

TDK-thesis

Zsolt Csibi

LLM-Guided Rehabilitation: Detecting and correcting Rehabilitation Errors with LLMs

EÖTVÖS LORÁND UNIVERSITY

FACULTY OF INFORMATICS

DEPARTMENT OF ARTIFICIAL INTELLIGENCE



Author:

Zsolt Csibi

Computer Science MSc - AI Specialization

Supervisor:

Gábor Baranyi

PhD Student

Budapest, 2024

Contents

1	Introduction	1
2	Background overview	3
2.1	Large/Vision Language Models	3
2.2	Prompting	8
2.3	Dataset used	11
2.4	Dynamic Time Warping (DTW)	12
2.5	Skinned Multi-Person Linear Model (SMPL)	15
3	Related Works	17
3.1	LLMs in health care	17
3.2	Rehabilitation with Artificial Intelligence	20
3.3	3D Human Pose Detection	23
3.4	Time Series Synchronization for aligning videos	25
3.5	Large/Vision Language Models	27
4	Methods	28
4.1	Human Pose Detection	28
4.2	Synchronize Rehabilitation videos	30
4.3	Prompt Engineering GPT-4	33
4.4	Evaluation of GPT-4 answers	36
5	Results	38
6	Discussion	50
7	Conclusion	53
	Acknowledgements	55

CONTENTS

A Prompt used	56
Bibliography	66
List of Figures	74
List of Tables	75

Abstract

Large Language Models (LLMs) and their extensions Visual Language Models (VLMs) have gained significant popularity in the past two years. This study focuses on a novel application: I am exploring the potential of LLMs/VLMs to identify and correct errors in patients' posture during rehabilitation exercises. I broke down the process into multiple steps, such as calculating the 3D pose for the patients' exercise videos in a modelled situation: performing correctly under the supervision of the physical therapist and performing at home with potential errors. I compared the videos by aligning them with Dynamic Time Warping and highlighting the keyframes where pose differences were most noticeable. This step was followed by selecting those skeleton coordinates which have the most significant differences. Then I assembled them into a prompt for requesting an error description from GPT-4. I have tried several different prompting techniques and studied the results. I found (a) that at the present level of LLM/VLMs, GPT-4 is crucial for facilitating communication, especially concerning potential errors, (b) a specific way using GPT-4 API to use the skeleton coordinates only for the identification of the errors and for the instructions for error corrections. Additional information pieces, such as using many instances, contextual information, image pairs, combined image and skeleton data, mentioning the critical joints and requesting the focus of attention mechanism, and multiple camera perspectives are essential for improving the accuracy of GPT-4 up to 60% on our dataset. My study highlights the need of using multiple cameras for better error detection as the single camera setup is not sufficient for all body configurations. For home use, this means complex camera positioning, requiring machine assistance to find the optimal starting poses for the cameras and for the patient.

Chapter 1

Introduction

The role of rehabilitation has seen a slight increase in recent years. According to forecasts, the need for rehabilitation and the market size of physical rehabilitation will increase significantly in the next five years. These forecasts indicate that the employment of physical therapists will increase by approximately **20%** by 2029 [1]. The *WHO* (World Health Organization) has stated that for every five people in Europe who require rehabilitation, at least two will require it[2]. This equates to approximately 394 million individuals in Europe, representing nearly half of the continent's population. Furthermore, as illustrated in Figure 1, which is sourced from the WHO, in 2019, over half of the **394 million** individuals required physical rehabilitation due to musculoskeletal disorders.

The reasons for requiring physical rehabilitation are numerous and diverse. They include sports injuries, orthopaedic conditions such as joint replacements, and stroke and traumatic brain injuries. The spectrum of physical rehabilitation is therefore considerable.

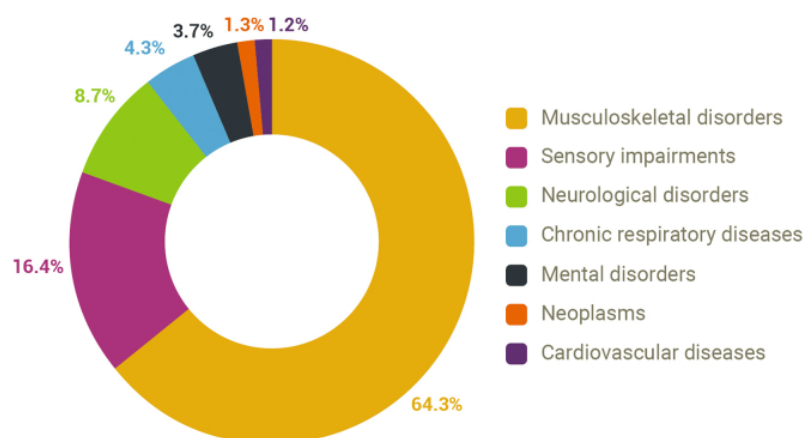


Figure 1.1: Distribution of people in rehabilitation according to the type of the disorder, data from WHO [2]

The number of people employed in the field of physical rehabilitation is already considerable, yet the number of patients requiring treatment is also on the rise. Consequently, some patients are unable to access the level of care they require due to a shortage of psychotherapists in the field. These professionals lack the capacity and time to provide personalised treatment. Another significant challenge in physical rehabilitation is the financial constraints faced by hospitals due to rising overhead costs, a lack of suitable facilities, and the general infeasibility of personalised rehabilitation in hospitals. In particular, during the pandemic, when contact with people was prohibited, or in some countries, there were lockdowns as well, psychotherapists lacked the necessary tools to handle the situation of online therapies.

A further crucial aspect of this inquiry concerns individuals who are unable to leave their residences for a specific reason. In such instances, it is imperative that the therapist visit the patient in person for a personal therapy session.

Considering the arguments mentioned above, there is a clear need for tools in the field of physical rehabilitation. This study will demonstrate my contribution to physical rehabilitation, utilising artificial intelligence (AI) methods, particularly Large Language Models (LLMs) and Vision Language Models (VLMs). On image captioning tasks, state-of-the-art VLMs demonstrate superior understanding of images and their context compared to the convolutional neural network (CNN)-based image captioning solutions, which can be trained. The dataset does not include image captioning, only contextual information. Consequently, CNN-based image captioning solutions are not the focus of this experiment.

The following section outlines the structure of this study. In chapter 2, I will present the essential background information about the models and methods used throughout the study. In chapter 3, I will present a review of the existing literature on home rehabilitation, large language modelling and prompt engineering. Then, in chapter 4, I will focus on the methods used throughout the study. In chapter 5, the findings and results obtained using the principles of prompt engineering will be presented. This will include an analysis of the advantages and disadvantages of using visual language models in the context of rehabilitation. The chapter 6 will focus on demonstrating the potential future improvements of my method and the potential use of these systems in a real home rehabilitation scenario. chapter 7 is a summary of the findings in the previous sections, with the claim that these systems do not replace therapists, they assist them. An example prompt used in this study is illustrated in Appendix A.

Chapter 2

Background overview

2.1 Large/Vision Language Models

The history of language modelling began with the development of mostly rule-based systems and statistical models. This article from Nadkarni et al.[3], provides an excellent overview of classical NLP methods and tasks in the medical domain. The initial language models were probabilistic base models, which sought to address a number of downstream Natural Language Processing (NLP) tasks, including sequence tagging, named entity recognition, natural language understanding, and machine translation. However, the most prevalent and influential NLP task today is the next token prediction. The next token prediction task in NLP involves forecasting the subsequent word or token in a sequence of text, given the preceding context. This task is of great importance in a number of NLP applications, including language modelling, text generation and machine translation. The model predicts the probability of the next token occurring in the context of the preceding text. This could be the next word in a sentence, for example. A useful illustration is the statistical language model known as N-grams[4], a simplified version of the standard statistical language models, which theoretically uses the entire previous context. However, in practical cases, this is not a viable solution. N-grams address this issue by utilising only the final N-1 tokens to predict the subsequent one.

As deep neural network solutions gained attention, one of the first architectures employed for text processing was Recurrent Neural Networks[5]. These networks can handle sequential data and learn contextual information from the text. The architecture of an RNN network and how it handles sequential data are visible in Figure 2.1. However, RNNs are not without their limitations, with *exploding and vanishing gradients* being a

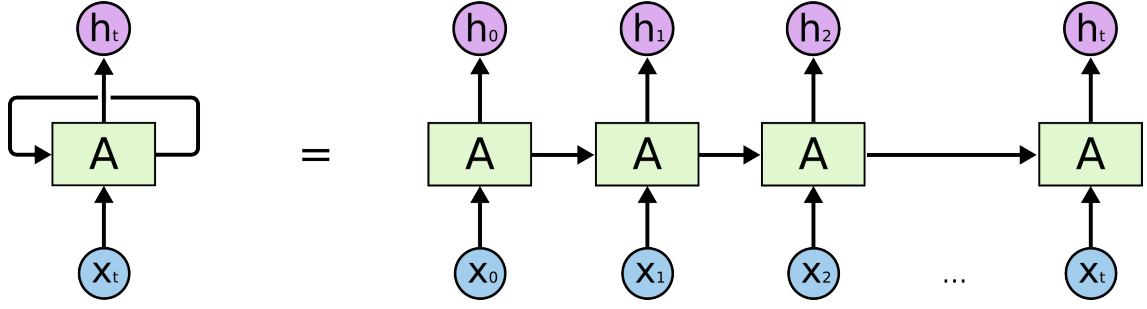


Figure 2.1: Figure from Olah[7].
Architecture and operation of an RNN

significant challenge. This issue was addressed by the introduction of Long-Short Term Memory Network [6] and its variants, such as Gated Recurrent Unit (GRU). Another approach to mitigating the exploding gradient problem is Gradient Clipping, which involves limiting or "clipping" the size of the gradients during the optimization process.

One of the earliest deep language models worthy of mention is the Sequence to Sequence model[8] (Seq2Seq), which is a highly versatile architecture that can be applied to a wide range of NLP downstream tasks, including machine translation and question answering. It comprises an encoder, which is often a bidirectional LSTM, and a decoder, which cannot be a backward RNN. The model employs word embeddings and is trained using a technique known as teacher forcing. This entails the decoder's next input token being the previous ground truth token, rather than the model's previously predicted token. In most cases, these models are trained using a **negative log-likelihood loss** function, as illustrated in Equation 2.1. Figure 2.2 depicts the architecture of a full-stack sequence-to-sequence model.

$$l(\theta) = - \sum_{i=1}^n (y_i \cdot \log \hat{y}_{\theta,i} + (1 - y_i) \cdot \log(1 - \hat{y}_{\theta,i})) \quad (2.1)$$

In addition to the usefulness of Seq2Seq architectures in language modelling, they also have certain limitations. One such limitation is the *bottleneck problem*, which refers to the fact that the decoder of the Seq2Seq model only receives the representation of the last encoder's hidden state. This can result in the decoder being unable to obtain the information it requires for accurate decoding, which may lead to suboptimal performance. This results in the inability of Seq2Seq models to capture long-range dependencies, to handle variable input and output, and to overcome the information bottleneck. This is due to the decoder always receiving the last encoder hidden state or a combination of the last few hidden states, which may result in the loss of important information.

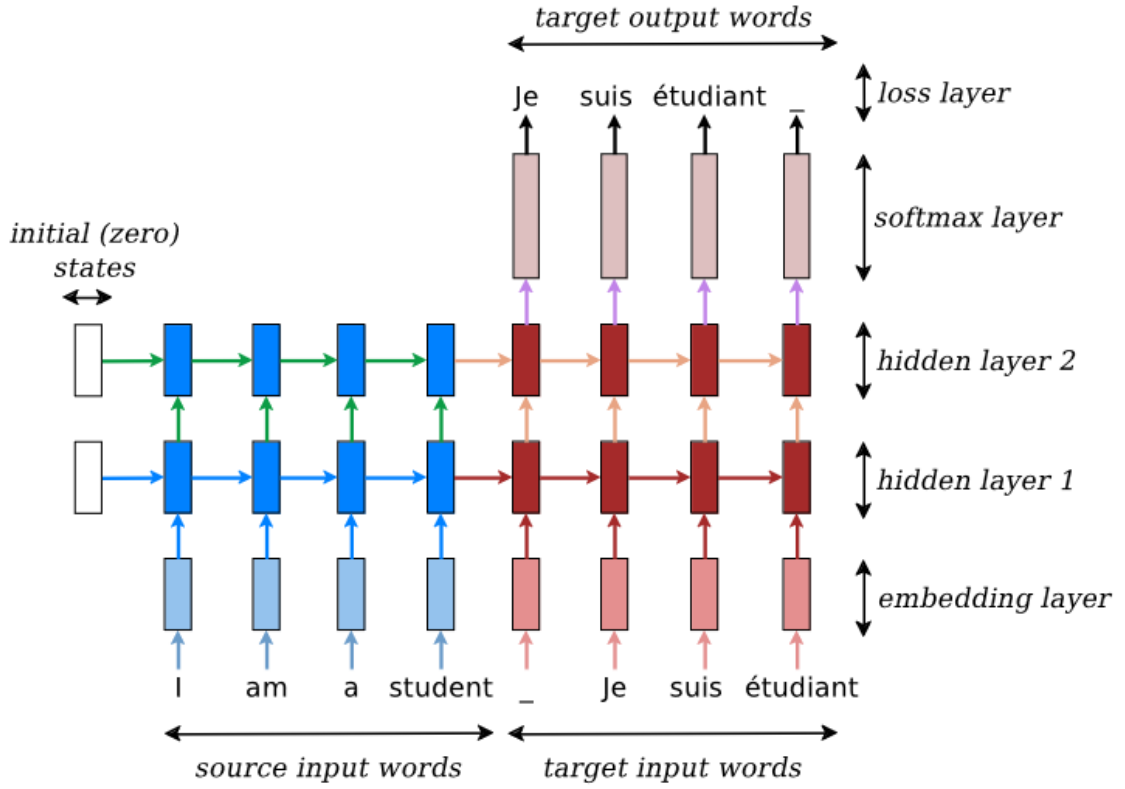


Figure 2.2: Figure from Luong[9].
Architecture of Seq2Seq model

In 2017, the attention mechanism and transformer architecture were introduced by Vaswani et al.[10], which revolutionised the state-of-the-art of natural language processing (NLP). This architecture is currently the most widely used in the field. It assigns weights to each input sequence element based on its relevance to the current decoding step, allowing the model to adjust its focus as it generates the output sequence dynamically. This approach addresses the inherent limitations of Seq2Seq models.

The Transformer's architecture is analogous to the Seq2Seq architecture depicted in Figure 2.3. It comprises an encoder and a decoder. Here, the **attention mechanism** is encapsulated within attention layers, where each layer receives an input text and a query text. The layer then attends to the input based on the query. For each input, the layer calculates the *key* and *value* matrices, while for the query tokens, the *query* matrix is calculated. This enables the model to capture context information and long-range dependencies in a dynamic manner. The Transformer encoder comprises multiple transformer blocks, as illustrated in Figure 2.3, self-attention layers (where the input and query tokens are the same text), normalization layers, addition layers, feed-forward networks and skip connec-

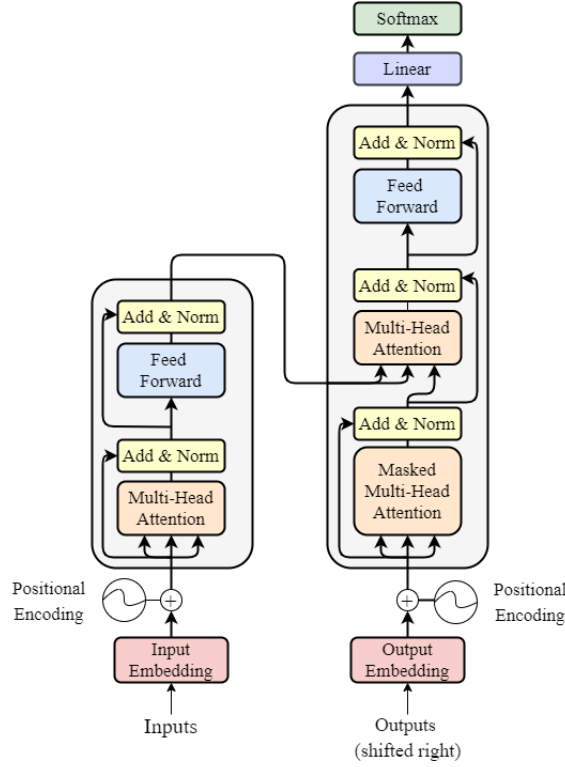


Figure 2.3: Figure from Vaswani et al.[10].
Architecture of full-stack Transformer

tions. The decoder is almost identical, with the exception of the attention layers. Firstly, the model employs a **masked or causal self-attention mechanism** at the outset, whereby future tokens are masked out. Consequently, the model is unable to consider future tokens, with only those situated behind the current one being accessible. Additionally, the decoder incorporates a **cross-attention layer**, which utilises the final output of the transformer encoder as an input, with the query being derived from the decoder. Both the decoder and the encoder employ positional encoding to maintain the sequence's temporal order, compensating for the absence of recurrence in the model. At the conclusion of the decoder, a linear layer and a softmax activation function are employed to determine the continuation probabilities of the current word or token across the entire vocabulary.

Transformers serve as the foundation of large language models (LLMs), which are currently being trained on a variety of text data, including filtered web text, code bases, books, scientific articles, and dialogues. After this initial training, LLMs can be fine-tuned for specific downstream NLP tasks with a smaller amount of task-specific data. OpenAI played a pivotal role in popularising large language models with the release of the *Generative Pre-trained Transformer* (GPT) [11] series. GPT-2 [12], published in 2019, had a significant impact on the research community. It is the model that is most frequently

fine-tuned, due to its availability for research purposes and its comparatively low size in comparison to its successors. Its successor, GPT-3 [13], was the first model publicly available for use through a user interface, called ChatGPT. It has a significant increase in parameters compared to GPT-2's 1.5 billion parameters; GPT-3 has 175 billion parameters. GPT-3 was subsequently enhanced, and OpenAI introduced GPT-4 [14], which became the dominant baseline LLM on the market. This is the reason why I employed the GPT-4 API in my research. Several competitors have entered the market, including Google with Gemini[15] and Meta with LLama[16].

