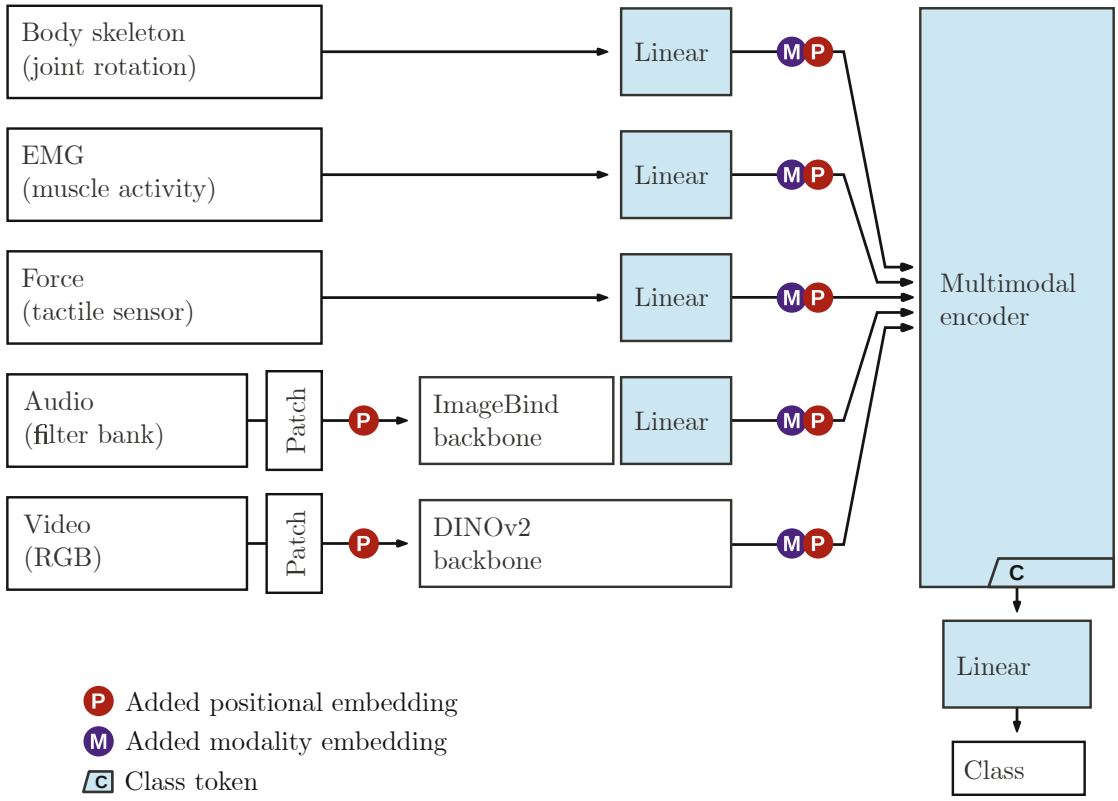


Figure 3.7: Overview of the multimodal transformer model using early fusion. Modules with trainable parameters are highlighted in blue.

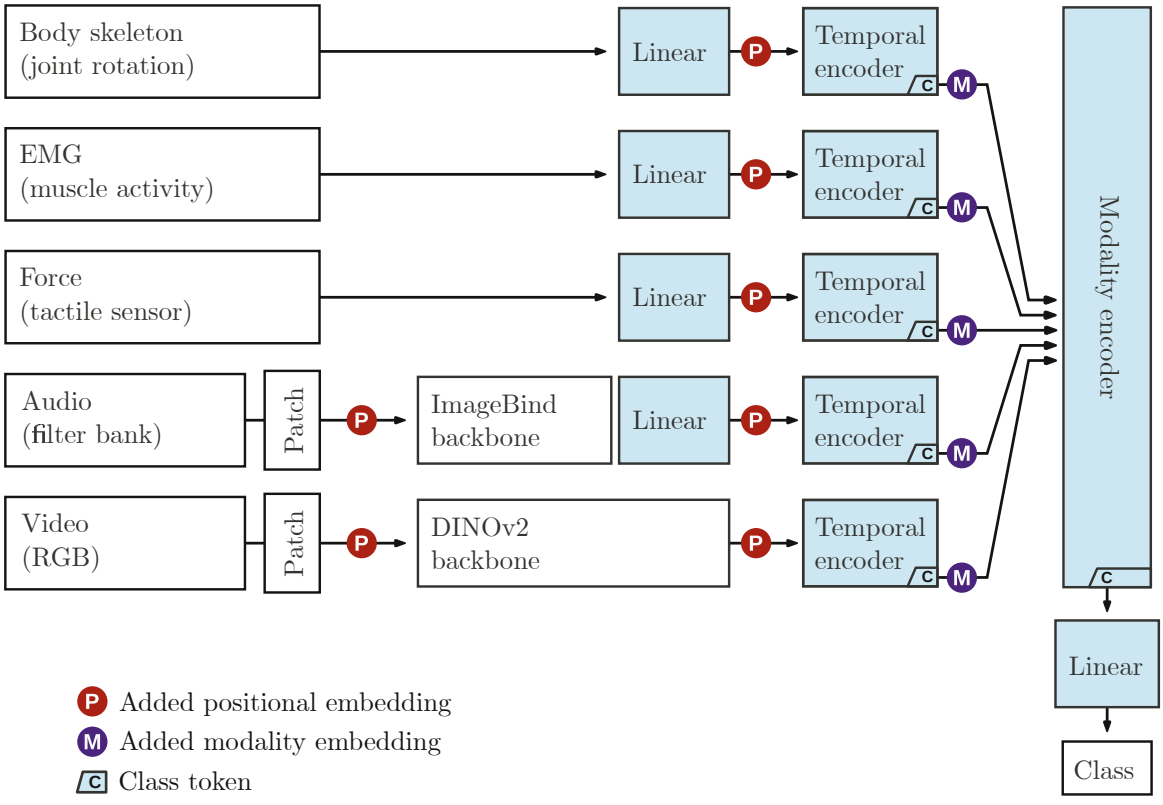


Figure 3.8: Overview of the multimodal transformer model using late fusion. Modules with trainable parameters are highlighted with a light blue background.

3.3.6 Decoder

Different modalities view the same action from different feature spaces and, therefore, correlate with each other. This particularly applies to internal modalities: muscle activity always results in force, and physical force requires muscle movement. The redundancy can be used to infer (reconstruct) a selected modality from other modalities.

Modality inference is a self-supervised sequence modelling task. Most implementations for this problem use an encoder-decoder structure. The encoder extracts relevant information from the input features, creating a new representation in a latent space. The decoder then takes tokens from this latent space and outputs features of the desired modality (in this case, tactile force). Both the encoder and the decoder consist of attention blocks [2], although the decoder is usually shallower [17] (has fewer layers).

Our early fusion encoder receives tokens from different modalities. Therefore, both modality and positional embedding are applied. These tokens are forwarded to the decoder after l layers of encoding. Additionally to the encoder outputs, the decoder receives another set of tokens, namely the mask tokens. After decoding, these tokens represent the inferred modality. The decoder aims to match these tokens to the corresponding timesteps determined by the positional embedding. The embedding of the decoder is identical to that of the encoder. A linear layer ensures the output has the exact dimensions and value range as the original input features. The detailed structure of the model is displayed in Figure 3.9.

3.4 Training the model

The model is trained using the AdamW optimiser [31] in PyTorch. We use a cosine learning rate scheduler starting at η_0 with $n_{e,w}$ warm-up epochs and descending to η_{end} following a sinusoidal curve. The default configuration is set up without applying regularisation techniques. Data gets loaded in batches of size B. We choose the cross entropy formula as the loss function. Since the dataset is balanced regarding the number of instances in each class, we do not enforce an additional balancing strategy in our loss terms. The training is performed with n_e epochs. Details about the training implementation are given in Table 3.2.

Classification metrics are obtained using cross-validation. Out of the five complete subjects in the dataset, four are used for training, and the remaining one is held out for the validation set. This training process is repeated for all subjects, with a different validation subject each time. The resulting metrics are averaged to avoid bias caused by the subjects. The model with the lowest loss is selected to be the final model of the training session.

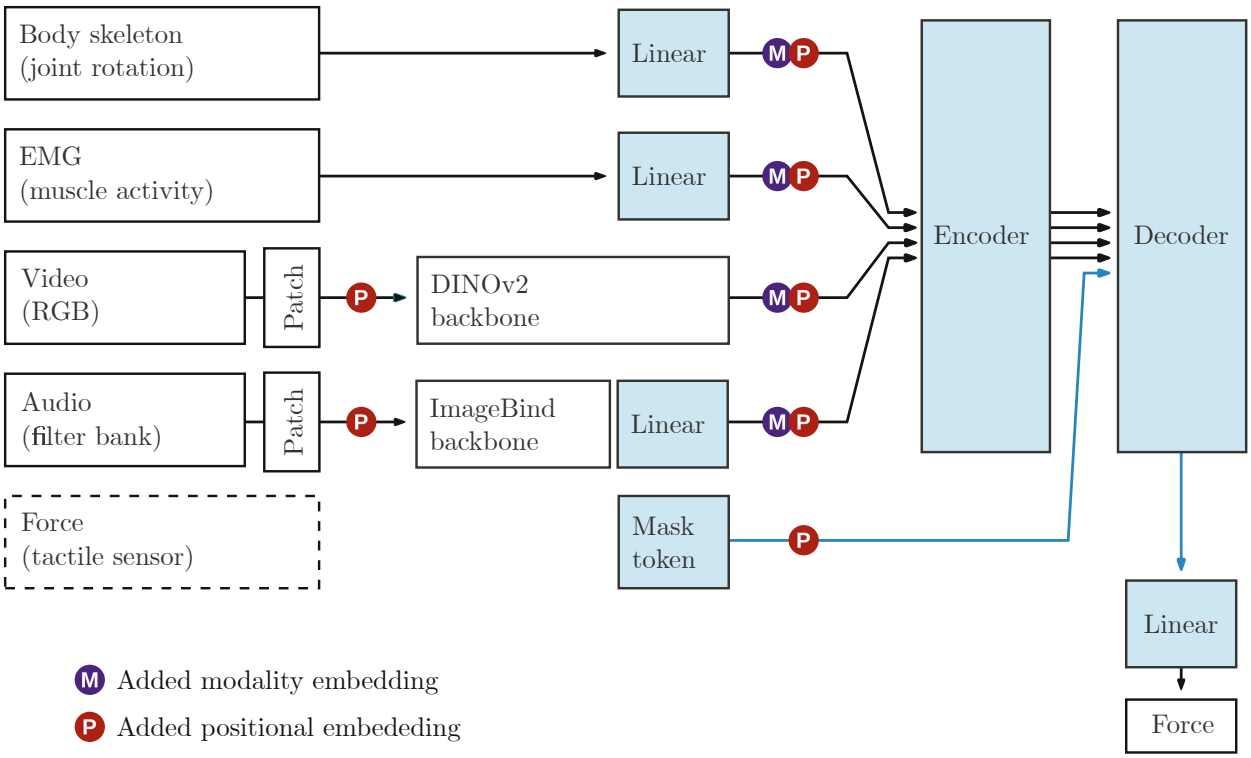


Figure 3.9: Extending the model with a decoder to infer a selected modality. Modules with trainable parameters are highlighted with a light blue background.