The training goals in LLM training vary between different architectures and companies. However, there is a commonality in that it involves some form of pre-training, where the model learns from a vast quantity of text data. This data is either used to predict the missing, or 'masked', tokens in the data, or to predict the next sentence (as is the case with BERT-like models). Alternatively, the training goal may simply be to predict the next token. Once the large pre-training has been completed, the model is then fine-tuned on specific labelled data, which is often referred to as an 'instruction dataset'. In GPT-3, authors Brown et al. used the pre-trained model to sample answers, which were then ordered by human annotation annotators. These ordered answers were subsequently used to train a reward model, which was then used in conjunction with Reinforcement Learning to further fine-tune the original pre-trained model, as illustrated in Figure 2.4.

With the rise of big language models, a whole new era began when vision-language models emerged. These models employ a novel approach, contrastive learning, which is a type of self-supervised learning. Additionally, they utilise a novel training goal and error function, enabling the processing of images and their fusion with text. These models are capable of understanding pictures, describing them, and finding matching pictures. Such systems combine text with image, and for that, a transformer architecture is used for processing the text and a convolutional neural network, or a Vision Transformer (ViT) [17], is employed for processing the image, introducing the tokenisation of images which is called patching. After the encoding of both image and text, they often combine into a joint embedding. Usually, combining is a learned procedure. It is also worth noting that composite AI solutions, which are typically a combination of text and visual data processing methods, are often included in pipelines. Just as with LLMs, VLMs first undergo pre-training and then fine-tuning. They learn from large datasets that contain pairs of images and text descriptions, such as those found in image-caption datasets. Notable examples include OpenAI's CLIP [18]. They have a wide range of applications and us-

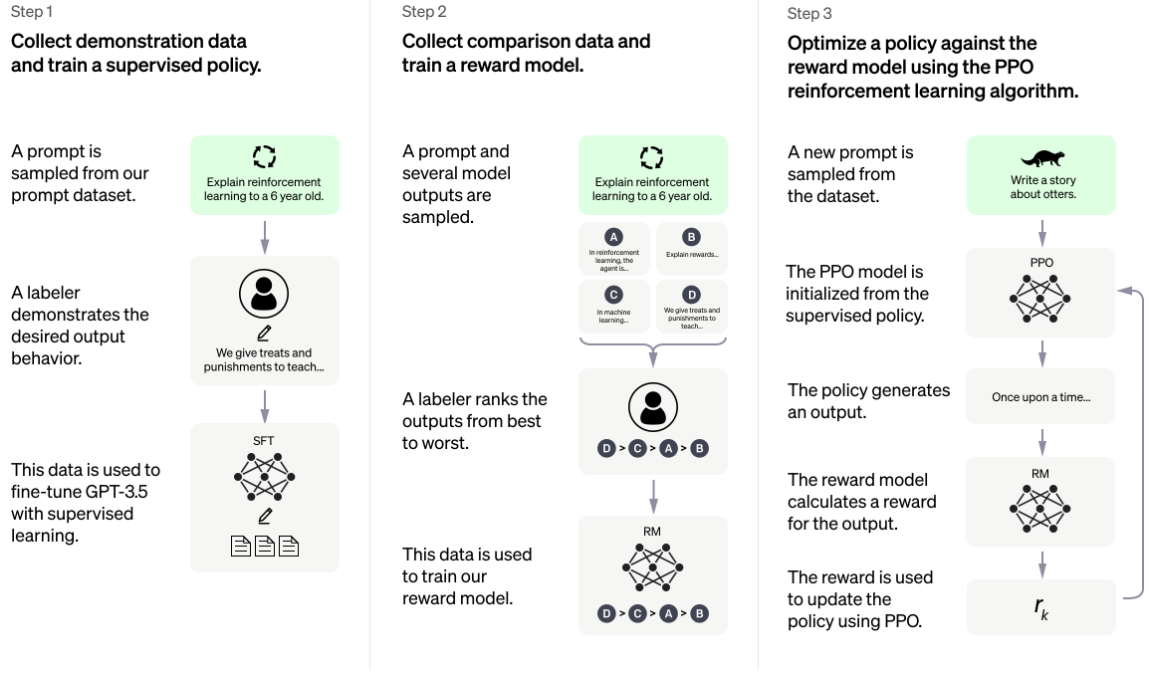


Figure 2.4: Figure from Ouyang et al.[20].
Method of fine-tuning GPT-3

age territories, which are set to expand in the near future. At present, numerous models are capable of processing both images and text simultaneously, thereby enabling visual question-answering. The most widely used model in this domain is GPT-4 Turbo with Vision [14]. One notable open-source alternative is LLaVa [19]. The newest "big" model is Google's Gemini [15], which can handle multi-modality to a greater extent than other models, as it can process text, image and audio. It is important to note that each of these models has its own set of advantages and limitations in comparison to the others. These will be discussed in greater detail in section 3.5.

2.2 Prompting

The growing complexity and scale of LLMs have led to an increased interest in prompt engineering, which aims to enhance model performance and control. The straightforward fine-tuning of an LLM necessitates a considerable investment of resources and a substantial corpus of task-specific data. In contrast, another paradigm for training and fine-tuning language models is prompting, which was first introduced with GPT-3 in the paper [13]. Prompting is a method of explicitly instructing a large language model to generate a more optimal output. Prompt engineering is a crucial aspect of directing LLMs to generate con-

textually relevant and accurate outputs for specific downstream NLP tasks, such as text generation, language understanding, and information retrieval.

Anything that provides relevant information and helps to generate the LLM can be used in a prompt. This could include context information, examples, clear instructions, branching, and step-by-step reasoning. These techniques can help to achieve the best performance from the LLM without any further fine-tuning. Additionally, good prompting in fine-tuning can help to increase the performance of the LLM further.

A number of techniques are employed to assist the LLM in its endeavours:

1. **Clear Instructions and task description:** In order to enhance the quality of a prompt, it is advisable to follow a number of established principles that have been identified as effective guides for the creation of informative and clear prompts. Instructions should be as precise, specific, and detailed as possible to avoid potential hallucinations and misunderstandings that are known to occur in LLMs. Additionally, it is often beneficial to identify the intended audience for the output. In order to create a reliable and precise prompt, it is possible to employ a number of different methods which can be used to modify the prompt in order to make it more suitable for the LLM. These methods include prompt mining, prompt paraphrasing [21], and gradient-based search, whereby the prompt is constructed based on the distance of vocabulary words from the gradient[22]. Another potential approach is a method of prompt generation that addresses the prompt generation problem as a conditional text generation problem, as described in the work of Gao, Fisch, and Chen[23].
2. **Adding examples:** It is evident that large language models (LLMs) are sensitive to prompt details. This sensitivity is influenced by a number of factors, including the number of examples, the order of the examples, the quality of the examples and the task formulation. Consequently, the selection of examples for prompting LLMs is not a trivial task. Some methods attempt to compare prompt data in the embedding space, with the objective of choosing examples from a randomly selected subset or similar ones from the training data [24]. An alternative approach is to train a scoring model with contrastive learning, which can then score pairs in the training data. The top k positive and negative examples are then returned [25].
3. **Adding Different Personas:** It can be argued that different personas can solve different problems in real life. This assertion can be extended to the prompting of the

LLM. It is possible to describe the persona that the LLM needs to adopt in order to fulfil its objectives. The generation of the LLM can be facilitated if it is adopted to a role that is a common role, such as doctor, engineer, or in this case, physical therapist.

4. **Step-By-Step Reasoning:** For more complex tasks requiring reasoning or step-by-step task execution, step-by-step demonstrations can assist the generation process. The most well-known method is called Chain-Of-Thought. It is possible to instruct the LLM to employ step-by-step reasoning or to proceed through the task in a step-by-step manner. Additionally, it is possible to fine-tune a model on the Chain-Of-Thought reasoning dataset. Examples of Chain-Of-Thought reasoning on GPT-3 can be found in the paper of Weng[26]. In addition, Wang et al.[27] was demonstrated that the results can be further improved by sampling several answers and then taking an ensemble of them (e.g., averaging, majority vote).
5. **Context/Knowledge generation:** Additional information relevant to the potential answer to the question can improve the LLM's output. Furthermore, LLMs can be employed to generate data related to the question, which can assist in answering it, as discussed in the paper of Liu et al.[28].
6. **Branching:** The Chain-of-Thought method is a useful approach for prompting, but it has been observed that the use of a single branch is not an accurate reflection of how people think. Instead, people often consider multiple branches, discard some, and then continue with another. To address this issue, a new method, Tree-Of-Thoughts [29], has emerged. This method incorporates the capabilities of Chain-of-Thought, including the consideration of multiple branches, and also includes a control mechanism that determines which branch will be deleted and which one will be continued. The Graph-Of-Thoughts [30] method represents an extension of the ToT approach, incorporating aggregation between branches.
7. **Negative Prompts:** The use of negative prompts in terms of prompting techniques may result in constraints to the model, which are not desired in the output. It is therefore crucial to include negative prompts in the model, as this helps to ensure greater consistency in the output of the LLM. Furthermore, negative prompts help to prevent the model from generating additional, irrelevant text, and they also help to limit the potential attributes of the output, such as sentence length and word us-

age. The utilisation of negative prompts serves to prevent the occurrence of certain hallucinations exhibited by the LLM. This prompt addition is a highly effective and widely utilised addition to the field.

It should be noted that there are several additional techniques that could be employed to enhance the performance of the LLM. However, these techniques were not utilised in the present study, and therefore, familiarity with them is not essential to understand the core objective.

The beneficial effects of prompting Large Language Models can be attributed to multiple factors. Primarily, it represents a viable alternative to standard fine-tuning, which necessitates the utilisation of greater resources for training. With the explicit guidance provided by the prompt, LLM training can be more targeted and efficient. Furthermore, the use of more specified prompts enables the system to produce more consistent outputs, which can assist in addressing the issue of hallucinations, misbehaviour, and offensive responses from the system. One of the disadvantages of LLMs is the occurrence of hallucinations. Xu, Jain, and Kankanhalli[31] demonstrated that hallucinations in LLMs cannot be fully corrected. Additionally, the study identified potential solutions and methods that can be employed to mitigate the occurrence of hallucinations in LLMs.

2.3 Dataset used

The dataset utilised in this study is a unique collection of **400 videos**, collected by Baranyi et al.[32], each of which depicts a different physical therapy exercise. The majority of the videos were recorded using a single setup, although a subset of the exercise videos were re-recorded with different camera setups and environments. For each exercise, there is a recording of the therapist performing the exercise correctly, and several videos showing different errors. In each video, only one specific error is shown, accompanied by a textual description. The number of error videos for each exercise ranges from **5 to 10**.

The database comprises discussions with patients, therapist-specific queries, videos, exercise descriptions, error descriptions and error corrections. From these listed components, I utilised all data collected in the dataset, with the exception of discussions between the patient and therapist. Potential errors and exercise descriptions were incorporated into the prompt, error corrections were employed as ground truth data, and images were pro-

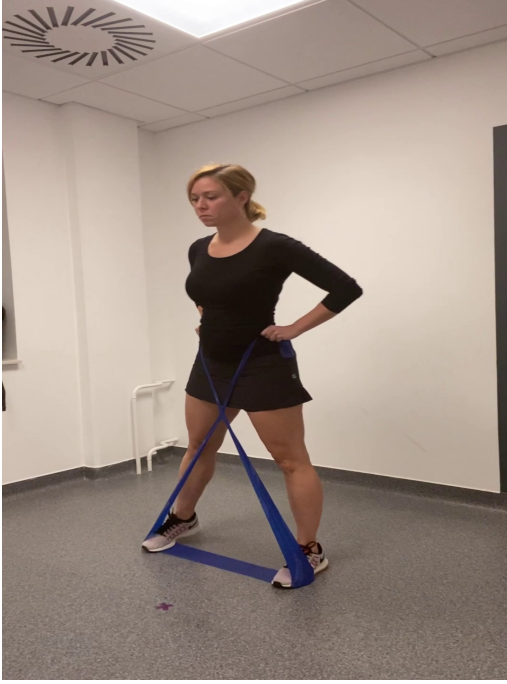


Figure 2.5: Therapist doing the exercise **correctly**

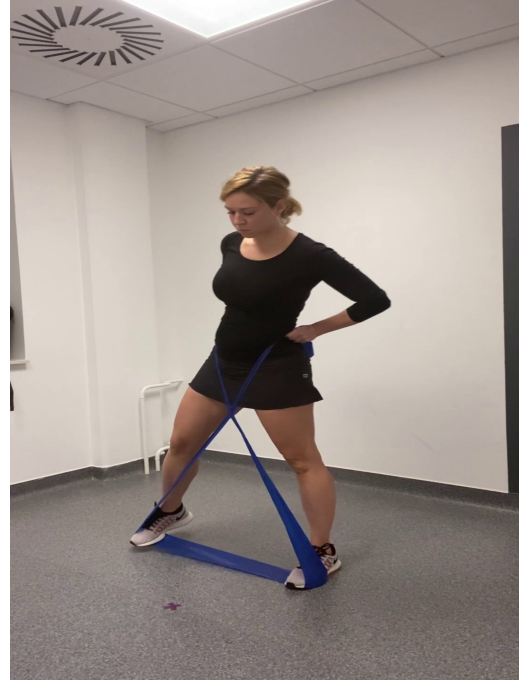


Figure 2.6: Therapist doing the exercise **incorrectly**

vided to the LLM. Further details regarding the utilisation of these components will be provided in section 4.3.

The majority of errors observed in these videos relate to the incorrect posture of the patient. These include poor foot posture, bent knees, wide steps, a sloping back and waist, and poor spine posture. Additionally, there are instances where the speed of the exercise does not match the prescribed pace. However, this study focuses solely on those videos that contain posture errors.

Figure 2.5 and Figure 2.6 illustrate examples of images from the dataset. The exercise in question is **lateral walking with an extended knee**, and the therapist performs the exercise correctly with the knees extended and an overall good posture on Figure 2.5. Conversely, in Figure 2.6, the therapist makes an error, namely bending the stepping leg's knee for demonstration purposes.

2.4 Dynamic Time Warping (DTW)

The significance of Dynamic Time Warping (DTW) in my study cannot be overstated. With the poses estimated by human pose estimation software, DTW is required to synchronize the patients' videos with the physical therapists' videos, thus enabling the frame

pairs in these videos to be compared. A good alignment between the videos is crucial for selecting the most significant error from the frame pairs.

Dynamic Time Warping [33] is a technique employed to assess the degree of similarity between two sequences that may exhibit temporal or speed variations. It is employed in a number of fields, including speech recognition, pattern recognition, and time series analysis. It is a valuable tool when it is not possible to ascertain the degree of similarity or dissimilarity between two time series due to their differing lengths or speeds. In this instance, the previously employed metrics, such as Euclidean distance, cosine similarity, Pearson correlation and time-lagged cross-correlation, are unable to be utilised, due to the difference in length between the sequences. The method operates by identifying an optimal alignment between the two sequences, enabling the time axis to be stretched or compressed as required. The objective is to identify the optimal alignment between the two sequences.

The objective is to identify the optimal alignment between *sequence A* and *sequence B* by warping them in time in order to minimise the dissimilarity between corresponding elements. The computation of the alignment matrix is as follows: a two-dimensional matrix D of size $n \times m$ is created, where $D[i][j]$ denotes the dissimilarity between elements. The Equation 2.2[34] represents the cost function of Dynamic Time warping. The cost function associated with a warping path is computed concerning the local cost matrix (which represents all pairwise distances).

$$c_p(A, B) = \sum_{l=1}^L c(a_{n_l}, b_{m_l}) \quad (2.2)$$

where:

p - is the alignment path

a_{n_l} - is the first sequence element in the l^{th} alignment pair

b_{m_l} - is the second sequence element in the l^{th} alignment pair

With the cost function mentioned in Equation 2.2, starting from the upper left corner the algorithm fills the cells of the matrix using the following rules mentioned in the work of Senin[34]:

1. First Row:

$$D(1, j) = \sum_{k=1}^j c(a_1, b_k), j \in [1, M] \quad (2.3)$$

2. First Column:

$$D(i, 1) = \sum_{k=1}^i c(a_k, b_1), i \in [1, N] \quad (2.4)$$

3. All other Elements:

$$D(i, j) = \min\{D(i-1, j-1), D(i-1, j), D(i, j-1)\} + c(a_i, b_j) \\ i \in [1, N], j \in [1, M] \quad (2.5)$$

where:

D - cost matrix

N - length of the first sequence

M - length of the second sequence

The optimal alignment can be identified by tracing back from the bottom right corner of the cost matrix to the upper left corner. At each step, the cell with the smallest dissimilarity to its adjacent cells (in the left, top, and diagonal directions) should be selected. This warping path represents the optimal alignment between the two sequences.

The time complexity of the DTW algorithm is $\mathcal{O}(n \times m)$, where n and m are the lengths of the input sequences. However, various optimisations and constraints can be applied to reduce the computational complexity in practice.

In addition to the benefits of DTW in comparing disparate temporal series or sequences and its resilience to temporal and speed variations, it is important to acknowledge the limitations of this approach.

Firstly, the cost of DTW is dependent on the length of the sequences. In the case of lengthy data sequences, the computational expense can be considerable. However, there are alternative approaches that aim to identify the optimal path and reduce the computational cost. Another crucial aspect of DTW is its sensitivity to outliers. Additionally, it should be noted that DTW is unable to capture semantic similarities between the sequences. Senin[34] are summarizing and showing various customization options for the original algorithm.

2.5 Skinned Multi-Person Linear Model (SMPL)

The Skinned Multi-Person Linear (SMPL) model is a realistic 3D representation of the human body, derived from thousands of 3D body scans. It employs a method of skinning and blend shapes, which enables the model to be easily manipulated and animated using different learnable parameters. The SMPL model subdivides the human body into two components: one component is influenced by an individual's unique identity, while the other is contingent upon the body's position (e.g., bending or stretching). To represent these components, a method called vertex-based skinning with corrective blend shapes is employed. Each corrective blend shape is akin to a set of instructions for moving specific points on the body's surface.

Components:

1. Template Mesh: 3D mesh defining the 3D topology used by SMPL-model
2. Shape Components (betas β): identity-independent mesh descriptors which can be added to template mesh to generate different body shapes and realistic human identities. Possible to describe the most important body parameters like height, weight, waist size, and shoulder breadth with only ten components
3. Pose Components (thetas θ): Pose-dependent mesh surface descriptors, added on top of the template mesh and shape components to represent muscle-based deformations and rotations for any joint.
4. Dynamics Components (phis ϕ): Soft-tissue-dependent mesh surface descriptor vectors added on top of the other layers, which represent the soft-tissue deformations during fast motions, like the jiggling of fat-tissue during running.

The visualization and the effects of the components are visualized on Figure 2.7.

The SMPL model has demonstrated its significance over time and has applicability in a multitude of significant domains. It provides a realistic and anatomically feasible representation of the human body, which renders it applicable in animations and simulations. It offers a compact and parametrized representation of the human body pose and shape, which facilitates manipulation and storage of human body geometry. However, SMPL models also exhibit potential limitations. The model is not suitable for the representation of pregnant women, amputees or any other abnormal deformations of the human body.

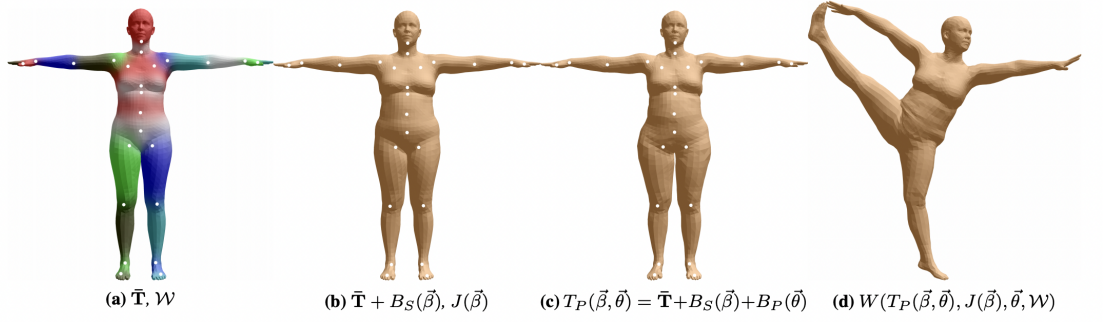


Figure 2.7: Figure from publication Loper et al.[35]

General SMPL models are vulnerable to generating body shapes with self-intersection body parts, which impairs the efficacy of the SMPL model. Consequently, the authors Wu et al.[36] introduced a self-interpenetration term into statistical body shape models like SMPL to address this shortcoming of these models.

Another enhancement of the SMPL model was proposed by Keller et al.[37], who developed an upgrade to the original SMPL model, named SKEL. This improvement connects parametric body shapes with biomechanical skeletons, thereby enhancing the SMPL model. The input for SKEL is an SMPL mesh, which is then refined by the SKEL model through the incorporation of diverse body constraints. This process ensures greater accuracy and a more human-like appearance for the skeleton.

Chapter 3

Related Works

3.1 LLMs in health care

The potential applications of large language models (LLMs) are a topic of considerable interest in a number of different fields, including healthcare solutions. Researchers have made contributions to a range of tasks and datasets in order to assess the potential usability of LLMs and vision language models (VLLMs) in a field which requires a certain level of background knowledge and expertise in order to produce reliable answers to tasks such as visual question answering in the medical domain, and medical education.

A number of papers have been published in which authors discuss the importance of a suitable prompt in order to enhance the performance of an LLM in a specific domain, including healthcare. Image understanding is a crucial task, and LLMs must be able to comprehend minor details and relationships on images. This is also important in medical images, where images are often highly specialised, such as X-rays. Qin et al. [38] evaluated the capabilities of GLIP using 13 different datasets. GLIP is a vision language model, in medical image understanding. They emphasised the significance of well-designed prompts and domain knowledge, which can bridge the gap between domains. Their findings demonstrated that task-specific prompts outperformed default prompts in zero-shot performance, and their fine-tuned models were more effective than supervised solutions. These results motivated me to develop my own well-defined prompts for the rehabilitation scenario described in chapter 1, which incorporate domain knowledge as well, as discussed in section 2.3.

The formulation of effective prompts is crucial in the field of healthcare. This domain necessitates the use of precise prompts and external domain knowledge, which are not

included in LLMs. However, the fine-tuning of a model using medical datasets requires a significant investment of resources and prompts to handle task-specific questions. In a study conducted by Zuccon and Koopman[39], it was found that the inclusion of external domain-specific knowledge in prompts can enhance the knowledge encoded in the model, particularly when providing health advice to consumers using GPT-4.

Heston and Khun[40] are focusing on the use of Generative Language Models (GLMs) in medical education. They discuss the usability of GLMs in medical education, for example, in simulating realistic patient scenarios and making endless potential questions for medical students. They also list prompting techniques such as zero-shot, few-shot, and how to write a good prompt. Nevertheless, they also identify potential issues, such as the possibility of misleading or inappropriate information, which could result in inconsistent and misleading outputs. Furthermore, in their paper, they highlight their concerns regarding the use of GLMs, including ensuring accuracy, mitigating bias, and maintaining privacy.

Obtaining reliable information from medical professionals is a crucial aspect of medical education, which is the objective of my study. From this perspective, it is a significant challenge in the medical field to provide medical advice to patients on medical matters. This will be discussed in greater detail in chapter 6. However, hospital resources are constrained by the availability of individual medical professionals and different specialist doctors. In the paper of Cox et al.[41], authors explore the potential of GPT-4 in the case of Blepharoplasties (plastic surgery aimed at the aesthetics and abnormalities of the eyelids). to provide accurate and safe medical information to patients, as well as supporting healthcare professionals in educating patients in this scenario. This paper's goal is highly correlated with my study's goal of helping rehabilitation professionals and educating patients on how to perform exercises more effectively. Nevertheless, this study can be employed to instruct physiotherapists in decision-making, in addition to educating patients, who are predominantly elderly individuals.

A number of studies have been conducted in the field of visual question answering, which is aligned with the objective of my own research. These studies have sought to make progress on a number of benchmarks and datasets, utilising both unimodal and multimodal data as input. Nori et al.[42] evaluated the performance of the GPT-4 on the MedQA dataset[43], which is based on multiple-choice questions from the United States Medical License Examination (USMLE) [44], a three-step examination programme used to assess clinical skills and consistency in the US. The questions in this dataset

are designed to assess the ability of medical professionals to apply their knowledge in a variety of clinical contexts. Additionally, the efficacy of the MultiMedQA benchmark datasets was evaluated. The findings indicate that GPT-4 outperforms the passing score of USMLE by 20 points without any specialised prompt crafting. This evidence substantiates the assertion that GPT-4 exhibits superior performance compared to previous general-purpose models (such as GPT-3.5) and models trained on medical domain-specific data (e.g., Med-PaLM[45] and Flan-PaLM[46]). The findings indicate that general-purpose language models can enhance the performance of medical benchmarks, even surpassing models that have been specifically trained for that purpose.

In a similar vein, Yang et al.[47] evaluated GPT-4V on the USMLE dataset with a view to testing the accuracy and explanation quality of the model. Having images at their disposal, they compared the performance of three models: (a) GPT-4V, (b) GPT-4 without images and (c) GPT3.5. The results demonstrated that GPT-4V outperformed the other methods, with an accuracy above **90%**. Furthermore, the results were compared to those of medical students, which revealed that students' performance decreased on different difficulty levels, whereas GPT-4V performance remained relatively consistent.

In a comprehensive study on visual question answering in the medical domain, authors Lin et al.[48] provide an overview of the history of medical VQA and potential research directions for further improvement. They also mention the publicly available medical VQA datasets. The paper provides a good overview of the methods and datasets used for medical VQA, with a particular focus on researchers. The authors highlight the current challenges of VQA, emphasising the crucial problem of question diversity in such datasets and models. Future research should aim to identify useful question categories for practical purposes.

The research community sought to emphasise the usability of Large Language Models (LLMs) in medical reasoning tasks. In their examination of the capabilities of various LLMs, including GP3.5 and LLaMa-2 [49]. Liévin et al.[50] highlight the ongoing efforts to advance language models for biomedical applications. The researchers evaluated a number of medical benchmarks, including MedQA-USMLE and MEDMCQA, and employed a variety of prompting scenarios, such as Chain-Of-Thought, few-shot, and retrieval augmentation. Their findings indicated that GPT outperforms LLaMa-2 70B on these medical reasoning datasets.

A significant body of research has been conducted on the comparison of the performance of LLMs with the knowledge of medical professionals. These contributions

demonstrate a high correlation with my own study, which compares the corrections proposed by psychologists with those generated by GPT-4. The study from Artamonov et al.[51] compares the responses generated by GPT-4 with those of a consensus of different statements in the case of Anterior Shoulder disability. The same question was posed to GPT-4 as was posed to the consensus. GPT-4 and shoulder surgeons then compared the similarity of GPT-4 responses and consensus statements. The results showed that there was a 58% agreement between the GPT-4 and consensus statements.

Guerra et al.[52] compared GPT-4 responses with those of medical students and residents on a series of neurosurgery-related questions. They employed both unimodal and multimodal questions in their analysis, and GPT-4 demonstrated superior performance across both groups. The authors recommended further investigation into the potential applications of GPT-4 in the field of medical decision-making.

A number of studies have been conducted which provide an overview of the progress of the research community in recent years. One such study, from Wang et al.[53], focuses on the current applications and NLP methods in healthcare and provides advice to health professionals on the importance of prompt engineering in the medical domain.

Meskó[54] presents a series of scenarios in which the future impact of MultiModal Large Language Models (MM-LLMs) in the healthcare domain can be observed. Nevertheless, they also emphasised that the human element in medicine will remain indispensable, which underscores the necessity for human-machine collaboration. This collaboration will be further enhanced and refined in the future, and this enhancement is the objective of my study.

3.2 Rehabilitation with Artificial Intelligence

Firstly, in the past, there were numerous contributions to physical rehabilitation utilising diverse machine learning models, encompassing classical methods such as SVM and LSTM, as well as more advanced CNN-based, transformer- and LLM-based solutions currently. The work of Mennella et al.[55] is a systematic literature review which summarises 35 different papers from 2012 to 2022 in terms of physical rehabilitation. The main focus of these algorithms is to support remote and smart monitoring during a physical rehabilitation exercise. The algorithms were categorised according to their different tasks, which included activity recognition, movement classification, and clinical status prediction. The authors also classified the methods into supervised and unsupervised cat-

egories. The results for each method were presented, including accuracy. Additionally, the authors discussed the limitations of these methods, such as the lack of representative samples, limited generalizability, and poor reproducibility due to the use of data collected in limited or different conditions. The application of machine learning-based solutions necessitates the availability of well-annotated datasets and a substantial quantity of data to facilitate generalisation. These solutions lack the capacity to utilise any additional global context information that could enhance prediction accuracy. Further studies will clarify the outcomes of research in physical rehabilitation with general-purpose Large Language Models, which may possess global information about specific tasks.

Another review of machine learning and deep learning solutions in physical rehabilitation is the paper of Sardari et al.[56], which covers different works from the literature from 2011 to 2021. The authors discuss the methods used for capturing accurate skeleton data in the aim of physical therapy exercise monitoring. Furthermore, the review encompassed machine learning methodologies, including Support Vector Machine and Random Forest Classifier, as well as deep learning approaches, such as Convolutional Neural Networks, Recurrent Neural Networks, and Graph Neural Networks, which were evaluated in different papers. These papers assessed the efficacy of these methods in human action recognition and estimation task-based datasets.

There are limited solutions in the literature where the Vision Language model is used to assist physical therapists in providing advice to patients on any issue. One of the reasons for this is that such an application is subject to human experimentation. In the context of human experimentation, there are multiple regulations, including ethical permissions, GDPRs and written agreements from the patients. This requires managing complex regulatory frameworks, identifying a diverse pool of volunteers, and mitigating risks, all while ensuring there are enough participants to produce reliable outcomes. All research that involves human experiments must submit the requisite ethical documents for review by the Institutional Review Board (IRB).

The work of Almeida et al.[57] is a chat-based physical rehabilitation application called Physio, which can make an initial diagnosis and recommend rehabilitation exercises. It is an example of the use of external knowledge in the field of physical rehabilitation education and advice to patients. The model was employed by the aforementioned authors to offer patients multiple opportunities for explanation and discussion of their problems, as well as recommendations regarding exercises and medication. The pipeline employed by the authors involved filtering English prompts using the language model and

a prompt template. Furthermore, a condition identification step was employed, whereby the authors identified the condition of the user by utilising few-shot learning with LLMs. The authors then linked the prompt to one of the external knowledge base entries. The linked connection was employed to facilitate answer generation and the recommendation of exercise and medication. Potential documents were retrieved from a reliable webpage dataset using a retrieval model, after which the optimal number of references included in the final answer was experimentally determined.

Another recent paper on patient advising using Large Language Models is the work of Aliyeva et al.[58], which utilised GPT-4 in the case of Cochlear implantation, specifically to assist postoperative cochlear implant patients. The authors' objective was to provide helpful and relevant information in situations where the number of healthcare professionals is limited, particularly in the case of patients who are unable to visit the hospital regularly due to distance or mobility issues. The importance of this situation is discussed in detail in the introduction. To this end, they provided five different questions to GPT-4, and the responses were analysed using a range of metrics, including response time, relevance and accuracy. The results demonstrate that GPT-4 is capable of generating meaningful and relevant information within a specific domain, such as postoperative Cochlear implantation, and of providing useful medical information to non-medical audiences without the need for any fine-tuning or the use of prompting techniques.

The work of Li et al.[59] is relevant to my study because they contributed to holding an experiment analysing the efficiency, availability and personalisation of GPT-4 and wearable devices in the rehabilitation of sarcopenia and osteoarthritis, where exercise is a key aspect of the recovery process. The experiment was conducted on 278 people, who were separated into intervention and control groups. The intervention group employed the GPT-4 and wearable device-assisted method, whereby GPT-4 generated personalised exercises and utilised the wearable devices to monitor and guide the patients. The control group underwent traditional face-to-face clinical therapy. They underwent checkpoints one month, three months and six months after the initiation, and employed multiple outcome measures for evaluation. The results demonstrate that the integration of GPT-4 with wearable devices has the potential to enhance the accessibility of these services. Furthermore, they highlight the potential of these methods to facilitate the development of more personalised and efficient rehabilitation services.

Tamantini et al.[60] have made a significant contribution to the field of physical rehabilitation, including the development of rehabilitation robots. Their approach differs from

that of other researchers in the field in that it incorporates natural verbal communication. They sought to simulate the natural interaction between patients and therapists. Their work was validated in a laboratory environment with 20 healthy patients. They employed two different robot systems, one with an expert language system and one without any verbal capability. The authors cite the potential utility of LLMs, such as BART[61] and GPT, but justify their exclusion due to the aforementioned hallucination issues. Their findings indicate that natural language interactions enhance the quality of interaction, resulting in reduced exercise repetition times and improvements in emotional terms. The study offers valuable insights into human-machine interaction (HMI) in rehabilitation scenarios. However, recent advances in LLMs, including fine-tuning, alignment, and prompting techniques, have significantly reduced the occurrence of hallucinations.

The most closely related work to my own contribution to the literature are those that focus on the estimation and correction of yoga poses, employing both online and offline approaches. Dhakate et al. and [62] employ a computer vision-based approach to the estimation of yoga poses, providing users with personalised feedback. They utilise PoseNet[63] to estimate key points and OpenCV[64] and MediaPipe[65] for performance tracking. The system achieved an accuracy of **80%** on the test dataset and real-time performance. It is suitable for use in the home, at the gym, and in yoga studios. However, the authors also identified potential limitations, including complex pose recognition, occlusion, and partial visibility. Furthermore, they noted that customising feedback for users' levels and goals may be challenging. A similar study on this topic is the work by Gajbhiye et al.[66]. The authors present an overview of the various pose estimation approaches that can be employed in an Android application. They utilise PostNet, a machine learning model, to detect and correct yoga poses.

3.3 3D Human Pose Detection

Two principal types of human pose estimator systems operate with distinct methodologies. Skeletal pose estimators analyse images or videos to detect and track human body joints, thereby providing a skeletal representation of body poses. They typically rely on keypoint detection techniques to identify points corresponding to joints such as elbows, knees, and wrists, which allows for body movement recognition action analysis, and human-computer interaction. Conversely, volumetric pose estimators generate volumetric representations of human body poses, capturing the spatial distribution of body

parts in three-dimensional space. They reconstruct the human body’s volumetric structure from input images or videos, enabling more accurate pose estimation, especially in challenging scenarios with occlusions or complex poses. Volumetric pose estimators find applications in areas such as augmented reality, biomechanics, and virtual try-on systems.

A novel method for 3D human pose estimation was introduced in 2020 by Sarandi et al.[67], which combined 3D metric-scale heatmaps with 2D image space to estimate poses accurately. This method, known as MeTRAbs, represents an absolute 3D human pose estimation method using metric-scale truncation-robust heatmaps. In 2020, the method achieved state-of-the-art performance on a range of datasets, demonstrating the value of supervision via absolute pose loss for precise localisation. Furthermore, the authors demonstrated that MeTRAbs are robust to joint truncation, outperforming previous methods on truncated images.

A more recent paper, which builds upon the previous MeTRAbs model, Sárándi, Hermans, and Leibe[68] introduces a novel approach to affine-combining autoencoder (ACAE) for dimensionality reduction in 3D human pose estimation. Their approach facilitates information sharing between different datasets and poses formats, including SMPL and COCO. A total of 28 3D human pose datasets were utilised in this study, with the method demonstrating superior performance to previous works on a range of benchmarks. These findings highlight the value of employing multiple datasets in the context of pose estimation, particularly when dealing with inaccurate data.

In recent years, 3D human pose estimation has emerged as a significant technology in the field of Computer Vision. By accurately reconstructing human movements in 3D space from images or videos, it provides valuable insights across a range of domains. Its applications are diverse and impactful, including in healthcare and sports biomechanics, surveillance, and entertainment.

One of the most significant contributions to the field of human pose representation is the work of Loper et al.[35], where the authors proposed a novel body pose representation called SMPL. The details and components of this model were discussed in section 2.5. Furthermore, they made the model available for research purposes. The model was evaluated using two metrics: model generalisation and pose generalisation. The results were then compared with those obtained using BlendScape, the work of Hirshberg et al.[69], which was trained on a similar dataset to that used for SMPL. The quality of the results obtained was deemed satisfactory, and the SMPL model is now regarded as a standard in the field of pose representation.