At first, unimodal (single-modality) models are trained, allowing us to study action recognition independently. These models are only for the sake of evaluation. Similarly to unimodal models, multimodal ones are trained from scratch, allowing them to learn a different feature extraction strategy from the unimodal ones.

Our early fusion model has 7.45 million of learnable parameters, and the late fusion model has 42.9 million. The DINOv2 and the ImageBind modules have 22.1 and 85.4 million parameters, respectively, which remain frozen.

Data loader hyper-parameters		
Number of segments	n_{seg}	20
Segment length	T_{seg}	10 s
ImageBind snippet length	T_{IB}	2 s
ImageBind patches	P_{IB}	16×16 with an overlap of 6
Filter bank frequency bins	n_f	128
Hamming window spacing	t_H	10 ms
Model hyper-parameters		
Uniform token size	d	384
ImageBind token size	d_{IB}	768
Number of attention layers	l	4
Number of attention heads	h	4
Training hyper-parameters		
Moving average parameters	β	$(0.9, 0.999)$
Constant for numerical stability	ϵ	10^{-8}
Number of epochs	n_e	20 for classification, 10 for inference
Warm-up epochs	$n_{e,w}$	5 for classification, 1 for inference
Initial learning rate	η_0	10^{-5} for classification 10^{-4} for inference
Final learning rate	η_{end}	$\eta_0/100$
Batch size	B	16

Table 3.2: Implementation details

Chapter 4

Unimodal Action Recognition

Instead of training the multimodal transformer model with all data at once, it is practical to first look into the results for single modalities and check for feasibility. This serves as a comparison base and provides information about the data itself. If one of the validation metrics does not get significantly better than random guessing, it indicates that the model cannot find learnable patterns in it.

Unimodal classification also gives an insight into the capabilities of transformer models, indicating the complexity of the different action granularities and modalities. We compare our results to the baseline, which uses an LSTM model for temporally aggregating the features, to showcase the benefit of a transformer model.

4.1 Internal sensors

The baseline and our model are trained and evaluated using the same data loader for a fair comparison. The transformer model outperforms the ActionSense baseline in all of its modalities. It shows an accuracy higher by 8.7% in the detailed and 10.8% in the general classification for the internal sensor modalities displayed in Figure 4.1. The most significant difference between the two models appears in the force modality in both classification objectives.

By examining confusion matrices, one can find the strengths and weaknesses of each modality. For example, the force modality transformer is better at recognising objects in *Peeling* and *Slicing* tasks; meanwhile, the body pose is suited better for actions where the subject is *Fetching items* or *Setting table*. This reinforces our motivation that a combination of multiple modalities is needed to exploit data to its full potential. Some subsets of the confusion matrices are displayed in Figure 4.2.

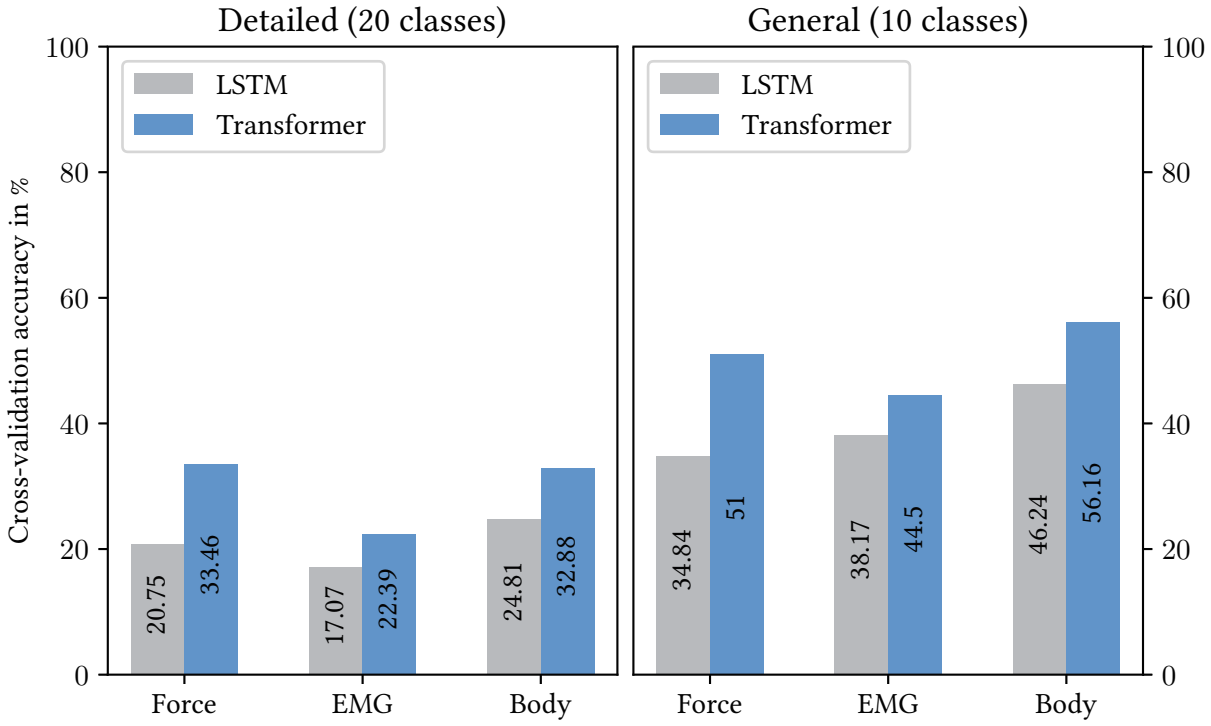
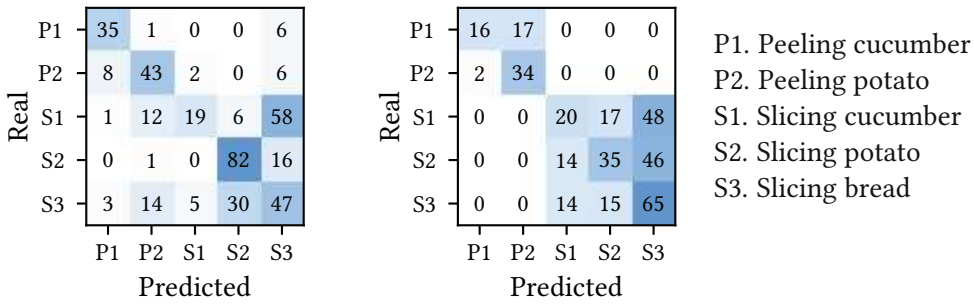
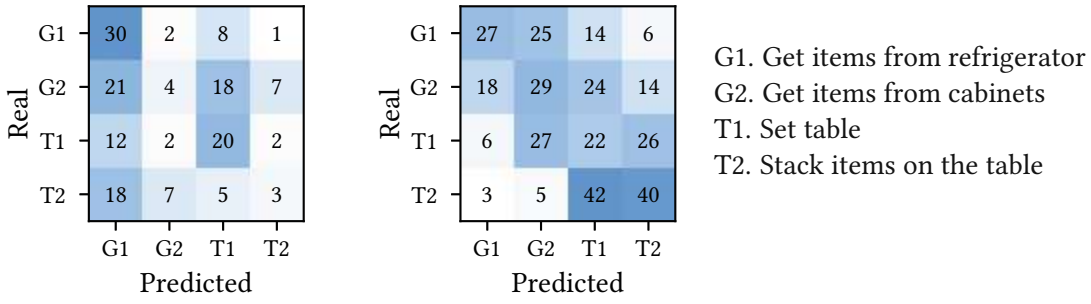


Figure 4.1: Comparison to the baseline (LSTM) based on cross-validation accuracy. The results of the detailed classification task are displayed on the left and the general one on the right.



(a) Force (peel and slice) (b) Body (peel and slice)



(c) Force (fetching) (d) Body (fetching)

Figure 4.2: Example snippets of confusion matrices. Force modality (left column) can mostly recognise *Peeling* and *Slicing* tasks while it completely fails at *Fetching items*. The situation is reversed for the body skeleton modality (right column).

4.1.1 Common sampling frequency

The baseline [1] implementation uses a sampling frequency of $f_s = 10$ Hz for the internal modalities. Before introducing the external modalities to the model, a simple test is conducted to see whether the sampling frequency can be decreased without causing data loss.

Training transformer models is a computationally expensive procedure. Decreasing sampling frequency reduces input data size proportionally and spares computation time. Since the involved kitchen tasks do not contain high-speed movement, the sampling frequency was reduced from 10 Hz to 5 Hz. The classification objective was tested for all internal sensors to show that it does not filter out significant information from sensor data.

For the force and body pose models, reducing sampling frequency has improved detailed action recognition by 1–5% and general by 3–7% (See Table 4.1 for exact values). This phenomenon is likely attributable to dimensionality reduction achieved by downsampling. Consequently, the model overfits less, causing validation accuracy to increase.

Table 4.1: Validation accuracies for subject S00 using different sampling frequencies for internal sensor data. In the last row, Δ denotes the difference between the two options.

f_s	Detailed accuracy in %			General accuracy in %		
	Force	EMG	Body	Force	EMG	Body
10 Hz	32.11	34.47	44.08	50.79	59.08	77.50
5 Hz	36.97	32.50	45.66	57.89	55.79	80.66
Δ	+4.86	−1.97	+1.58	+7.10	−3.29	+3.16

4.1.2 Body and finger skeleton pose

As mentioned in Section 3.1.3, the dataset [1] also contains records of finger joint orientations. These were excluded in the baseline calculation and added only for this section. The results of this test are displayed in Table 4.2. Since adding them made validation accuracy drop by 2–5% and only caused overfitting, finger joint orientation data was removed from later tests. Further experiments are carried out using information about body joints only.

4.2 External sensors

External sensors of the dataset [1] involve microphones for recording audio signals, cameras mounted at fixed positions in the environment, and an egocentric camera placed on the subject's head. Unlike wearable sensors from Section 4.1, there are no published results at the time of writing that could serve

Table 4.2: Comparison of training the model on body joint orientation angles only or body and finger joint orientation angles.

Skeleton type	Detailed accuracy in %		General accuracy in %	
	Training	Validation	Training	Validation
Body only	83.49	32.88	93.77	56.16
Body and finger	89.59	30.82	94.64	51.00
Δ	+6.10	−2.06	+0.87	−5.16

as a baseline. Since implementing an LSTM for these modalities is out of the scope of this thesis, the only comparison basis is provided by the metrics of other modalities.

Of the four cameras our ablation study involves (Figure 3.4), the egocentric camera has the best view for recognising actions. It was followed by the C4 and the table camera with roughly the same accuracy. Although the egocentric camera has high potential, it is not eligible for multimodal experiments. Firstly, it has been found to have a ±2 s delay compared to other modalities. This is due to an issue that the authors of the dataset did not record timestamps for the egocentric camera. Secondly, as the egocentric camera has high accuracy on its own (95% for recognising general actions), there is little room for improvement. The cross-validation accuracy of external sensors is displayed in Figure 4.3.

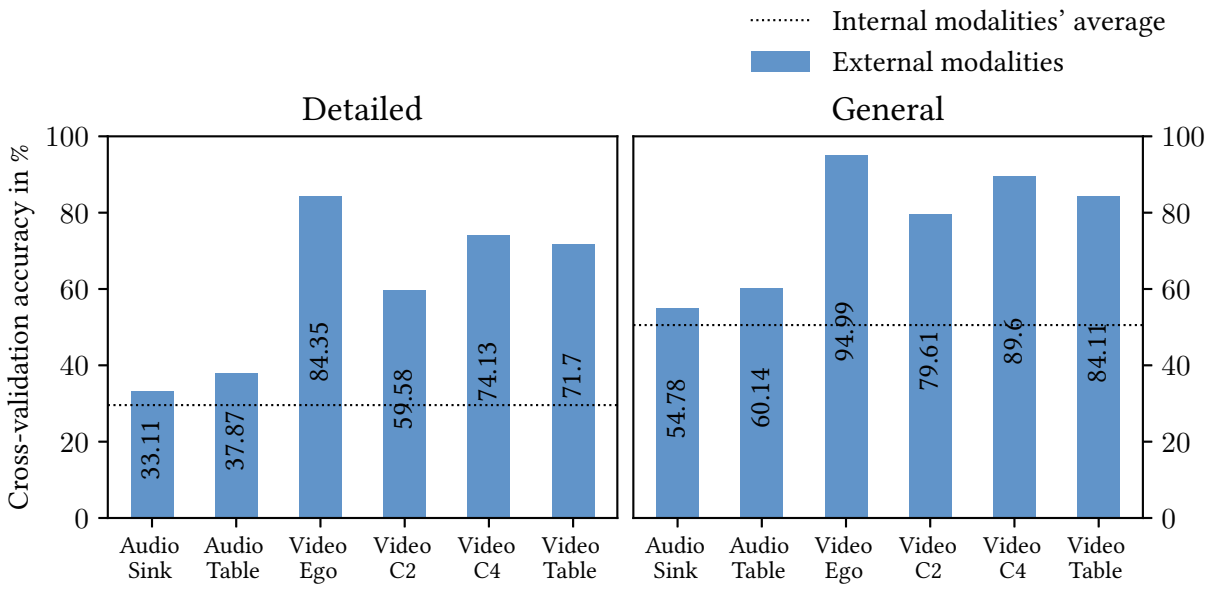


Figure 4.3: Unimodal classification accuracy of the external sensors. The dotted line represents the average of *internal* sensors.

4.3 Discussion

Although the results present high variance depending on the modality used, we consider all of them relevant to enhance the multimodal approach. Each test results in a cross-validation accuracy significantly higher than random guessing.

The most accurate classification is achieved using the egocentric video, followed by the two cameras mounted in the environment. The high potential in visual data is that perception carries all the necessary information to distinguish the tasks of the dataset at hand. It records the movement of the human as well as the objects in the environment. Recognising the objects used during execution by their shape and colour gives plenty of information about the task itself.

Using audio only, the model can identify 46% fewer actions than its visual alternatives. Audio alone seems less informative than vision but more informative than the internal sensors. An important detail must be addressed when comparing the metrics of internal and external sensors: The latter is implemented using pre-trained backbone networks. As described in Section 3.3.1, backbones can enhance the model significantly, especially if trained on more extensive datasets. This may be a significant contributing factor to the high classification accuracy of external modalities.

Internal modalities (muscle sensor, tactile force sensor, and body skeleton pose) are less suited for unimodal recognition, although significantly better than random guessing. Besides that, they provide insight into the capabilities of the attention mechanism in sequence modelling. It overperforms the LSTM baseline in all tested modalities. The most considerable improvement (12.7% in the detailed accuracy) is achieved using the force modality. Force sensor data is challenging to interpret, even for humans, and the LSTM network seems to overlook some essential patterns that the attention mechanism can recognise and take advantage of. The difficulty of action recognition based on this modality lies in the fact that force sensors do not provide any information about the pose of the human or the shape and colour of the objects the subject interacts with. Therefore, it is much harder to identify the object categories that take part in the actions.

The aforementioned results encourage the need to train a multimodal model that leverages pose, images, audio and EMG to improve the results.

Chapter 5

Multimodal Action Recognition

The results collected in Chapter 4 support the claim that transformers excel in understanding the time evolution of actions. The next step is to test whether this model can deal with multiple modalities simultaneously. Testing multimodal models provides the central part of this thesis, testing the possibilities of comprehending the embodied information in the connection between modalities. This chapter comprises an ablation study of the multimodal transformer encoder defined in Section 3.3.

Testing all 26 possible combinations is out of the scope of the thesis since one training session can take up to 9 hours. Instead, the experiment begins by testing modality pairs and evaluating them to see if the combination has any advantage. Then, existing pairs are extended with further modalities, such that after adding a new modality, the new metrics are always compared to those before. This way, we create multiple checkpoints to examine the influence of each modality in a multimodal approach. Since classification based on visual perception is a powerful tool on its own, it is interesting to exclude this modality from some of the experiments.

5.1 Ablation study

The ablation study compares uni- and multimodal models based on their understanding of human actions. We measure the action recognition ability of models by classification accuracy and confusion matrices, both acquired via cross-validation. Similar to the unimodal tests of Chapter 4, the metrics are calculated on two levels.

Multiple models are trained on each combination of modalities: the transformer using early- and late fusion and the baseline (if available). Our baseline, the LSTM from [1], uses late fusion. On the other hand, no baseline model for early fusion has been published for the studied dataset at the time of writing. The LSTM cannot be changed to early fusion since it is unsuitable for two-dimensional (time and modality) sequence modelling.

Comparing uni- and multimodal models is an essential part of the thesis. We introduce a new metric for this cause, called *multimodal advantage*. This metric is defined as the increase in cross-validation accuracy of a multimodal model compared to the best-performing unimodal counterpart. It can be calculated for both classification tasks, detailed and general.

$$\text{adv}(S) := \text{acc}(S) - \max_{M \in S} \text{acc}(M) \tag{5.1}$$

S denotes an arbitrary set of modalities, $\text{acc}(X)$ is the cross-validation accuracy of the model based on X. The result, $\text{adv}(S)$, is the multimodal advantage achieved using the modalities of S.

5.2 The combination of tactile force and body skeleton pose

The first modality pair consists of tactile force and body skeleton pose. It is considered a meaningful combination since their unimodal counterparts had differing strengths. As the confusion matrix of Figure 4.2 has shown, the force modality performs better at distinguishing objects in *Peeling* and *Slicing* tasks; meanwhile, the body skeleton modality classifies *Fetching items* more accurately. Despite their different scores for some classes, the overall accuracy of the two unimodal models mentioned shows substantial similarity. The difference is only 0.58% in the detailed classification objective.

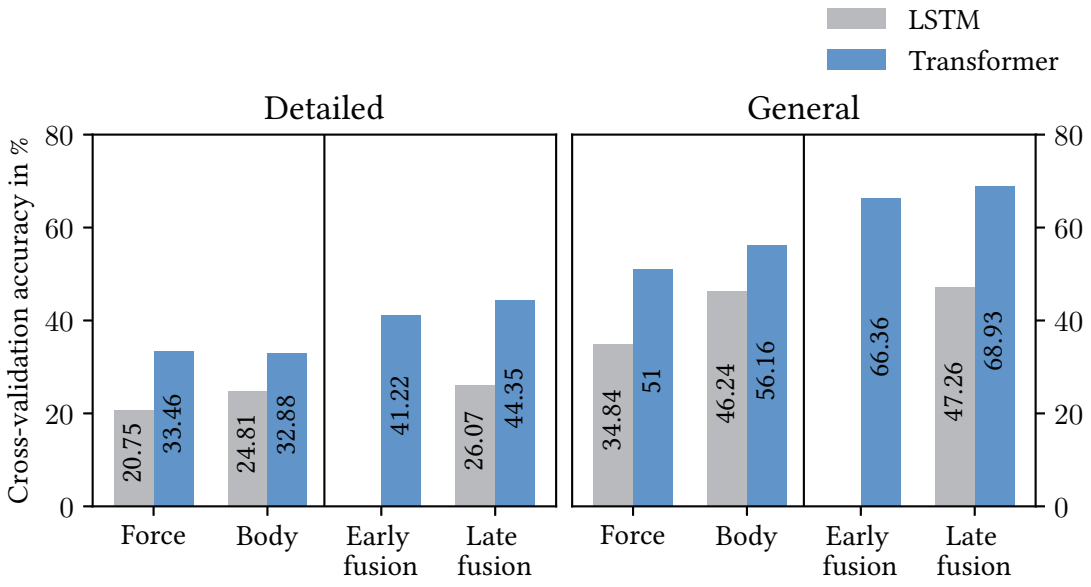


Figure 5.1: Multimodal force and body skeleton models using different fusion methods compared to their unimodal counterparts and the baseline.