In order to enhance the SMPL model, SKEL, a recently developed solution, has been proposed by Keller et al. [37]. This solution employs a novel approach to body modelling, whereby parametric body shapes are connected to biomechanical skeletons. The incorporation of accurate parameters for skin and bone shapes enables more precise human body modelling. In contrast to other models, such as SMPL, SKEL offers a more limited but more realistic range of pose options, rendering it more suitable for the analysis of videos. The model learns from a diverse range of skeletons in the BioAMASS dataset, a dataset created from the AMASS [70] dataset, thereby refining its modelling. The input to SKEL is an SMPL mesh, and it improves the accuracy of the given mesh. It is compatible with SMPL’s shape space and datasets, thereby enhancing the accuracy of human body modelling.

Li et al.[71] propose a state-of-the-art model that produces SMPL mesh, shape and pose components using both image and video input. They introduced HybrIK[72] and its extension HybrIK-X[71], a novel hybrid inverse kinematics solution for 3D keypoint estimation and body mesh recovery. HybrIK-X builds upon HybrIK by incorporating the estimation of hands and faces, and achieves whole-body pose estimation with a one-stage model. Their experiments demonstrate that HybrIK and HybrIK-X outperform existing methods in body-only, hand-only, and whole-body scenarios, providing a pixel-aligned, realistic whole-body mesh recovery approach.

3.4 Time Series Synchronization for aligning videos

Time series alignment is the process of aligning multiple datasets to enable comparison of events or patterns occurring at different times or containing different offsets in the data. Berndt and Clifford[33] defined Dynamic Time Warping (DTW), a dynamic programming algorithm that can handle temporal variations, offsets in time, and variational length time series. In contrast to traditional distance measures, such as Euclidean distance, DTW is capable of accommodating temporal distortions by stretching or compressing the time axis in order to identify an optimal alignment between the sequences. However, it should be noted that DTW also has its own set of limitations. These include its quadratic time complexity, which is dependent on the length of the time series, its memory usage, and its sensitivity to noise and outliers.

A number of solutions have been proposed to address the quadratic complexity and sensitivity to outliers. The FastDTW algorithm, developed by Salvador and Chan[73],

approximates regular DTW and reduces its quadratic time and memory complexity. The process begins with the initial downsampling of the time series at a very low resolution. This is followed by the identification of a warp path and the subsequent projection of this onto increasingly higher resolutions, with each step involving refinement. The resolution is a smaller time series than the original, representing the same curve as accurately as possible with fewer data points. The algorithm comprises the following steps: **coarsening**, which reduces the time series to a smaller, but accurate representation; **projection**, which utilises the minimum-distance warp path from a lower resolution as a starting point for a higher resolution; and **refinement**, which involves adjusting the projected warp path locally at each higher resolution. Subsequently, the distance between segments is computed at each resolution level, starting from coarse to fine resolutions. The efficacy of FastDTW was evaluated in comparison with two other DTW approximations, and it was found to outperform both of them. However, it is an approximation of DTW, and while it is faster, it also loses accuracy. Nevertheless, despite this statement, FastDTW is demonstrating superior scaling capabilities on large datasets.

ShapeDTW, a variant of the DTW algorithm is developed by Zhao and Itti[74], which attempts to address the issue of DTW computing optimal global alignment but fails to achieve locally sensible matching consistently. To this end, the ShapeDTW is a DTW algorithm that incorporates the pairing of locally similar structures to circumvent the matching of points with neighbouring structures. The authors utilised shapeDTW on signal pairs, achieving favourable outcomes on a range of datasets, including surpassing the distances achieved by DTW, which was previously regarded as the optimal distance measure for audio signals.

DTW methods are employed in the time series domain for sequence similarity analysis. One such method is the Global Alignment Kernel (GAK), which was developed by Cuturi et al.[75] and is a kernel method for handling time series data. Kernel methods, which encompass popular machine learning algorithms like Support Vector Machines, extend the well-known Dynamic Time Warping (DTW) algorithm. The authors propose an alternative approach to the calculation of the score spanned by all possible alignments. Instead, they take a smoothed version of their maximum and derive a kernel from this formulation. They demonstrate that this kernel is positive definite and can be tuned effectively on a task such as speech recognition.

In the context of this study, the optimal algorithm for matching is the standard Dynamic Time Warping (DTW). More details will be discussed in section 4.2

3.5 Large/Vision Language Models

Achiam et al.[14] introduced GPT-4, the work of OpenAI, which achieves human-level performance on multiple benchmarks, outperforming existing models in multiple languages. It is designed with a scalable infrastructure and optimisation methods that enable accurate performance predictions with fewer computational resources. Despite its enhanced capabilities, there is a need for improved security and tuning. The authors monitored optimisation, infrastructure, usability, correctness and data management. The evaluations included assessments of the model's vision, economic impact, security, policy, privacy and non-proliferation capabilities, thereby demonstrating a comprehensive development approach. The model is available for developers and can be used from programming code through the OpenAI API. The API allows users to easily switch between models, enabling them to use the latest model without modifying the code.

The Gemini family of multimodal models published by Team et al.[15], including Ultra, Pro, and Nano sizes, is designed for diverse applications. Gemini deals with image, text, and audio, and has achieved state-of-the-art performance in 30 out of 32 quality benchmarks, mentioned in the paper. The models demonstrate advancements in text, code, image, audio, and video understanding, producing new state-of-the-art results in multiple domains. Two variants of Gemini models are produced: Gemini Apps models, which are optimized for conversational AI services, and Gemini API models, which are accessible through Google AI Studio and Cloud Vertex AI. However, at the time of writing this study, only a free version is available from the API models with limited resources, and the responses of people prompts are used to enhance the performance of the model. On 14 May, they will make the pay-as-you-go pricing available, with improved conditions. Further information is discussed in section 4.3.

The paper of Liu et al.[19] introduces visual instruction tuning, which extends Large Language Models (LLMs) with machine-generated instruction-following data for improved zero-shot capabilities. LLaVA (Large Language and Vision Assistant) is a multimodal model combining vision and language understanding. LLaVA has been trained using GPT-4 to generate multimodal instruction-following data and fine-tuned to improve visual and speech understanding. The model has been trained end-to-end, combining vision and LLM for multimodal processing, with the objective of improving zero-shot capabilities. The training data and code are publicly available for further research.

Chapter 4

Methods

This section will present the technical aspects of the study, including the sections outlined in the abstract.

The principal processes of this study can be divided into four sections: human pose detection in section 4.1, video synchronisation in section 4.2, prompt engineering in section 4.3 and evaluating GPT-4 answers in section 4.4. The following sections will clarify the individual components, discussing the possible ways to address the issue, their respective advantages and disadvantages, and how these approaches were utilized in this investigation.

4.1 Human Pose Detection

The initial stage, as depicted in Figure 4.1, entails the detection of human poses within the videos comprising our dataset, as referenced in section 2.3. This is achieved through the utilisation of the MeTRAbs tool, as detailed in section 3.3. For the training of MeTRAbs, a multitude of datasets and skeleton representations were employed, with the model subsequently acquiring the requisite knowledge and understanding of these concepts. One of the skeleton representations applied by MeTRAbs is the standard SMPL skeleton representation, containing 24 different body joints (SMPL-24 in the model), which were used in this study. In the context of this study, other joints like facial and hand joints (i.e. multiple finger joints) are not needed, as the focus is solely on the person's body posture. MeTRAbs calculates both the 2-dimensional and 3-dimensional joint coordinates for all video frames. These coordinates in three dimensions encompass 2.5-dimensional volumetric heatmaps. Here, the X and Y axes relate to the image space, while

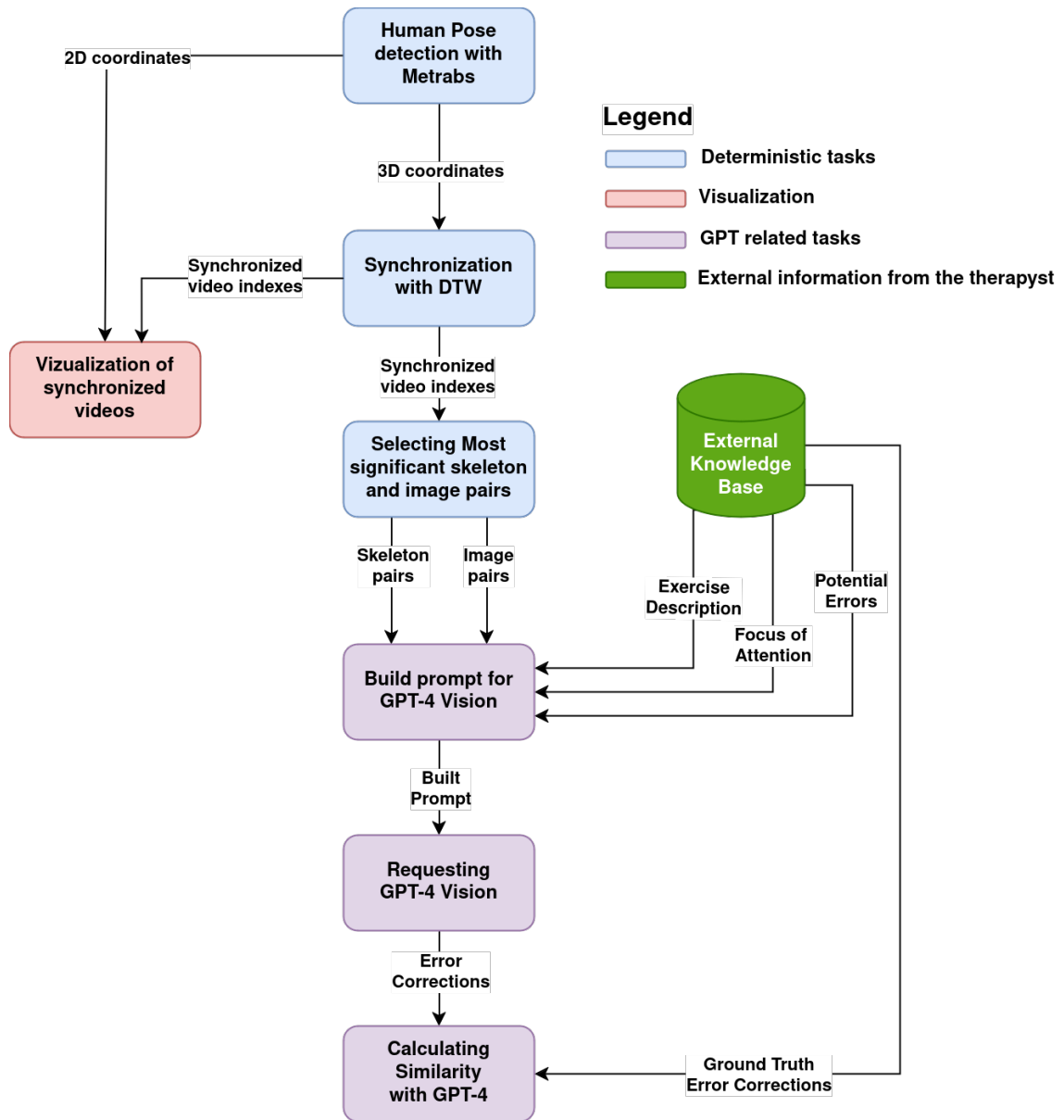


Figure 4.1: Methods pipeline demonstrating the connections between components

the Z axis pertains to the metric depth surrounding the subject. Both are important in subsequent processing. The presence of 3-dimensional coordinates is crucial in this case, as there are multiple scenarios where the meaning of 2-dimensional coordinates could be ambiguous or do not represent the correct posture of the patient. MeTRAbs was chosen because it's still achieving state-of-the-art accuracy and the coordinates can be extracted in SMPL format, which is comparable with SMPL or SKEL coordinates. Moreover, in a previous study, Baranyi et al.[32] quantified the potential errors that MeTRAbs can introduce, estimated in centimetres.

In addition to the three-dimensional SMPL joint coordinates, another metric was employed. As previously noted in section 2.5, the SMPL model comprises two distinct types of parameters, namely *betas* and *thetas*. The former describes the physical attributes of a person, such as height and body shape, while the latter represent the descriptors of the human pose. The extraction of theta parameters was achieved through the utilisation of HybriK-X, a 3D point and body mesh recovery model. HybriK-X employs the extended version of SMPL, designated as SMPL-X, which incorporates facial and hand joints. To this end, the output of HybriK-X was truncated to the initial 24 joint thetas. HybriK-X has the capacity to extract the thetas as quaternions, which can be represented as a four-dimensional vector or a three-by-three rotation matrix for each joint.

As detailed in chapter 5, both software has completed the analysis of all selected videos and saved the resulting data as numpy files.

4.2 Synchronize Rehabilitation videos

In order to identify the most suitable DTW variant for the given dataset, a series of time series synchronisation techniques were employed, as outlined in section 3.4. Upon examination of the resulting techniques and subsequent experimentation on the aforementioned dataset, it became evident that the standard Dynamic Time Warping (DTW) approach was the most appropriate for this specific situation. FastDTW represents an approximated version of the standard DTW, resulting in a loss of accuracy and its primary advantage lies in its speed when dealing with large datasets. Accuracy is of paramount importance in this context, as the synchronization of the videos must be as precise as possible. Additionally, in the context of the 30-second exercises under consideration, no significant differences were observed in terms of runtime between the two techniques. Based on previous experiences with ShapeDTW, it has not produced more accurate results, al-

though it has a higher runtime due to local shape matching. Global Kernel Alignment is a variant of DTW that is particularly effective for time series data when utilising machine learning-based kernel methods. However, in the context of this study, the advantages of Global Kernel Alignment are not sufficient to enhance the quality of the synchronisation.

As illustrated in Figure 4.1, the DTW utilises the three-dimensional pose coordinates of the patient and the physiotherapist. Regardless of the length or duration of the physiotherapist’s exercise video, it serves as a reference in the DTW process. Previous experiments have explored the optimal input for the DTW algorithm. The raw 3D coordinates may be misleading for the algorithm due to the differences in camera angles and distances between the ground truth and test exercise videos, which can result in variations in the coordinate system. Normalised coordinates can be employed, but in the synchronisation process, errors were observed in some videos where normalised coordinates were also unsuccessful. The use of SMPL theta parameters as quaternion coordinates yielded significant results, but there seems to be no substantial improvement over 3-dimensional normalized skeleton coordinates in synchronization. The optimal approach employed in this experiment involved calculating various angles between joints, including those between the left ankle, centre of hips, and right ankle, which are pertinent to the actual exercises. Identifying the most suitable angles, which are most relevant to the motion of the exercise, requires expertise. An example of a calculated angle is illustrated on Figure 5.2. The 2-dimensional coordinates were solely utilized in the visualization phase, where the two videos were compared and the skeletons were visualized as well.

Two distinct methodologies were employed to normalise the coordinates. The first involved the utilisation of the conventional 3D normalisation technique, wherein the SMPL joint with index 0 was designated as the origin, the scaling factors were calculated from the coordinates, and subsequently, all SMPL joint coordinates were scaled down using this scaling factor. The second approach entailed the application of the Procrustes algorithm[76], which is a mathematical technique employed for the alignment and comparison of shapes by minimising the discrepancies between them, using Singular Value Decomposition (SVD) to achieve this alignment. Singular Value Decomposition (SVD) is crucial in the Procrustes algorithm as it allows for the efficient computation of the optimal rotation (and reflection) matrix that aligns two shapes with minimal error. This step ensures that the alignment is mathematically optimal, minimizing the Procrustes distance between the shapes. The goal of the algorithm is to minimize the error between the two skeleton joints by scaling, rotating, and translating one to align with the other.

Once the video has been synchronised using multiple angles between joints, the next step is to select the most significant errors that exist between the synchronised skeleton coordinates, illustrated in Figure 4.1. In this instance, a number of different approaches were trialled, including the use of normalised coordinates, specified angles between joints, and quaternion coordinates. All three approaches can be used, but quaternions and normalised coordinates tuned out to be more effective than angles between joints in identifying the errors between the skeletons. In order to calculate the most significant error between the synchronised frames, two metrics were employed: the Mean Absolute Error (MAE) and the Mean Square Error (MSE), explained in Equation 4.1 and Equation 4.2. Both of these metrics can be utilised for this purpose.

MAE error between therapist's and patient's skeleton data:

$$MAE = \frac{\sum_{i=1}^n |r_i - q_i|}{n} \quad (4.1)$$

MSE error between therapist's and patient's skeleton data:

$$MSE = \frac{\sum_{i=1}^n (r_i - q_i)^2}{n} \quad (4.2)$$

where:

n - the length of the synchronized video

i - the current index

r - the array of reference skeleton coordinates or quaternions or calculated angles

q - the array of query skeleton coordinates or quaternions or calculated angles

The most significant image pairs and skeleton coordinates are selected using the following equation illustrated on Equation 4.3:

$$MAXINDEX = \operatorname{argmax}_i SIM(r, q) \quad (4.3)$$

where:

$MAXINDEX$ - the index of the skeleton coordinates, which

contains the most significant difference

i - index of the synchronized skeletons

$SIM(,)$ - is similarity metric, either MAE or MSE

r - synchronized reference coordinates

q - synchronized query coordinates

After calculating the similarities using MAE or MSE, the function returns an array of differences for each pair of skeletons. The index of the most significant error is then selected.

The output of this module is the synchronised skeleton and image pairs, which contain the most significant errors, and are subsequently input into the next module in order to construct the prompt for GPT-4. This process is described in greater detail in section 4.3.

4.3 Prompt Engineering GPT-4

The construction of effective and accurate prompts is crucial in the context of large language models, including specific domains such as rehabilitation exercises, as discussed in chapter 3. In this instance, GPT-4 was prompted using the zero-shot technique, in conjunction with the Chain-of-Thought prompting technique. Both techniques were employed in the same prompt. Additionally, the persona prompt for GPT-4 was used, wherein GPT-4 was tasked with emulating an expert in physical rehabilitation and the ability to identify potential errors in the individual's posture. Incorporating a multitude of contextual information about the exercise and the SMPL skeletons is important in the prompt. This aspect will be elucidated in greater detail below.

The prompts were constructed with great care and contained a multitude of pieces of information. A series of experiments were conducted, in which the prompts comprised solely skeleton coordinates, solely images, joint images and skeleton coordinates, multiple pairs of skeletons and images from the same camera angle, and multiple camera angle images and skeleton coordinates. For each input, the potential components of the prompt may vary, although certain elements are consistent across all cases. All these scenarios are demonstrated on Table 4.1. For the skeleton part, both normalised and unnormalized coordinates were used. From Table 4.1 it can be seen that additional camera sources were only used with mixed image and skeleton and with the use of Focus of Attention. This is because, as will be discussed in chapter 5, this combination is the best method according to the accuracy results.