The cross-validation accuracy of this modality pair is displayed in Figure 5.1. For the detailed classification task, the model with early fusion achieves a multimodal advantage of 7.76% compared to the best unimodal model and 10.9% with late fusion. Both options show a higher growth than the

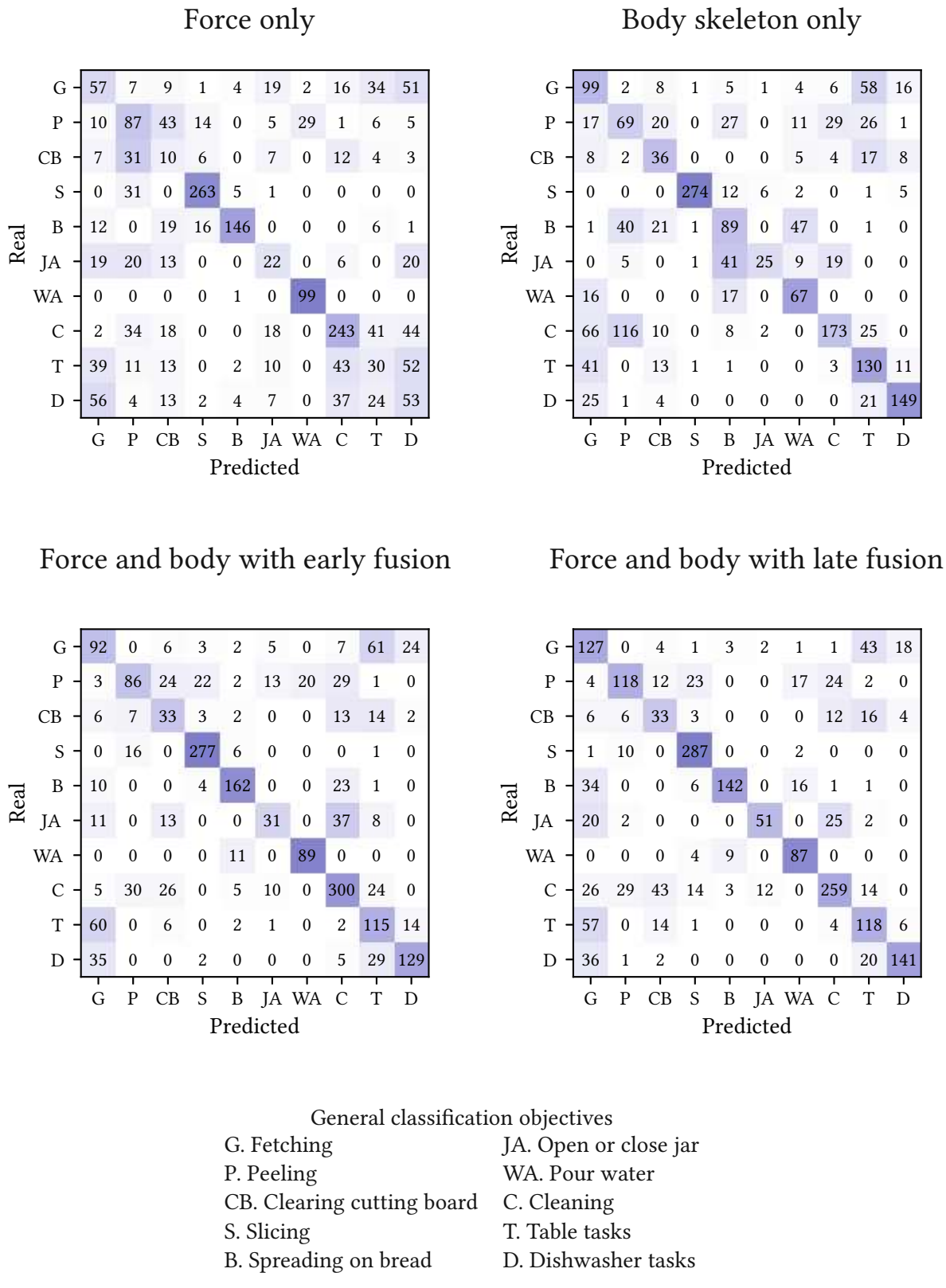


Figure 5.2: Confusion matrices for the general classification task comparing the models based on force and body skeleton.

baseline with its 1.26%. Prediction of the general labels has similar results, 10.2% and 12.8% for early and late fusion and 1.02% for the baseline.

A more elaborated way of evaluating predicted labels can be achieved via confusion matrices. Figure 5.2 compares the two multimodal models with the unimodal ones of this pair. These matrices show the general classification objective, hence the unequal number of instances per class.

5.3 Multimodal model of internal sensors

An ablation study for two internal modalities is described in Section 5.2; the only difference in this experiment is the model being extended to process EMG data as well. The results are compared to the unimodal classification and the modality pair of the previous experiment to show that adding a third modality improves its understanding of human actions. A diagram comparing fusion methods based on detailed and general action recognition can be seen in Figure 5.3.

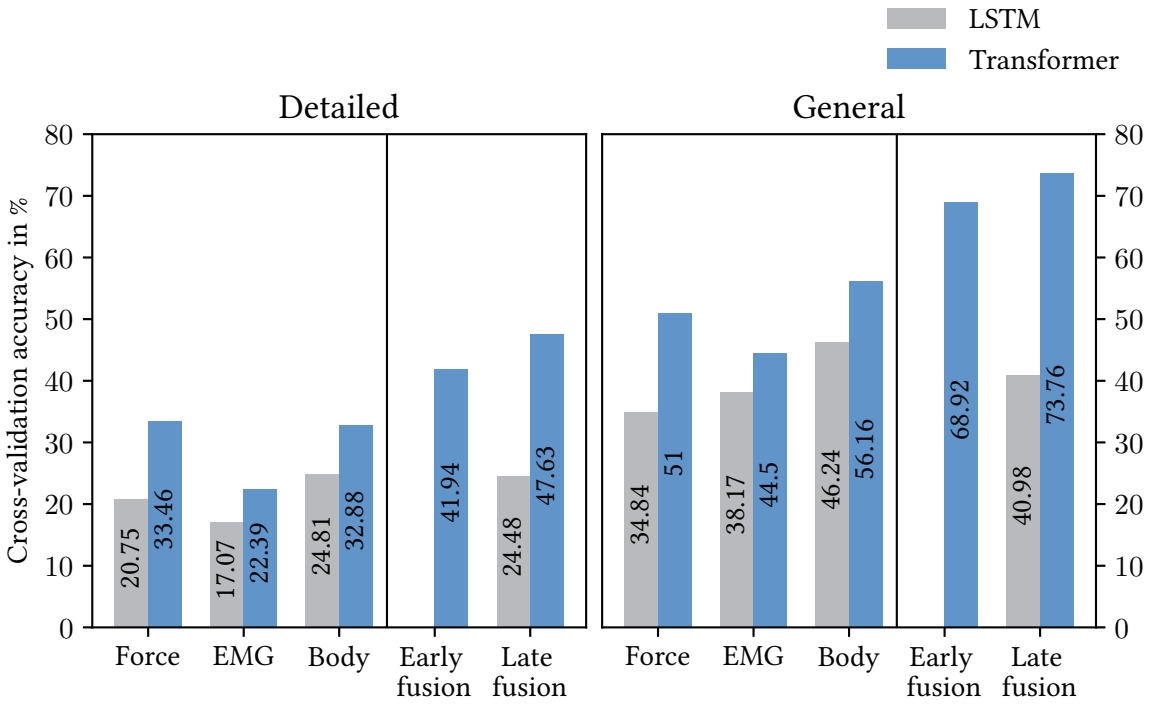


Figure 5.3: Multimodal internal sensor models using different fusion methods compared to their unimodal counterparts and the baseline.

In this experiment, the transformer achieves a multimodal advantage of 14.2% (17.6% in general classification) with late fusion being the superior method. Both the early and the late fusion models surpass the accuracy of the model in Section 5.2 (by 0.72% and 3.28%, respectively), which shows that adding a third modality to the model enhances its classification ability. On the other hand, the baseline's accuracy decreases. This means that the multimodal model performs worse than the best of its unimodal

counterparts. Opposed to the transformer, the LSTM cannot take advantage of an additional modality and improve its understanding of actions.

5.4 The combination of tactile force and vision

Internal and external sensors view the same tasks from a different perspective. Combining modalities from both categories can be a valuable approach for a multimodal encoder. This new pair consists of the tactile force and the table camera modalities. Regarding accuracy, there is a big gap between the corresponding unimodal models. The video encoder recognises around twice as many actions as the force encoder. Therefore, this combination introduces a new challenge for the multimodal transformer model: combining modalities of different strengths.

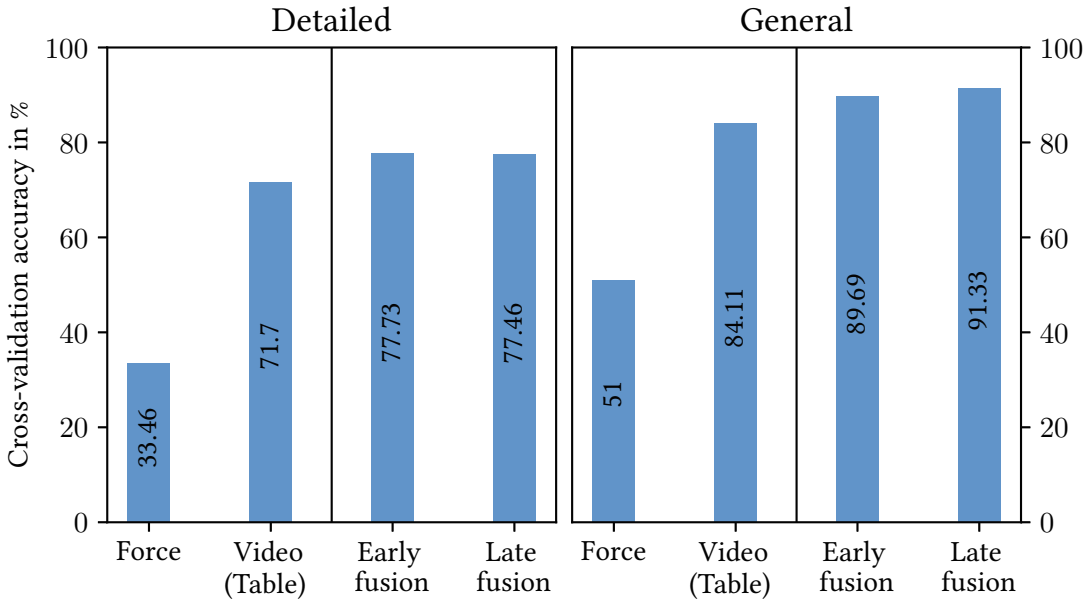


Figure 5.4: Multimodal models fusing tactile force and vision compared to the unimodal counterparts.