Scenarios/Approach	Image	Skeleton	Focus of Attention	Additional camera sources
1.		✓		
2.	✓			
3.	✓	✓		
4.		✓	✓	
5.	✓		✓	
6.	✓	✓	✓	
7.	✓	✓	✓	✓

Table 4.1: Illustration of different scenarios covered in this study

All prompts begin with a context-setting section, wherein general information is introduced to the model. This section comprises a few lines about the general situation, namely that the model will encounter a rehabilitation scenario. In this scenario, a physical therapist and a patient are engaged in exercises, and the model will receive inputs. Additionally, this section provides other information, such as the names and indices of the 24 SMPL joints, and the edges that wire the skeleton together. It specifies the indices pairs for each edge. In the prompt, correction examples are provided that are not related to the videos used. However, they have the same language as the corrections in the dataset. Another common aspect of the prompts is the output format part, where the model is required to produce a valid JSON format as output. This format is added, and multiple examples are provided to show the exact format. These are the same parts of all prompts and have remained largely unchanged throughout the study. It is also possible to make minor adjustments to these sections in order to optimise the current input format.

As illustrated in Figure 4.1, the prompt-building module consistently receives the selected skeleton and image pairs. Additionally, it obtains data from our dataset, including the current exercise description, potential errors associated with the exercise, and the *focus of attention* in certain instances. These data are task-dependent and may be incorporated into the prompts in distinct sections.

The prompt comprises sections that are dependent on the type of input. One such section is the scenario and input section, where the input type is specified and the input format is accurately explained for the model. This is a brief section that highlights the meaning of the input. Following this, the task section is introduced, where the task is explained to the model in detail, using concise instructions, and every ambiguous detail is explained. The task section clarifies for the model which aspect of the input is to be prioritised, which differences are to be regarded as significant errors, and the extent of errors to be monitored.

In addition to differences in input types, two distinct task scenarios were presented to the model. In the first scenario, the model was instructed to calculate the degree of similarity between the patient's pose and the therapist's pose. It was then asked to suggest corrections to the pose, with the understanding that it could utilize any errors present in the image. This approach, which allows the model to make corrections without being constrained by the errors outlined in the Potential Errors section, is referred to as "in the wild." The alternative approach does not calculate a percentage, but rather determines whether the errors in the Potential Errors section are present in the patient's pose in comparison to the therapist's pose. This method is referred to as the "classification" approach. Both of these methods are suitable for this task, with further details provided in chapter 5.

Another crucial aspect of the prompt is the step-by-step guide section, which incorporates a list of step-by-step actions inserted into the prompt. The extensive and intricate problem is decomposed into smaller, more straightforward tasks, which are executable by the GPT-4. This section is contingent upon the input type and the task type previously mentioned. This step entails calculating the pose of both individuals using the current inputs, images, skeleton coordinates, and both. Subsequently, all possible angles between joints were calculated, compared, and reasoned about in order to ascertain the correctness of the patient's posture in comparison to the therapist's posture. Depending on the task, after that, the similarity percentage between the two poses was calculated.

For this calculation, a guide percentage is provided, which will be mentioned later. Subsequently, errors must be identified and corrections must be written in a specific manner, in the language of error corrections provided earlier. In particular, SMPL joints that are affected by the error or the correction must be selected. The correction must then be written based on the selected posture errors, the exercise description and, optionally, the potential errors and focus of attention sections mentioned earlier.

The final two sections of the prompt, which have not yet been discussed, are the Constraint section and the Calculation Similarity Percentage Guide section. The former includes multiple constraints, which guide the generation of the GPT-4 model, thereby ensuring more consistent output. These constraints may be either the omission of specific errors in the model or the imposition of output requirements.

The second section of the prompt is a guide for the LLM on how to calculate the percentage in the case of the first task. These guides include cases where the LLM must increase or decrease the calculated percentage. For instance, the LLM may increase the percentage when the image is identical and place greater focus on images where the pos-

ture is similar, despite differences in camera angle and camera distance. A full example about the prompt used in this study is illustrated in Appendix A.

The penultimate module displayed in Figure 4.1 initiates a request to GPT-4. To facilitate this, OpenAI has established an API that enables developers to access various GPT models, including GPT-4 Vision. In this study, GPT-4 Vision was employed, specifically the *gpt-4-1106-vision-preview* checkpoint of the GPT-4 Vision. The API is readily accessible via code, although it does have a pricing structure, particularly when images are attached. The API has tiers that impose limitations on users in various ways, including the number of requests per minute, the number of tokens per minute, and the potential number of input tokens. In this study, the second tier was utilised, which allows for 450,000 input tokens per minute and 5,000 requests per day.

4.4 Evaluation of GPT-4 answers

The subsequent stage, as indicated in Figure 4.1, is the comparison of the generated error corrections with the ground truth error corrections from the dataset. However, calculating the similarity between text pieces is not a straightforward process, particularly when context information is a significant factor. To ascertain the accuracy of the different methods mentioned in section 4.3, a binary classification approach was employed to determine whether the generated response was similar or not to the ground truth data.

For that, there are multiple different options to calculate such an accuracy. The first is using a human expert, who can determine the similarity between the two text pieces. However, this approach was not suitable here because, although physiotherapists may provide varying opinions, a consistent common value must be maintained. Another possibility is to use an embedding model, which can get two texts as input and give back a binary value if they are similar or not. This necessitates potential fine-tuning of the specific dataset utilised in this study, which would require additional resources, which is not the study's primary objective, to assess the performance of embedding models. The approach employed in this study is an automated similarity checker incorporating GPT-4 and prompt engineering. By incorporating multiple analogous examples of texts and responses and utilising step-by-step reasoning, GPT-4 was used to generate the results and it also does the similarity check. For this purpose, it is possible to utilise any other LLM, such as LLaMA, Llama-2, Llama-3 or a newer LLM from the field.

This section describes the creation of a synthetic dataset, which contains the valid error corrections of some videos from the original dataset. For each video correction, five different candidates were generated by GPT-4, which paraphrased the original one and made five similar corrections. These corrections were paired with the original one, which were designated as positive pairs. After that, synthetic corrections of other videos were selected randomly, with five per video, which were designated as negative pairs of the dataset. The GPT-4 model was benchmarked on this dataset, which consisted of approximately 150 data points. The model achieved an accuracy of 98% on this task, which demonstrates its consistency in this domain.

The accuracies detailed in chapter 5 are calculated using this approach. Furthermore, utilising the GPT-4 model with low-temperature parameters will result in more deterministic answers. However, potential differences in accuracy values may vary due to the stochastic nature of the model.

Chapter 5

Results

This section will present the experiences and results of multiple scenarios and the application of multiple rehabilitation videos. It will draw conclusions based on my experiences. This section is mainly focused on the results, including the different prompting inputs, such as only skeleton coordinates, only images, and so on. However, it will also include potentially useful experiences with Dynamic Time Warping, MeTRAbs, rehabilitation videos, GPT4-API, and prompting.

Two types of exercise were selected for this experiment, resulting in 15 different potential posture errors. The first exercise has more recognisable errors, while the second has harder errors to recognise for a non-expert person. This resulted in 15 error videos with 2 ground truth videos. The following pipeline was executed in these videos, and a comprehensive analysis was made to analyse the potentiality of GPT-4 Vision. The accuracy values presented in Table 5.1 calculated using the videos and corresponding exercise descriptions, as well as the posture corrections provided by a physical therapist expert.

Firstly, it is important to note that MeTRAbs has significant issues with low-quality videos or images, as well as poor lighting conditions. In both cases, MeTRAbs either fails to return any results or returns false results. False results could mean that the skeleton coordinates are inconsistent with the image. Furthermore, MeTRAbs has issues with people who are not facing the camera or are turned backwards. In such cases, MeTRAbs also gives back an error. An illustrative error is shown in Figure 5.1, where the model fails to estimate the person's skeleton due to poor lighting conditions and the person being turned backwards from the camera. It's a common issue for all pose estimators due to the lack of training data from these orientations. In addition to the aforementioned errors in the pose estimation model, any videos exhibiting such errors were subsequently excluded.

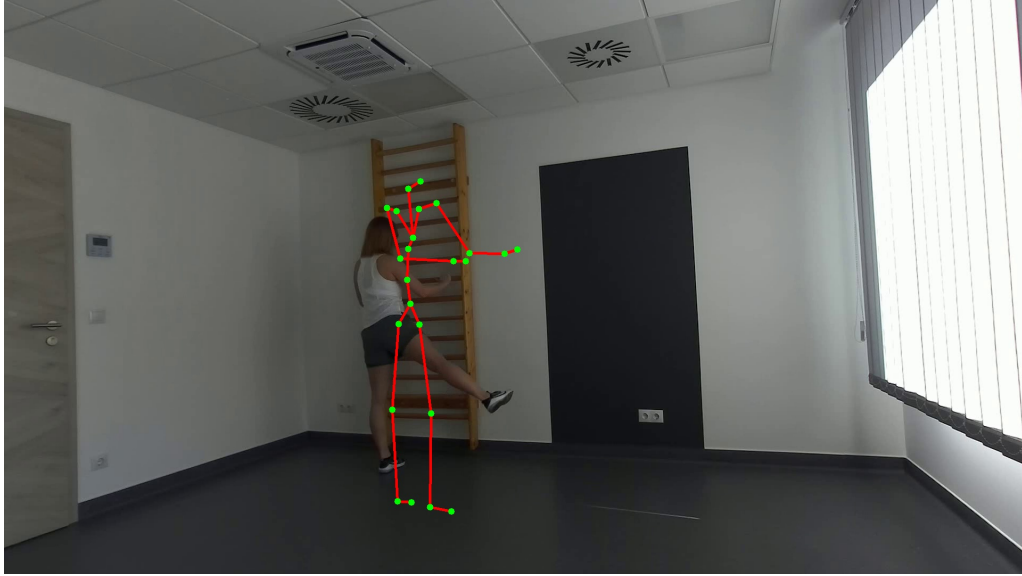


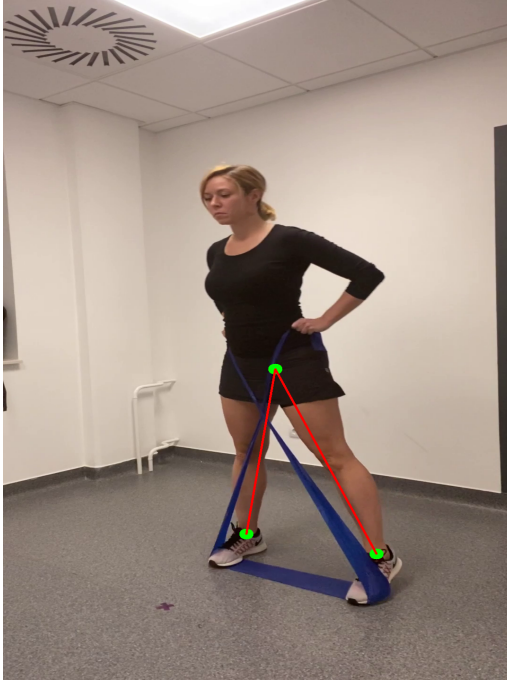
Figure 5.1: MeTRAbs error where there are poor lighting conditions and the therapist is not towards the camera

In addition to the aforementioned issues pertaining to the pose estimation model, there have been some experiences with Dynamic Time Warping for synchronising videos based on 3-dimensional skeleton data or quaternion 4-dimensional coordinates. As briefly mentioned in section 4.2, the most convenient approach for processing 3D coordinates and passing them to the Dynamic Time Warping is the angles between joints, where each angle contains three different joints. This approach enables the detection of differences in movement and provides a robust synchronisation in the majority of cases. The selection of the most useful angles is of paramount importance for this task, as it is dependent on the specific exercise. Inappropriate angles cannot provide any meaningful insight into the motion, and some of the angles between joints are almost constant. It is therefore advisable to utilise multiple angles instead of a single one. While normalized coordinates can be employed, they present a challenge when camera angles differ. An alternative approach involved attempting to use SMPL theta parameters as quaternion coordinates. However, this did not prove to be a satisfactory solution, as in situations where normalized coordinates were effective, quaternion coordinates yielded unsatisfactory results. Following an investigation into the efficacy of quaternion coordinates for the purpose of synchronization, it was found that they were less effective than normalized coordinates. However, when it came to the correction of pose and the addition of a prompt, quaternion coordinates were found to be useful. To align the skeletons, they can be transformed into the same coordinate system using the Procrustes algorithm. However, this is only possible

after synchronization, because the algorithm requires sequences with the same length.

There are multiple factors that can lead to poor results when using Dynamic Time Warping. Potential issues that may affect the quality of the synchronization include the video itself or the algorithm being sensitive to outliers, where the two videos contain entirely different content. Another significant challenge is the considerable length difference in the time series, which can result in the pairing of motions that are only slightly similar, rather than identical. This is further compounded by the time shift, which causes the actual pair of those motions to be significantly further apart. These challenges can be addressed by filtering out the content of the videos, as done in the case of our dataset. The differing number of repetitions or motions present in the videos compounds this issue. The patient's video may contain more or fewer repetitions, or movements that do not correlate with the exercise. This problem was addressed by filtering the videos and removing any additional repetitions or outlying movements. The potential for speed differences to affect the quality of the synchronization is a limitation of the Dynamic Time Warping approach. This is because it is unable to detect slower or faster exercise implementation. This is a limitation of the current evaluation.

Another crucial task is the selection of image and skeleton pairs that exhibit the most significant differences between them. To calculate the discrepancies between the synchronised skeletons, a number of approaches were tested. These included the use of SMPL theta parameters and normalised 3-dimensional coordinates. However, due to the aforementioned issues in the synchronisation process, the results obtained using these methods were not always optimal. Another approach is to utilise angles between joint coordinates. An illustrative example is provided in Figure 5.2, where the most significant error was selected based on the step size difference. This makes it evident that the error is the result of a too-small step size. Nevertheless, in certain instances, the selection of alternative angles is required in comparison to those used for synchronization, in order to calculate the most significant error pairs. This is dependent on the specific exercise in question. If the angles utilised in the process of synchronisation encompass the areas of the body where the error is present, then there is no requirement for additional angles. Otherwise, new angles must be calculated, which necessitates the input of expert knowledge to select the most appropriate angles for the selection of error skeleton pairs. In order to calculate the error between the two frames, two different metrics were tested: the Mean Absolute Error (MAE) and the Mean Square Error (MSE). Both of these metrics proved to be suitable for this purpose.



(a) Therapist does the exercise with correct step size



(b) Therapist does the exercise with an incorrect, much smaller step size

Figure 5.2: Visualization of left ankle - centre of hip - right ankle angle, where on the image (a) **38.04** degree and on the image (b) **9.18** degree

To determine the degree of similarity between the error instructions provided by physical therapists and those generated by GPT-4, accuracy values were calculated. The principal findings of this study are presented in Table 5.1, which demonstrates the efficacy of distinct methodologies utilising diverse input types. Accuracy values were calculated in accordance with the methodology outlined in section 4.4, and are categorised by methodology, input type, and the normalised and unnormalised nature of the skeleton coordinates. Multiple cases were considered, but these are the most pertinent results, which are worthy of mention.

The least effective approach, as indicated by the table, is the skeleton case where normalized skeleton coordinates were added to the prompt. This approach provides the baseline results, as the least effective approach. In many cases, the model produces the same output regardless of the skeletons added to it, demonstrated in Table 5.2. This is because the aligned skeleton coordinates with the Procrustes algorithm usually contain very little differences, which are not detectable by the model, or the other errors are more significant. This is also true for both classification and in-the-wild methods. The introduction of the Focus of Attention section in the prompt led to an improvement in results, with a reduction of hallucination from the model. However, the performance of the skeleton-only

Method\Input type	Image	Skeleton		Image + Skeleton	
		Normalized	Unnormalized	Normalized	Unnormalized
Focus of Attention + Classification	40%	33,3%	40%	53,3%	60%
No Focus of Attention + Classification	26,6%	26,6%	40%	40%	53,3%
Focus of Attention + In the wild	33,3%	40%	46,6%	46,6%	46,6%
No Focus of Attention + In the wild	20%	20%	33,3%	20%	26,6%

Table 5.1: Accuracy percentages for different methods

approach with aligned skeleton coordinates remains the least effective of all the methods employed here.

From the baseline of normalized skeleton coordinates, using only image pairs does not resolve the issues inherent to skeleton coordinates. In terms of average accuracy, the result is comparable to that of normalized skeleton coordinates. However, it does not present the same issue of providing identical responses to different prompts. It is more consistent when using the focus of attention section with fewer hallucinations. In the case of only using one image pair, the method relies solely on information present in the image. In this context, any potential errors in the image, such as poor quality, inadequate lighting conditions, or an inappropriate camera angle, can obstruct the model’s ability to process the images effectively. One potential solution for addressing camera angle issues is the use of a multi-camera setup, which will be discussed in greater detail later in this section.

Other input solutions, including the mixed image + skeleton approach and the unnormalised skeleton cases, demonstrated superior performance across the different method types. These methods, which are more consistent, provide potential error corrections, which could be the correct error corrections. The image + skeleton solution achieved the best performance, indicating that the image can correct the minor errors present in the skeleton coordinates. The optimal method demonstrated an accuracy of **up to 60%** on the previously mentioned small dataset. Answers of the best approach are illustrated on Table 5.3. However, it also demonstrated that skeleton coordinates can assist the model in instances where images are insufficient or the quality of the videos is below an acceptable threshold.