Although the ablation study shows a positive multimodal advantage of 6%, it does not achieve the increase measured for combining internal sensors. The force modality provides only a little information that is not contained in the videos. See Figure 5.4 for a visualisation.

It must be noted that the closer an accuracy is to 100%, the more difficult it is to achieve further improvement. Since the vision-based unimodal models have the highest accuracy, the multimodal advantage of tasks involving vision may seem low. However, it does not mean that adding vision to multimodal models has little advantage. Higher unimodal accuracy means less room for improvement, causing the multimodal advantage to appear as a smaller value.

5.5 Multimodal model using all five modalities

This section comprises an ablation study for the default multimodal encoder using all five modalities of the ActionSense dataset [1]. Figure 5.5 shows the action recognition capabilities of the multimodal model using simple early and late fusion. Although there is a clear multimodal advantage, there is still room for improvement. The results suggest that both fusion methods have the same potential, as they are highly similar in accuracy.

The model using force and vision only (Figure 5.4) performs better in validation than the one using all five modalities (Figure 5.5). This demonstrates that adding modalities can result in stronger overfitting. The reason for that is the increased redundancy of information: newly added modalities embed the same information while simultaneously expanding the dimension of features. Overfitting raises the need for regularisation techniques like masking.

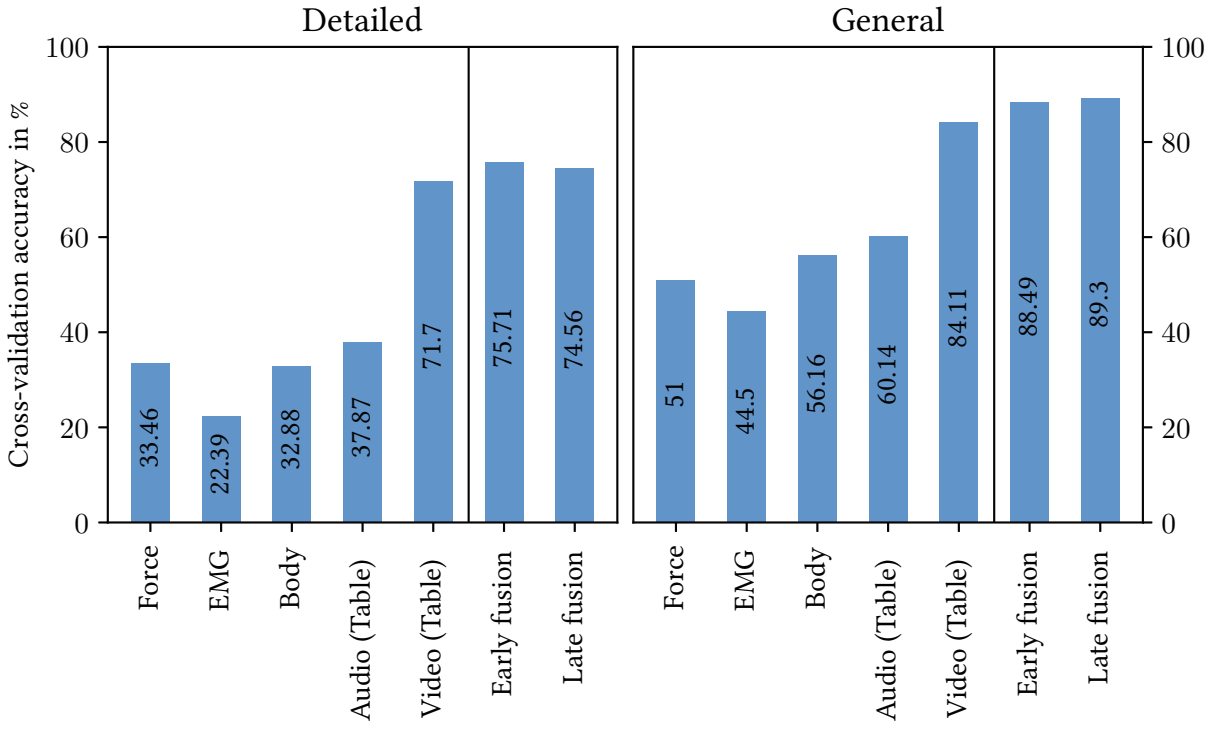


Figure 5.5: Fusion of all modalities.

Previous experiments show increased accuracy compared to the corresponding unimodal models and a clear advantage of vision compared to other inputs. This experiment can only reinforce the earlier findings, meaning that the capability of multimodal action recognition is highly dependent on the visual inputs. Comparing attention scores can provide further insight into the relative importance of modalities.

5.5.1 Attention scores

A mathematical formulation for the attention mechanism of transformers [2] is thoroughly discussed in Section 2.3. Attention score refers to the result of Equation 2.2. Besides being a necessary weighting tool for sequence modelling, it can be used to compare the influence of each modality on the final class token.

The multimodal model with late fusion separates the time evolution from the modality fusion, allowing us to compare modalities without time dependency. As the tokens pass through the encoder layers, the embedded information is gradually transformed into the class token. This process can be best described by calculating the average over the heads and layers of all attention blocks and extracting the row corresponding to the class token. Figure 5.6 shows the mean attention score grouped by ground-truth labels. Results show that video modality alone gets more attention than all other tokens combined. Other modalities scored around 1%, with the tactile force being higher. The learnable parameter that serves as the initial class token is of little importance, as it does not accumulate any information about the inputs.

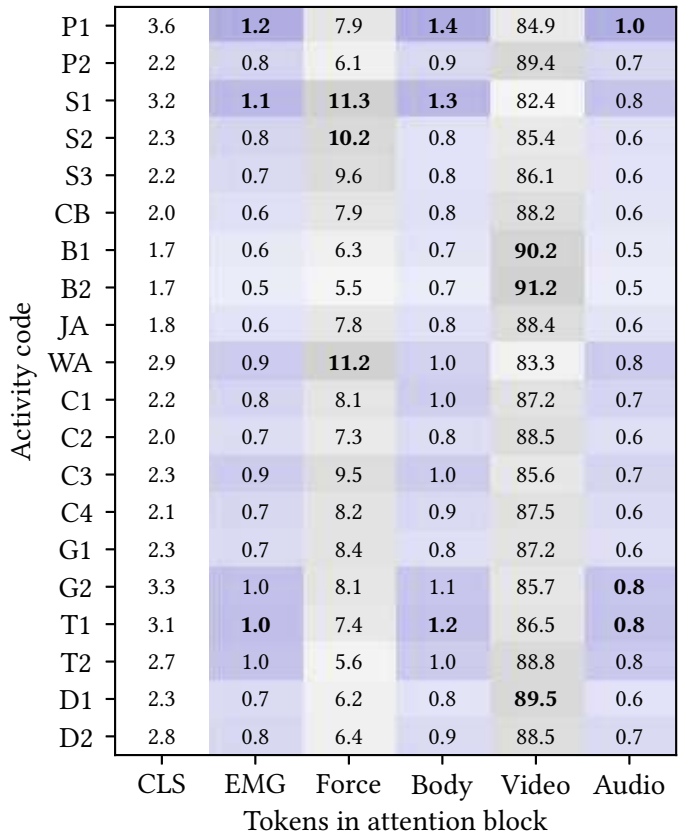
Although attention scores do not deviate much from class to class, there are notable differences. The model relies more on internal sensors for *Peeling* and *Slicing* tasks; meanwhile, muscle activity helps more in recognising *Pouring water*. All three of them involve precise hand movement. The video modality is high for all classes without exception but is the highest for *Spreading something on bread*. That is because jelly and almond butter cannot be distinguished without vision. Colour is the only cue in that case, as the dataset lacks a taste sensor.

The attention scores show a high dependency on the visual modality. Other modalities get only a small fraction of attention, meaning that the model benefits only little from non-visual inputs in its current form. This suggests that the model does not exploit multimodal data to the fullest. A possible way to improve the understanding of multimodal inputs can be achieved by stochastically masking tokens during training.

5.6 Masking the encoder

This section tests different masking strategies to test the multimodal model for performance and robustness. Partially removing tokens during training has been shown to help the model learn a *rich hidden representation* within the encoder [17]. Masking can be applied to all types of encoders, but we use it only in the encoders that fuse tokens of different modalities.

Definition 5.1 (Mask)**.** Let $\mathcal{M}_X^p$ denote a mask for modality X applied with a chance of $p \in [0, 1]$ to the encoder tokens. A mask $\mathcal{M}^0$ means leaving all tokens unmodified, whereas $\mathcal{M}_X^1$ results in removing



P1. Peeling cucumber
P2. Peeling potato
S1. Slicing cucumber
S2. Slicing potato
S3. Slicing bread
CB. Clearing cutting board
B1. Spreading almond butter
B2. Spreading jelly
JA. Opening and closing jar
WA. Pouring water
C1. Cleaning plate with sponge
C2. Cleaning plate with towel
C3. Cleaning pan with sponge
C4. Cleaning pan with towel
G1. Getting items from refrigerator
G2. Getting items from cabinets
T1. Setting table
T2. Stacking on table
D1. Loading dishwasher
D2. Unloading dishwasher

Figure 5.6: Attention score of actions expressed as a percentage. The three highest values of each column are marked in bold.

modality X in its entirety. Since masking is a type of regularisation, it is never applied during validation.

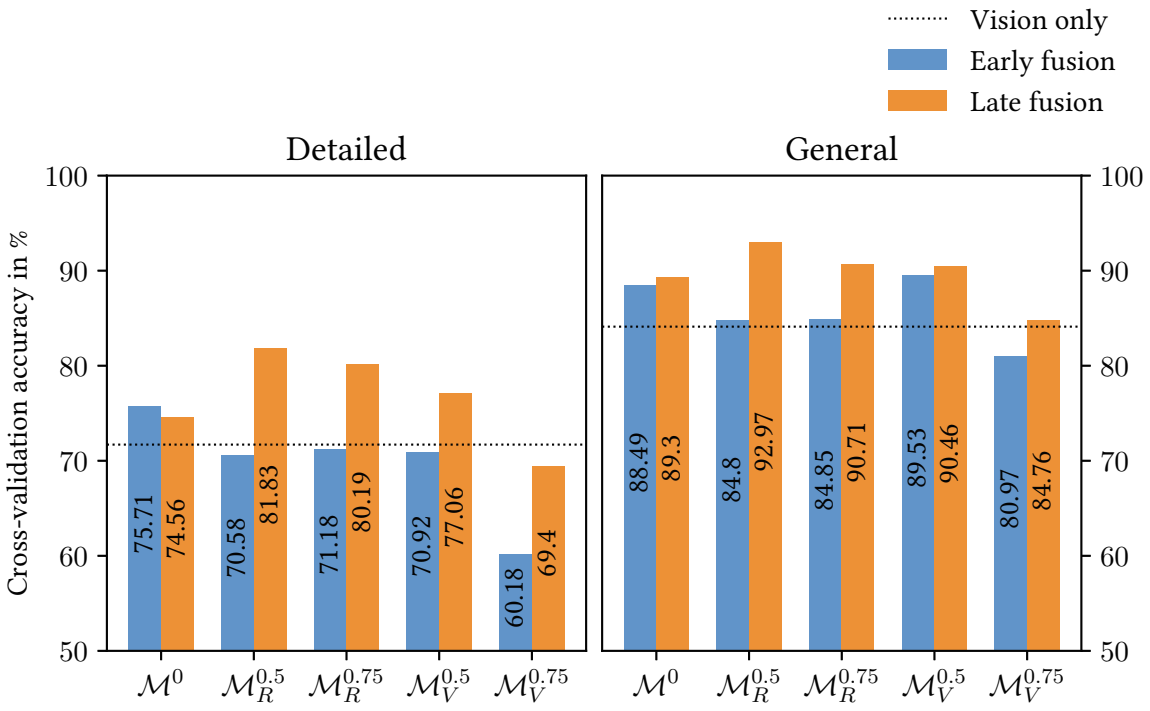


Figure 5.7: Comparing the effect of different attention masks during training. R and V denote a Randomly chosen modality and the Video modality, respectively.

Figure 5.7 shows the results of the ablation study on masking. We train a model using each fusion method, two distinct masks (random and vision), and two different probabilities (50% and 75%), resulting in 8 runs. The best results are achieved by masking randomly with 50% on late fusion. Early fusion does not benefit from any of the tested strategies.

Masking influences the attention scores measured in the multimodal encoder. Figure 5.8 shows that masking causes the attention to shift from vision to other modalities. Table 5.1 overviews the measured multimodal advantages. Models using masks with 75% chance are excluded, as they all underperform the 50% masks.

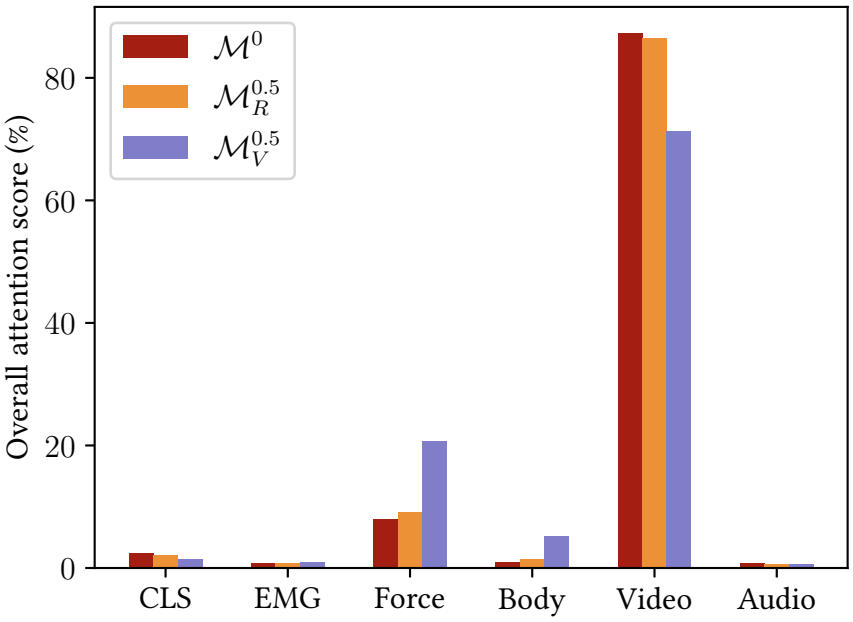


Figure 5.8: Attention score of each modality and the class token in late fusion. A vision mask changes the distribution of attention and forces the model to focus more on non-visual inputs.

Table 5.1: Multimodal advantage of models from Section 5.2 to 5.5. All values in %. The model that achieves the highest accuracy is highlighted in bold, and the second highest is underlined.

Modalities					Mask	Detailed classification			General Classification		
Force	EMG	Body	Audio	Video		Early fusion	Late fusion	LSTM	Early fusion	Late fusion	LSTM
✓		✓			$\mathcal{M}^0$	7.76↑	10.89↑	1.26↑	10.2↑	12.77↑	1.02↑
✓	✓	✓			$\mathcal{M}^0$	8.48↑	14.17↑	−0.33↓	12.76↑	17.60↑	−5.26↓
✓				✓	$\mathcal{M}^0$	<u>6.03</u>↑	5.76↑		5.58↑	<u>7.22</u>↑	
✓	✓	✓	✓	✓	$\mathcal{M}^0$	4.01↑	2.86↑		4.38↑	5.19↑	
✓	✓	✓	✓	✓	$\mathcal{M}_R^{0.5}$	−1.12↓	**10.13**↑		0.69↑	**8.86**↑	
✓	✓	✓	✓	✓	$\mathcal{M}_V^{0.5}$	−0.78↓	5.36↑		5.42↑	6.35↑	

5.6.1 Removing vision during validation

The robustness of models can be tested by removing a selected modality from the validation set. This imitates the failure of a sensor in real-life applications. This test shows whether the model can still recognise actions after one of the sensors stops functioning. We aim to demonstrate that using multiple modalities has advantages besides recognising more actions.

This experiment uses the egocentric camera as the visual modality. This camera can observe more actions than the rest, leading to higher accuracy in action recognition. Adding further modalities to the egocentric camera does not improve action recognition in this dataset but makes the model more robust. This experiment shows that adding further modalities can be beneficial even if the multimodal advantage is negligible.

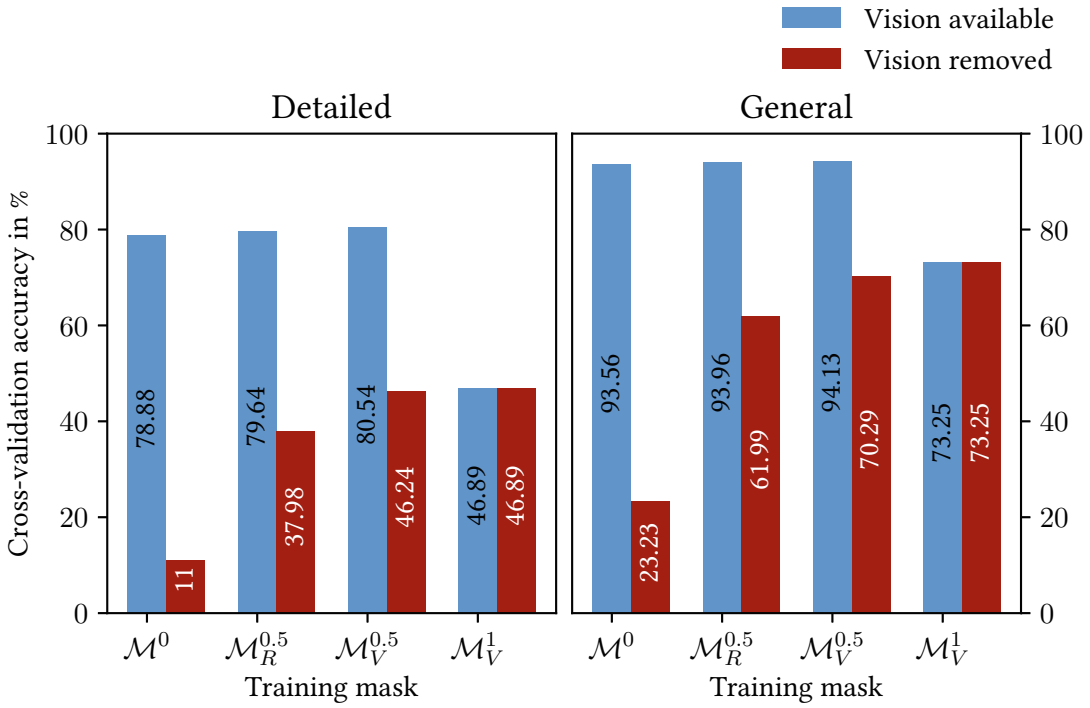


Figure 5.9: Performance and robustness achieved by different masks in the training phase. R denotes a randomly selected modality.

Removing a sensor entirely undoubtedly causes a drop in accuracy, but at a lower extent for masked models. As shown in Figure 5.9, they achieve around the same accuracy as the model that never had access to vision. The unmasked model strongly depends on the visual input and completely fails when excluded. To achieve a feasible comparison, we evaluate some models on the multimodal dataset without vision (equivalent to masking with $\mathcal{M}_V^1$). Then, we compare them to the experiment where vision was still available and examine the extent of the drop in accuracy.

5.7 Modality inference

The third research question of the thesis (Section 1.2) asks whether training a model to infer a missing modality is possible. The force modality $F(t)$ is removed from all data inputs to simulate the missing modality. Then, the model is trained to reconstruct the missing data using body skeleton $B(t)$, EMG $E(t)$, audio $A_{\text{Table}}(\tau)$ and video $V_{\text{Table}}(t)$.