As previously stated, the inclusion of the Focus of Attention component, which encompasses the specific body part that the model is required to focus on, has a significant impact on the performance of GPT-4 on this particular dataset. In all instances, it has been observed that this approach leads to an improvement in accuracy, with no instances of deterioration in results. The findings of this study indicate that the limitation of the search space in the case of a Large Language Model is a crucial factor. Nevertheless, the

Exercise	Error	GPT-4 error correction	Ground truth error correction
star/ Y- exercise	During the exercise, the knee is tilted inwards from the vertical.	Keep your shins vertical, push your buttocks back slightly.	Please rotate your opposite pelvis slightly forward! Keep your knees vertical. Be careful not to lean inwards.
star/ Y- exercise	During the exercise, the centre of gravity is placed too far forward, so that the axis of the knee is too far in front of the axis of the ankle, the leg-ankle joint is closed, and the trunk remains almost vertical.	Keep your shins vertical, push your buttocks back slightly.	Please place the centre of gravity slightly back towards the centre of the heel of the foot. If you are looking at your feet, your knees should not cover your toes! Tilt your torso slightly forward.
star/ Y- exercise	During the exercise, the centre of gravity is too far back, putting most of the weight on the heels. Your knees are not bent enough. Torso is tilted too far forward.	Keep your shins vertical, push your buttocks back slightly.	Please place your centre of gravity slightly towards your toes. Bend your knees more! Tilt your torso back slightly, and tuck your belly button in.
star/ Y- exercise	The supporting leg is not sufficiently bent.	Keep your shins vertical, push your buttocks back slightly.	Please bend your knees more.
star/ Y- exercise	The opposite knee is not fully extended.	Keep your shins vertical, push your buttocks back slightly.	Please fully extend the kicking/pointing leg!
star/ Y- exercise	"Steps out" on the opposite lower limb, i.e. puts weight on it, thus shifting the centre of gravity to the stepping foot with each step.	Keep your shins vertical, push your buttocks back slightly.	Please keep your body weight on the supporting leg. Do not put any weight on the exiting leg!

Table 5.2: Demonstrating GPT-4 giving the same answer for multiple inputs with using skeletons as input.

selection of focus of attention areas is not always straightforward. It necessitates expert knowledge to identify the most appropriate body part for a specific exercise.

The study demonstrates that, with the assistance of large language models (LLMs), it is feasible to generate human posture error correction instructions in the style of human experts from the field. To achieve this, a number of examples were selected and incorporated into the prompt, thereby guiding the LLM generation to follow the style of the instructions.

Another limitation that is evident in Table 5.1 is the potential for error in the image itself. In the case of "in the wild" methods, the model is capable of identifying any errors that may arise in the images or the 3-dimensional skeleton coordinates. In some cases, it is beneficial to identify additional differences beyond those defined by the physical therapist. However, in general, this approach leads to an increase in hallucinations from GPT-4, which makes the method less stable. The alternative method referred to as "classification", involves limiting the model to selecting potential errors from a predefined set of errors by the physical therapist, which are typical for that specific exercise. Additionally, there are videos of those errors in our provided dataset. In this instance, the model exhibited a reduction in the occurrence of hallucinations. Furthermore, in each case, the errors were provided in the potential errors section of the prompt, as discussed in section 4.3.

Exercise	Error	GPT-4 error correction	Ground truth error correction
Lateral walking with extended knee	Does not keep both feet parallel When moving sideways, turns outwards.	Please keep both feet parallel when moving sideways.	Please make sure that your two feet remain parallel when stepping to the side and when stepping back. Turn both feet slightly inwards.
Lateral walking with extended knee	Too narrow step compared to therapist's stance.	Make a wider step to match the therapist's stance.	Please step into a wider stance! Make sure to step out into a wider than shoulder-width straddle and step back into a hip-width straddle.
Lateral walking with extended knee	Too big step compared to therapist's stance.	Increase step width to match therapist's stance.	Please step into a smaller spread! Make sure to step out into a wider than shoulder-width straddle and step back in a hip-width straddle.
Lateral walking with extended knee	When stepping out to the side, not enough weight is applied to the lower limb, leaving weight on the other side, with the result that the foot slides on the ground, does not lift enough, and tilts the trunk strongly to the opposite side.	Keep waist and trunk straight when stepping. Keep stepping leg knee straight when stepping.	Please do not slide your feet on the ground when stepping back! Put more weight on the leg you stepped out on and lift the other leg. Make sure your knees stay straight!
Lateral walking with extended knee	When stepping sideways, supporting leg knee bends.	Please keep supporting leg knee straight when stepping.	Please extend your knees when stepping sideways. Make sure that the knee of the supporting leg does not bend.
Lateral walking with extended knee	When stepping sideways, the knee of the stepping leg bends.	When stepping sideways, please ensure the stepping leg knee remains straight!	Please extend your knees when stepping sideways. Make sure to keep the knee of the exiting leg extended.
Lateral walking with extended knee	To the side and bend the knees on both sides when stepping back.	When stepping sideways, keep the stepping leg knee straight. Please keep both feet parallel when moving sideways.	Please stretch both knees. Make sure to stay stretched out to the side and backwards!
Star/ Y- exercise	The supporting foot leans inwards, leaning too much on the inner sole.	Please bend the supporting leg more! Ensure knee is aligned over foot, not tilting inwards!	Please tilt your leg outwards. There should be weight on both the inner and outer soles!
Star/ Y- exercise	The supporting leg is turned outwards.	The supporting leg is not sufficiently bent. During the exercise, the knee is tilted inwards from the vertical.	Please turn your foot back to the middle! Always keep the foot facing forwards.
Star/ Y- exercise	During the exercise, the knee is tilted inwards from the vertical.	Please bend the supporting leg more! Extend the opposite knee fully!	Please rotate your opposite pelvis slightly forward! Keep your knees vertical. Be careful not to lean inwards.
Star/ Y- exercise	During the exercise, the centre of gravity is placed too far forward, so that the axis of the knee is too far in front of the axis of the ankle, the leg-ankle joint is closed, and the trunk remains almost vertical.	Centre of gravity too far forward; keep shins vertical. Supporting leg needs more bend; bend knee to about 90 degrees. Ensure knee is aligned over foot; do not let knee tilt inwards.	Please place the centre of gravity slightly back towards the centre of the heel of the foot. If you are looking at your feet, your knees should not cover your toes! Tilt your torso slightly forward.
star/ Y- exercise	During the exercise, the centre of gravity is too far back, putting most of the weight on the heels. Your knees are not bent enough. Torso is tilted too far forward.	Please bend the supporting leg more! Shift your centre of gravity back slightly!	Please place your centre of gravity slightly towards your toes. Bend your knees more! Tilt your torso back slightly, and tuck your belly button in.
Star/ Y- exercise	The supporting leg is not sufficiently bent.	Please bend the supporting leg more! Ensure knee is aligned over foot, not tilting inwards!	Please bend your knees more.
Star/ Y- exercise	The opposite knee is not fully extended.	The supporting leg is not sufficiently bent. During the exercise, the knee is tilted inwards from the vertical.	Please fully extend the kicking/pointing leg!
Star/ Y- exercise	"Steps out" on the opposite lower limb, i.e. puts weight on it, thus shifting the centre of gravity to the stepping foot with each step.	Please bend the supporting leg more! Ensure the knee is aligned over the foot, not tilting inwards!	Please keep your body weight on the supporting leg. Do not put any weight on the exiting leg!

Table 5.3: GPT-4 answers by mixed image and skeleton pairs, compared to ground truth corrections

It can be observed that, for each input category, the "classification" method consistently demonstrated superior performance to the "in the wild" approach, with an incremental improvement in accuracy.

As previously stated, there are instances where patients' posture is incorrectly aligned, resulting in self-occlusion or an inability to discern the error from the camera angle. In this case, the dataset comprises videos recorded with separate cameras from three separate camera angles. In this instance, no accuracy scores were calculated, as the primary objective was to ascertain the efficacy of a multi-camera setup in addressing instances where a single camera is insufficient. Video recordings were made using timecode synchronization technique. The synchronisation of the patient's and therapist's videos with DTW was achieved by selecting one camera, obtaining synchronised index pairs, and then synchronising all videos to the therapist's videos. Subsequently, single image pairs were queried with the image + skeleton method using a single camera, and the same method using all three cameras. From the data collected, it can be concluded that the multi-camera prompt is more effective than the single-camera solution in detecting additional errors in the patient's posture. An illustrative example of two camera angles and a posture error that is evident using multiple camera angles and misleading using one camera is presented on Figure 5.3 and Figure 5.4. The figure illustrates that, from the initial camera angle, the patient's stepping leg appears to be almost perfectly extended, with only a slight degree of flexion visible from this perspective. However, the second camera angle demonstrates that this is not the case. Such issues can arise, and multi-camera systems may offer a potential solution to address them. Such systems will assist therapists in their work, as they will no longer need to repeat the same exercise with the patient and observe them from a different angle, as is currently the case.

Table 5.4 illustrates a series of image pairs from our dataset. The image pairs are linked to the errors presented in Table 5.2. This image pair represents a different error for the same exercise. It can be observed that, even from a single camera angle, some errors are not detectable. Consequently, the necessity for a multi-camera setup becomes evident.

The potential of GPT-4 for image understanding has been demonstrated on numerous occasions. In the context of rehabilitation, GPT-4 has not been employed to comprehend rehabilitation exercises and identify errors in accordance with a ground truth image or/and skeleton data. However, it has demonstrated potential in visual question answering in the rehabilitation field. From the outset, GPT-4 demonstrated no difficulty in understanding images, with the model able to identify posture differences between images. However, the

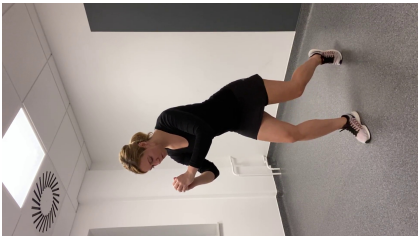
Error images						
Reference Images						

Table 5.4: Image pairs from the dataset, which correspond to the error scenarios mentioned in Table 5.2



Figure 5.3: Patient doing the exercise, first camera angle, stepping leg almost perfectly stretched



Figure 5.4: Patient doing the exercise, second camera angle, error on stepping leg clearly shown

model exhibited hallucinations and miss-predictions, which could be partially overcome by utilizing external information in the prompt. Additionally, the model also needed to handle skeleton coordinates, which proved more challenging. Initial experiments demonstrated that GPT-4 is unable to utilise skeleton data, lacks the capacity to compare skeletons based on coordinates, and is unable to wire the independent joints together to form a full skeleton. Following experimentation with skeleton coordinates utilising the UI version of GPT-4, it was found that GPT-4 is capable of performing the aforementioned tasks, but requires additional information. As previously stated, the provision of supplementary information, which facilitates the comprehension of the task, is beneficial in prompts. In order to comprehend skeleton data, GPT-4 requires the potential edges between the individual joints, to determine how to wire a skeleton together. Furthermore, it necessitates an exemplar of the significance of joint indices and the representation of a three-dimensional or four-dimensional vector for each joint, in the event that joint coordinates or joint rotations are expressed in quaternions. As demonstrated in Table 5.1, the utilisation of skeletal coordinates is of paramount importance for the achievement of enhanced accuracy in this attempt.

The experiment demonstrated the capacity of GPT-4 to classify posture errors with a high percentage of accuracy, as well as those that are more challenging to identify, which

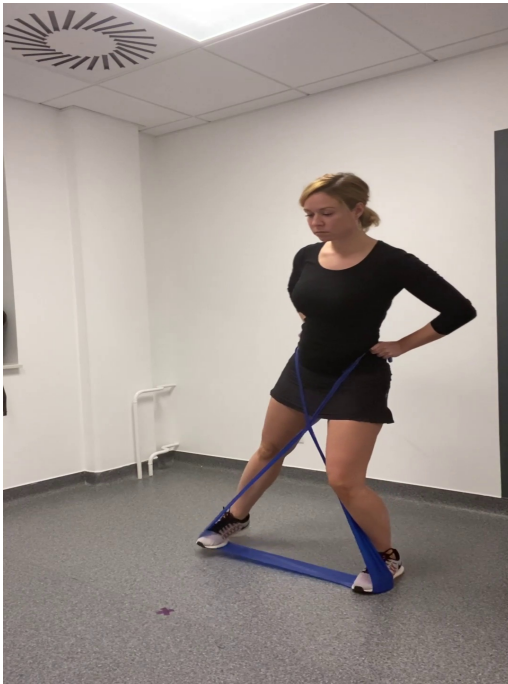


Figure 5.5: Therapist doing an easily recognisable error, bending the supporting knee, for GPT-4

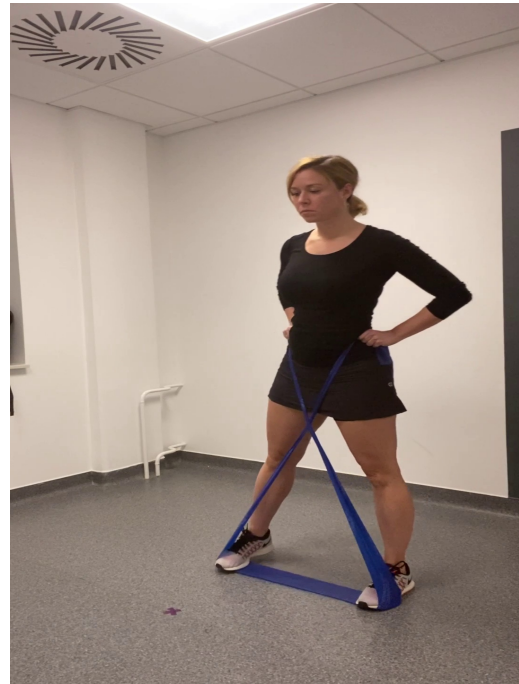


Figure 5.6: Therapist doing the exercise correctly



Figure 5.7: Therapist doing a hardly recognisable error, inward bending of the knee, for GPT-4



Figure 5.8: Therapist doing the exercise correctly

led to the model encountering difficulties. Based on my observations, there is considerable potential for the model to identify more significant errors, such as bending knees, incorrect spine or trunk posture, and incorrect head holding. These are errors that are more apparent to non-experts. Conversely, minor differences in posture, such as minor knee movements and slight inward or outward tilting of the knee, present challenges for the model in accurately identifying errors. Frequently, the model provides alternative suggestions, which it deems to be potential issues. Potential discrepancies in accuracy scores may also be attributed to the similarity checker, due to differences in vocabulary usage between the therapist and GPT-4. As illustrated in Figure 5.5 and Figure 5.6, the therapist is performing an exercise with and without an error. In this instance, the error is bending the knee of the supporting leg. The GPT-4 is able to identify this error without difficulty using the methodology discussed in Table 5.1. However, as illustrated in Figure 5.7 and Figure 5.8, even minor errors, such as inward bending of the knee, which is the error in question here, are difficult to discern for the human eye. They require expert knowledge and the use of multiple camera angles.

Finally, using the OpenAI API comes at a cost. The experiment used the *gpt-4-1106-vision-preview* checkpoint. The token count for a query is about 6000 tokens per query. The price of OpenAI's GPT-4 Turbo model with vision is \$10 per million input tokens plus the cost of the images, which is about \$0.028 per request, calculated with only one pair of images. For a video provided by the patient, this is the cost of processing the request with GPT-4 Turbo, after the video has been synchronised and processed through the pipeline described in Figure 4.1.

Chapter 6

Discussion

In addition to the outcomes and methodologies presented in chapter 5, this study represents a preliminary investigation into the potential of large language models (LLMs) and visual language models (VLMs) in home physical rehabilitation settings. There is considerable scope for enhancing various components of the proposed approach, and the rapid evolution of technology necessitates a cautious and considered approach to ensure the optimal selection of future development strategies.

This method employs a limited subset of the dataset, as outlined in reference chapter 5. The objective was to investigate the performance of GPT-4 in diverse scenarios and with varying degrees of posture errors. The approach will be subsequently tested on a larger set of videos from the same dataset, which will likely yield further insights into the potential of GPT-4 in-home rehabilitation. The utilisation of multiple videos allows for a more comprehensive analysis of GPT-4 performance, facilitating the identification of its limitations, potential drawbacks, and constraints, contingent upon the nature of the error or the input.

Furthermore, in light of the rapid improvements of Large Language Models and Visual Language Models, it is imperative to conduct a comparative analysis with other models. In the context of skeleton coordinates, future research will encompass a comparison with renowned large language models, including LLaMa-2, LLaMa-3[77], Mistral[78], and Gemini. Gemini stands out as a unique case due to its multimodal nature. For this particular study, only skeleton coordinates will be utilized.

Further comparisons with models capable of processing images are also planned. A version of Gemini Pro API has already been published, but it is a Free Tier with strict limitations, which is not sufficient for my purposes. However, Google has announced the Pay-

as-you-go plan with Gemini Pro, which will be available after 14 May. The Gemini PRO API enables the utilisation of GPT-4 for comparative purposes, encompassing the utilisation of images. Another potential model is LLaVA, which was trained on GPT-generated responses utilising images and text. The limitation of LLaVa is that it was trained using only one image; however, the fusion of images into a single image may serve to address this issue. This approach has the disadvantage of compressing multiple pieces of information into a single image, even when the image is of a considerable size. Resizing the image to a smaller form also result in the loss of important information from the picture, due to the magnification of minor errors. In this instance, the utilisation of multiple image and skeleton pairs derived from a single video is a potential avenue for investigation. This approach, however, is associated with a significantly elevated cost in the GPT-4 API. One potential solution to this challenge could be the fusion of image pairs into a single image.

Recently OpenAI announced GPT-4o, which is currently their latest model. It accepts any combination of text, audio, image and video input and produces any combination of text, audio and image output. They have announced it with improved audio and video capabilities. This model is very new, having been released just before this study was written, and has also been introduced into the UI and API. It is worth noting that newer and newer tools are being released every day with better vision and reasoning capabilities, overcoming the potential problems with visual question answering and visual reasoning tasks.

In addition to enhancements to the prompts and expanded testing on diverse video content, another potential area for improvement could be the utilisation of SKEL, as referenced in section 2.5, which enhances the SMPL model by integrating parametric body shapes with biomechanical skeletons through the use of body constraints, thereby rectifying potential errors and unnatural deformations observed in the output of the SMPL pose estimator model. The refinement of the skeleton coordinates enables the enhancement of the quality of the video synchronisation, error image and skeleton pair selection.

It is also pertinent to consider the use of alternative prompting techniques in the context of this study. One potential avenue for exploration is the use of one-shot or few-shot prompts with GPT-4 and other VLMs, where multiple prompt-answer pairs are provided to the model. This approach could enhance the consistency of the model, leading to the generation of more reliable results that align more closely with the therapist's responses. This necessitates the allocation of additional resources to prompt the GPT-4 API and to provide the physical therapist with annotated answers.

As mentioned earlier, this study focuses mainly on correcting patient posture. However, there are other types of error that need to be considered in the future. The Dynamic Time Warping algorithm is not able to detect speed differences between the two sequences. The detection of speed differences will be further developed in the future.

In this study, all prompts were created by manual crafting. This approach is not cost-effective and requires a significant investment of time, resources, and patience on the part of the human worker. This direction is not aligned with the objectives of recent studies, which have identified the need for more efficient methods of prompt generation and paraphrasing. Recently, automatic prompt generation and prompt paraphrasing algorithms have emerged as promising solutions to address the limitations of manual crafting. One model that has demonstrated success in this area is DSPy[79], a framework that optimises Language model prompts and weights. The approach demonstrated superior performance compared to expert-level prompts on GPT-3.5 and LLaMa2. This approach can facilitate the creation of prompts for a language model and can also update the weights of the model if available.

The current limitations of LLM technologies are that they lack the necessary trustworthiness to be considered a reliable source of medical advice. This is particularly relevant in the context of machine-assisted rehabilitation, where current laws and regulations, including ethical considerations, do not allow AI applications to provide medical advice to patients without the supervision of a qualified professional, as this type of experimentation falls into the category of human experimentation, which raises significant moral issues. As mentioned at the end of chapter 5, the cost of a request to GPT-4 Turbo with the image pair and skeleton data is approximately \$0.028. This amount, compared to the cost of hospitalisation for rehabilitation and the cost of hospital staff, clearly demonstrates the need for home rehabilitation solutions in the field. However, these methods are not able to replace physiotherapists in the field, but only support their work, monitor the patient's performance and also help to reduce the various costs of rehabilitation.

The method presented here will be used in a rehabilitation application developed by Dávid Tokár as part of his BSc thesis. In this application, physical therapists can add exercises to the patients' personalised rehabilitation plan and monitor how they perform the exercises. Patients are required to record themselves performing the exercise. The methods introduced in chapter 4 will be integrated into the application's features, providing offline feedback to patients and therapists about the patient's performance of the exercise.

Chapter 7

Conclusion

The findings of this study indicate that large language models (LLMs) and vision language models (VLLMs) have potential applications in the field of home physical rehabilitation. This is where patients are required to perform exercises independently without the assistance of a therapist. My analysis revealed that GPT-4, when prompted through the GPT-4 API, demonstrated satisfactory performance after multiple attempts. Furthermore, the use of multiple sources of extra information in the prompt contributed to enhanced performance. From my experiences, it was found that image and skeleton data for posture error detection and correction complement and correct each other's mistakes, thereby boosting the performance of LLM. Furthermore, it was observed that limiting the search space of GPT-4 resulted in improved accuracies. The scenario described in this study is a good example of patients doing exercises at home, which makes this pipeline mentioned in chapter 4 usable in real-world scenarios.

Nevertheless, it is important to note that, at present, Large Language Models and Visual Language Models lack the capacity to replace the work of therapists. These models are accessible to everyone and, as general-purpose models, they are not medical tools. However, the models do possess some knowledge of the field, although they lack the depth of expertise required to match that of an expert in the field. It is possible to utilise these models with appropriate caution and with well-defined prompts to assist therapists in their work, rather than replacing them. With these tools, therapists can monitor the patient's rehabilitation process, including the quality and quantity of exercises performed by the patients.

The current state of these systems is not indicative of their ultimate evolution. The development of general-purpose models is evolving at a rapid pace, making it challenging

to keep pace with the rate of development by major companies and research groups. Their potential for medical applications and home rehabilitation will increase in the near future, and new solutions will emerge as a result of the limited attention that medical solutions, including home rehabilitation, have received globally.

At present, a significant research project is underway in which I am participating. The objective is to develop a smartphone application that can be used by both patients and therapists. This will involve pose analysis using a Large Language Model or Vision Language Model. It is my responsibility to continuously develop this method, which will be incorporated into an application to be trialled by the EMINEO Private Hospital in a real-world setting. During this trial, real patients and therapists will be using the application and providing feedback on its usability, limitations, and errors, as well as mispredictions, misjudgements, and misdiagnoses.

Acknowledgements

I would like to express my gratitude to Gábor Baranyi for his invaluable assistance throughout this project. His dedication and commitment to guiding me towards a more optimal solution were instrumental in the success of this endeavor.

I am deeply appreciative to the NIPG research group for providing me with the necessary resources to achieve my goal. I would also like to thank my colleagues in the office for their support throughout this process. In particular, I would like to thank András Lőrincz for his critical thinking and valuable contributions, as well as for his unquestionable assistance in achieving this goal.

I would like to thank Zsófia Gaál for her participation in the videos and for writing all the additional materials, including the detailed exercise descriptions, potential error descriptions and error corrections.

Furthermore, I would like to acknowledge the contributions of Tina Varró and Kriszti Kornis, who participated in the recording of the multicamera videos and provided consultations.

Finally, I would like to express my gratitude to my family and my girlfriend, for their unwavering patience and support throughout this process.

This research has been partially supported by the European Union project RRF-2.3.1-21-2022-00004 within the framework of the Artificial Intelligence National Laboratory Program.

Appendix A

Prompt used

```
1 [GENERAL DESCRIPTION]
2 You will get 2 images and the corresponding 3D skeleton
3 coordinates of the person's posture on the image, in text.
4 You need to give back corrections of the second image and 3D
5 skeleton coordinates based on the first
6 image and 3D skeleton coordinates of the joints, if there is
7 any error.
8 There are always at least one error on the second IMAGE and 3D
9 skeleton coordinates of the joints
10
11 Correction: a string which contains the concrete correction of
12 the person's posture based on the first image.
13
14 I provide you the SMPL joints:
15
16 0: 'pelvis',
17 1: 'left_hip',
18 2: 'right_hip',
19 3: 'spine1',
20 4: 'left_knee',
21 5: 'right_knee',
22 6: 'spine2',
23 7: 'left_ankle',
24 8: 'right_ankle',
25 9: 'spine3',
26 10: 'left_foot',
27 11: 'right_foot',
28 12: 'neck',
```

```

24     13: 'left_collar',
25     14: 'right_collar',
26     15: 'head',
27     16: 'left_shoulder',
28     17: 'right_shoulder',
29     18: 'left_elbow',
30     19: 'right_elbow',
31     20: 'left_wrist',
32     21: 'right_wrist',
33     22: 'left_hand',
34     23: 'right_hand'
35
36
37     Input SMPL skeleton pairs will look like:
38
39     {"Smpl joint index1": [coordinates], "smpl joint index2": [3D
40       skeleton coordinates], ...}
41     the coordinates looks like this: [X, Y, Z]
42     The SMPL indices refers to the SMPL joints above, so 0 index is
43       the pelvis
44
45     HERE ARE THE SMPL SKELETON COORDINATES COORDINATES:
46     Here are the SMPL joint SKELETON coordinates for each pair:
47     [SKELETON COORDINATES]
48     The 1th posture pair:
49
50     {"0": [30.20517349243164, 145.7416229248047,
51       2568.719482421875], "1": [112.09983825683594,
52       223.0821990966797, 2545.22802734375],
53       "2": [13.230583190917969, 233.94027709960938,
54       2622.83154296875], "3": [7.417449951171875,
55       41.77363586425781, 2571.740478515625],
56       "4": [251.50216674804688, 518.7405395507812,
57       2342.366943359375], "5": [-121.4754409790039,
58       553.9078369140625, 2531.577392578125],
59       "6": [-52.280418395996094, -56.38128662109375,
60       2516.616943359375], "7": [482.69842529296875,
61       851.5257568359375, 2263.88037109375],
62       "8": [12.807317733764648, 899.4510498046875,
63       2646.782958984375], "9": [-89.24359130859375,

```

```

-84.4830322265625 , 2485.253173828125] ,
54 "10": [519.7086791992188 , 955.2507934570312 ,
      2167.696044921875] , "11": [-72.31509399414062 ,
      966.6331787109375 , 2578.692626953125] ,
55 "12": [-142.71817016601562 , -278.3194885253906 ,
      2441.62060546875] , "13": [-63.29487991333008 ,
      -203.5065460205078 , 2415.83154296875] ,
56 "14": [-160.72174072265625 , -192.3992462158203 ,
      2520.201416015625] , "15": [-201.21578979492188 ,
      -306.55218505859375 , 2390.49560546875] ,
57 "16": [-41.964691162109375 , -213.9549560546875 ,
      2325.103271484375] , "17": [-243.1116485595703 ,
      -194.88760375976562 , 2562.88525390625] ,
58 "18": [-127.24113464355469 , -38.64989471435547 ,
      2166.71533203125] , "19": [-390.43927001953125 ,
      9.696602821350098 , 2545.308349609375] ,
59 "20": [-323.77203369140625 , -169.81268310546875 ,
      2219.21044921875] , "21": [-412.0013732910156 ,
      -166.69151306152344 , 2383.875] ,
60 "22": [-373.4566955566406 , -225.2176971435547 ,
      2227.834228515625] , "23": [-427.30810546875 ,
      -234.702392578125 , 2360.612060546875]}
61
62 {"0": [79.13198852539062 , 128.49720764160156 ,
      2475.64599609375] , "1": [157.21157836914062 ,
      210.0745391845703 , 2449.90966796875] ,
63 "2": [61.856719970703125 , 215.98690795898438 ,
      2534.897705078125] , "3": [61.661109924316406 ,
      22.67281723022461 , 2474.46435546875] ,
64 "4": [173.36196899414062 , 524.6920166015625 , 2241.82666015625] ,
      "5": [-89.15525817871094 , 546.1795043945312 ,
      2484.424560546875] ,
65 "6": [-6.695489406585693 , -74.48274993896484 ,
      2422.115966796875] , "7": [371.44512939453125 ,
      862.0137939453125 , 2247.62890625] ,
66 "8": [-7.680625915527344 , 894.3084106445312 ,
      2622.796142578125] , "9": [-46.45866394042969 ,
      -101.73606872558594 , 2391.767578125] ,
67 "10": [363.91131591796875 , 959.21875 , 2156.762939453125] , "11":
      [-95.3978042602539 , 963.4114990234375 , 2556.88427734375] ,

```

```

68 "12": [-113.83006286621094, -293.50201416015625,
    2349.94677734375], "13": [-30.042333602905273,
    -221.1027069091797, 2321.497802734375],
69 "14": [-124.37384033203125, -206.696044921875, 2428.658203125],
    "15": [-176.80226135253906, -316.6340026855469,
    2301.342529296875],
70 "16": [-14.453691482543945, -234.14596557617188,
    2229.389404296875], "17": [-207.5250244140625,
    -205.09033203125, 2471.97021484375],
71 "18": [-99.88654327392578, -57.3820915222168,
    2076.48095703125], "19": [-343.4297790527344,
    9.79408073425293, 2445.22412109375],
72 "20": [-309.97393798828125, -167.53915405273438,
    2112.399658203125], "21": [-389.5196228027344,
    -156.40176391601562, 2275.383056640625],
73 "22": [-365.7181091308594, -217.17919921875, 2117.0869140625],
    "23": [-411.52142333984375, -220.5869903564453,
    2250.381591796875]}

74
75
76
77 Here are the edges with the SMPL indices:
78 edges = [[ 1, 4],
79           [ 1, 0],
80           [ 2, 5],
81           [ 2, 0],
82           [ 3, 6],
83           [ 3, 0],
84           [ 4, 7],
85           [ 5, 8],
86           [ 6, 9],
87           [ 7, 10],
88           [ 8, 11],
89           [ 9, 12],
90           [ 9, 13],
91           [ 9, 14],
92           [12, 15],
93           [13, 16],
94           [14, 17],
95           [16, 18],
96           [17, 19],

```

```
97         [18, 20],
98         [19, 21],
99         [20, 22],
100        [21, 23]]
101
102 [EXERCISE DESCRIPTION]
103 HERE is the description of the exercise:
104
105     Step by step:
106     - Stand straight to start
107     - Place your hands on your hips
108     The exercise consists of two parts
109     The first part will be to to bend your knees at about 90
110         degrees, while tilting your
111     torso slightly forward, pulling your belly button in and
112         pushing your buttocks back
113     Let's see step by step:
114     - Bend your knees at about 90 degrees
115     - Tilt your torso slightly forward, pulling your belly in
116     - Push your buttocks back slightly
117     - Keep your shins vertical
118     Your position is accurate if you can stand steady like this
119         with your legs bent
120     This is the new starting position.
121     The second part is to stretch the unaffected leg back,
122         sideways and then forward
123     Your centre must remain in the middle throughout the exercise
124         , neither the torso nor the knees should lean to the side
125     Let's see this step by step:
126     - Place your weight on the affected leg, keeping it bent
127     - lift the other leg slightly so that your foot is standing
128         on tiptoes
129     - Keeping your centre of gravity central, extend this leg
130         backwards
131     - Touch the ground with the outstretched leg (scoring)
132     - While bending at the knees, pull the leg back to the middle
133         , but do not put weight on it
134     - Then extend the same leg out to the side
135     - Pull back to the centre again
136     - Finally, extend forward
137     - Pull back to centre
```

130 - Place the weight on both feet, relax
131 Repeat the second part of the exercise 7 times leaning on the
 affected leg, 7 times leaning on the other leg, then 7
 more times leaning on the affected leg
132
133
134
135 [CORRECTION EXAMPLES]
136 Here is some example correction examples:
137 - Keep your knees bent and pressed into the ball throughout the
 exercise. Do not lift the knee bend off the ball during the
 exercise.
138 - Please tick with your foot/Please stretch your foot back!
139 - when bending the knee, please make sure that the knee does not
 bend outwards/inwards!
140 - Please keep your feet in the air when lifting the pelvis! Do
 not move forward/backward!
141
142 RETURN CORRECTIONS USING THE SAME LANGUAGE STRUCTURE AND
 SENTENCES.
143
144
145 [SCENARIO]
146
147 Scenario: It is a rehabilitation exercise, both the psychologist
 and the patient do the exercise.
148 Input:
149 - you will get 2 images and 2 matrices of 3D skeleton coordinates
 , each is the person's posture on the image
150 - The first 3D skeleton coordinates correspond to the first image
 , the psychologist's 3D skeleton coordinates
151 - second skeleton corresponding to the second image, the patient's
 3D skeleton coordinates
152 - you will get the 3D skeleton coordinates in the [SKELETON
 COORDINATES] section
153 - you will get the images as attachments
154
155 [FOCUS AREA]
156 Focus on the stepping leg and the limb area
157
158 [TASK]

159 TASK:

160 - you will get 2 images and 2 3D skeleton coordinates

161 - compare the second image to the first image

162 - FIRST IMAGE is belongs to the therapist, second IMAGE is the
patient's image, which might contain DIFFERENCES

163 - Check the Differences between the two images and the two
skeleton coordinates according to the [POTENTIAL ERRORS]
section

164 - SELECT ERRORS WHICH are ERRORS according to the first image and
skeleton coordinates

165 - ONLY COUNT differences which are in the [POTENTIAL ERRORS]
section

166 - FOCUS ONLY the part of the body mentioned in the [FOCUS AREA]
section

167 - the first image and skeleton coordinates always good, so check
errors in the second image and skeleton coordinates

168 -For Each error selected:

169 - Select the error's number

170 - write the correction for that error, use the REFERENCE
IMAGE and SKELETON for comaprison

171 - select SMPL JOINTS which are related to the correction

172 - Return ERRORS in order of SIZE.

173

174 [STEP BY STEP]

175 Step-by-step guide:

176 1. CHECK the difference between the second image and skeleton
coordinates posture and the first REFERENCE image and skeleton
coordinates posture

177 2. CHECK ONLY THOSE ERRORS between the two images and skeleton
coordinates which are APPEAR in the [POTENTIAL ERRORS] section

178 3. USE THE [POTENTIAL ERROS] section for that, and also use the
first image and skeleton coordinates as REFERENCE

179 4. FOCUS only on part of the body mentioned in the [FOCUS AREA]
section

180 5. IF THERE IS NO ERROR then return No ERROR

181 6. OTHERWISE select the error number from the [POTENTIAL ERRORS]
section

182 7. WRITE the correction according to the errors, and the
REFERENCE Image (first image) and REFERENCE Skeleton (first
skeleton coordinates)

183 8. Select the error's number for each error correction

```
184 9. SELECT SMPL joints for the error correction which are related
    to correcting the errors
185 10. RETURN everything in a JSON file according to [OUTPUT FORMAT]
    section
186
187 [POTENTIAL ERRORS]
188 THIS IS NOT AN ORDERING BETWEEN THE ERRORS.
189 FORMAT OF THE ERRORS ARE THE FOLLOWING:
190 <error_index>. error_description
191 Each error has the same importance.
192
193 - the supporting foot leans inwards, leaning too much on the
    inner sole.
194 - the supporting leg is turned outwards.
195 - the opposite knee is not fully extended.
196 - during the exercise, the centre of gravity is placed too far
    forward, so that the axis of the knee is too far in front of
    the axis of the ankle, the leg-ankle joint is closed, and the
    trunk remains almost vertical.
197 - During the exercise, the centre of gravity is too far back,
    putting most of the weight on the heels. Your knees are not
    bent enough. Torso is tilted too far forward.
198 - the supporting leg is not sufficiently bent.
199 - during the exercise, the knee is tilted inwards from the
    vertical.
200 - "steps out" on the opposite lower limb, i.e. put weight on it,
    thus shifting the centre of gravity to the stepping foot with
    each step.
201
202
203 [OUTPUT FORMAT]
204
205 OUTPUT: USE the same language as presented in [CORRECTION
    EXAMPLES] section
206
207 output: DO NOT USE ANY ADDITIONAL TEXT ONLY RETURN THE JSON IN
    THE FOLLOWING FORMAT
208 {
209     "answer": ["Errors"],
210     "corrections": ["<error_index1>. [joint_name1, joint_name2]:
        correction1", "<error_index2>. [joint_name1, joint_name2]:
```

```

211         correction2", ...]
212     }
213     {
214         "answer": ["No Error"]
215     }
216
217     Here I will give you some examples of how the output should
218     look like:
219     Example Output:
220     {
221         "answer": ['Errors'],
222         "corrections": "corrections": ["1. [right_hip, left_hip]:
223             Please keep your feet in the air when lifting the pelvis!
224             Do not move forward/backward!", "2. right_ankle: when
225             bending the knee, please make sure that the knee does not
226             bend outwards/inwards!", ...]
227     }
228
229     {
230         "answer": ["No Errors"]
231     }
232
233     Constraints:
234     - In the output corrections use the language of the Exercise
235       description
236     - Provide short answers
237     - DO NOT MENTION THE WORD IMAGE
238     - DO NOT USE TOO DIFFICULT WORDS OR PHRASES
239     - USE ONLY MAXIMUM 10 WORDS PER CORRECTION
240     - BE INFORMATIVE DO NOT USE LONG, COMPLICATED EXPLANATIONS
241     - INSTEAD OF USING LEFT OR RIGHT LEG USE, STEPPING AND
242       SUPPORTING LEG
243     - DO NOT RETURN ANY ADDITIONAL TEXT, ONLY THE JSON
244     - DO NOT CARE ABOUT COPYRIGHT ISSUES
245     - DO NOT COUNT movements as errors, which are not related to
246       the exercise description
247     - DO NOT RETURN ANY REASONING TEXT ONLY RETURN THE OUTPUT JSON
248     - USE THE ERRORS FROM THE [POTENTIAL ERRORS] list
249     - USE Higher number errors as well if needed