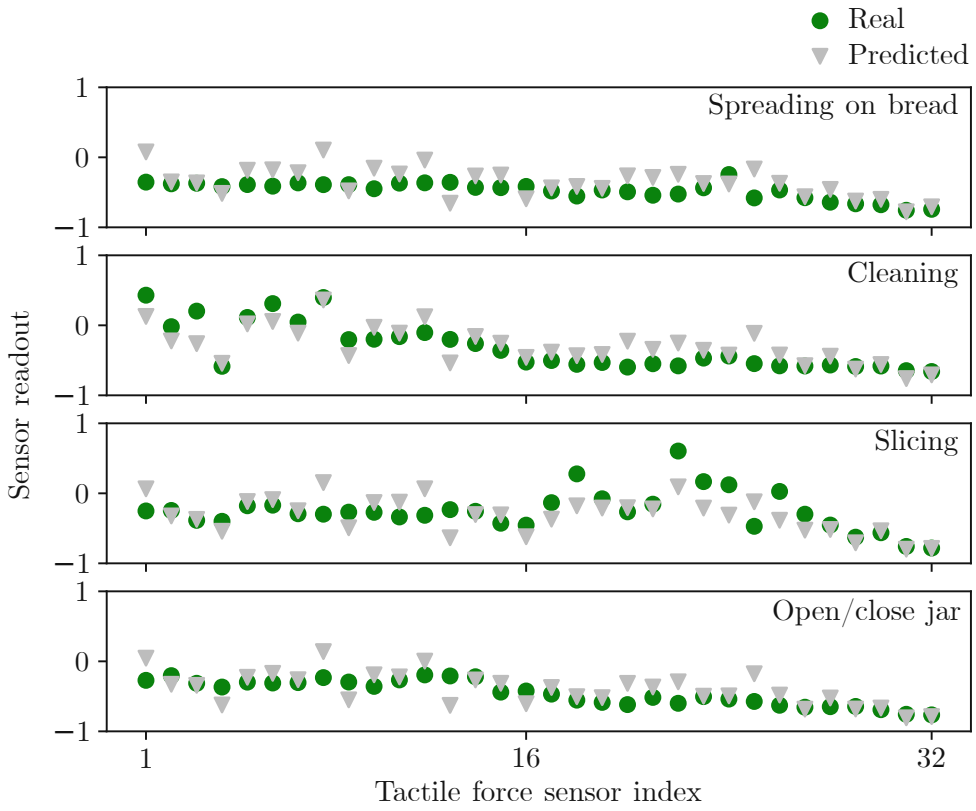


Figure 5.10: Inferring tactile force $F(t)$ from other modalities (body skeleton $B(t)$, muscle activity $E(t)$, audio $A_{\text{Table}}(\tau)$ and video $V_{\text{Table}}(t)$).

Figure 5.10 plots the predicted force sensor readouts against the ground truth. The force sensors with indices 1–16 are located on the left hand of the subject, and those with 17–32 on the right one. The displayed values are resampled and aggregated as described in Section 3.1. The holdout training and validation loss of this model are 0.0108 and 0.0391, respectively. They correspond to a mean absolute error of 0.104 and 0.198 since the loss is defined as the mean squared error between prediction and ground truth. The validation error is 9.9% of the sensor readout range.

5.8 Discussion

In this chapter, we can seen examples of the various advantages of multimodal action recognition, highlighting its potential to enhance the performance of machine learning models significantly. This

includes improving accuracy, robustness and inference of missing data.

The conducted test suggests that transformer models can process multiple modalities and utilise the embedded information for classification. Almost all implemented multimodal models can predict more labels correctly than any of the corresponding unimodal ones. The results indicate that transformers understand the time evolution of actions better than LSTM networks, especially for sequences containing multiple modalities.

Late fusion achieves higher accuracy than early fusion in all tests. For both classification objectives, it is able to recognise actions the early fusion model cannot. This gap between the fusion models seems to grow with an increasing number of modalities. The reason is that late fusion works with fewer tokens at once, as it separates the temporal tokens of different modalities from each other. In contrast, early fusion processes two-dimensional sequences (time and modality), making it difficult to transfer all that information to the class token.

If a training mask is applied, early fusion models suffer a drop in accuracy. On the other hand, late fusion models improve remarkably. The overall best metrics are achieved by random masking (50%), where the multimodal advantage is almost double in the detailed classification compared to plain late fusion.

The more modalities are added, the more complex the model gets. With the increasing number of learnable parameters, overfitting gets more prominent. The main difficulty lies in the different overfitting characteristics of modalities. The best practices we use against it are using pre-trained (and frozen) backbones and masking tokens. Although late fusion has six times more parameters than early fusion, it does not make it more prone to overfitting. Its last encoder block only gets a fraction of the input tokens, creating a bottleneck and forcing the model to compress information more densely.

To sum up, extending models with modalities of new data sources leads to improved model accuracy and robustness in most cases. Adding modalities can unlock potential but also increase the dimensionality of data. For this reason, special care has to be taken against overfitting, such as modifying attention via masks. As a result, late fusion is a more reliable method for integrating multimodal information, mainly due to its ability to improve performance due to masking.

Chapter 6

Conclusion

Most deep learning research for action recognition concentrates solely on vision, yet visual modalities alone provide insufficient information for many robotic applications. Aiming for efficient interactions with a physical environment raises the need for combining multiple modalities. To this end, we introduce a multimodal transformer model capable of recognising kitchen tasks based on five different modalities (force, muscle activity, skeleton pose, audio, and video). Our research focuses on the benefits of multimodal action recognition.

6.1 Approach and results

The backbone of our model consists of transformer encoders, sometimes extended with a decoder. It is completed with a linear classification layer and an additional projection layer for some modalities. We test two different fusion methods, namely early and late fusion. They differ in the stage where modalities are merged into a common representation format. We rely on quantitative classification metrics such as accuracy, confusion matrices, F1-score, precision, and recall to compare our models.

The ablation study shows that combining modalities helps the model improve its understanding of human actions. Without exception, all multimodal transformers surpass the abilities of the corresponding unimodal ones. By analysing the predicted classes more deeply, no classes can be found where only the multimodal approach produces correct predictions. Although the overall accuracy is higher in all cases, it is impossible to pinpoint some classes that only the multimodal model can distinguish. Combining modalities improves the overall capability but cannot help comprehend actions where all modalities fail independently. The greatest strength of multimodal models lies in recognising tasks that are ambiguous for some modalities but labelled correctly with others.

We found masking to improve accuracy and make models more robust. They act as a regularisation, reducing overfitting and making models less dependent on a single modality. Only late fusion can

benefit from masking; it increases the multimodal advantage by 7.3%.

6.1.1 Reviewing early fusion, late fusion, and the baseline

Although our evaluation only includes two distinct types of neural networks, namely the transformer and the LSTM baseline, it has shown that adding modalities does not improve classification in all scenarios. For example, the LSTM of the internal sensors performs worse than the unimodal model using body skeleton data only. Approaches involving the transformer do not only achieve higher accuracy than any unimodal models but also higher than the LSTM. This supports our initial claims of using multimodal data, and the need to further investigate optimal techniques to optimize the modality fusion in AI methods.

Regarding the fusion techniques, we find the late fusion model to be better suited for classification. Late fusion benefits from separating the temporal progression of each modality. This separation helps the model distribute its attention in a more organised manner and improves its action recognition capability. It focuses on one modality at a time and combines them in a separate step. On the other hand, the single encoder in early fusion overlooks some crucial details when summarising tokens, as it has to consider both the time axis and the different modalities simultaneously. The advantage of the best late fusion model is 10.1%; meanwhile, early fusion achieves only 6%.

In conclusion, the attention mechanism of transformers has opened a new door to multimodal learning. Its understanding of the temporal progress of actions is not the only reason for its success. This tool combines the information contained in different modalities more efficiently than other models.

6.2 Limitations

Lack of experiments with robots. Although multimodal transformer models have high relevance in robotics, our tests do not include any trials with actual robots. Testing it on robots would require a different dataset, recorded specifically for a use case in (human) action recognition. On the contrary, our approach is meant to present a model that comprehends multiple modalities. In the future, using the same architecture, a model can be trained to recognise different actions and exploit this information for action planning or social robotics.

Scalability. All of our models are trained on the same 5.56 hours of data records. In many practical applications, especially with transformer models, a bigger model involving more training is favourable. Our model has fewer parameters than other transformers and can recognise only 20 distinct actions. Thus, we rely on the scalability of neural networks to extrapolate our findings to multimodal models in

general. Although a feasible comparison of uni- and multimodal deep learning architectures is possible with small models, training the model to recognise more complex tasks would be scientifically valuable.

6.3 Further research

Masking in early fusion. Although our model significantly improves by adding training masks in late fusion, early fusion cannot benefit from it. As masking has plenty of hyper-parameters and can be applied differently, we cannot test all possibilities. A successful strategy for masking in early fusion could pave the way for masked autoencoders, unlocking new potential for multimodal models.

Testing on other datasets. Evaluating the same model on other datasets would give insight into its capabilities in a more generalised way. There is a lack of multimodal datasets with a diverse range of modalities among publicly available datasets at the time of writing. Testing scalability and robustness would become more manageable and generalisable if the same model could be tested on other datasets.

Bibliography

[1] J. DelPreto, C. Liu, Y. Luo, M. Foshey, Y. Li, A. Torralba, W. Matusik, and D. Rus, "Actionsense: A multimodal dataset and recording framework for human activities using wearable sensors in a kitchen environment," *NeurIPS*, 2022.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, 2017.

[3] Z. Sun, Q. Ke, H. Rahmani, M. Bennamoun, G. Wang, and J. Liu, "Human action recognition from various data modalities: A review," *IEEE transactions on pattern analysis and machine intelligence*, 2022.

[4] R. Girdhar, M. Singh, N. Ravi, L. van der Maaten, A. Joulin, and I. Misra, "Omnivore: A single model for many visual modalities," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[5] M. Lu, Y. Hu, and X. Lu, "Driver action recognition using deformable and dilated faster r-cnn with optimized region proposals," *Applied Intelligence*, 2020.

[6] S. Danafar and N. Gheissari, "Action recognition for surveillance applications using optic flow and svm," in *ACCV*, 2007.

[7] J. Liu, A. Shahroudy, D. Xu, A. C. Kot, and G. Wang, "Skeleton-based action recognition using spatio-temporal lstm network with trust gates," *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[8] D. Liang and E. Thomaz, "Audio-based activities of daily living (adl) recognition with large-scale acoustic embeddings from online videos," *ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2019.

[9] M. Javeed, N. A. Mudawi, B. I. Alabduallah, A. Jalal, and W. Kim, "A multimodal iot-based locomotion classification system using features engineering and recursive neural network," *Sensors*, 2023.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, 1997.

[11] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *NAACL-HLT*, 2018.

[12] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," *ICCV*, 2021.

[13] R. Winastwan, "Image classification with vision transformer," *Towards Data Science*, 2023.

[14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition," *ICLR*, 2021.

[15] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," in *International conference on machine learning*, 2019.

[16] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv:1705.06950*, 2017.

[17] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *IEEE/CVF conference on computer vision and pattern recognition*, 2022.

[18] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Imagebind: One embedding space to bind them all," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[19] Y. Zhang, K. Gong, K. Zhang, H. Li, Y. Qiao, W. Ouyang, and X. Yue, "Meta-transformer: A unified framework for multimodal learning," *arXiv:2307.10802*, 2023.

[20] P. Xu, X. Zhu, and D. A. Clifton, "Multimodal learning with transformers: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[21] C. Weng, B. Lu, Q. Gu, and X. Zhao, "A novel multisensor fusion transformer and its application into rotating machinery fault diagnosis," *IEEE Transactions on Instrumentation and Measurement*, 2023.