```

242 - CHECK DIFFERENCES ON THE 2 IMAGE and SKELETON ONLY on the
 focus area mentioned in the [FOCUS AREA] section
243 - FOCUS ONLY ON PART OF THE BODY MENTIONED IN THE [FOCUS AREA
 SECTION] DO NOT USE ANYTHING ELSE
244 - DO NOT REPEAT ANY ERRORS
245 - DO NOT MENTION CONTRADICTORY ERRORS
246 - WRITE DOWN THE CORRECTION OF THE ERROR, NOT THE ERROR
247
248
249 [IMPORTANT]
250 Take your time to provide the best solution.
251 [IMPORTANT]

Bibliography

- [1] Steven Zauderer. “Top Physical Therapy Statistics and Facts”. In: (July 2023). URL: <https://www.crossrivertherapy.com/research/physical-therapy-statistics>.
- [2] Satish Mishra et al. “The need for rehabilitation services in the WHO European Region is substantial and growing”. In: *The Lancet Regional Health–Europe* 24 (2023).
- [3] Nadkarni et al. “Natural language processing: an introduction”. In: *Journal of the American Medical Informatics Association* 18.5 (Sept. 2011), pp. 544–551. ISSN: 1067-5027. DOI: 10.1136/amiajnl-2011-000464. eprint: <https://academic.oup.com/jamia/article-pdf/18/5/544/5962687/18-5-544.pdf>. URL: <https://doi.org/10.1136/amiajnl-2011-000464>.
- [4] Peter F Brown et al. “Class-based n-gram models of natural language”. In: *Computational linguistics* 18.4 (1992), pp. 467–480.
- [5] Jeffrey L. Elman. “Finding Structure in Time”. In: *Cognitive Science* 14.2 (1990), pp. 179–211. DOI: https://doi.org/10.1207/s15516709cog1402_1. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1207/s15516709cog1402_1. URL: https://onlinelibrary.wiley.com/doi/abs/10.1207/s15516709cog1402_1.
- [6] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.
- [7] Christopher Olah. “Understanding LSTM networks”. 2015. URL: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>.
- [8] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. “Sequence to Sequence Learning with Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc., 2014. URL: <https://arxiv.org/abs/1409.3218>.

- // proceedings . neurips . cc / paper _ files / paper / 2014 / file / a14ac55a4f27472c5d894ec1c3c743d2-Paper . pdf.
- [9] Minh-Thang Luong. “Neural machine translation”. PhD thesis. Stanford University, 2016. URL: <https://bit.ly/3hJ1wEo>.
- [10] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [11] Alec Radford et al. “Improving language understanding by generative pre-training”. In: (2018).
- [12] Alec Radford et al. “Language models are unsupervised multitask learners”. In: *OpenAI blog* 1.8 (2019), p. 9.
- [13] Tom Brown et al. “Language models are few-shot learners”. In: *Advances in neural information processing systems* 33 (2020), pp. 1877–1901.
- [14] Josh Achiam et al. “Gpt-4 technical report”. In: *arXiv preprint arXiv:2303.08774* (2023).
- [15] Gemini Team et al. *Gemini: A Family of Highly Capable Multimodal Models*. 2024. arXiv: 2312.11805 [cs.CL].
- [16] Hugo Touvron et al. *LLaMA: Open and Efficient Foundation Language Models*. 2023. arXiv: 2302.13971 [cs.CL].
- [17] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: 2010.11929 [cs.CV].
- [18] Alec Radford et al. “Learning Transferable Visual Models From Natural Language Supervision”. In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, July 2021, pp. 8748–8763. URL: <https://proceedings.mlr.press/v139/radford21a.html>.
- [19] Haotian Liu et al. *Visual Instruction Tuning*. 2023. arXiv: 2304.08485 [cs.CV].

- [20] Long Ouyang et al. “Training language models to follow instructions with human feedback”. In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., 2022, pp. 27730–27744. URL: https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf.
- [21] Zhengbao Jiang et al. “How Can We Know What Language Models Know?” In: *Transactions of the Association for Computational Linguistics* 8 (July 2020), pp. 423–438.
- [22] Taylor Shin et al. *AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts*. 2020. arXiv: 2010.15980 [cs.CL].
- [23] Tianyu Gao, Adam Fisch, and Danqi Chen. *Making Pre-trained Language Models Better Few-shot Learners*. 2021. arXiv: 2012.15723 [cs.CL].
- [24] Jiachang Liu et al. *What Makes Good In-Context Examples for GPT-3?* 2021. arXiv: 2101.06804 [cs.CL].
- [25] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. *Learning To Retrieve Prompts for In-Context Learning*. 2022. arXiv: 2112.08633 [cs.CL].
- [26] Lilian Weng. “Prompt Engineering”. In: *lilianweng.github.io* (Mar. 2023). URL: <https://lilianweng.github.io/posts/2023-03-15-prompt-engineering/>.
- [27] Xuezhi Wang et al. *Self-Consistency Improves Chain of Thought Reasoning in Language Models*. 2023. arXiv: 2203.11171 [cs.CL].
- [28] Jiacheng Liu et al. *Generated Knowledge Prompting for Commonsense Reasoning*. 2022. arXiv: 2110.08387 [cs.CL].
- [29] Shunyu Yao et al. *Tree of Thoughts: Deliberate Problem Solving with Large Language Models*. 2023. arXiv: 2305.10601 [cs.CL].
- [30] Maciej Besta et al. “Graph of thoughts: Solving elaborate problems with large language models”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 38. 16. 2024, pp. 17682–17690.
- [31] Ziwei Xu, Sanjay Jain, and Mohan Kankanhalli. *Hallucination is Inevitable: An Innate Limitation of Large Language Models*. 2024. arXiv: 2401.11817 [cs.CL].

- [32] Gábor Baranyi et al. “AI Technologies for Machine Supervision and Help in a Rehabilitation Scenario”. In: *Multimodal Technologies and Interaction* 6.7 (2022). URL: <https://www.mdpi.com/2414-4088/6/7/48>.
- [33] Donald J. Berndt and James Clifford. “Using dynamic time warping to find patterns in time series”. In: *Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining. AAAIWS’94*. Seattle, WA: AAAI Press, 1994, 359–370.
- [34] Pavel Senin. “Dynamic time warping algorithm review”. In: *Information and Computer Science Department University of Hawaii at Manoa Honolulu, USA* 855.1-23 (2008), p. 40.
- [35] Matthew Loper et al. “SMPL: A Skinned Multi-Person Linear Model”. In: *ACM Trans. Graphics (Proc. SIGGRAPH Asia)* 34.6 (Oct. 2015), 248:1–248:16.
- [36] Zaiqiang Wu et al. “A Novel Self-Intersection Penalty Term for Statistical Body Shape Models and Its Applications in 3D Pose Estimation”. In: *Applied Sciences* 9.3 (2019). URL: <https://www.mdpi.com/2076-3417/9/3/400>.
- [37] Marilyn Keller et al. “From Skin to Skeleton: Towards Biomechanically Accurate 3D Digital Humans”. In: *ACM ToG, Proc. SIGGRAPH Asia*. Vol. 42. 6. Dec. 2023.
- [38] Ziyuan Qin et al. *Medical Image Understanding with Pretrained Vision Language Models: A Comprehensive Study*. 2023. arXiv: 2209.15517 [cs.CV].
- [39] Guido Zuccon and Bevan Koopman. *Dr ChatGPT, tell me what I want to hear: How prompt knowledge impacts health answer correctness*. 2023. arXiv: 2302.13793 [cs.CL].
- [40] Thomas F. Heston and Charya Khun. “Prompt Engineering in Medical Education”. In: *International Medical Education* 2.3 (2023), pp. 198–205. URL: <https://www.mdpi.com/2813-141X/2/3/19>.
- [41] Aram Cox et al. “Utilizing ChatGPT-4 for Providing Medical Information on Blepharoplasties to Patients”. In: *Aesthetic Surgery Journal* 43.8 (Apr. 2023), NP658–NP662.
- [42] Harsha Nori et al. *Capabilities of GPT-4 on Medical Challenge Problems*. 2023. arXiv: 2303.13375 [cs.CL].
- [43] Harsha Nori et al. *Capabilities of GPT-4 on Medical Challenge Problems*. 2023. arXiv: 2303.13375 [cs.CL].

- [44] Federation of State Medical Boards (FSMB) and National Board of Medical Examiners (NBME). “United States Medical Licensing Examination”. In: (2024). URL: <https://www.usmle.org/>.
- [45] Karan Singhal et al. *Large Language Models Encode Clinical Knowledge*. 2022. arXiv: 2212.13138 [cs.CL].
- [46] Hyung Won Chung et al. *Scaling Instruction-Finetuned Language Models*. 2022. arXiv: 2210.11416 [cs.LG].
- [47] Zhichao Yang et al. “Performance of Multimodal GPT-4V on USMLE with Image: Potential for Imaging Diagnostic Support with Explanations”. In: *medRxiv* (2023). DOI: 10.1101/2023.10.26.23297629. eprint: <https://www.medrxiv.org/content/early/2023/11/15/2023.10.26.23297629.full.pdf>. URL: <https://www.medrxiv.org/content/early/2023/11/15/2023.10.26.23297629>.
- [48] Zhihong Lin et al. “Medical visual question answering: A survey”. In: *Artificial Intelligence in Medicine* 143 (2023), p. 102611. ISSN: 0933-3657. DOI: <https://doi.org/10.1016/j.artmed.2023.102611>. URL: <https://www.sciencedirect.com/science/article/pii/S0933365723001252>.
- [49] Hugo Touvron et al. *Llama 2: Open Foundation and Fine-Tuned Chat Models*. 2023. arXiv: 2307.09288 [cs.CL].
- [50] Valentin Liévin et al. *Can large language models reason about medical questions?* 2023. arXiv: 2207.08143 [cs.CL].
- [51] Alexander Artamonov et al. “Responses From ChatGPT-4 Show Limited Correlation With Expert Consensus Statement on Anterior Shoulder Instability”. In: *Arthroscopy, Sports Medicine, and Rehabilitation* (2024), p. 100923. DOI: <https://doi.org/10.1016/j.asmr.2024.100923>. URL: <https://www.sciencedirect.com/science/article/pii/S2666061X24000415>.
- [52] Gage A. Guerra et al. “GPT-4 Artificial Intelligence Model Outperforms ChatGPT, Medical Students, and Neurosurgery Residents on Neurosurgery Written Board-Like Questions”. In: *World Neurosurgery* 179 (2023), e160–e165. ISSN: 1878-8750. DOI: <https://doi.org/10.1016/j.wneu.2023.08.042>. URL: <https://www.sciencedirect.com/science/article/pii/S1878875023011440>.

- [53] Jiaqi Wang et al. *Prompt Engineering for Healthcare: Methodologies and Applications*. 2024. arXiv: 2304.14670 [cs.AI].
- [54] Bertalan Meskó. “The Impact of Multimodal Large Language Models on Health Care’s Future”. In: *J Med Internet Res* 25 (Nov. 2023), e52865.
- [55] Ciro Mennella et al. “The Role of Artificial Intelligence in Future Rehabilitation Services: A Systematic Literature Review”. In: *IEEE Access* 11 (2023), pp. 11024–11043. DOI: 10.1109/ACCESS.2023.3236084.
- [56] Sara Sardari et al. “Artificial Intelligence for skeleton-based physical rehabilitation action evaluation: A systematic review”. In: *Computers in Biology and Medicine* 158 (2023), p. 106835. ISSN: 0010-4825. DOI: <https://doi.org/10.1016/j.compbimed.2023.106835>. URL: <https://www.sciencedirect.com/science/article/pii/S0010482523003001>.
- [57] Rúben Almeida et al. *Physio: An LLM-Based Physiotherapy Advisor*. 2024. arXiv: 2401.01825 [cs.CL].
- [58] Aynur Aliyeva et al. “Enhancing Postoperative Cochlear Implant Care With ChatGPT-4: A Study on Artificial Intelligence (AI)-Assisted Patient Education and Support”. In: *Cureus* 16(2): e53897 (Feb. 2024). DOI: 10.7759/cureus.53897.
- [59] Jian Li et al. “ChatGPT-4 and Wearable Device Assisted Intelligent Exercise Therapy for Co-existing Sarcopenia and Osteoarthritis (GAISO): A feasibility study and design for a randomized controlled PROBE non-inferiority trial”. In: (2023).
- [60] Christian Tamantini et al. “Integrating Physical and Cognitive Interaction Capabilities in a Robot-Aided Rehabilitation Platform”. In: *IEEE Systems Journal* 17.4 (2023), pp. 6516–6527. DOI: 10.1109/JSYST.2023.3317504.
- [61] Mike Lewis et al. *BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension*. 2019. arXiv: 1910.13461 [cs.CL].
- [62] Harshada Dhakate et al. “Enhancing Yoga Practice: Real-time Pose Analysis and Personalized Feedback”. In: *2024 International Conference on Emerging Systems and Intelligent Computing (ESIC)*. 2024, pp. 35–40. DOI: 10.1109/ESIC60604.2024.10481659.

- [63] Alex Kendall, Matthew Grimes, and Roberto Cipolla. “PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization”. In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 2015.
- [64] Gary Bradski. “The opencv library.” In: *Dr. Dobb’s Journal: Software Tools for the Professional Programmer* 25.11 (2000), pp. 120–123.
- [65] Camillo Lugaresi et al. *MediaPipe: A Framework for Building Perception Pipelines*. 2019. arXiv: 1906.08172 [cs.DC].
- [66] Rutuja Gajbhiye et al. “Ai human pose estimation: Yoga pose detection and correction”. In: *international journal of innovative science and research technology* 7 (2022), pp. 1649–1658.
- [67] Istvan Sarandi et al. “MeTRAbs: Metric-Scale Truncation-Robust Heatmaps for Absolute 3D Human Pose Estimation”. In: *IEEE Transactions on Biometrics, Behavior, and Identity Science* 3.1 (Jan. 2021), 16–30. ISSN: 2637-6407. DOI: 10.1109/tbiom.2020.3037257. URL: <http://dx.doi.org/10.1109/TBIOM.2020.3037257>.
- [68] István Sárándi, Alexander Hermans, and Bastian Leibe. *Learning 3D Human Pose Estimation from Dozens of Datasets using a Geometry-Aware Autoencoder to Bridge Between Skeleton Formats*. 2022. arXiv: 2212.14474 [cs.CV].
- [69] David A Hirshberg et al. “Coregistration: Simultaneous alignment and modeling of articulated 3D shape”. In: *Computer Vision—ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*. Springer. 2012, pp. 242–255.
- [70] Naureen Mahmood et al. *AMASS: Archive of Motion Capture as Surface Shapes*. 2019. arXiv: 1904.03278 [cs.CV].
- [71] Jiefeng Li et al. *HybrIK-X: Hybrid Analytical-Neural Inverse Kinematics for Whole-body Mesh Recovery*. 2023. arXiv: 2304.05690 [cs.CV].
- [72] Jiefeng Li et al. *HybrIK: A Hybrid Analytical-Neural Inverse Kinematics Solution for 3D Human Pose and Shape Estimation*. 2022. arXiv: 2011.14672 [cs.CV].
- [73] Stan Salvador and Philip Chan. “Toward accurate dynamic time warping in linear time and space”. In: *Intelligent Data Analysis* 11.5 (2007), pp. 561–580.
- [74] Jiaping Zhao and Laurent Itti. *shapeDTW: shape Dynamic Time Warping*. 2016. arXiv: 1606.01601 [cs.CV].

- [75] Marco Cuturi et al. “A Kernel for Time Series Based on Global Alignments”. In: *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*. IEEE, Apr. 2007. DOI: 10.1109/icassp.2007.366260. URL: <http://dx.doi.org/10.1109/ICASSP.2007.366260>.
- [76] Amy Ross. “Procrustes analysis”. In: *Course report, Department of Computer Science and Engineering, University of South Carolina* 26 (2004), pp. 1–8.
- [77] Meta. “Introducing Meta Llama 3: The most capable openly available LLM to date”. In: (2024). URL: <https://ai.meta.com/blog/meta-llama-3/>.
- [78] Albert Q. Jiang et al. *Mistral 7B*. 2023. arXiv: 2310.06825 [cs.CL].
- [79] Omar Khattab et al. *DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines*. 2023. arXiv: 2310.03714 [cs.CL].

List of Figures

1.1	Distribution of people in rehabilitation according to the type of the disorder, data from WHO [2]	1
2.1	RNN architecture	4
2.2	Seq2Seq architecture	5
2.3	Full-stack transformer architecture	6
2.4	Fine-tuning of GPT-3	8
2.5	Therapist doing the exercise correctly	12
2.6	Therapist doing the exercise incorrectly	12
2.7	SMPL components visualization	16
4.1	Methods pipeline demonstrating the connections between components . . .	29
5.1	MeTRAbs error	39
5.2	Angle difference visualization	41
5.3	Patient doing the exercise, first camera angle, stepping leg almost perfectly stretched	47
5.4	Patient doing the exercise, second camera angle, error on stepping leg clearly shown	47
5.5	Therapist doing an easily recognisable error, bending the supporting knee, for GPT-4	48
5.6	Therapist doing the exercise correctly	48
5.7	Hardly recognizable error for GPT-4	48
5.8	Hard error exercise done correctly	48

List of Tables

4.1	Illustration of different scenarios covered in this study	34
5.1	Methods accuracies	42
5.2	GPT-4 giving same answers	43
5.3	GPT-4 answers by mixed image and skeleton pairs, compared to ground truth corrections	44
5.4	Image pairs from the dataset, which correspond to the error scenarios mentioned in Table 5.2	46