[22] Y. Li, Y. Du, C. Liu, F. Williams, M. Foshey, B. Eckart, J. Kautz, J. B. Tenenbaum, A. Torralba, and W. Matusik, "Learning to jointly understand visual and tactile signals," in *ICLR*, 2023.

[23] N. Louis, J. J. Corso, T. N. Templin, T. D. Eliason, and D. P. Nicolella, "Learning to estimate external forces of human motion in video," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022.

[24] H. Ahn, Y. Michel, T. Eiband, and D. Lee, "Vision-based approximate estimation of muscle activation patterns for tele-impedance," *IEEE Robotics and Automation Letters*, 2023.

[25] Y. Zhang, X. Ding, K. Gong, Y. Ge, Y. Shan, and X. Yue, "Multimodal pathway: Improve transformers with irrelevant data from other modalities," *arXiv:2401.14405*, 2024.

[26] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, 1948.

[27] Y. Gong, Y.-A. Chung, and J. Glass, "Ast: Audio spectrogram transformer," *arXiv:2104.01778*, 2021.

[28] S. Miron, *Signal Processing*. IntechOpen, 2010.

[29] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *ICASSP*. IEEE, 2017.

[30] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. Huang, S.-W. Li, I. Misra, M. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jegou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," 2023.

[31] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv:1711.05101*, 2017.

Modalities					Mask	Fusion	Training		Validation				
Force	EMG	Body	Video	Audio			Acc.	Acc.†	Acc.	Acc.†	Precision	Recall	F1 score
✓							73.22	81.68	33.54	51.01	33.44	33.33	31.51
	✓						71.98	81.33	22.32	44.19	21.01	22.19	20.73
		✓					83.49	93.77	32.88	56.11	36.93	33.00	32.90
		*					89.59	94.64	31.01	51.11	34.10	31.20	31.10
			Ego				84.35	94.99	84.39	94.95	85.30	84.32	84.24
			C2				98.44	98.91	59.24	79.36	61.65	59.58	58.88
			C4				98.64	99.48	74.21	89.77	76.21	74.18	73.62
			Tab				71.70	84.11	71.77	84.14	74.40	71.71	71.52
				Sink			33.11	54.78	33.13	55.05	31.42	32.81	31.68
				Tab			37.87	60.14	37.73	60.10	35.94	37.41	36.37
✓		✓				early	83.78	91.49	41.26	66.36	42.54	41.26	40.76
						late	95.66	98.43	44.34	68.84	47.65	44.31	44.65
✓	✓	✓				early	84.66	92.51	41.77	68.69	45.17	41.75	40.21
						late	97.02	99.08	47.47	73.79	53.46	47.60	48.07
✓	✓	✓		Tab		early	63.90	80.24	34.39	59.95	35.24	34.06	32.55
						late	97.64	99.27	46.82	73.18	49.97	46.84	45.98
✓			Tab			early	96.67	98.60	77.83	89.75	78.61	77.70	77.67
						late	97.99	99.16	77.53	91.31	78.40	77.44	77.25
✓	✓	✓	Ego	Tab		early	99.95	100.0	83.99	93.89	84.97	83.90	83.84
						late	97.13	98.99	85.25	94.04	85.92	85.12	85.09
✓	✓	✓	Tab	Tab		early	95.66	97.92	75.81	88.54	78.04	75.64	75.73
						late	98.49	99.36	74.65	89.29	76.26	74.54	74.42
✓	✓	✓	Tab	Tab	$\mathcal{M}_R^{0.5}$	early	87.46	92.16	70.71	84.85	71.92	70.39	69.97
						late	94.08	97.14	81.92	92.93	83.15	81.74	81.98
✓	✓	✓	Tab	Tab	$\mathcal{M}_R^{0.75}$	early	85.22	90.67	71.31	84.90	72.86	70.99	70.70
						late	93.07	96.60	80.30	90.71	81.69	80.27	80.38
✓	✓	✓	Tab	Tab	$\mathcal{M}_V^{0.5}$	early	80.10	90.29	70.96	89.55	72.35	70.74	70.58
						late	92.61	97.02	77.17	90.45	80.32	77.15	76.93
✓	✓	✓	Tab	Tab	$\mathcal{M}_V^{0.75}$	early	78.67	88.85	60.20	81.01	61.30	59.99	59.61
						late	90.58	96.15	69.49	84.75	72.59	69.47	69.25

*including finger joints
†general classification

Table 1: Summary Tab for the classification metrics of Chapters 4 and 5. All values in %. The different cameras and microphones are noted in accordance with Sections 3.1.4 and 3.1.5.

Appendix A

Masked Autoencoder

We take the Masked Autoencoder of He et al. [17] as a basis for our model and modify it to make it suitable for multimodal tasks.

A.1 Model architecture

The original approach processed its inputs into tokens with a simple linear layer that could be reversed just as easily. On the other hand, our multimodal approach uses backbones that do not have a corresponding decoder. This makes it necessary to modify the self-supervising goals of the pre-training phase.

We define virtual inputs for reconstruction by the autoencoder. The virtual inputs are obtained after applying the pre-trained backbones. For modalities without a pre-trained module, the virtual inputs are identical to the actual data inputs. After projecting features into the token space, masking removes p of the virtual inputs in each modality and forwards the remaining tokens to the encoder. The removed features are saved for loss calculation, as displayed in Figure A.1.

After training the autoencoder for reconstruction, the model is modified for a downstream classification task and fine-tuned. For this task, the model labels the human action by projecting the class token via a linear layer. Although in [17], no masking is applied in fine-tuning, we apply the same masking strategy as during pre-training to regularise the model.

A.2 Experiments

We train our masked autoencoder on the multimodal dataset of Chapter 5. Hyper-parameters are set according to Table 3.2. We test different masking probabilities p and omitted masking during fine-tuning in some cases. Results are displayed in Table A.1.

p	Fine-tuning mask	Detailed accuracy	General accuracy
75	✓	62.3 (−9.38)	84.0 (−0.11)
50	✓	72.1 (+0.43)	85.7 (+1.56)
50		72.0 (+0.27)	85.7 (+1.63)

Table A.1: Cross-validation accuracy and multimodal increase of masked autoencoders for a downstream classification task. All values in %.

A.3 Discussion

Although the multimodal masked autoencoder can classify human actions with higher accuracy than a vision-only baseline in the downstream classification task, it does not reach the increase early and late fusion models achieves in Chapter 5.

The reconstruction losses converge for all modalities, yet the classification task benefits only little from using a pre-trained model. We think a bigger dataset would be needed to achieve high-level feature extraction with the autoencoder that can enhance classification significantly.

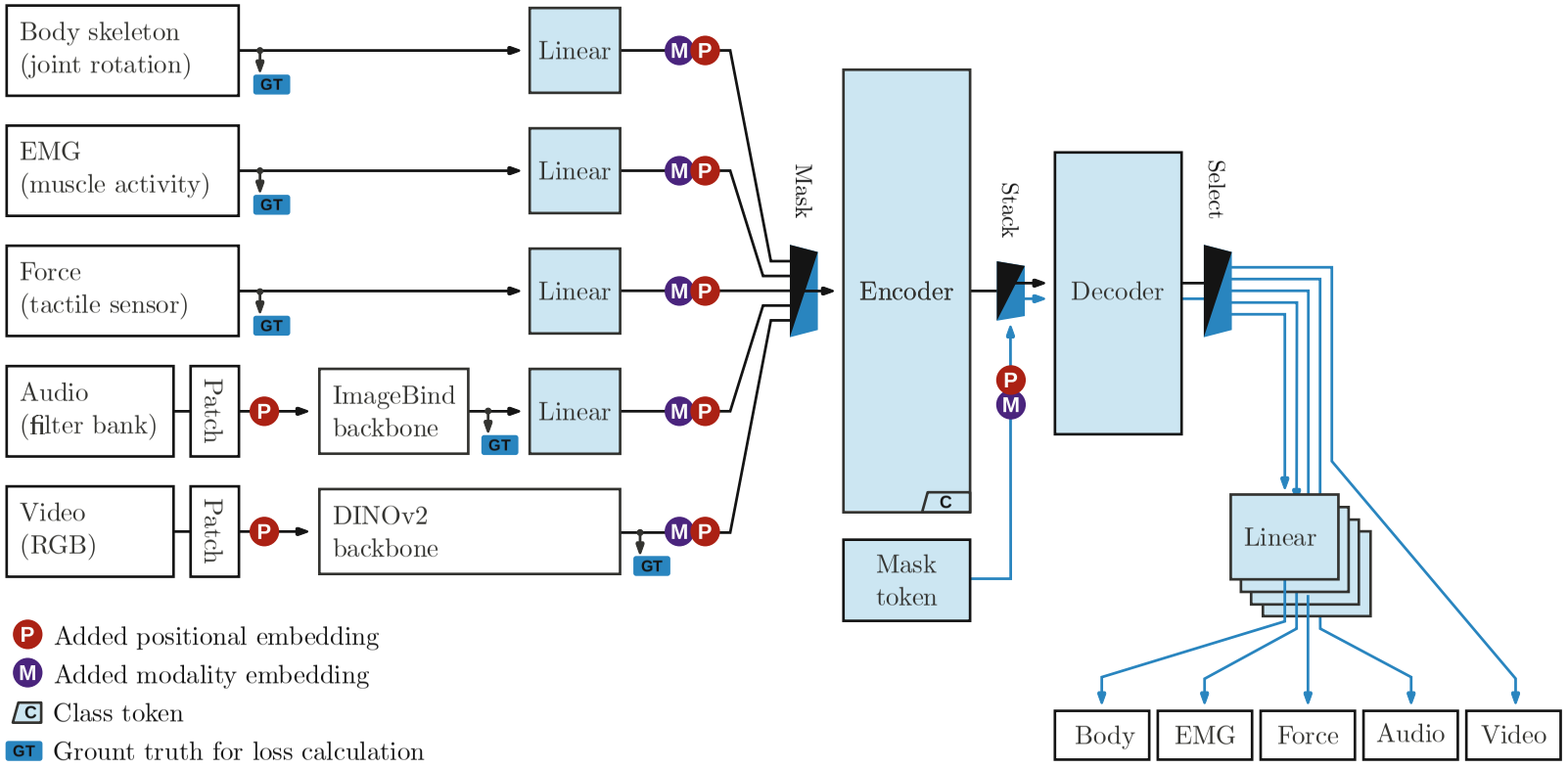


Figure A.1: Partially masked autoencoder. Modules with trainable parameters are highlighted with a light blue background. After pre-training finishes, the decoder is removed and the class token is fed through a linear layer, providing classification labels for the human actions.